

Article

Novel BSSO-Based Deep Convolutional Neural Network for Face Recognition with Multiple Disturbing Environments

Neha Soni ^{1,*}, Enakshi Khular Sharma ¹ and Amita Kapoor ²

¹ Department of Electronic Science, University of Delhi South Campus, Delhi 110021, India; enakshi54@yahoo.co.in

² Shaheed Rajguru College of Applied Sciences for Women, University of Delhi, Delhi 110091, India; amita.kapoor@rajguru.du.ac.in

* Correspondence: nsoni@south.du.ac.in or soni.neha2191@gmail.com

Abstract: Face recognition technology is presenting exciting opportunities, but its performance gets degraded because of several factors, like pose variation, partial occlusion, expression, illumination, biased data, etc. This paper proposes a novel bird search-based shuffled shepherd optimization algorithm (BSSO), a meta-heuristic technique motivated by the intuition of animals and the social behavior of birds, for improving the performance of face recognition. The main intention behind the research is to establish an optimization-driven deep learning approach for recognizing face images with multiple disturbing environments. The developed model undergoes three main steps, namely, (a) Noise Removal, (b) Feature Extraction, and (c) Recognition. For the removal of noise, a type II fuzzy system and cuckoo search optimization algorithm (T2FCS) is used. The feature extraction is carried out using the CNN, and landmark enabled 3D morphable model (L3DMM) is utilized to efficiently fit a 3D face from a single uncontrolled image. The obtained features are subjected to Deep CNN for face recognition, wherein the training is performed using novel BSSO. The experimental findings on standard datasets (LFW, UMB-DB, Extended Yale B database) prove the ability of the proposed model over the existing face recognition approaches.

Keywords: face recognition; occlusion; bird swarm algorithm; convolution neural network; shuffled shepherd optimization



Citation: Soni, N.; Sharma, E.K.; Kapoor, A. Novel BSSO-Based Deep Convolutional Neural Network for Face Recognition with Multiple Disturbing Environments. *Electronics* **2021**, *10*, 626. <https://doi.org/10.3390/electronics10050626>

Academic Editor: Tarek Gaber

Received: 30 January 2021

Accepted: 26 February 2021

Published: 8 March 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Due to the evolution and advancement in biometric system technology, face recognition has become very popular in image and computer vision [1]. In the last few years, the analysis based on face recognition has grown rapidly where building access control, video surveillance, and autonomous vehicles are the few instances of the concrete applications that are gaining more attention in industries [2–4]. Several approaches, such as holistic, local, and hybrid are developed for providing face image description with only fewer face image features or the whole facial features [5]. Numerous researchers have shown a better recognition rate for face recognition [6–8]. However, during the biometric validation, the face is affected by the illumination with several intensities of the light and different angles [9]. This makes face recognition complex in real-time applications. When the effect of illumination is controlled, it is very easy to recognize the face [10]. In the video scenes, the regions of extracted face normally have low resolution (LR), and they are sensitive to illumination and pose variations; these flaws degrade subsequent recognition tasks.

In the last few years, face recognition has been showing promising performance in several applications with various challenging conditions, such as the variation in pose, occlusion, expression, and illumination. Several face recognition systems are introduced to recognize the face images in the controlled conditions; there are some studies focused on surveillance systems with the LR faces [6]. A major challenge that lies in these applications is that high resolution (HR) probe images are not available because of the huge

camera distance from the subject [11]. The performance of the face recognition system is significantly affected under an uncontrolled environment. The problem of misalignment heavily affects the face recognition system [12]. Previous works show that the presence of occlusions degrades the performance of the face recognition system; hence, the initial task for recognizing the face with the occlusion is to eliminate or suppress occlusions effectively [13].

Deep learning is utilized to provide accurate results for pre-processing and face recognition. In the deep learning techniques, the original data are considered as the input in convolve filters with various multiple levels for detecting high representation from unlabeled or labeled data [14,15]. The deep face recognition system aligns faces initially with affine transformations; after that, the aligned faces are subjected to Convolutional Neural Networks (CNN) for extracting identity-preserving features. As affine transformations remove in-plane pose variations but fail to eliminate intra-class appearance variations from the out-plane poses, this results in the face misalignment issue [12]. The deep learning algorithm is utilized for extracting shallow structures from facial images [14].

The main intention behind the research is to establish an optimization-driven deep learning approach for recognizing face images. In our work, initially, the input image undergoes a noise removal phase to eliminate noise to make them suitable for subsequent processing. The noise removal is performed using a type II fuzzy system and cuckoo search optimization algorithm (T2FCS), which helps to detect noisy pixels from the image for improved processing. After the noise removal phase, the feature extraction is carried out using the CNN model and landmark-enabled 3D morphable model (L3DMM). Here, the CNN is the multi-layered neural network (NN) for detecting the complex features in the data and the L3DMM feature is utilized to fit a 3D face from a single uncontrolled image. The obtained features are subjected to Deep CNN for face recognition, which is trained by a novel bird search-based shuffled shepherd optimization algorithm (BSSSO). Here, the developed BSSSO is designed by combining the shuffled shepherd optimization algorithm (SSOA) [16] and bird swarm algorithm (BSA) [17]. The major contributions of the paper are:

Proposed BSSSO-based Deep CNN for face recognition: The proposed BSSSO-based Deep CNN is the combination of the proposed BSSSO strategy to train the Deep CNN. Here, the integration of SSOA is done along with the BSA for an efficient face recognition system. Both BSA and SSOA are bio-inspired meta-heuristic algorithms designed to solve optimization algorithms. Both algorithms are based on simple concepts and can be used to solve a wide range of problems. In this paper, the developed BSSSO combines both these algorithms efficiently by modeling the BSA's bird intelligence (foraging) and SSOA's sheep instinct (to find the way). The novel algorithm is effective, efficient, and stable for face recognition.

The other sections of research are as follows: Section 2 describes the conventional face recognition techniques used in literature and faced challenges. The developed model for face recognition is shown in Section 3. The results of the developed model with other schemes are portrayed in Section 4, and Section 5 concludes the paper.

2. Related Work

The existing face recognition techniques along with their limitations are explained in this section, which motivates researchers to develop a new method to perform face recognition.

2.1. Literature Survey

The eight classical strategies based on face recognition along with its limitations are deliberated below: Li, X.X. et al. [13] developed a joint probabilistic generative model for improving the quality of error images in the Image Gradient Orientations-embedded Structural Error Coding (IGO-SEC). Further, the novel reconstruction approach was introduced by incorporating the advantages of the pixel as well as IGO features. The method kept more stable recognition performances but failed to consider error coding models into the deep learning method. He, M. et al. [12] presented Deformable Face Net (DFN) for handling

pose variations for recognizing the face. The deformable convolution module was utilized for learning identity enabled preserving feature extraction and face-recognition-oriented alignment. In addition, Displacement Consistency Loss (DCL) was introduced as the regularization term for enforcing learned displacement fields to align faces to be consistent both in amplitude and orientation. Here, the performance was found better, but the risk of over-fitting was increased. Yang, W. et al. [18] developed the Local Multiple Patterns (LMP) feature descriptor for feature extraction and Weber's law for face recognition. In the LMP, Weber's ratio was quantized into various intervals for generating the multiple feature maps to describe the changes, and then LMP concatenated non-overlapping regions of feature maps to represent the image. After that, the LMP captured the small-scale structures for generating discriminative visual features. The method was complex to fix the pre-defined threshold in the LMP. Zangeneh, E. et al. [11] employed the coupled mappings approach for face recognition based on Deep CNN with low resolution. Here, the distance among features of the corresponding low- and high-resolution images was back-propagated for training the networks. The method can lead to over-fitting issues due to general features that can sparsely change.

Gao, G. et al. [19] presented multilayer locality-constrained structural orthogonal Procrustes regression (MLCSOPR) for recognizing the face. This method learns pose-robust discriminative representation features for reducing the resolution gap among the LR and the HR image space, but still strengthens consistency among the HR and LR image space. However, the method failed to address multiple pose variations. Chen, Z. et al. [20] developed Slack Block-Diagonal (SBD) structure, in which the target structure matrix was updated dynamically, but their block diagonal nature was preserved. Further, the dictionary learning method was introduced using the mixed-noise model for depicting noise in the face images. The method failed to capture global structures of data. Vishwakarma, V.P. and Dalal, S [10] presented a face recognition approach using adaptive illumination normalization. In addition, the illumination normalization was carried out over Discrete Cosine Transform (DCT) coefficients that were adaptively estimated based on alteration significance of coefficients. The method failed to include various color face images for color image illumination normalization. Iran Manesh, S.M. et al. [21] developed coupled Generative Adversarial Network (CpGAN) for solving matching problems against the gallery of the visible faces. This architecture contains two sub-networks where one is given to the visible spectrum, whereas the other belongs to the non-visible spectrum. The method prevents information flow in the network due to a combination of the identity function.

2.2. Challenges

The challenges of existing face recognition techniques are illustrated below:

A desirable image feature should be both discriminative and robust to different variances, e.g., image noise and illumination changes. Variation in the illumination and Gaussian white noise causes serious problems [13].

Unconstrained face recognition remains a challenging task due to various factors such as pose, expression, illumination, partial occlusion, etc. In particular, the most significant appearance variations are stemmed from poses that lead to severe performance degeneration [12].

While many face recognition systems have been developed for recognizing high-quality face images in controlled conditions, there are a few studies focused on face recognition in real-world applications such as surveillance systems with low-resolution faces. One important challenge in these applications is that high-resolution (HR) probe images may not be available due to the large distance of the camera from the subject. Thus, the performance of traditional face recognition systems that are developed for high-quality images degrades considerably with low-resolution (LR) face regions [19].

During biometric checks, a face can be affected by illumination with different intensities of light from different angles. Face recognition is difficult due to varying lighting conditions [10].

Proposed here is the bird search-based shuffled shepherd optimization-based Deep Convolutional Network for face recognition

This section elaborates on the proposed BSSSO for face recognition. The schematic view of the developed model is illustrated in Figure 1.

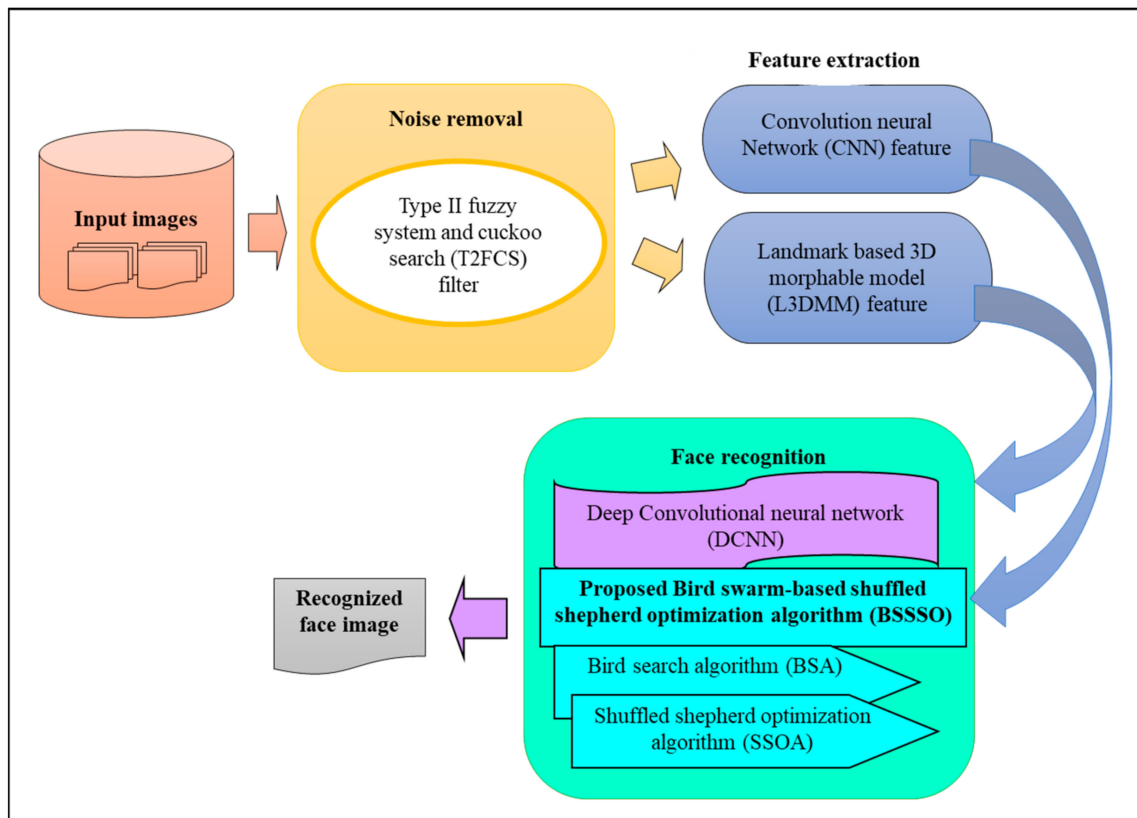


Figure 1. Schematic view of the proposed methodology of face recognition

Initially, the input image undergoes a noise removal phase to eliminate noise to make them suitable for subsequent processing. The noise removal is performed using T2FCS [22], which helps to detect noisy pixels from the image for improved processing. After the noise removal phase, the feature extraction is carried out using the convolution neural network (CNN) model and L3DMM [23]. Here, the CNN is a multi-layered neural network with a special architecture for detecting the complex features in the data and the L3DMM feature is utilized to fit a 3D face from the single uncontrolled image. The obtained features are subjected to Deep CNN for face recognition. The training of the deep neural network is performed using the bird search-based shuffled shepherd optimization algorithm (BSSSO). Here, the proposed BSSSO is designed by combining the shuffled shepherd optimization algorithm (SSOA) [16] and bird swarm algorithm (BSA) [17] for training Deep CNN.

Let us consider the dataset G with b number of images, which is expressed as,

$$G = \{K_b\}; (1 \leq v \leq b) \quad (1)$$

where, the term b represents the total images and K_b indicate the b^{th} image. Each image K_b is forwarded to the noise removal module, which carries noise removal in the input images. The size of database is expressed as $[M \times N]$.

3. Proposed Bird Search-Based Shuffled Shepherd Optimization-Based Deep Convolutional Network for Face Recognition

3.1. Noise Removal Using Type II Fuzzy System and Cuckoo Search Filter

The initial step involved in face recognition is noise removal. The main aim of noise removal is to remove the noise for improving the quality of the image and to recognize the face accurately. Here, the input image K_b is passed to the noise removal where noise removal is carried out based on T2FCS filter [22]. Here, the noisy pixels in the image are detected using the circular-based searching scheme and the detected corrupted pixels are removed using the cuckoo search algorithm.

The noise removal process is utilized for modifying input image by changing a visual impact. The enhancement image provides detailed information about boundaries, edges with increased dynamic range. In this work, the enhancement procedure is carried based on noisy pixels. Once the noisy pixels position K is identified, the identified pixels are applied for the enhancement process. After that, a novel image matrix is produced based on nearest neighbor for the noisy pixel, expressed by,

$$J_{n1}(x, y) = \frac{1}{9} \sum_{u=1}^1 \sum_{v=1}^1 J_n(x+u, y+v) \quad (2)$$

In order to solve the uncertainty issues, the generated new pixel values are subjected to the neuro-fuzzy system.

$$J_{n2}(x, y) = N_{fuzzy}(J_n(x, y)) \quad (3)$$

where, the term N_{fuzzy} signifies the neuro-fuzzy system. Once the T2FCS set model is generated, the new pixel $J_n(x, y)$ is considered as the input and the new image is generated using the noise pixel-based location. At last, the restored image is obtained from the below equation.

$$J_G(x, y) = \left[\frac{J_{n1}(x, y) + J_{n2}(x, y)}{2} \right] \quad (4)$$

The structure of T2FCS [22] is composed of consequence and premise parts. The T2FCS is trained for determining the parameters related to the parts employed in the optimization approach. Here, the T2FCS uses output and input data pairs in the training. The layers present in T2FCS are the fuzzification, rule, normalization, defuzzification, and summation layers. Here, the fuzzification layer employs the membership functions for obtaining fuzzy clusters from the input values where the firing strengths are produced based on the membership values evaluated in the fuzzification, whereas the normalization layer is utilized to compute the firing strengths corresponding to every rule. The normalized value refers to the ratio of firing strength to i^{th} rule to the total number of firing strengths. The defuzzification layer computes the weighted values for every node, while the summation layer sums the results obtained from the defuzzification layer. Hence, the output obtained from noise removal output is denoted as T .

3.2. Feature Extraction Based on CNN Features and L3DMM Feature

After noise removal, the feature extraction is required for extracting the appropriate features using CNN features and L3DMM feature. The segmentation image T is fed to feature extraction phase to perform feature extraction effectively. The feature extraction step performed in this research is explained below.

(i) CNN features

CNN is a multi-layered network, which is utilized for detecting the complex feature from the pre-processed image. The CNN architecture comprises convolutional (Conv)

layers, pooling (POOL) layers, and the Fully Connected (FC) layers, where each performs a specific task. The initial layer is convolution layer that is utilized to extract essential features from pre-processed images. It provides the relationship between the pixel values and the image features. It takes the pre-processed result T as input and the CNN features get extracted using the convolutional layer. The features that are extracted from the initial convolutional layer are known as the CNN features and are denoted as f_1 with dimension $[1 \times 256]$.

The features obtained from CNN are denoted as E_1 . In this work, VGG19 is used for the extraction of features. VGG19 architecture contains 16 CNN layers, 3 fully connected layers and a final layer for softmax function.

(ii) L3DMM features

The 3D lighting environment is computed from the person's face based on the following steps: the initial step is to achieve a 3D face shape. For solving limitations of 3D face fitting, the L3DMM [23] is employed to fit 3D face efficiently from the single uncontrolled image. This approach is initially developed by Blanz and Thomas for estimating the 3D face model from a single image based on the analysis-by-synthesis method. The 3DMM is designed through principal component analysis of the textures and face shapes. Then, multiple image features are employed for improving the convergence property of the 3DMM fitting method with a better precision level. According to 3DMM, the shape of the expressional face is given by,

$$B = \bar{B} + J_{ef}\beta_{ef} + J_{exp}\beta_{exp} \quad (5)$$

where, the term B refer to average shape vector matrix, the terms J_{ef} and J_{exp} consists of principle components of the shape variations corresponding to expression and identity, and the terms β_{ef} and β_{exp} refer to weight coefficients. The main aim is to morph 3DMM for reducing distance among projected 3D facial landmarks, as well as its 2D ones. Hence, objective function is expressed by,

$$S(\phi) = \|m_{t-land} - m_{land}(\phi)\| \quad (6)$$

where, m_{t-land} land is the position of ground truth 2D facial landmarks detected from the image based on explicit shape regression approach. $m_{land}(\phi)$ is the projected position of the morphable model's 3D facial landmarks, which is obtained by,

$$m_{land}(\phi) = LM(B_{land} + g) \quad (7)$$

where, the terms M and g represents the rotation matrix, and translation vector of 3D facial landmarks is B_{land} . The features obtained from L3DMM are denoted as E_2 . Finally, the feature vector obtained from the feature extraction step is expressed as,

$$E = \{E_i; 1 < i < m\} \quad (8)$$

3.3. Face Recognition Using the Proposed Bird Search-Based Shuffled Shepherd Optimization-Based Deep Convolutional Neural Network (DCNN)

In this section, the face recognition is presented based on the developed BSSSO-based Deep CNN method, and detection is carried out based on the feature vector. The feature extracted output is given to the classification module based on the Deep CNN [3] and training is carried out based on the training algorithm, termed BSSSO, which is the integration of SSOA [16] with the BSA [17]. BSSSO aims to recognize a face from input images based on extracted features.

SSOA: Animals use their intuition to find the best way for living and human beings learn animal's instincts for their purposes. Horses have an instinct to find the best way, and shepherds use one or more horses in their herd to find the way. Shepherds guide sheep behind the horses to pasture and bring them back. The SSOA [16] is a simple meta-

heuristic technique that is motivated by utilizing an intuition of animals by a shepherd. Here, the first step is to separate the agents into different communities, and optimization is progressed by shepherd behavior operating in each community. Thus, the proposed algorithm enhances the convergence speed, offering the optimal solution.

BSA: BSA [17] is the bio-inspired optimization that is based on the social interactions and the social behaviors of birds. It consists of three behaviors, namely, flight behavior, vigilance behavior, and foraging behavior. The birds try to forage the food using social interactions and escape it from predators. The birds can switch between foraging and vigilance behavior such that the birds foraging and the vigilance behavior of birds are modeled as the stochastic decision. During foraging, each records and updates the previous best experience regarding the food patch. However, social information can be instantaneously shared among the whole swarm. The birds seek to shift to the center of the swarm using vigilance behavior. Birds switch between scrounging and producing while flying to another site. The mathematical modeling of this behavior of birds is BSA.

The integration of SSOA and BSA is done to enhance the overall system performance of the algorithm. The structure of Deep CNN and algorithmic procedure of the developed model are given below.

3.3.1. Architecture of Deep CNN

The architecture of the Deep CNN is deliberated in Figure 2 and the architecture of DCNN comprises three layers: pooling (POOL), convolutional (conv), and the Full Connected (FC) layers. Among the three layers of DCNN, each constitutes a specific function. The main function of the conv layers is to generate feature maps from segments of pre-processed images and these feature maps are sub-sampled down in pool layers. The third layer is the FC layer, where the classification is progressed. The convolutional layer engages in mapping the input such that the input maps undergo convolution with the convolutional kernels in order to develop the output map. The size of the output map is similar to the kernel number and the kernel matrix dimension is $[3 \times 3]$. Hence, this makes it clear that the conv layers are the multilayer input maps loop, output maps, and kernel weights. In the first conv layer, there are several inputs and outputs, whose size reduces in the successive conv layers such that classification accuracy of the objects depends on the layers in the Deep CNN.

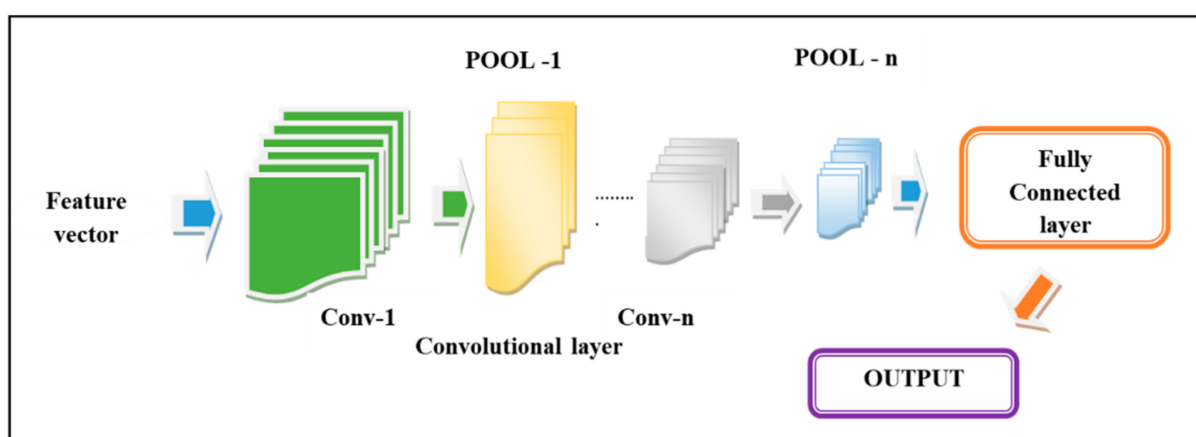


Figure 2. Structure of Deep Convolutional Neural Network (DCNN) for the construction of prediction map.

Conv layers: This layer extracts the features and provides pattern extraction using the feature vector obtained by input images. The neurons that are linked with the trainable weights and those trained weights get convolved with inputs to build feature maps. After that, the outcome is subjected to the non-linear activation function to ease the functional

mappings between input and the response variables. Let us assume input to the Deep CNN is G and, therefore, the output from the conv layer is given by,

$$(M_c^b)_{i,j} = (B_c^b)_{i,j} + \sum_{d=1}^{\omega_1^{d-1}} \sum_{k=-\omega_1^l}^{\omega_1^l} \sum_{n=-\omega_1^l}^{\omega_2^l} (\vartheta_{c,d}^b)_{k,n} * (J)_{i+k,j+n} \quad (9)$$

where, the symbol $*$ indicates the convolutional operator, and $(B_c^b)_{i,j}$ signifies fixed feature maps. The output from $(b-1)^{th}$ layer forms the input to l^{th} conv layer. The conv layer weight is denoted as, $\vartheta_{c,d}^b$, which is b^{th} conv layer weights, and b^{th} conv layer bias is expressed by B_c^b . Assume d, k , and n denote the feature maps notations.

ReLU layer: ReLU ensures simplicity and effectiveness, and works faster when dealing with huge networks. The ReLU layer output when subjected with the feature maps is given by,

$$G_c^b = Afn(M_c^{b-1}) \quad (10)$$

where, $Afn()$ refers to activation function in b^{th} layer.

POOL layers: It is a non-parametric layer without bias and has weights that follow fixed operation. The importance of the POOL layer is to mitigate the spatial dimensions of the input and minimize the computational complexity.

FC layers: The generated patterns based on conv layers and pooling are given to fully connected layers. The fully connected layer output is expressed as,

$$H_c^b = \delta(G_c^{b-1}) \text{ with } M_c^b = \sum_{d=1}^{\omega_1^{d-1}} \sum_{k=-\omega_1^l}^{\omega_1^l} \sum_{n=-\omega_1^l}^{\omega_2^l} (\vartheta_{c,d}^l)_{k,n} * (J)_{i+k,j+n} \quad (11)$$

where, $(\vartheta_{c,d}^l)_{k,n}$ denotes the weight.

3.3.2. Training of DCNN Based on BSSSO Algorithm

This section illustrates the developed BSSSO algorithm for face recognition. Here, the proposed BSSSO is developed by integrating the SSOA [16] and BSA [17]. The SSOA [16] is devised for introducing a new multi-community that is motivated by the shepherd [16]. Here, the first step is to separate the agents into different communities, and optimization is motivated by the shepherd's behavior acting on each neighborhood. This method is utilized to generate improved features for solving multi-objective optimization issues. The SSOA is simpler because it works in a single phase.

Here, the updated equation of the BSA algorithm is modified based on the SSOA optimization updated equation. The modification makes the solution update to be effective, and it further enhances the convergence of the optimization algorithm. The procedure of the developed model is illustrated below:

- (i) Initialization of population: The initial step is the initialization of shepherd population, which is expressed as,

$$Y = \{Y_1, Y_2, \dots, Y_l, \dots, Y_j\} \quad (12)$$

where, Y_l refer to the position of l^{th} shepherd and j represents the total shepherds.

- (ii) Compute fitness function: The optimal solution is determined using fitness, which is known as minimization problem and hence solution generating the least Mean Square Error (MSE) is chosen as an optimal solution. Thus, the MSE is computed as,

$$MS_{err} = \frac{1}{J} \sum_{j=1}^J [\zeta_j - M_c^b]^2 \quad (13)$$

where, ζ_j signifies expected output and M_c^b represented predicted output, J indicates count of input data, where $1 < j < J$.

- (iii) Generate new solutions: The position Y_i^{t+1} are updated using previous position. According to SSOA [16], the updated equation is given as,

$$Y_g^{s+1} = \alpha \times rand \times (Y_e^s - Y_g^s) + \beta \times rand \times (Y_e^s - Y_g^s) \quad (14)$$

where, the term Y_e^s refer to solution vectors of the selected horse, the term Y_g^{s+1} signifies solution vector of the shepherd in the m-directional search space, $rand$ represents the random number, and α, β denotes the step size which is a random number.

$$Y_g^{s+1} = (\alpha + \beta)rand \times Y_e^s - (\alpha + \beta)rand \times Y_g^s \quad (15)$$

According to BSA [17], the updated equation of foraging behavior is represented as,

$$Y_g^{s+1} = Y_g^s + (G - Y_g^s) \times B \times rand(0, 1) + (K - Y_g^s) \times M \times rand(0, 1) \quad (16)$$

where $rand(0, 1)$ denotes the independent uniformly distributed numbers in $(0, 1)$. B and M are two positive numbers, also called cognitive and social accelerated co-efficient.

Rearrange Equation (16),

$$Y_g^{s+1} = Y_g^s [1 - B \times rand(0, 1) - M \times rand(0, 1)] + G \times B \times rand(0, 1) + (K \times M \times rand(0, 1)) \quad (17)$$

$$Y_g^s = \frac{Y_g^{s+1} - G \times B \times rand(0, 1) - K \times M \times rand(0, 1)}{1 - B \times rand(0, 1) - M \times rand(0, 1)} \quad (18)$$

Substitute Equation (18) into Equation (15),

$$Y_g^{s+1} = (\alpha + \beta)rand \circ Y_g^s - (\alpha + \beta)rand \circ \left(\frac{Y_g^{s+1} - G \times B \times rand(0, 1) - K \times M \times rand(0, 1)}{1 - B \times rand(0, 1) - M \times rand(0, 1)} \right) \quad (19)$$

$$Y_g^{s+1} + \frac{(\alpha + \beta)rand \circ Y_g^{s+1}}{1 - B \times rand(0, 1) - M \times rand(0, 1)} = (\alpha + \beta)rand \circ Y_g^s + (\alpha + \beta)rand \circ \left(\frac{G \times B \times rand(0, 1) - K \times M \times rand(0, 1)}{1 - B \times rand(0, 1) - M \times rand(0, 1)} \right) \quad (20)$$

The final updated equation of the proposed BSSSO is given by,

$$Y_g^{s+1} = \left[\frac{1 - B \times rand(0, 1) - M \times rand(0, 1)}{1 - B \times rand(0, 1) - M \times rand(0, 1) + (\alpha + \beta)rand} \right] \left[(\alpha + \beta)rand \circ Y_g^s + (\alpha + \beta)rand \circ \left(\frac{G \times B \times rand(0, 1) - K \times M \times rand(0, 1)}{1 - B \times rand(0, 1) - M \times rand(0, 1)} \right) \right] \quad (21)$$

- (iv) Ranking the shepherds: The shepherds are ranked using the values of fitness, with maximal fitness chosen to obtain optimal solution.
- (v) Termination: Steps (ii) to (iv) are iterated till maximal iterations. Figure 3 shows the pseudo-code of the proposed BSSSO:

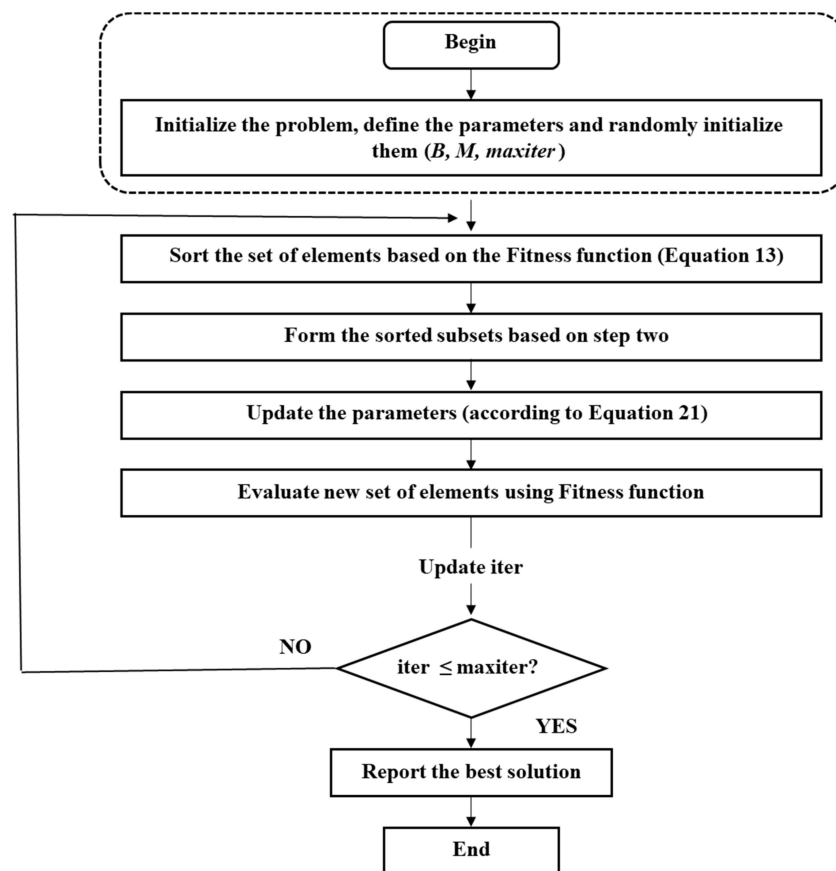


Figure 3. Pseudo code of developed bird search-based shuffled shepherd optimization algorithm (BSSSO).

4. Results and Discussion

The results of the proposed BSSSO-based Deep CNN for face recognition are elaborated in this section.

4.1. Experimental Setup

The implementation is carried out using PYTHON with Keras tool using Labelled Faces in the Wild Home (LFW) dataset [24], The University of the Milano Bicocca 3D Face Dataset (UMB-DB) database [25], and Extended Yale B database [26].

LFW database: It contains labeled faces for face verification with the original and the aligned images.

UMB-DB database: It consists of 1473 pairs of color and depth images with 143 subjects. Every subject is collected from various facial expressions and with partially occluded objects, like hats, hands, and scarves. It contains a total of 578 occluded acquisitions.

Extended Yale B database: It consists of 16,128 images of the human subjects with 64 illumination statuses and 9 poses.

4.2. Evaluation Metrics

The performance is computed by considering metrics, like accuracy, false acceptance rate (FAR), and false rejection rate (FRR).

Accuracy: It measures the closeness of exact or true values and is computed as,

$$A_y = \frac{D_p + D_n}{D_p + D_n + F_p + F_n} \quad (22)$$

where D_p denotes true positive, D_n refers to true negative, F_p signifies false positive, and F_n specifies false negative.

False Acceptance rate (FAR): It measures the likelihood that may incorrectly accept the access attempt by the unauthorized one.

$$FAR = \frac{F_a}{T_a} \quad (23)$$

where F_a represents the false acceptance and T_a represents the total acceptance.

False rejection rate (FRR): It measures the likelihood that incorrectly rejects the access attempt by unauthorized one.

$$FRR = \frac{F_r}{T_r} \quad (24)$$

where, F_r represents the false rejections and T_r represents the total rejections.

4.3. Experimental Results

Figure 4 represents the image results of the LFW database. Figure 4a–c represent image results of a person acquired using the LFW database.

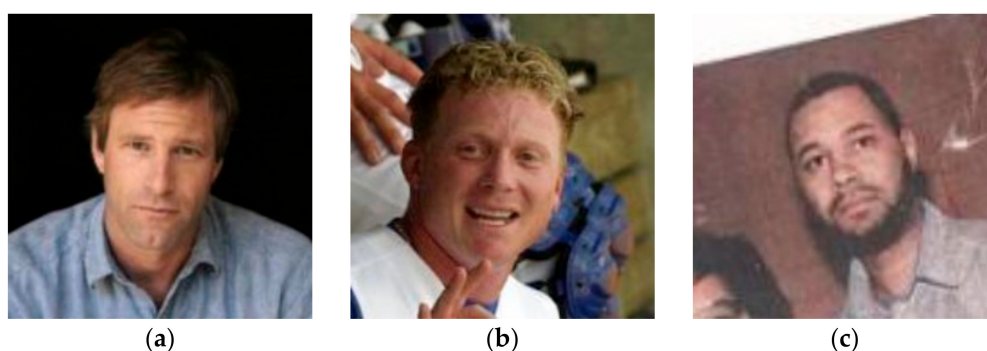


Figure 4. Image results using Labelled Faces in the Wild Home (LFW) database, (a) image-1, (b) image-2, (c) image-3.

Figure 5 depicts the image results of UMB-DB database. Figure 5a–c refer to expression image-1, 2, and 3, respectively. Figure 5d, 5e, and 5f portray occlusion image-1, 2, and 3, respectively.

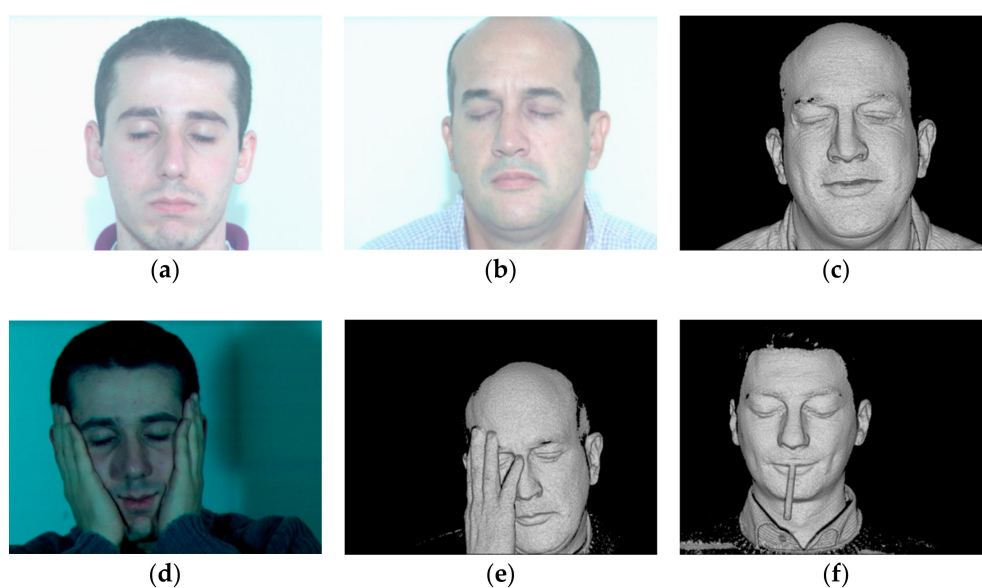


Figure 5. Image results based on The University of the Milano Bicocca 3D Face Dataset (UMB-DB) database, (a) expression image-1, (b) expression image-2, (c) expression image-3, (d) occlusion image-1, (e) occlusion image-2, (f) occlusion image-3.

Figure 6 portrays the image results of the Extended Yale B database. Figure 6a,b represents illumination image-1 and image-2, respectively. Figure 6c,d represent pose image-1 and image-2, respectively.

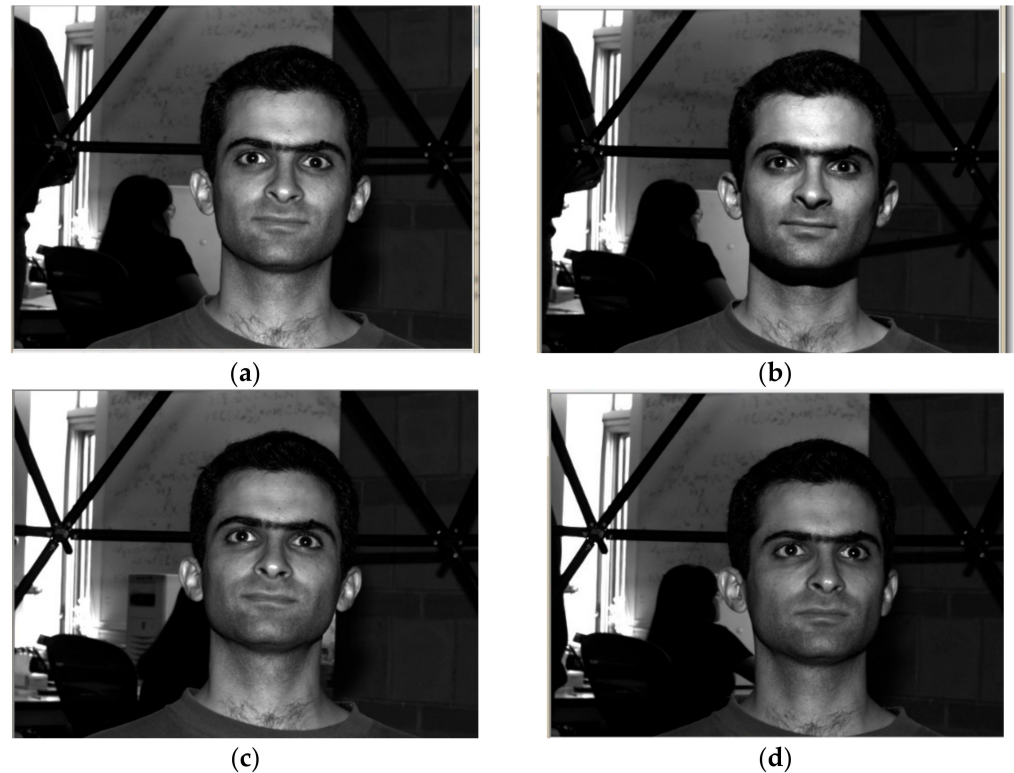


Figure 6. Image results using Extended Yale B database, (a) illumination image-1, (b) illumination image-2, (c) pos image-1, (d) pos image-2.

4.3.1. Comparative Methods

The methods utilized for the comparison are Deformable Face Net (DFN) [12], multi-scale block local multiple patterns (MB-LMP) [18], bird swarm algorithm based Deep CNN (DCNN+BSA), and the proposed BSSSO-based Deep CNN.

4.3.2. Comparative Analysis

This section describes a comparative analysis of the developed approach using three different datasets in terms of the metrics accuracy, FAR, and FRR.

Analysis Using LFW Database

The proposed BSSSO-based-DCNN is tested on the LFW dataset [24] under the different amounts of training data and varying noise. Figure 7 illustrates the performance of the proposed BSSSO-based-DCNN and other traditional methods, namely, DFN, MB-LMP, and DCNN+BSA in terms of accuracy, FAR, and FRR under the different percentages of training data. It can be observed that the proposed BSSSO-based DCNN outperforms other traditional methods in all settings for training data ranging from 50% to 90%. It shows the maximum accuracy, minimum FAR and FRR. Considering a particular case when training data is 90%, the false accepts was reduced to 50% i.e., 0.2190 when compared with other methods (Figure 7b). The statistical data presented in Figure 7a–c indicate that our proposed BSSSO-based DCNN has achieved state-of-the-art results on the LFW dataset under different amounts of training data.

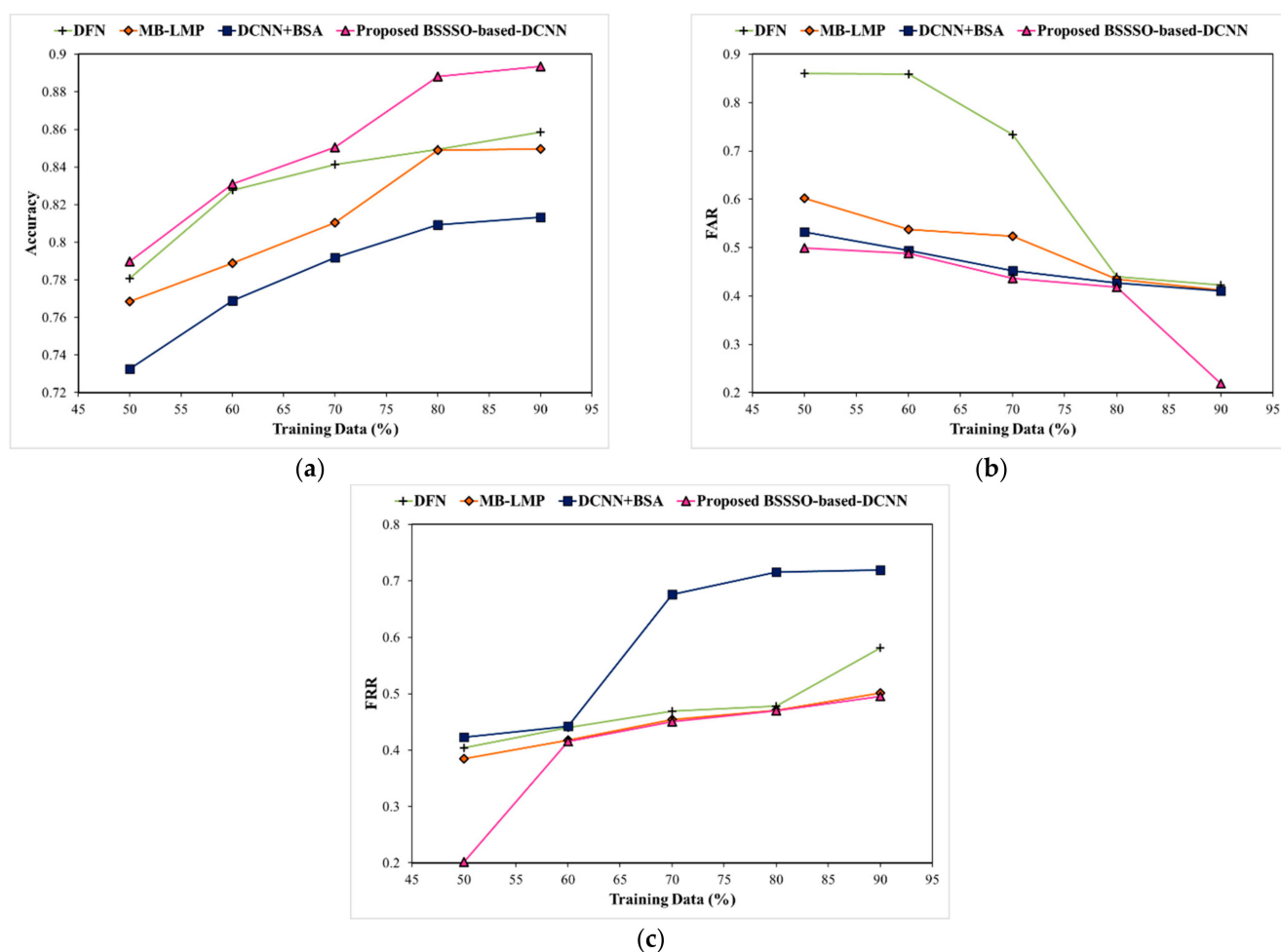


Figure 7. Analysis based on LFW database based on training data.

Figure 8 shows the performance of the methods using the LFW database by varying the noise. Figure 8a–c depict the accuracy, FAR, and FRR, respectively, with respect to the noise variance ranging from 50% to 90%. Compared to all algorithms, the proposed BSSSO-based Deep CNN achieved higher accuracy and dropped false accepts and false rejects. When tested for 90% noisy face images, the proposed algorithm shows an improved +1.09 accuracy in terms of relative performance when compared with DFN.

Analysis Using UMB-DB Database

Next, the UMB-DB database is used to evaluate the performance of the proposed BSSSO-based Deep CNN under expression and occlusion variations. Figures 9 and 10 depict an analysis of the proposed method using the UMB-DB database by varying the expressions and occlusions, respectively, in terms of accuracy, FAR, and FRR. A significant improvement in accuracy is depicted. The expressions considered are normal, sad, smile, angry, and hate; a significant drop in the false accepts and false rejects is also achieved by the proposed BSSSO-based Deep CNN. For a particular case, when the expression is sad, the false rejects dropped by 0.80 for the proposed BSSSO-based Deep CNN in comparison to BSA + DCNN. Figure 10 portrays the improvement of the proposed method using the UMB-DB database by varying the occlusion. Figure 10a–c portray the analysis of accuracy, FRR, and FAR, respectively, with respect to different occlusion like finger, cup, hand (in front of the face image), mask, hat (on the face). Better accuracy, reduced FAR, and reduced FRR demonstrate that the novel BSSSO-based Deep CNN is robust to expression and occlusion variations.

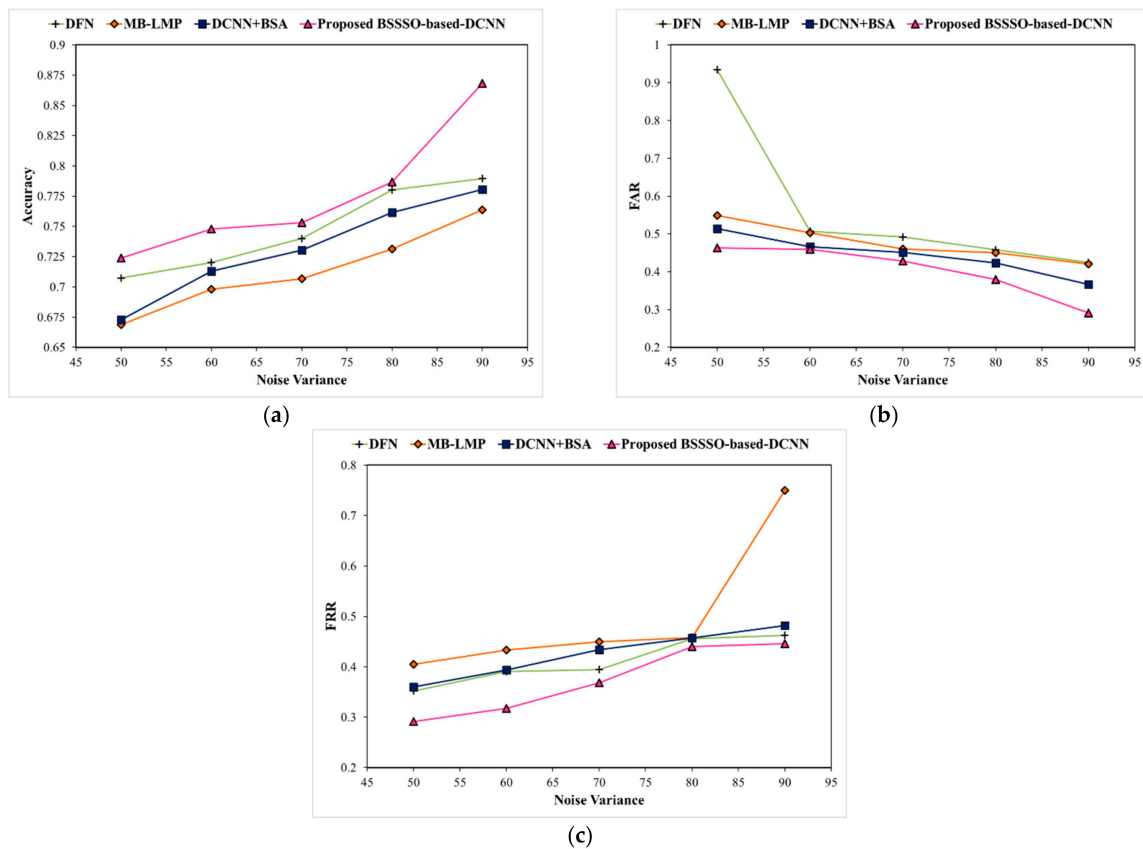


Figure 8. Analysis based on LFW dataset by varying the noise variance.

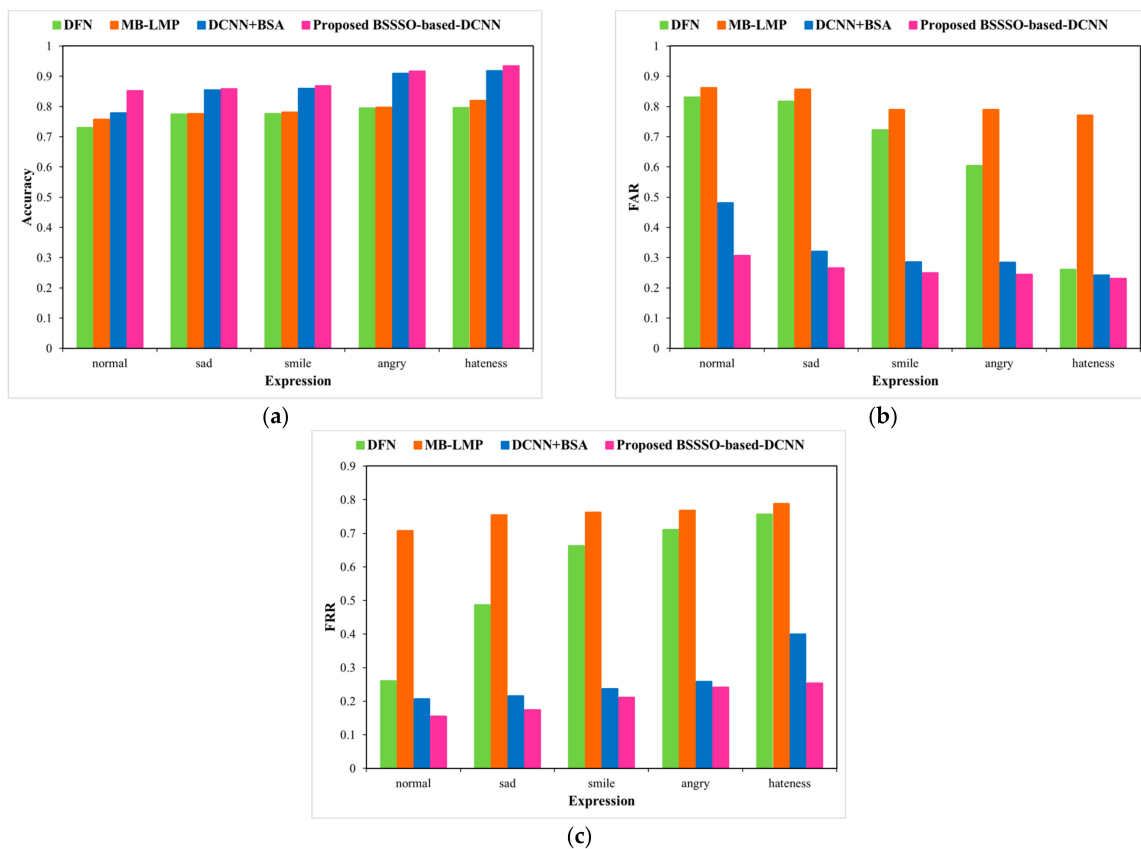


Figure 9. Analysis using UMB-DB database by varying expression.

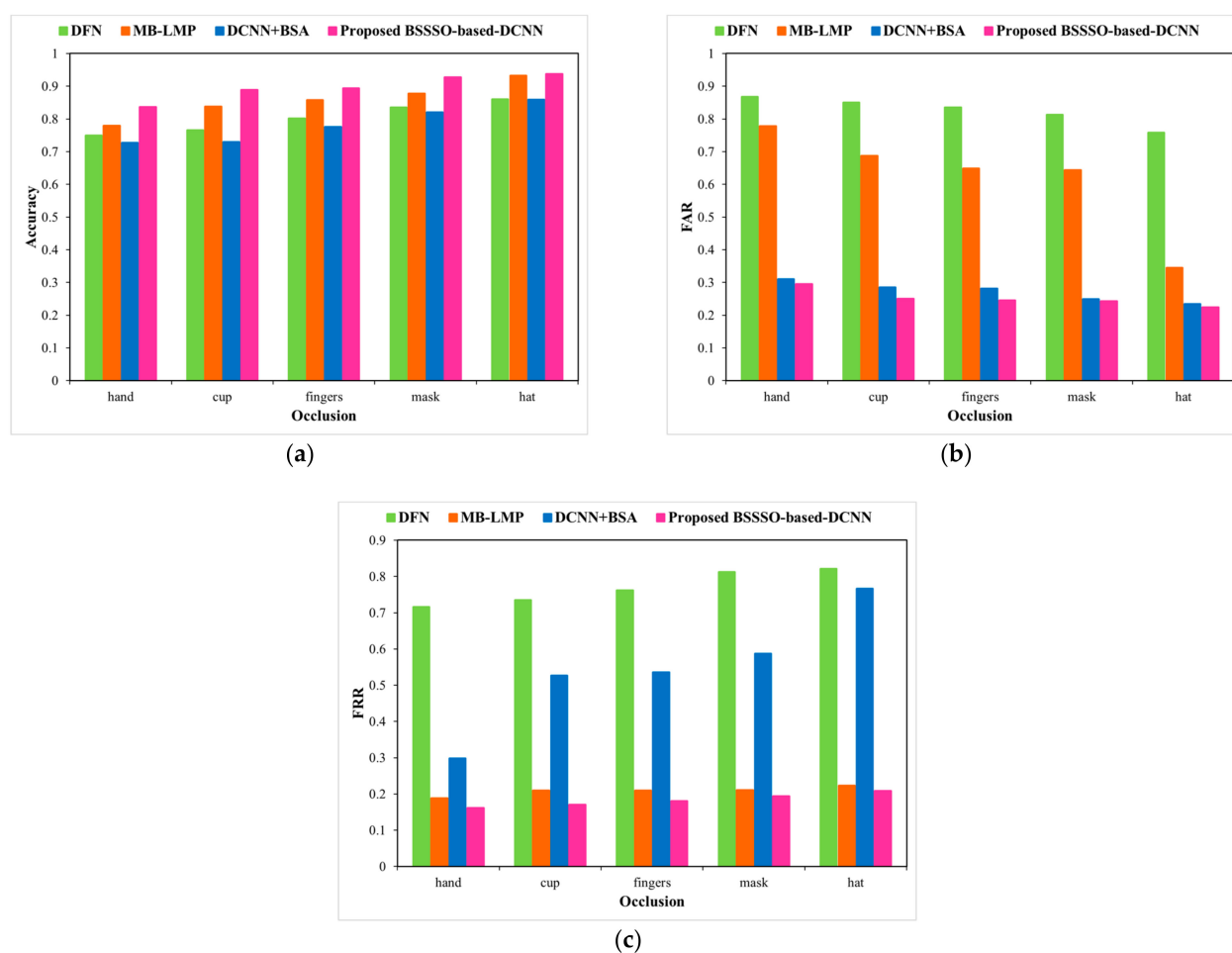


Figure 10. Analysis using UMB-DB database by varying occlusion.

Analysis using Extended Yale B database

We evaluate our model on the Extended Yale B database by illumination and pose variations. Figure 11a–c portray the accuracy, FAR, and FRR, respectively, for the developed method and the traditional methods for different illumination angles. When the illumination angle is 45, the proposed algorithm shows an improved +1.15 accuracy in terms of relative performance when compared with BSA + DCNN.

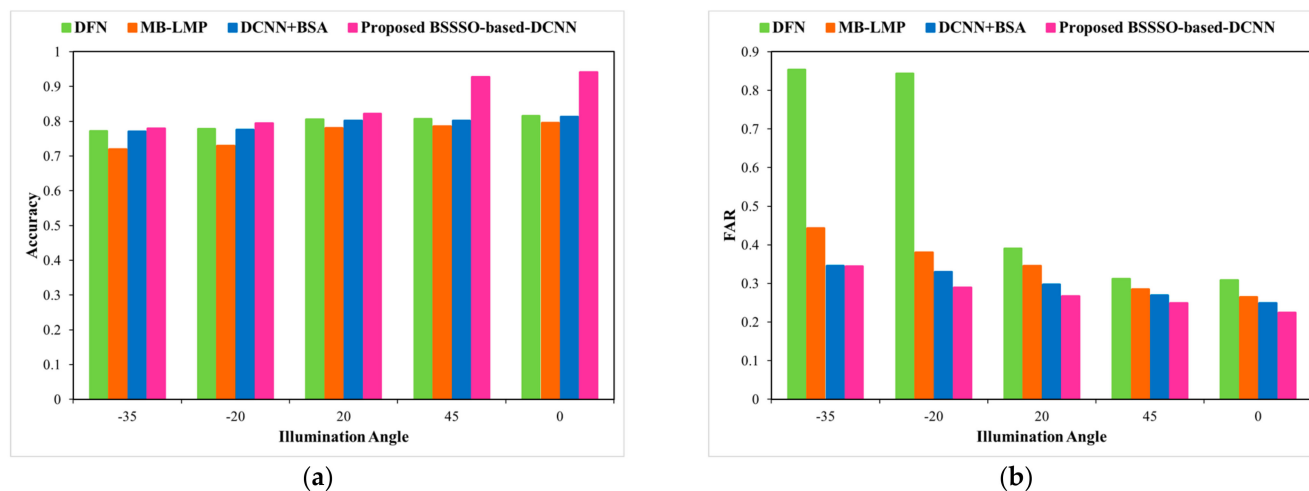


Figure 11. Cont.

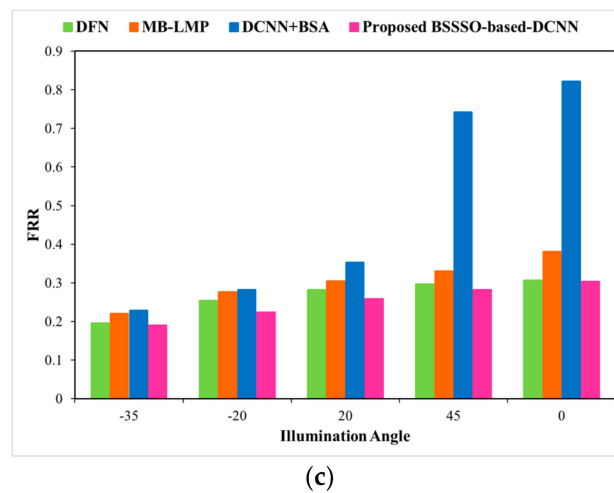
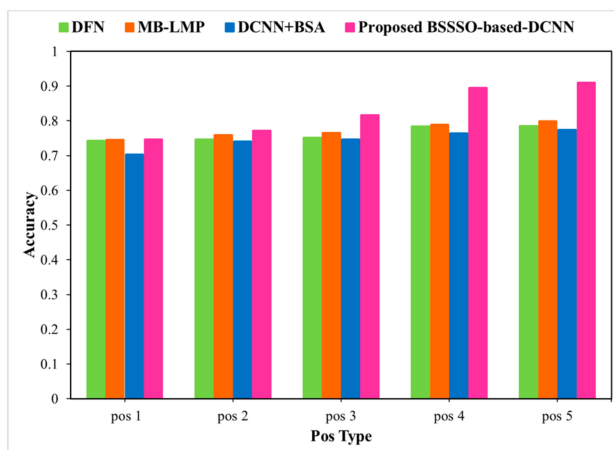
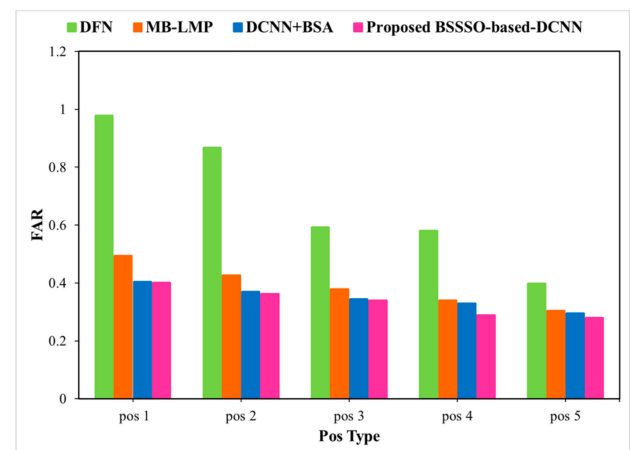


Figure 11. Analysis using Extended Yale B database by illumination angle.

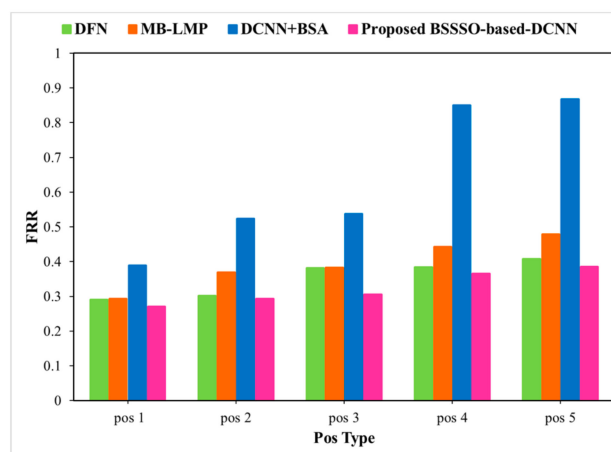
Figure 12 portrays the analysis of the proposed method using the Extended Yale B database by varying pose type. Figure 12a–c portray the analysis of accuracy, FAR, and FRR, respectively, with respect to pose types. Improved accuracy, reduced FAR, and reduced FRR demonstrate that the novel BSSO-based Deep CNN is robust to illumination and pose variations.



(a)



(b)



(c)

Figure 12. Analysis with respect to Extended Yale B database by pos type.

4.4. Comparative Discussion

The results illustrated in Figures 7–12 indicate the generalization ability of the proposed BSSSO-based Deep CNN for face recognition in the presence of noise, varying expressions, occlusions, illuminations, and pose variations. The proposed method outperforms the state-of-the-art face recognition techniques on the standard LFW, UMB-DB, and extended Yale-B database. When compared with other techniques, BSSSO-based Deep CNN shows significant improvement in terms of accuracy, FAR, and FRR.

Table 1 illustrates a comparative discussion of the developed model. By considering the training data of 90%, the accuracy measure computed by the traditional DF, MB-LMP, and DCNN+BSA is 0.8586, 0.8495, and 0.8132, whereas developed BSSSO-based Deep CNN achieved higher accuracy of 0.8935 using LFW database by varying training data. Similar improvements have been achieved with respect to noise variance using the LFW database for the proposed method.

Table 1. Comparative discussion.

Database	Metrics	DF	MB-LMP	DCNN + BSA	Proposed BSSSO-Based Deep CNN
LFW database	Accuracy	0.8586	0.8495	0.8132	0.8935
	FAR	0.4222	0.4125	0.4101	0.2190
	FRR	0.5810	0.5010	0.7190	0.4953
	Accuracy	0.7896	0.7637	0.7804	0.8683
	FAR	0.4245	0.4208	0.3667	0.2908
	FRR	0.4622	0.7501	0.4816	0.4459

5. Conclusions

This research aims to establish an optimization-driven deep learning approach for face recognition in multiple disturbing environments. The major contribution of the research is the development of a new deep learning strategy based on the optimization algorithm for face recognition. The proposed novel bird search-based shuffled shepherd optimization algorithm (BSSSO) is a meta-heuristic technique motivated by the intuition of animals and the social behavior of birds.

Initially, the input image undergoes a noise removal phase to eliminate noise to make them suitable for subsequent processing. The noise removal is performed using T2FCS, which helps to detect noisy pixels from the image for improved processing. After the noise removal phase, the feature extractions are carried out using the CNN model and L3DMM. Here, the CNN is the multilayered NN with the special architecture for detecting the complex features in the data, and the L3DMM feature is utilized to efficiently fit 3D face from a single uncontrolled image. The obtained features are subjected to Deep CNN for the face recognition. The training of DCNN is performed using BSSSO. Here, the proposed BSSSO is designed by combining the shuffled shepherd optimization algorithm (SSOA) and bird swarm algorithm (BSA). The experimental findings on standard datasets (LFW, UMB-DB, Extended Yale B database) prove the ability of the proposed model over the existing face recognition approaches. For instance, the proposed method obtained higher accuracy of 0.8935, and minimum FAR and FRR of 0.2190 and 0.4953, respectively, using the LFW database concerning training data.

Author Contributions: All authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: One of the authors, Neha Soni, wants to thank the Department of Science & Technology (DST), Ministry of Science & Technology, New Delhi, India, for the financial support as DST Inspire Fellow.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yan, C.; Xie, H.; Chen, J.; Zha, Z.; Hao, X.; Zhang, Y.; Dai, Q. A fast Uyghur text detector for complex background images. *IEEE Trans Multimed.* **2018**, *20*, 3389–3398. [CrossRef]
2. Soni, N.; Sharma, E.K.; Singh, N.; Kapoor, A. Assistance System (AS) for Vehicles on Indian Roads: A Case Study. In *International Conference on Human Systems Engineering and Design: Future Trends and Applications*; Springer: Cham, Switzerland, 2018; pp. 512–517.
3. Gulli, A.; Kapoor, A. *TensorFlow 1.x Deep Learning Cookbook: Over 90 Unique Recipes to Solve Artificial-Intelligence Driven Problems with Python*; Packt Publishing Ltd.: Birmingham, UK, 2017.
4. Soni, N.; Sharma, E.K.; Singh, N.; Kapoor, A. Artificial Intelligence in Business: From Research and Innovation to Market Deployment. *Procedia Comput. Sci.* **2020**, *167*, 2200–2210. [CrossRef]
5. Kortli, Y.; Jridi, M.; Al-Falou, A.; Atri, M. Face recognition systems: A Survey. *Sensors* **2020**, *20*, 342. [CrossRef] [PubMed]
6. Soni, N.; Singh, N.; Kapoor, A.; Sharma, E.K. Low-resolution image recognition using cloud hopfield neural network. In *Progress in Advanced Computing and Intelligent Engineering*; Springer: Singapore, 2018; pp. 39–46.
7. Soni, N.; Singh, N.; Kapoor, A.; Sharma, E.K. Face recognition using cloud Hopfield neural network. In *Proceedings of the International Conference on Wireless Communications, Signal Processing and Networking*, Chennai, India, 23–25 March 2016; pp. 416–419.
8. Ibrahim, A.; Tharwat, A.; Gaber, T.; Hassanien, A.E. Optimized superpixel and AdaBoost classifier for human thermal face recognition. *Signal Image Video Process.* **2018**, *12*, 711–719. [CrossRef]
9. Chen, Z.; Huang, W.; Lv, Z. Towards a face recognition method based on uncorrelated discriminant sparse preserving projection. *Multimed. Tools Appl.* **2017**, *76*, 17669–17683. [CrossRef]
10. Vishwakarma, V.P.; Dalal, S. A novel non-linear modifier for adaptive illumination normalization for robust face recognition. *Multimed. Tools Appl.* **2020**, *79*, 1–27. [CrossRef]
11. Zangeneh, E.; Rahmati, M.; Mohsenzadeh, Y. Low resolution face recognition using a two-branch deep convolutional neural network architecture. *Expert Syst. Appl.* **2020**, *139*, 112854. [CrossRef]
12. He, M.; Zhang, J.; Shan, S.; Kan, M.; Chen, X. Deformable face net for pose invariant face recognition. *Pattern Recognit.* **2020**, *100*, 107113. [CrossRef]
13. Li, X.X.; Hao, P.; He, L.; Feng, Y. Image gradient orientations embedded structural error coding for face recognition with occlusion. *J. Ambient Intell. Hum. Comput.* **2020**, *11*, 2349–2367. [CrossRef]
14. Prasad, P.S.; Pathak, R.; Gunjan, V.K.; Rao, H.R. Deep learning based representation for face recognition. In *ICCCE*; Springer: Singapore, 2019; pp. 419–424.
15. Gulli, A.; Kapoor, A.; Pal, S. *Deep Learning with TensorFlow 2 and Keras: Regression, ConvNets, GANs, RNNs, NLP, and More with TensorFlow 2 and the Keras API*; Packt Publishing Ltd.: Birmingham, UK, 2019.
16. Kaveh, A.; Hamedani, K.B.; Zaerreza, A. A set theoretical shuffled shepherd optimization algorithm for optimal design of cantilever retaining wall structures. *Eng. Comput.* **2020**, 1–18. [CrossRef]
17. Meng, X.B.; Gao, X.Z.; Lu, L.; Liu, Y.; Zhang, H. A new bio-inspired optimisation algorithm: Bird Swarm Algorithm. *J. Exp. Theor. Artif. Intell.* **2016**, *28*, 673–687. [CrossRef]
18. Yang, W.; Zhang, X.; Li, J. A local multiple patterns feature descriptor for face recognition. *Neurocomputing* **2020**, *373*, 109–122. [CrossRef]
19. Gao, G.; Yu, Y.; Yang, M.; Chang, H.; Huang, P.; Yue, D. Cross-resolution face recognition with pose variations via multilayer locality-constrained structural orthogonal procrustes regression. *Inf. Sci.* **2020**, *506*, 19–36. [CrossRef]
20. Chen, Z.; Wu, X.J.; Yin, H.F.; Kittler, J. Noise-robust dictionary learning with slack block-diagonal structure for face recognition. *Pattern Recognit.* **2020**, *100*, 107118. [CrossRef]
21. Iranmanesh, S.M.; Riggan, B.; Hu, S.; Nasrabadi, N.M. Coupled generative adversarial network for heterogeneous face recognition. *Image Vis. Comput.* **2020**, *94*, 103861. [CrossRef]
22. Kumar, S.V.; Nagaraju, C. T2FCS filter: Type 2 fuzzy and cuckoo search-based filter design for image restoration. *J. Vis. Commun. Image Represent.* **2019**, *58*, 619–641. [CrossRef]
23. Peng, B.; Wang, W.; Dong, J.; Tan, T. Automatic detection of 3d lighting inconsistencies via a facial landmark based morphable model. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, Phoenix, AZ, USA, 25–28 September 2016; pp. 3932–3936.
24. Labeled Faces in the Wild Home Database. Available online: <http://vis-www.cs.umass.edu/lfw/#download> (accessed on 25 July 2020).
25. The University of Milano Bicocca 3D Face Database. Available online: <http://www.ivl.disco.unimib.it/minisites/umbdb/request.html> (accessed on 23 July 2020).
26. Extended Yale B Database. Available online: <http://vision.ucsd.edu/~lke/ExtYaleDatabase/ExtYaleB.html> (accessed on 29 July 2020).