

## A Neural Network in Convolution with Constant Error Carousel Based Long Short Term Memory for Better Face Recognition

P. Ramaraj

Assistant Professor, Department of Computer Science,  
Government Arts and Science College, Veppanthattai-621116  
(Affiliated Bharathidasan University- Tiruchirappalli)  
Perambalur - Dt, TAMILNADU, INDIA.

**Article History:** Received: 10 November 2020; Revised: 12 January 2021; Accepted: 27 January 2021;  
Published online: 05 April 2021

**Abstract:** Unconstrained face identification, facial periocular recognition, facial land marking and pose prediction, facial expression recognition, 3D facial model design, and other facial-related problems require robust face detection in the wild. Despite the fact that the face recognition issue has been researched intensively for decades with different commercial implementations, it nevertheless faces problems in certain real-world scenarios due to multiple obstacles, such as severe facial occlusions, incredibly low resolutions, intense lighting, exceptionally pose inconsistencies, picture or video compression artefacts, and so on. To solve the problems described above, a face detection technique called Convolution Neural Network with Constant Error Carousel dependent Long Short Term Memory (CNN-CEC-LSTM) is proposed in this paper. This research implemented a novel network structure and designed a special feature extraction that employs a self-channel attention (SCA) block and a self-spatial attention (SSA) block that adaptively aggregates the feature maps in both channel and spatial domains to learn the inter-channel and inter-spatial connection matrices; additionally, matrix multiplications are conducted for a This approach first smoothed the initial image with a Gaussian filter before measuring the gradient image. The Canny-Kirsch Method edge detection algorithm was then used to identify human face edges. The proposed method is evaluated against two recent difficult face detection databases, including the IIT Kanpur Dataset. The experimental findings indicate that the proposed approach outperforms the most current cutting-edge face recognition approaches.

**Keywords:** Face detection, Image processing, Convolution Neural Network with Constant Error Carousel based Long Short Term memory, feature extraction

### 1. Introduction

Face detection applies to technologies that can recognise or validate the identification of subjects in photographs or recordings. Face recognition's non-intrusive existence is one of the distinctive characteristics that renders it more attractive than other biometric modalities. For eg, fingerprint recognition requires users to put their finger on a sensor, iris recognition necessitates users moving really close to a camera, and speaker recognition necessitates users speaking out loud. Modern facial recognition devices, on the other hand, only enable users to be inside the field of view of a camera (provided that they are within a reasonable distance from the camera). Face detection is therefore the most user-friendly biometric modality. That also ensures that facial detection has a broader variety of possible uses, since it may be used in situations where people are not likely to comply with the technology, such as security devices. Face recognition is also used for access management, fraud prevention, identity checking, and social networking. Owing to the high uncertainty of facial images in the modern world, face identification is one of the most difficult biometric modalities to apply in unconstrained settings (these type of face images are commonly referred to as faces in-the-wild). Head poses, ageing, occlusions, lighting patterns, and facial movements are some of the variants.

The challenge of designing features that were resistant to the various variations found in unconstrained environments prompted researchers to concentrate on specialised methods for each form of variance, such as age-invariant methods [1], [2], pose-invariant methods [3], and so on. Deep learning approaches focused on convolutional neural networks have recently overcome conventional facial recognition methods (CNNs). Deep learning models have the biggest benefit of being able to be trained for very broad datasets to learn the right features to reflect the results. CNN-based facial recognition [4] methods educated on these databases have recently attained very high precision because they can acquire features that are resilient to the real-world differences found in the face photos used during testing. Furthermore, as CNNs are used to solve many other computer vision tasks, such as object detection and identification, segmentation, optical character recognition, facial expression analysis, age prediction, and so on, the popularity of deep learning approaches for computer vision has intensified face recognition science.

The effect of this is that the memory needs are limited, as is the amount of criteria to be learned. As a result, the algorithm's accuracy is enhanced. At the same time, other machine learning algorithms need us to do pre-

processing or feature extraction on the images. However, when using CNN for image processing, these operations are rarely used. This is something that other deep learning systems are incapable of. In-depth learning has several flaws as well. This thesis suggested a Convolution Neural Network with Constant Error Carousel dependent Long Short Term Memory (CNN-CEC-LSTM) for human face recognition based on this inspiration. Initially, this approach smoothed the original image with a Gaussian filter and computed the gradient image as a pre-processing step. The Canny-Kirsch Method edge detection algorithm was then used to identify human face edges. After that, the global feature dependencies in both spatial and channel dimensions are captured using a self-residual attention-based network (SRANet) for discriminative face feature embedding. To the best of our understanding, this is the first time a self-attention mechanism has been used to optimise visual features of image-based face recognition. The rest of this paper goes into the specifics of the facial recognition method. Section 2 discusses the associated work, which is the current Face Recognition technology. Section 3 explains the proposed classifier's architecture. Section 4 describes experimental findings from testing the existing methods, as well as discussions. Finally, in Section 5, the findings are outlined with possible suggestions.

## 2. Related Work

Many facial recognition algorithms have been suggested. [5] proposes a patch-based approach for generating a simulated frontal view from a non-frontal face picture using MRFs and an effective version of the BP algorithm. A collection of potential warps for each patch in the input picture is obtained by aligning it with images from a training database of frontal faces. The alignments are then done effectively in the frequency domain using an illumination invariant extension of the Lucas-Kanade (LK) algorithm. The algorithm's aim is to find the globally optimal range of local warps that can be used to predict picture patches in frontal view. However, a different strategy is needed to minimise the impact of patch size on the performance.

[6] introduces a modern human face recognition algorithm focused on bidirectional two-dimensional principal component analysis (B2DPCA) and intense learning system (ELM). The suggested approach is focused on curvelet picture decomposition of human faces, and a subband with the highest standard deviation is dimensionally decreased using a new dimensionality reduction technique. In particular, three significant contributions were made in [7]: 1) A clear and effective pre-processing chain is provided that removes the majority of the effects of shifting lighting while maintaining the critical appearance information needed for recognition; 2) local ternary patterns (LTP), a generalisation of the local binary pattern (LBP) local texture descriptor that is more discriminant and less susceptible to noise in standardised areas, and sho

Machine learning and the generalisation capability of support vector models (SVMs) was used in [8] user authentication schemes to ensure a minor classification error. By training an SVM classifier on user facial features correlated with wavelet transforms and a spatially enhanced local binary pattern, this study created an online face-recognition framework. For solving classification precision issues, a cross-validation scheme and SVMs aligned with the Olivetti Testing Laboratory database of consumer facial features were used.

[9] identified a novel Gabor phase-based illumination invariant extraction approach that aimed to eradicate the impact of varying illumination on face recognition. For instance, it normalises varying lighting on face pictures, which may minimise the influence of varying illumination to a certain degree. Second, for image transformation, a collection of 2D actual Gabor wavelets of separate directions is used, and several Gabor coefficients are merged into one whole in terms of continuum and phase. Finally, by removing the phase function from the combined coefficients, the illumination invariant is obtained.

Centered on discriminant analysis, suggested a procedure for constructing a composite feature vector for face recognition in [10]. Using the discriminant feature extraction process, first remove the holistic and local features from the entire face picture and different forms of local pictures. Then, for face recognition, it measures the sum of discriminative knowledge in the human holistic and local features and constructs composite features with only discriminative features. [11] suggested a pixel sorting system for facial recognition in a face picture dependent on discriminant characteristics. Through evaluating the interaction between the pixels in face images and the features derived from them, the pixels with the most discriminative information are used, while the pixels with the least discriminative information are discarded.

[12] used subject-based SVM classifiers to classify individuals after fine-tuning a qualified base model of a symmetric BCNN to remove face characteristics. Pyramid CNN demonstrated a pyramid-like configuration with several CNNs in [13]. Two videos are fed into each CNN, and the SIAMESE network is used to train it. The output neurons equate the outputs and predict if the two face images are distinct. The pyramid CNN is greedily

educated; if the first layer is well-trained, the next layer is trained. The performance is a multi-scale landmark-based function with a highly compact characteristic. A discussion of similar facial recognition approaches focused on deep learning and other methods. Although the accuracies of many still image-based face matching have increased, there are still some difficulties in practise. As a result, the aim of this work is to investigate the curious issue of how a machine performs face recognition in the presence of incomplete facial knowledge as recognition cues. More importantly, this work seeks to examine how different aspects of the face perform on the role of face recognition.

### 3. Proposed Methodology

This paper proposes a face recognition approach focused on a Convolution Neural Network with Constant Error Carousel and Long Short Term Memory (CNN-CEC-LSTM). It is a three-layer architecture framework that recognises all image regions containing faces. Face detection is a step in the pre-processing of an automated face recognition device. The picture Enhancement using Retinex-based adaptive filter is used in the first stage to eliminate excess noise. The face edge is identified using the Canny-Kirsch Method edge detection, and attribute extraction is performed using the SCA and SSA. Finally, the CNN-CEC-LSTM categorises the undecided or non-face class as either face or non-face. Figure 1 illustrates the proposed CNN-CEC-LSTM-based facial recognition block diagram.

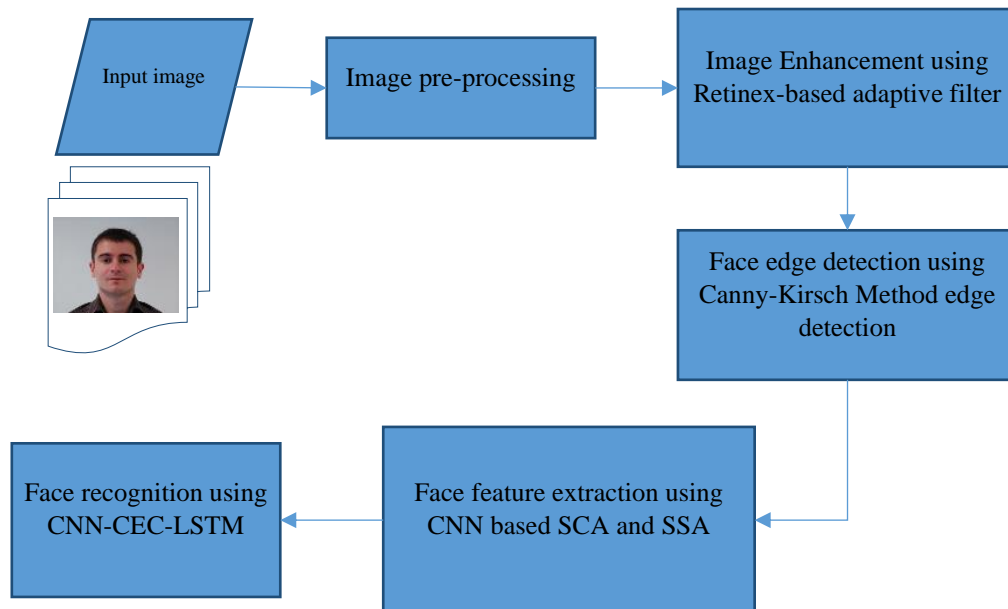


Figure 1. The block diagram of CEC-LSTM-CNN based face recognition

#### Image pre-processing

Since the various environments, such as the different sources, can cause the final outcome in picture pre-processing. The suggested method would change the light intensity of the facial expression captured in the image pictures. In this picture, higher brightness of the pixel interval was used to measure the mean, and this average value was used as the comparison and measurement in equations (1 & 2);

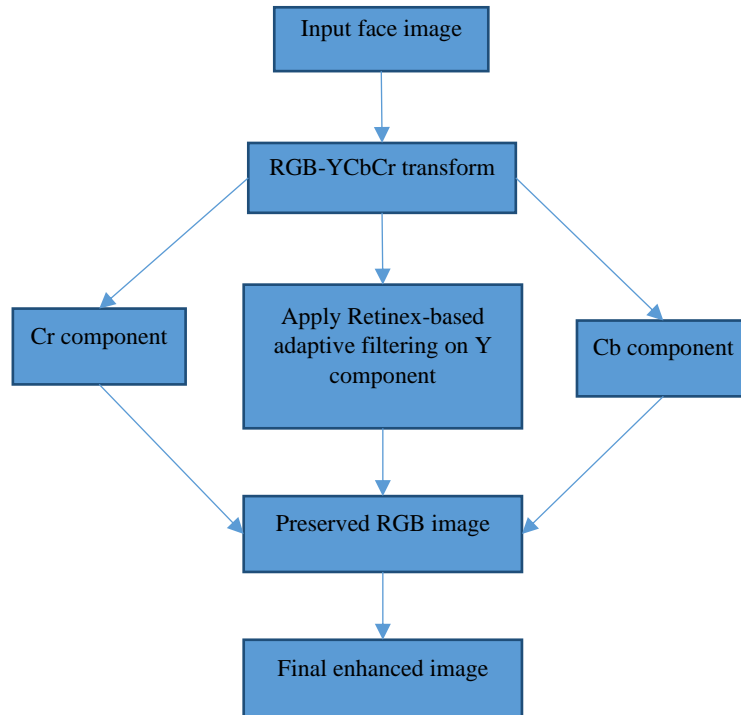
$$R_a = \frac{\sum_1^n \max N_R}{n}, G_a = \frac{\sum_1^n \max N_G}{n}, B_a = \frac{\sum_1^n \max N_B}{n} \tag{1}$$

$$R' = \frac{255}{R_a} * N_R, G' = \frac{255}{G_a} * N_G, B' = \frac{255}{B_a} * N_B \tag{2}$$

The original image pixel adjustments based on average values. Where  $N_R, N_G$  and  $N_B$  represent the original image pixel values. R, G, and B represent the average value of a pixel in the colour channel period of  $n$ , here  $n$  represent the total number of pixels used to retrieve a brightness set, and represent pixel values after modification. This picture pre-processing can be used to correct for pictures captured in poor light.

### Image Enhancement using Retinex-based adaptive filter

In this part, a system for improving colour images is suggested using a Retinex-based adaptive filter. This framework can be used to improve standard 24-bit images as well as compress high dynamic range images that are linear RGB images generated from raw format or multiple exposure technique. Figure 2 depicts the enhancement of a human face picture utilising Retinex-based adaptive filtering.



**Figure 2.** Human face image enhancement using Retinex-based adaptive filtering

The Retinex-based algorithm that is used on the luminance channel Y is defined in this segment. The algorithm derives its new pixel value from the Retinex principle by calculating the ratio of the treated pixel to a weighted average of other pixels in the picture. Let the treated luminance variable be described by Retinex theory as:

$$Retinex_Y = \log_{10}(I'_Y) - \log_{10}(mask) \tag{3}$$

where  $\log_{10}(I'_Y)$  is the Y component of the non-linear RGB image  $I$  and transformed into YCbCr color space. The last term mask, is a matrix that represents for each pixel the weighted average of its surround. An important point is how this surround and its corresponding weights are defined. A traditional approach is to define the mask using a convolution of the image with a filter.

$$mask = I'_Y * F \tag{4}$$

where  $F$  is a circularly symmetric low-pass filter that is completely determined by a 1-dimensional function rotated around the z axis, and the 1-dimensional curve is normally defined by a plain Gaussian or a mixture of Gaussian functions. The radial 1-dimensional feature is a Gaussian curve with a spatial constant that differs with the local contrast of the face image. The initial value of the spatial constant is provided by equation (5). If a high contrasted edge crosses the radius, is separated by 2.

$$\sigma = \frac{r_{max}}{8}, \text{ where } \max(I_{size}) = r_{max} \tag{5}$$

Since the filter's weights and support are adapted for each pixel, the mask is computed sequentially pixel after pixel and  $mask(x, y)$  is the weighted sum of elements in the surround of the pixel of coordinate  $(x, y)$ .

$$mask(x, y) = \int_{\theta=0}^{360} \int_{r=0}^{r_{max}} I_Y(x + \cos \theta), y + \sin(\theta) e^{-\frac{r^2}{\sigma^2}} \quad (6)$$

where  $\sigma$  is the Gaussian spatial constant that varies along the radial direction. In this way, the filter's support approximately follows the image's high contrast face edges. These face edges are detected using the Kirsch edge detection.

### Face Edge detection using Canny-Kirsch Method

Firstly, use Canny edge-detection algorithm to calculate gradient images of original images, then carry on Kirsch calculation to gradient images, instead of to original images directly and denoted as CK. From the use of Retinex-based adaptive filtering original images are smoothed, then calculate  $mask(x, y)$  at certain point and edge direction.

After that carry on Kirsch calculation to gradient image calculated. Suppose an image  $I$  has  $Height \times Width$  pixel points, its edge pixels are usually not in excess of  $5 \times Height$ . To the image that has a certain target, it is a comparatively loose defined value. Get the initial threshold value  $T_0$  to calculate Kirsch arithmetic operator of each pixel point  $i$ .

If  $CK(i) > T_0$ , then  $i$  is marginal point, the edge points are  $N + 1$ , once the edge points surpass  $5 \times Height$ , but  $i$  is less than pixel points of the whole image, that means the threshold values are so low that many pixels that are not marginal pointed are extracted. So threshold values need to be raised, and the minimum  $CK(i)$  which satisfy  $(i) > T_0$ , is accounted as  $K_{min}$ , then the minimum value will be taken as the new threshold value. The whole process is adjusted according to the following methods:

**Input:** image  $I$

**Output:** Edge detection results

1. Initialize  $I$ , Pixel points  $Height \times Width$ , mask Coordinates  $(x, y)$
2. Condition check  $\rightarrow$  if  $CK(i) > T_0$
3. Then record the marginal point  $i$  and set the minimum value of  $CK(i)$  as  $CK_{min}$
4. Increase the edges point  $N + 1$
5. If  $N \geq 5 \times Height$  and satisfy lowest edge requirement
6. Adjust the threshold values to minimum place as  $T_0 = CK_{min}$
7. Repeat step 1 with the new threshold points
8. If condition satisfies the new marginal point as  $newN$
9. Assign sum of new edge points to the counting of  $N$  as  $N = newN$
10. Continue this process until if  $N < 5 \times Height$
11. Finally make  $T_2 = T_1, T_1 = \beta T_0 // \beta$  is a constant  $0 < \beta < 1$
12. End
13. End

After above process the Edge extraction is done with the use of two threshold values  $T_1$  and  $T_2$  to do threshold values analysis of the gradient images produced from the first step, among which the pixels that are bigger than  $T_2$  are called strong edge pixels, the pixels that are between  $T_1$  and  $T_2$  are called weak edge pixels. Only when strong edge pixels and weak edge pixels are link together, weak edge pixels will be included in output.

### Face Feature Extraction and recognition using CNN-CEC-LSTM

This paper expands on the convergence of the Convolutional Neural Network (CNN) with the Constant Error Carousel-based Long Short-Term Memory (CEC-LSTM), culminating in a new framework in the well-explored area of visual processing and facial image recognition. LSTM is a form of Recurrent Neural Network (RNN) that can remember long-term dependencies. When used in a layered order, LSTMs were found to be capable of supplementing CNN's feature extraction capacity. LSTMs can selectively recall trends over long periods of time, and CNNs can remove the essential features from them. When used for facial image recognition, this CEC-

LSTM-CNN layered structure outperforms traditional CNN classifiers. Figure 3 depicts the proposed CEC-LSTM-CNN.

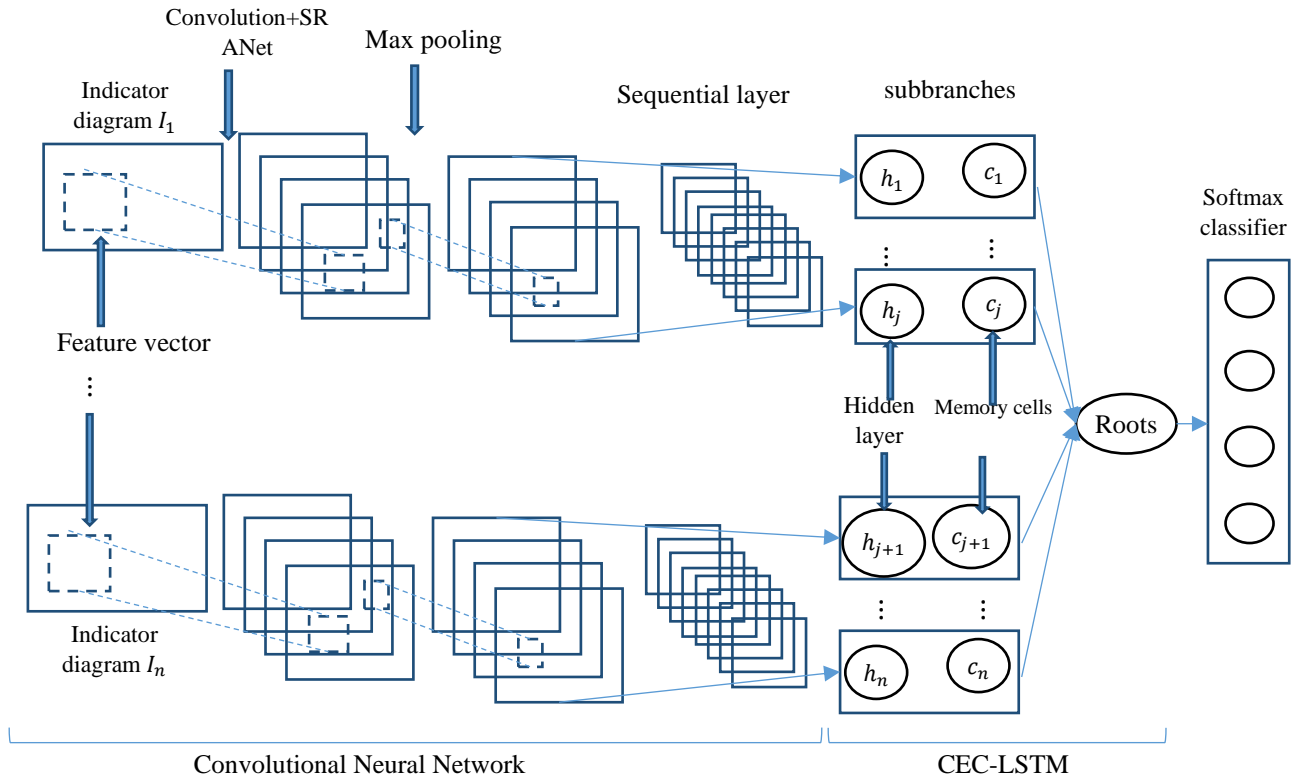


Figure 3. Face recognition using CNN-CEC-LSTM

**CNN based feature extraction**

This segment suggests a facial recognition system based on CNN-CEC-LSTM. And the network used in this case has nine levels. Convolution layers, pooling layers, full-connected layers, and a Softmax regression layer are among the layers included in this package. For feature extraction, convolution and pooling layers are used, accompanied by full-connected layers, and the final layer employs a CEC-LSTM classifier with good non-linear classification capabilities. Figure 3 portrays the CNN-based function extraction process.

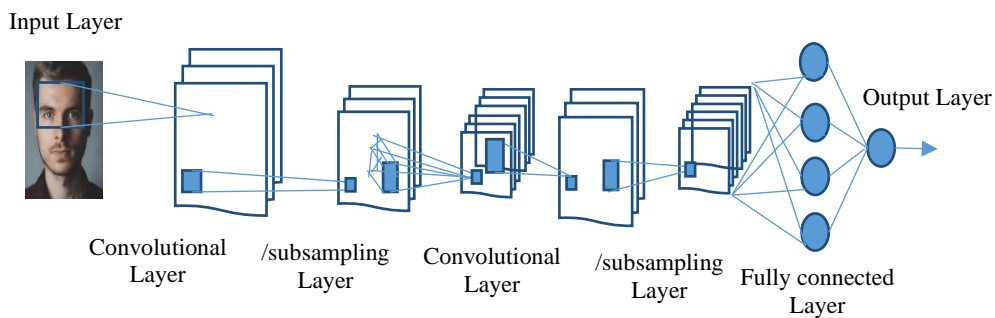


Figure 4. Typical convolutional network architecture for feature extraction

CNN contains three types of layers: input, convolution and pooling, and fully connected layers. In the input layer, information received consists of multiple image sequences  $\{I_1, I_2, \dots, I_n\}$ , and is made up of the predictor diagram dataset. To obtain a refined face function, expand on the original CNN architectures by inserting attention modules on top of each residual bottleneck of the ResNet structure. The proposed attention module, in particular, consists of two blocks called self-residual channel attention module and self-residual spatial attention module, which sequentially learns the channel relationship matrix and spatial relationship matrix, and then achieves the refined function by matrix multiplications.

For eg, provided an intermediate function map FM, the channel refined feature FC and the spatial refined feature  $F_s$  can be obtained sequentially. Furthermore, suggest that the features derived from the global average pooling layer are insufficiently discriminative for deep face recognition, but instead use a completely linked layer. With the above-mentioned improvements, it is possible to reduce knowledge redundancy across channels and learn the most important part of face images [14]. Finally, residual shortcut learning may be used to achieve the refined function. The feature vectors obtained are then fed into the sequential sheet. To catch the long-distance dependency, LSTM is inserted into the vector composition sequential sheet.

**Constant Error Carousel based Long Short Term Memory (CEC-LSTM) for Face Recognition**

The LSTM is an updated variant of the Recurrent Neural Network (RNN). To deal with the issue of disappearing and exploding gradients, LSTM employs memory blocks rather than traditional plain RNN modules. Long-term dependencies are handled even easier by LSTMs than by conventional RNNs. This ensures that LSTMs will recall and relate past knowledge (which is far older than the current) to the present. In LSTM, a memory block is a dynamic processing unit made up of one or more memory cells. As input and output gates, a pair of multiplicative gates are used. A series of adaptive multiplicative gates controls the entire activity of a memory block. The input gate conducts an authorise or discard procedure for a cell activation input flow to a memory cell. The output gate conducts an accept or discard procedure for a memory cell's output state to other nodes. As LSTM research advanced, forget gate and peephole links were added to the current LSTM network. Instead of the Persistent mistake carousel, the forget gate is used (CEC). The forget gate assists in the forgetting or resetting of brain cell conditions. A memory cell's peephole links are made to both of its gates. They discover the exact timing of outputs as well as the internal condition of a memory cell. The following is how the CEC-LSTM works.

The CNN function series is fed into the CEC-LSTM architecture. The performance sequence) of continuous write, read, and reset operations by three multiplicative units (input I output (o), and forget gate (f)) on memory cell (c) is calculated iteratively from  $j = 1, 2, \dots, j+1$  in the LSTM architecture's recurrent secret layer (h). The series of operations occurring in CEC-LSTMs at time phase j can be expressed succinctly by the equation below (7).

$$\begin{aligned}
 i_t &= \sigma(w_{xi}x_j + w_{hi}h_{j-1} + w_{ci}c_{j-1} + b_i) \\
 f_j &= \sigma(w_{xf}x_j + w_{hf}h_{j-1} + w_{cf}c_{j-1} + b_f) \\
 c_j &= f_j \odot c_{j-1} + i_j \odot \tan h(w_{xc}x_j + w_{hc}h_{j-1} + b_c) \\
 o_j &= \sigma(w_{xo}x_j + w_{co}h_{j-1} + w_{co}c_{j-1} + b_o) \\
 h_j &= o_j \odot \tanh(c_j) \\
 y_j &= w_{yh}h_j + b_y
 \end{aligned}
 \tag{8}$$

Where  $\odot$  is the scalar product of two vectors and  $\sigma()$  denotes the standard logistics sigmoid function defined as follows (9)

$$\sigma(x) = \frac{1}{1 + e^{-x}}
 \tag{9}$$

Here weight matrices denoted as  $w$  and bias vectors  $b$  are utilized to build connections between the input layer, output layer and memory block. In this CNN-CEC-LSTM, CNN consists of convolution layer and maxpooling layers only. The output of the maxpooling layer is fed to the subsequent LSTM layer.

$$y_j = CNN(x_i)$$

Here  $x_i$  is the initial input vector to the CNN network with the class label and  $y_j$  is the output of the CNN network to be fed to the next CEC-LSTM network  $x_i$  the feature vector formed from the max-pooling operation in CNN. It is fed to the LSTM to learn the long-range temporal dependencies.

**4. Experimental Results and Discussion**

**Indian Face Database:** In February 2002, the database was established on the campus of IIT Kanpur. There are forty separate photographs for each of the forty topics. Additional images are used for certain topics. Many of

the photographs have a vivid, homogeneous backdrop. The participants are in a frontal, erect role. JPEG format was used for the files. Each picture is 640x480 in size and has 256 grey levels per pixel. The photographs of men and women were put in two separate folders. Every topic has eleven separate images in both folders. The database contains variants depending on orientation and emotion. Face orientations include facing down, looking left, looking correct, looking up, looking up towards the left, looking up towards the right, and looking down. And the various feelings are neutral, smile, amusement, sad/disgust [15].

**Implementation Specifics:** Face detection is included in this section to aid in the introduction of the proposed CNN-CEC-LSTM classifier. The CNN-CEC-LSTM production is compared to established models such as SVM [16], CNN-LRC [17] using performance measures such as precision, recall, f-measure, and accuracy. If the face sample is positive and the classifier accepts it as positive, i.e., a correctly segmented positive sample, it is called a true positive (TP); if it is negative, it is considered a false negative (FN) (FN). Whether the sample is negative and segmented as negative, it is called real negative (TN); if it is positive and segmented as positive, it is considered false positive (FP) (FP).

**Accuracy:** It represents the proportion of correctly segmented positive samples to the total number of positive predicted samples, as shown in eq (9)

$$Precision = \frac{TP}{FP + TP} \quad (10)$$

**Recall:** The recall of a classifier represents the positive correctly segmented samples to the total number of positive samples, and it is estimated as follows in eq.(18)

$$Recall = \frac{TP}{TP + FN} \quad (11)$$

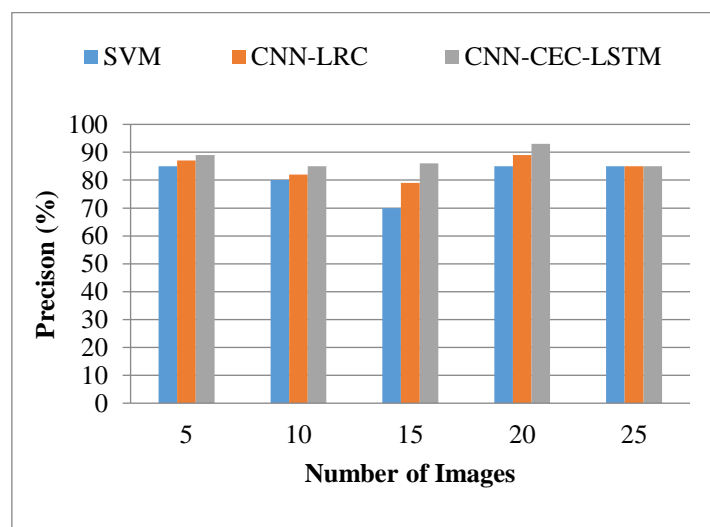
**F-measure:** this is also called  $F_1$ -score, and it represents the harmonic mean of precision and recall as follows in eq.(19)

$$F - measure = \frac{2 * (Recall * Precision)}{(Recall + Precision)} \quad (12)$$

**Accuracy:** It is one of the most commonly used measures for the classification performance, and it is defined as a ratio between the correctly segmented samples to the total number of samples as follows in eq.(20)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (13)$$

### Precision Rate comparison

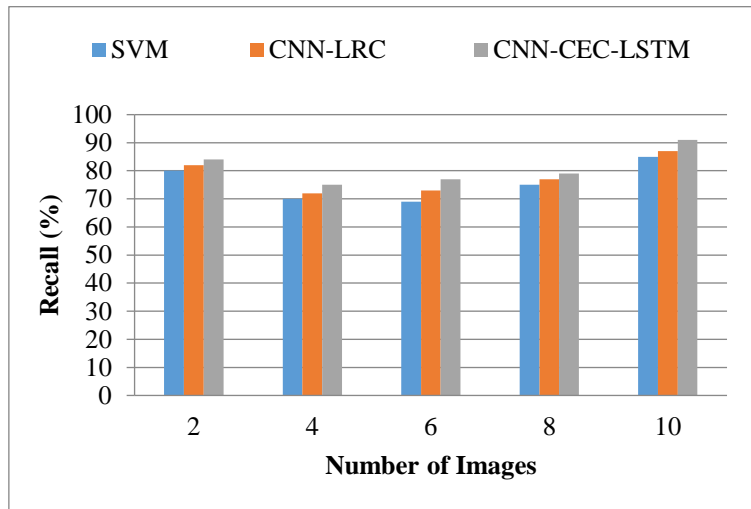




**Figure 5.** Result of Precision Rate

The graph in Fig.3 above illustrates the precision relation for the number of images in defined datasets. SVM, CNN-LRC, and CNN-CEC-LSTM are among the methods used. The number of images is then expanded in relation to the precision value. According to this graph, the proposed CNN-CEC-LSTM approach has higher precision than previous methods such as SVM and CNN-LRC, which yield better face recognition performance. The explanation for this is that the suggested approach uses CNN-based feature extraction, which improves CEC-LSTM detection results.

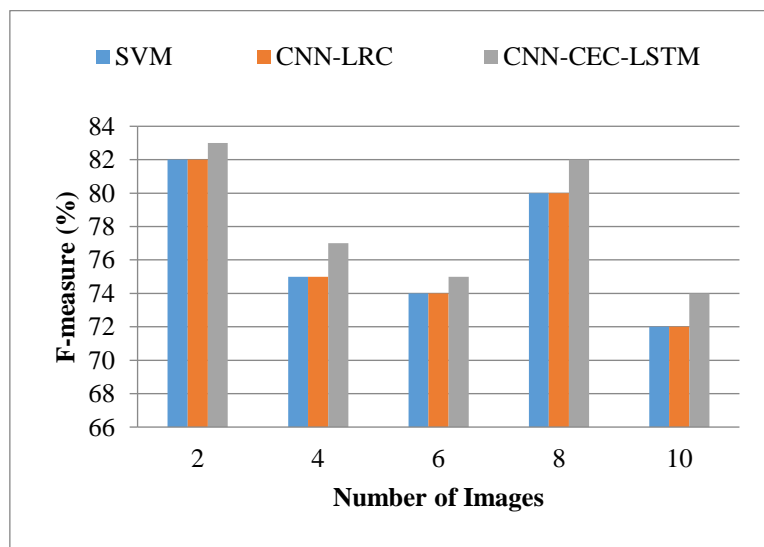
**Recall Rate comparison**



**Figure 6.** Result of Recall Rate

The graph in Fig.5 above illustrates the recall relation for the number of images in defined datasets. SVM, CNN-LRC, and CNN-CEC-LSTM are among the methods used. As the number of memories is raised, so is the associated recall value. This graph reveals that the proposed CNN-CEC-LSTM has a higher recall than previous approaches such as SVM and CNN-LRC. The explanation for this is that the CNN-CEC-LSTM can train the face pictures, improving recognition accuracy and reducing error.

**F-measure Rate comparison**



**Figure 7.** Result of F-measure Rate

The graph in Fig.6 above illustrates the f-measure relation for the number of images in defined datasets. SVM, CNN-LRC, and CNN-CEC-LSTM are among the methods used. As the number of pictures is raised, the f-

measure meaning rises in proportion. This graph shows that the proposed CNN-CEC-LSTM approach outperforms previous approaches such as SVM and CNN-LRC in terms of f-measure. As a consequence, the proposed CNN-CEC-LSTM algorithm outperforms the current algorithms in terms of segmentation performance. The explanation for this is that the CNN-CEC-LSTM filtering approach can boost the picture and reduce the noise value, allowing the CNN-CEC-LSTM method to achieve strong facial image recognition performance.

### Accuracy comparison

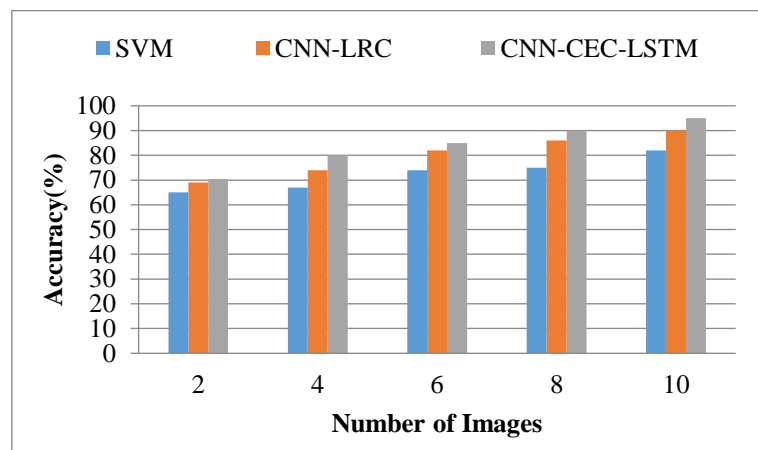


Figure 8. Result of Accuracy

The graph in Fig.7 above illustrates the processing time relation for the amount of images in listed datasets. SVM, CNN-LRC, and CNN-CEC-LSTM are among the methods used. The number of photographs is considered on the x-axis, and the precision value is considered on the y-axis. This graph shows that the proposed CNN-CEC-LSTM takes less time to process than previous approaches such as SVM and CNN-LRC. As a consequence, the performance explains that the proposed CNN-CEC-LSTM algorithm outperforms current algorithms in terms of improved segmentation performance with a strong accuracy score. The explanation for this is that the mechanism of gates and CEC units cooperating in the LSTM framework, along with tuned vectors and the design, has a promising capacity for collecting sequence knowledge by simulating complex interactions between features. As a consequence, the CNN-CEC-LSTM has greater versatility in simulating interactions between feature vectors and more facial recognition.

### 5. Conclusion and future work

Face detection is performed in this paper using the proposed CNN-CEC-LSTM. The final performances were obtained by varying the amount of training and test photographs. Following pre-processing, a Retinex-based adaptive filter is added to improve the face pictures. So far, convolutional neural networks have provided the strongest feature extraction outcomes. For face classification issues, this study suggested using an LSTM network and comparing its output to that of a normal MLP network. The CEC-LSTM network presented for face recognition will produce improved success in terms of accurate classification rates in all three suggested face classification tasks, indicating that it is an effective method in face recognition applications even with a limited training collection. The CNN-CEC-LSTM outperforms traditional schemes such as SVM and CNN-LRC in terms of recognition efficiency. The proposed solution has the benefit of achieving high face detection rates and real-time performance due to the lack of exhaustive searching on the whole picture. The proposed method can be expanded by integrating classifiers such as deep learning with different optimization schemes such as the Genetic Algorithm, NeuroGenetic Algorithm, and Ant Colony Algorithm, among others.

### References

1. U. Park, Y. Tong, and A. K. Jain, "Age-invariant face recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 5, pp. 947–954, 2010.
2. Z. Li, U. Park, and A. K. Jain, "A discriminative model for age invariant face recognition," *IEEE transactions on information forensics and security*, vol. 6, no. 3, pp. 1028–1037, 2011.
3. C. Ding and D. Tao, "A comprehensive survey on pose-invariant face recognition," *ACM Transactions on intelligent systems and technology (TIST)*, vol. 7, no. 3, p. 37, 2016.

4. Khalajzadeh, Hurieh, Mohammad Mansouri, and Mohammad Teshnehlab. "Face recognition using convolutional neural network and simple logistic classifier." In *Soft Computing in Industrial Applications*, pp. 197-207. Springer, Cham, 2014.
5. Ho, Huy Tho, and Rama Chellappa. "Pose-invariant face recognition using markov random fields." *IEEE transactions on image processing* 22, no. 4 (2012): 1573-1584.
6. Mohammed, Abdul Adeel, Rashid Minhas, QM Jonathan Wu, and Maher A. Sid-Ahmed. "Human face recognition based on multidimensional PCA and extreme learning machine." *Pattern Recognition* 44, no. 10-11 (2011): 2588-2597.
7. X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE transactions on image processing*, vol. 19, no. 6, pp. 1635–1650, 2010
8. Lin, Wen-Hui, Ping Wang, and Chen-Fang Tsai. "Face recognition using support vector model classifier for user authentication." *Electronic Commerce Research and Applications* 18 (2016): 71-82.
9. Fan, Chunian, Shuiping Wang, and Hao Zhang. "Efficient Gabor phase based illumination invariant for face recognition." *Advances in Multimedia* 2017 (2017).
10. Choi, Sang-Il, Sung-Sin Lee, Sang Tae Choi, and Won-Yong Shin. "Face recognition using composite features based on discriminant analysis." *IEEE Access* 6 (2018): 13663-13670.
11. S.-I. Choi, C.-H. Choi, G.-M. Jeong, and N. Kwak, "Pixel selection based on discriminant features with application to face recognition," *Pattern Recognit. Lett.*, vol. 33, no. 9, pp. 1083–1092, 2012.
12. Chowdhury, A.R., Lin, T.-Y., Maji, S., Learned-Miller, E., 2015. Face identification with bilinear cnns. arXiv preprint arXiv:1506.01342.
13. Fan, H., Cao, Z., Jiang, Y., Yin, Q., Doudou, C., 2014. Learning deep face representation. arXiv preprint arXiv:1403.2802.
14. Ling, Hefei, Jiyang Wu, Lei Wu, Junrui Huang, Jiazhong Chen, and Ping Li. "Self residual attention network for deep face recognition." *IEEE Access* 7 (2019): 55159-55168.
15. Sharma, Reecha, and M. S. Patterh. "Indian Face Age Database: A database for face recognition with age variation." *International Journal of Computer Applications* 126, no. 5 (2015).
16. Lin, Wen-Hui, Ping Wang, and Chen-Fang Tsai. "Face recognition using support vector model classifier for user authentication." *Electronic Commerce Research and Applications* 18 (2016): 71-82.
17. Khalajzadeh, H., Mansouri, M. and Teshnehlab, M., 2014. Face recognition using convolutional neural network and simple logistic classifier. In *Soft Computing in Industrial Applications* (pp. 197-207). Springer, Cham.