electronics

MDPI

Article

Automated Facial Emotion Recognition Using the Pelican Optimization Algorithm with a Deep Convolutional Neural Network

Mohammed Alonazi [1,*], Hala J. Alshahrani [2], Faiz Abdullah Alotaibi [3], Mohammed Maray [4], Mohammed Alghamdi [5] and Ahmed Sayed [6]

[1] Department of Information Systems, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj 16273, Saudi Arabia
[2] Department of Applied Linguistics, College of Languages, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[3] Department of Information Science, College of Humanities and Social Sciences, King Saud University, P.O. Box 28095, Riyadh 11437, Saudi Arabia
[4] Department of Information Systems, College of Computer Science, King Khalid University, P.O. Box 394, Abha 61421, Saudi Arabia
[5] Department of Information and Technology Systems, College of Computer Science and Engineering, University of Jeddah, Jeddah 23218, Saudi Arabia
[6] Research Centre, Future University in Egypt, New Cairo 11845, Egypt
* Correspondence: mn.alonazi@psau.edu.sa

Abstract: Facial emotion recognition (FER) stands as a pivotal artificial intelligence (AI)-driven technology that exploits the capabilities of computer-vision techniques for decoding and comprehending emotional expressions displayed on human faces. With the use of machine-learning (ML) models, specifically deep neural networks (DNN), FER empowers the automatic detection and classification of a broad spectrum of emotions, encompassing surprise, happiness, sadness, anger, and more. Challenges in FER include handling variations in lighting, poses, and facial expressions, as well as ensuring that the model generalizes well to various emotions and populations. This study introduces an automated facial emotion recognition using the pelican optimization algorithm with a deep convolutional neural network (AFER-POADCNN) model. The primary objective of the AFER-POADCNN model lies in the automatic recognition and classification of facial emotions. To accomplish this, the AFER-POADCNN model exploits the median-filtering (MF) approach to remove the noise present in it. Furthermore, the capsule-network (CapsNet) approach can be applied to the feature-extraction process, allowing the model to capture intricate facial expressions and nuances. To optimize the CapsNet model's performance, hyperparameter tuning is undertaken with the aid of the pelican optimization algorithm (POA). This ensures that the model is finely tuned to detect a wide array of emotions and generalizes effectively across diverse populations and scenarios. Finally, the detection and classification of different kinds of facial emotions take place using a bidirectional long short-term memory (BiLSTM) network. The simulation analysis of the AFER-POADCNN system is tested on a benchmark FER dataset. The comparative result analysis showed the better performance of the AFER-POADCNN algorithm over existing models, with a maximum accuracy of 99.05%.

Keywords: facial emotion recognition; deep learning; computer vision; emotions; pelican optimization algorithm

check for updates

Citation: Alonazi, M.; Alshahrani, H.J.; Alotaibi, F.A.; Maray, M.; Alghamdi, M.; Sayed, A. Automated Facial Emotion Recognition Using the Pelican Optimization Algorithm with a Deep Convolutional Neural Network. *Electronics* **2023**, *12*, 4608. https://doi.org/10.3390/electronics12224608

Academic Editors: Dong Zhang and Dah-Jye Lee

Received: 28 September 2023
Revised: 27 October 2023
Accepted: 1 November 2023
Published: 11 November 2023

1. Introduction

Facial emotion recognition (FER) methods are mainly utilized to recognize facial expressions on the human face [1]. Numerous kinds of emotions occur, but some may not be superficial to the human eye [2]. Therefore, with the help of appropriate mechanisms, any kind of suggestions can aid in identifying the classification. In the FER field, there are different types of universal facial expressions, like neutral, happiness, surprise, fear, anger, sadness, and disgust [3]. From facial expressions, emotion extraction is an active study in mental health, psychiatry, and psychology nowadays [4]. The automatic emotion recognition from facial expressions comes with numerous usages, likely HCI (human–computer interaction), modern augmented reality, healthcare, smart living, and HRI (human–robot interaction) [5]. Most of the researchers are employing FER because it comes with many customs [6]. The procedure to build emotion-specific features is difficult because of several factors which appear from the nonlinear interaction among dissimilar evidence, multidimensional data, and the modalities that one can rapidly face with emotions in different scenarios [7].

There is an increasing demand for robust and accurate systems able to automatically recognize and classify human emotions from facial expressions. Emotions have a major role in human communication, affect decision-making processes, and have applications in diverse domains, from human–computer interaction to mental health assessment. Existing FER models often face challenges related to noise, feature extraction, and generalization. Recently, machine learning (ML) and deep networks have established an effective technique to avoid such restrictions by distinguishing the most multipart nonlinear features linked in multimodal data [8]. The two utmost-serious techniques in emotion detection are feature extraction and classification [9]. Some of the foremost feature classifiers used for superior classification exactness are artificial neural network, ML, and deep-learning (DL) systems [10]. The established feature-engineering and ML methods attempt to remove complicated as well as nonlinear patterns from the multivariate time-series data [11]. However, selecting an effective characteristic from many feature sets is very difficult, so the dimensionality-reduction method will be needed. The feature-extraction as well as selection process takes more time. For instance, when the dimensionality feature increases, the calculating feature overhead selection develops radically [12]. DL techniques, namely, the recurrent neural network (RNN), autoencoder (AE), and convolutional neural network (CNN), have improved in all the areas of computing, especially computer vision, natural language processing (NLP), audio-recognition machine translation, etc. [13]. Recently, DL methods have been employed to deliver high-level data abstraction to develop a flexible structure for emotion detection. In the DL method, DNNs are utilized to gather unique qualities from the high-level data illustration [14].

This study introduces an automated facial emotion recognition using the pelican optimization algorithm with a deep convolutional neural network (AFER-POADCNN) model. The major intention of the AFER-POADCNN model lies in the automated recognition and classification of facial emotions. To accomplish this, the AFER-POADCNN method exploits the median-filtering (MF) approach to remove the noise present in it. Furthermore, the capsule-network (CapsNet) approach can be applied to the feature-extraction process and the hyperparameter tuning of the CapsNet model is carried out by the POA. Finally, the detection and classification of different kinds of facial emotions take place using a bidirectional long short-term memory (BiLSTM) network. The performance analysis of the AFER-POADCNN technique is tested on benchmark FER databases. In short, the key contributions of the paper are summarized as follows.

- An AFER-POADCNN technique comprising MF-based preprocessing, a CapsNet feature extractor, POA-based hyperparameter tuning, and BiLSTM classification has been developed for FER. To the best of our knowledge, the AFER-POADCNN technique has never existed in the literature;
- The CapsNet model has been employed for feature extraction, allowing for the capture of intricate and nuanced facial expressions;
- The POA is presented to tune the hyperparameters of the capsule network, enhancing the model's adaptability and generalization to different emotions and diverse populations;
- The BiLSTM model applied for emotion classification ensures the robust detection and categorization of various facial emotions.

2. Literature Review

Sarvakar et al. [15] proposed a neural network convolutionary (FERC) technique which uses two parts. At the initial stage, the image backdrop is removed, and then the face vector is removed. For the classification process, the expressional vector (EV) is employed. The two-layer CNN is constant, and the exponent and weight standards of the last perception layer may differ with each iteration. Then, the EV generation ensures the growth of issues before a novel background-removal procedure is utilized. Said and Barr [16] designed a face-sensitive CNN for human-emotion classification, which is collected from two phases. At the initial stage, the method is employed to identify faces in high-resolution images, and then the faces are trimmed for further processing. Next, the CNN is utilized to estimate facial expression, which is relayed on standard analytics. Then, it is implemented on pyramid images for processing scale invariance. In [17], the facial-expression-detection method has been designed. At an initial stage, an area of interest has been performed as face classification. Secondly, a DL-based CNN design is projected. In the third stage, some of the new data-augmentation methods have been applied.

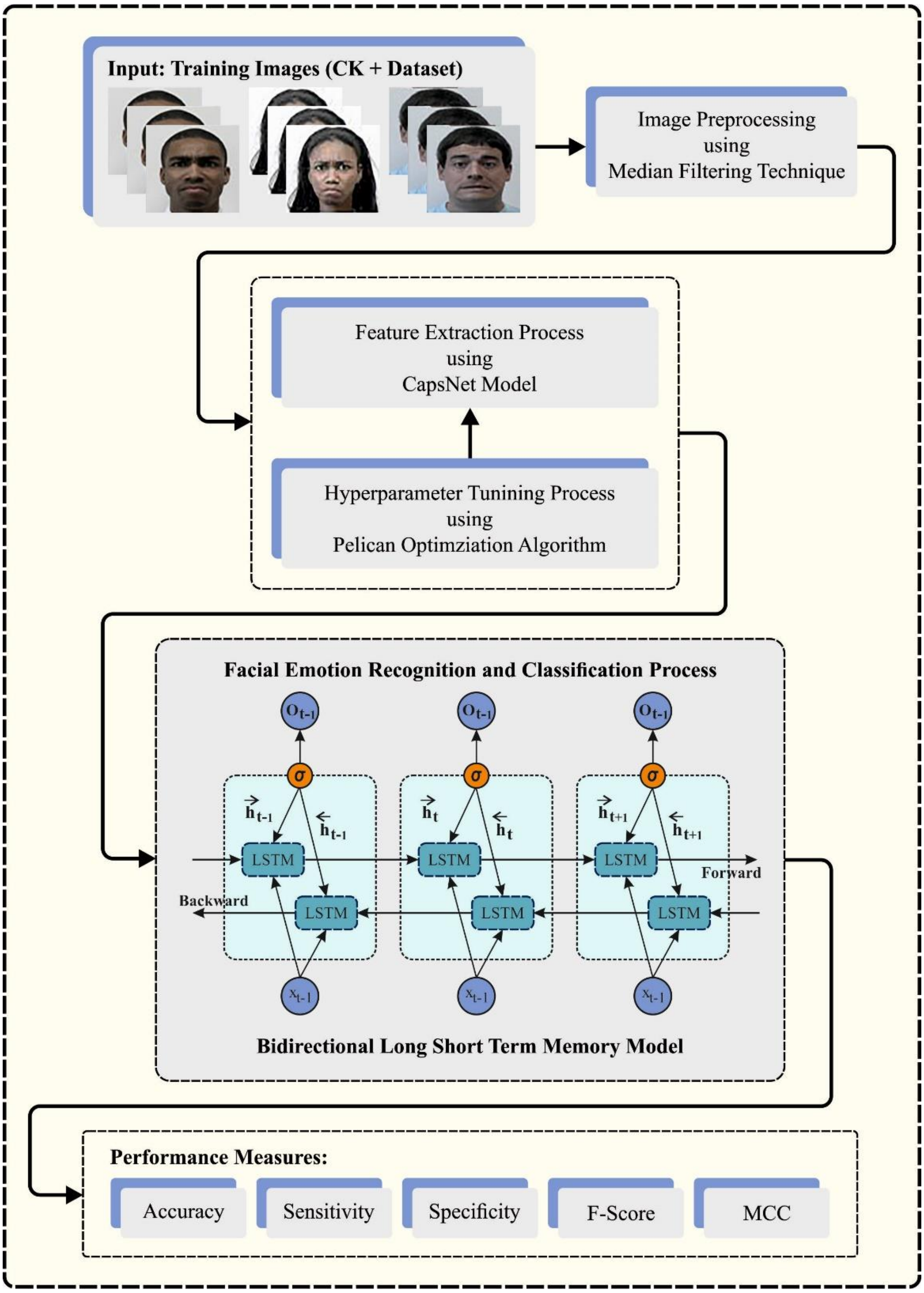
Talaat [18] proposed a real-time emotion-identification method that employs three phases of emotion classification. The selection designs an enhanced DL method to identify facial emotions by utilizing the CNN. The projected emotion-recognition framework took up the benefit of employing the IoT and fog for reducing the delays for real-time classification, with a quick response time as well as providing location awareness. Chowdary et al. [19] mainly dealt with emotion detection by making use of the transfer-learning (TL) method. The well-defined networks of Mobile Net, Vgg19, Resnet50, and Inception V3 are utilized in the research. The pretrained ConvNets are deleted, and then entire connected layers are added that are more appropriate for the totality of the instructions. At last, the fresh additional networks are skilled to improve the weights. In [20], an automated framework algorithm is used for facial recognition by employing an FD-CNN, which is developed with four convolutional layers as well as two hiding networks for enhancing the accuracy. An extensive CK+ dataset is mainly employed, including facial images of dissimilar females and males with various expressions. For validating the projected technique, K-fold cross-validation is executed.

Sikkandar and Thiyagarajan [21] presented an improved cat swarm optimization (ICSO) method. The deep-CNN technique is employed for the extraction process. The ICSO is mainly designed to select optimum features. Using DCNN with the ICSO method enhances the retrieval performance; then, the ensemble classification algorithm uses a support vector machine (SVM) and neural network (NN) that are performed to classify facial expressions. Helaly et al. [22] developed a DCNN method based on an intelligent computer-vision system which is capable of identifying the facial emotions on human faces. In the first stage, the DCNN designed using the TL method is mainly introduced to build up an accurate FER system. Secondly, the research suggests the ResNet18 method.

In [23], a new end-to-end facial-microexpression-recognition architecture termed Deep3DCANN has been developed to combine these modules for active microexpression discovery. The first module of our design is a deep 3D-CNN that learns beneficial spatiotemporal features from a series of facial images. Kansizoglou et al. [24] offer a new model for online emotion detection that features audio as well as visual modalities, and then offers a receptive forecast when the system is sufficiently self-assured. The author developed two deep CNN techniques for removing emotional features; one model for each modality, and a DNN for their fusion. Li et al. [25] presented a unique self-supervised exclusive–inclusive interactive-learning (SEIIL) technique to simplify the discriminative multilabel FER in the wild that effectually grip-coupled manifold thoughts with incomplete unrestrained training data. Kansizoglou et al. [26] presented a new method that slowly maps, as well as learns, human personalities by considering and following a person's emotional differences through communication. The developed network removes the facial landmarks of a subject, which are utilized to train a properly planned deep recurrent neural network (DRNN) framework.

3. The Proposed Model

In this study, we have designed a novel AFER-POADCNN model. The primary objective of the AFER-POADCNN model lies in the automated recognition and classification of facial emotions. To accomplish this, the AFER-POADCNN method includes different phases of operations, namely, MF-based preprocessing, the CapsNet model for feature extraction, the BiLSTM model for classification, and POA-based hyperparameter tuning. Figure 1 portrays the entire procedure of the AFER-POADCNN system.



Input: Training Images (CK + Dataset)
Image Preprocessing using Median Filtering Technique
Feature Extraction Process using CapsNet Model
Hyperparameter Tunining Process using Pelican Optimziation Algorithm
Facial Emotion Recognition and Classification Process
O_{t-1}
O_{t-1}
O_{t-1}
σ
σ
σ
$\overrightarrow{h}_{t-1}$
$\overleftarrow{h}_{t-1}$
$\overrightarrow{h}_{t}$
$\overleftarrow{h}_{t}$
$\overrightarrow{h}_{t+1}$
$\overleftarrow{h}_{t+1}$
LSTM
LSTM
LSTM
LSTM
LSTM
LSTM
Forward
Backward
x_{t-1}
x_{t-1}
x_{t-1}
Bidirectional Long Short Term Memory Model
Performance Measures:
Accuracy
Sensitivity
Specificity
F-Score
MCC

Figure 1. Overall process of the AFER-POADCNN algorithm.

3.1. Image Preprocessing

To remove the noise that exists in the input images, MF is used. It is a basic preprocessing method deployed in image processing to decrease noise and increase data quality [26]. Different from other smoothing techniques, MF methods excel at maintaining edges and fine details while efficiently mitigating impulse noise, like salt-and-pepper noise usually created in images. It works by changing all values of the pixels with the median value from a local neighborhood, making it specifically compatible with applications where noise reduction is important, without affecting the integrity of significant image features. MF is extensively utilized in domains like CV, remote sensing, and medical imaging for enhancing the data quality and robustness before additional visualization or analysis.

3.2. Feature Extraction

For deriving the feature vectors, the CapsNet model is applied. Capsule networks, also known as CapsNets, describe a new method for DL techniques, developed to address a few drawbacks of standard CNNs in tasks, namely image recognition [27]. Developed by Geoffrey Hinton and his team, CapsNets present capsules as the main building blocks. These capsules are smaller collections of neurons that function together to identify different parts of objects or visual patterns within an image. Dissimilar CNNs depend on max-pooling for extracting features, and CapsNets utilize dynamic-routing mechanisms to evaluate the spatial relationships among parts and their entire objects. This allows CapsNets to manage complex hierarchical connections among features that are specifically beneficial in conditions where object pose and orientation matter, like in understanding handwritten features or identifying objects in cluttered scenes. Another feature of CapsNets is the capability to manage variable-length pose vectors for all parts, permitting them to obtain rich information about the relative positions and orientations of object components. This makes CapsNets robust to various transformations, including scaling, rotation, and deformation, making them a compelling choice for tasks like image segmentation and object recognition in challenging real-world conditions. While CapsNets are still an evolving field of research, they hold significant promise in advancing the state-of-the-art in computer vision and pattern recognition. Figure 2 exemplifies the infrastructure of the CapsNet.

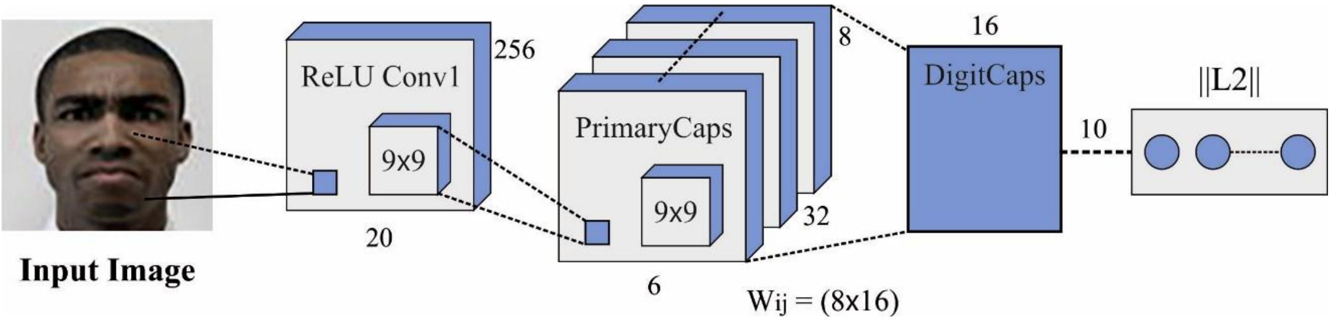


Figure 2. Structure of the CapsNet model.

3.3. Hyperparameter Tuning

In this work, the POA optimally chooses the hyperparameter values of the CapsNet model. POA is a new swarm intelligence (SI)-based optimization algorithm, and pelicans are its population [28]. The swarm member implies a candidate solution. Mainly, the swarm members are randomly initialized based on the problem limit:

$$z_{i,j} = l_j + rand \cdot (u_j - l_j), j = [1, 2, \ldots, m], i = [1, 2, \ldots, N] \tag{1}$$

In Equation (1), the upper and lower bounds of the jth variable of the problem are u_j and l_j, correspondingly. The amount of the jth parameter defined by the ith candidate solutions are $z_{i,j}$; the number of swarm members is N, an accidental amount between $(0,1)$ is *rand*, and the number of parameters in the solution space is m.

The pelican swarm's associates can be described by the matrix. In the following matrix, the value in each row describes a candidate solution; moreover, the value in the column depicts the presented amount for the variable in the solution space.

$$Z = \begin{bmatrix} Z_1 \\ \vdots \\ Z_i \\ \vdots \\ Z_N \end{bmatrix} = \begin{bmatrix} z_{1,1} & \cdots & z_{1,j} & \cdots & z_{1,m} \\ \vdots & \ddots & z_{i,j} & & \vdots \\ z_{i,1} & \cdots & \vdots & \cdots & z_{i,m} \\ \vdots & & \vdots & \ddots & \vdots \\ z_{N,1} & \cdots & z_{N,j} & \cdots & z_{N,m} \end{bmatrix} \tag{2}$$

In Equation (2), Z and Z_i are the swarm matrix and ith pelicans.

In this work, the cost function is calculated on any candidate solution. The cost-function vector (F_i) is used to define the attained amount for the cost function, as follows:

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(Z_1) \\ \vdots \\ F(Z_i) \\ \vdots \\ F(Z_N) \end{bmatrix} \tag{3}$$

1. Exploration Stage (Moving direction of Bait)

First, pelican members used to identify the hunting region; then, they move towards that place. The solution area is scanned due to its simulation of the pelican strategy; also, it gives rise to the exploration capability of the POA to explore different areas of the solution space. Consider that the hunting position is randomly generated in the searching region; the exploration ability rises once it finds the solution space. This can be mathematically expressed as follows:

$$z_{i,j}^{p_1} = \begin{cases} z_{i,j} + rand.\left(p_j - I.z_{i,j}\right), F_p < F_i, \\ z_{i,j} + rand.\left(z_{i,j} - p_j\right), else, \end{cases} \tag{4}$$

In Equation (4), the new situation of ith pelicans at the dimension is $z_{i,j}^{p_1}$, I indicates a random integer; the prey location in the jth dimension denotes the p_j. The parameter I is arbitrarily chosen for any member and any iteration. If the quantity of I is 2, then it increases in dislocation for a member; thus, the member conducts a new region of the problem. The exploration ability of these optimizers for the incorrect scanning of the problem is better than the I parameter.

The new position for the pelican is attained, which provides the cost-function value. The algorithm could not move towards the nonoptimum region using this type of upgrade, which is called the effective-update process. This method is simulated by the subsequent formula:

$$Z_i = \begin{cases} Z_i^{p_1}, F_i^{p_1} < F_i; \\ Z_i, else \end{cases} \tag{5}$$

In Equation (5), $Z_i^{p_1}$ shows the new position of ith pelicans and $F_i^{p_1}$ represents its cost-function amount.

2. Exploitation Stage (Winging on the Water Surface)

In this phase, the pelicans spread their wings to attain the water surface; therefore, the fish come out and they gather bait in their throat pouch. The statistical formula for the demeanor of pelicans during hunting is given below:

$$z_{i,j}^{p_2} = z_{i,j} + R \times \left(1 - \frac{t}{T}\right) \times (2 \times rand - 1) \times z_{i,j} \tag{6}$$

In Equation (6), the new location of the ith individuals at the jth parameter is $z_{i,j}^{p_2}$, and the neighborhood radius of $z_{i,j}$ is $R \times \left(1 - \frac{t}{T}\right)$, where the iteration number is t, R equals 0.2, and the maximal iteration counter is T. "$R \times \left(1 - \frac{t}{T}\right)$" is the coefficient representing the radii of the surrounding of the swarm individual to the local exploration, and the proximity to any member to converge for a promising solution. The nearby area and swarm member with more precise and shorter movements can be examined, and the PO is able to converge the answer closer to the global optima. In this stage, an effective update is used for taking or refusing the new pelican position, which is formulated as follows:

$$Z_i = \begin{cases} Z_i^{p_2}, F_i^{p_2} < F_i; \\ Z_i, else \end{cases} \tag{7}$$

In Equation (7), the novel situation of the ith pelican is $Z_i^{p_2}$ and the cost-function amount is $F_i^{p_2}$.

3. Repetition

Once the swarm individuals are updated, the optimal solution will be upgraded by the rate of the performance index and the new position of the swarm. The next iteration begins, and the different stages of the proposed PO, using the abovementioned formula, are repeated, finishing the whole performance. Eventually, the best solution candidate attained in the algorithm epoch is shown as a quasioptimal solution.

The POA algorithm derives an FF to obtain a high efficiency of classification. It determines a positive integer to represent the improved performance of the solution candidate. The decline of the classifier error rate is considered an FF.

$$fitness(x_i) = ClassifierErrorRate(x_i)$$

$$= \frac{number\ of\ misclassified\ samples}{Total\ number\ of\ samples} * 100 \tag{8}$$

3.4. Detection Using the BiLSTM Model

To detect the presence of emotions in distinct types, the BiLSTM model is applied. The input, forget, and output gates are the three gates of the LSTM unit [29]. The input gate defines what amount of the input data to remain in the existing state of the memory unit, the forget gate is a basic design for the LSTM to learn long-term dependency, and the output gate decides which state of the memory cell is transported to the hidden output state. The state of the memory unit at $t-1$ is C_{t-1}, the input of the unit at existing moment t is x_t, and the output of the hidden state at prior time $t-1$ is h_{t-1}, which are the three inputs of the LSTM neurons. The state of the memory cell at t is C_t and the output of the hidden layer (HL) at t is h_t, which are the two outputs of the LSTM neuron:

$$f_t = \sigma\left(w_{fx}x_t + w_{fh}h_{t-1} + b_1\right) \tag{9}$$

$$I_t = \sigma(w_{\mathrm{ix}}x_t + w_{ih}h_{t-1} + b_2) \tag{10}$$

$$Z_t = \tanh(w_{zx}x_t + w_{zh}h_{t-1} + b_3) \tag{11}$$

$$C_t = f_t \circ C_{t-1} + i_t \circ Z_t \tag{12}$$

$$O_t = \sigma(w_{ox}x_t + w_{oh}h_{t-1} + b_4) \tag{13}$$

$$h_t = O_t \circ \tanh(C_t) \tag{14}$$

From the equations, the sigmoid function is σ. The multiplication by elements is " $\circ$ "; w and b are the corresponding weight parameters and bias.

The prediction of the BiLSTM is based on time sequences and considers the negative and positive direction of prior data. The BiLSTM model comprises two layers of one-way LSTM, where HL in the positive time direction consists of prior data series and evaluates the present data sequence. HL in the reverse time direction is used to add the reverse data series in the calculation and read the future data series in the input. Next, the value defined by the two LSTM modules is feedforwarded into the output layer.

4. Results and Discussion

The proposed model is simulated using the Python 3.8.5 tool on a PC i5-8600k, GeForce 1050Ti 4 GB, 16 GB RAM, 250 GB SSD, and 1 TB HDD. The parameter settings are given as follows: learning rate: 0.01, dropout: 0.5, batch size: 5, epoch count: 50, and activation: ReLU. In this section, the FER outcome of the AFER-POADCNN algorithm is examined on the FER database [30], comprising 920 images and 8 classes, as illustrated in Table 1.

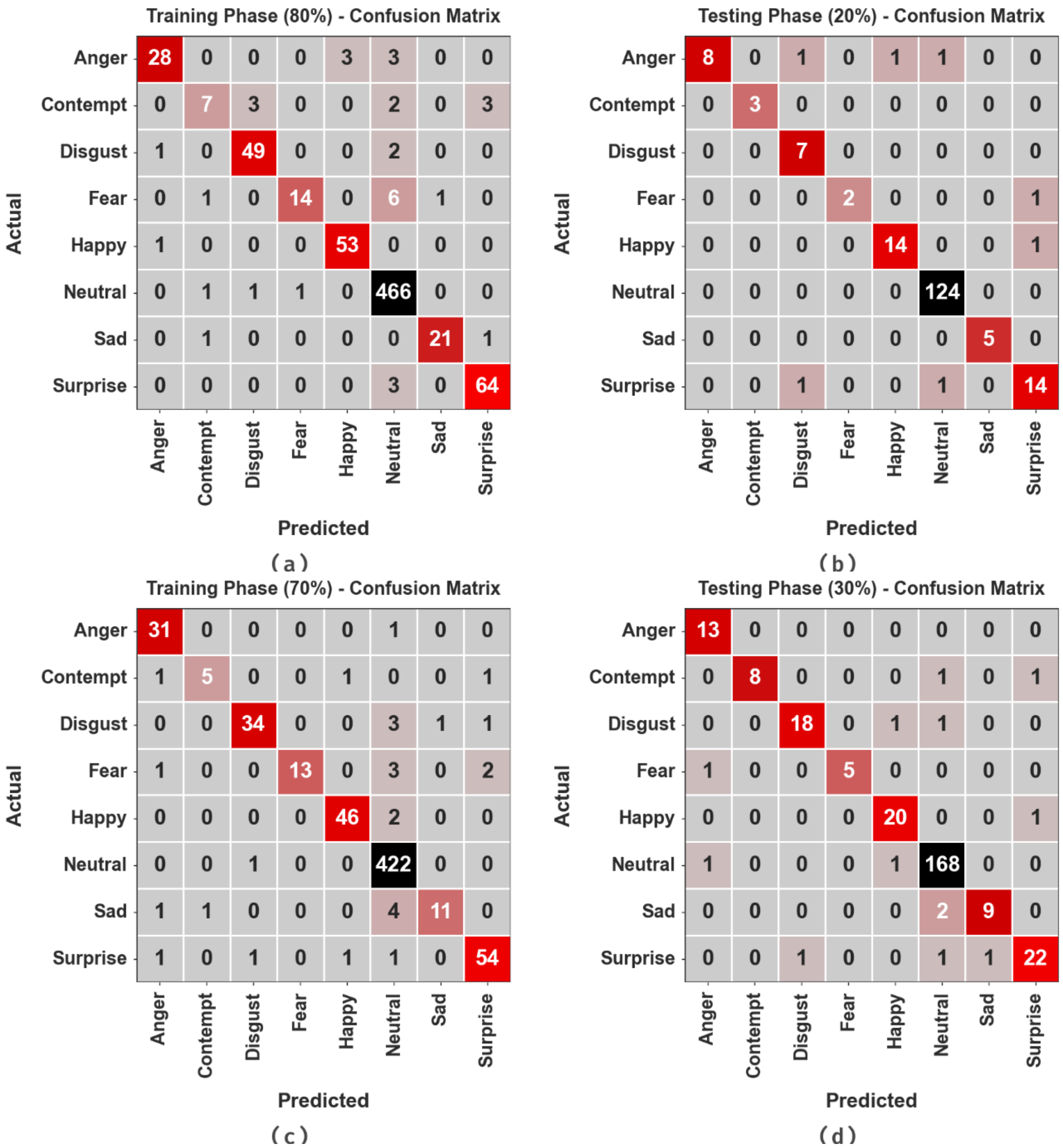
Table 1. Details of the database.

Classes	No. of Samples
Anger	45
Contempt	18
Disgust	59
Fear	25
Happy	69
Neutral	593
Sad	28
Surprise	83
Total No. of Sample Images	920

Figure 3 demonstrates the confusion matrices attained by the AFER-POADCNN algorithm at 80:20 and 70:30 of the TR phase/TS phase. The outcome denotes the effective recognition and classification of all eight classes.

In Table 2 and Figure 4, the FER results of the AFER-POADCNN technique at 80:20 of the TR phase/TS phase are presented. The simulation values highlight that the AFER-POADCNN method properly recognized facial emotions accurately. With 80% of the TR phase, the AFER-POADCNN technique offers an average $accu_y$ of 98.85%, $sens_y$ of 83.90%, $spec_y$ of 98.93%, F_{score} of 86.88%, and an MCC of 86.36%. Additionally, with 20% of the TS phase, the AFER-POADCNN approach achieves an average $accu_y$ of 99.05%, $sens_y$ of 90.03%, $spec_y$ of 99.22%, F_{score} of 91.47%, and an MCC of 91.28%.

In Table 3 and Figure 5, the FER outcome of the AFER-POADCNN approach at 70:30 of the TR phase/TS phase is presented. The outcome displayed that the AFER-POADCNN algorithm appropriately detected the facial emotions accurately. With 70% of the TR phase, the AFER-POADCNN system gains an average $accu_y$ of 98.91%, $sens_y$ of 83.55%, $spec_y$ of 98.92%, F_{score} of 87.37%, and an MCC of 86.99%. Furthermore, with 30% of the TS phase, the AFER-POADCNN approach reaches an average $accu_y$ of 98.82%, $sens_y$ of 89.65%, $spec_y$ of 99.02%, F_{score} of 91.43%, and an MCC of 90.74%.



Training Phase (80%) - Confusion Matrix

	Anger	Contempt	Disgust	Fear	Happy	Neutral	Sad	Surprise
Anger	28	0	0	0	3	3	0	0
Contempt	0	7	3	0	0	2	0	3
Disgust	1	0	49	0	0	2	0	0
Fear	0	1	0	14	0	6	1	0
Happy	1	0	0	0	53	0	0	0
Neutral	0	1	1	1	0	466	0	0
Sad	0	1	0	0	0	0	21	1
Surprise	0	0	0	0	0	3	0	64

(a)

Testing Phase (20%) - Confusion Matrix

	Anger	Contempt	Disgust	Fear	Happy	Neutral	Sad	Surprise
Anger	8	0	1	0	1	1	0	0
Contempt	0	3	0	0	0	0	0	0
Disgust	0	0	7	0	0	0	0	0
Fear	0	0	0	2	0	0	0	1
Happy	0	0	0	0	14	0	0	1
Neutral	0	0	0	0	0	124	0	0
Sad	0	0	0	0	0	0	5	0
Surprise	0	0	1	0	0	1	0	14

(b)

Training Phase (70%) - Confusion Matrix

	Anger	Contempt	Disgust	Fear	Happy	Neutral	Sad	Surprise
Anger	31	0	0	0	0	1	0	0
Contempt	1	5	0	0	1	0	0	1
Disgust	0	0	34	0	0	3	1	1
Fear	1	0	0	13	0	3	0	2
Happy	0	0	0	0	46	2	0	0
Neutral	0	0	1	0	0	422	0	0
Sad	1	1	0	0	0	4	11	0
Surprise	1	0	1	0	1	1	0	54

(c)

Testing Phase (30%) - Confusion Matrix

	Anger	Contempt	Disgust	Fear	Happy	Neutral	Sad	Surprise
Anger	13	0	0	0	0	0	0	0
Contempt	0	8	0	0	0	1	0	1
Disgust	0	0	18	0	1	1	0	0
Fear	1	0	0	5	0	0	0	0
Happy	0	0	0	0	20	0	0	1
Neutral	1	0	0	0	1	168	0	0
Sad	0	0	0	0	0	2	9	0
Surprise	0	0	1	0	0	1	1	22

(d)

Figure 3. Confusion matrices of (**a**–**c**) TR phase of 80% and 70% and (**b**–**d**) TS phase of 20% and 30%.

Table 2. FER outcome of the AFER-POADCNN algorithm at 80:20 of the TR phase/TS phase.

Classes	$Accu_y$	$Sens_y$	$Spec_y$	F_{Score}	MCC
		TR Phase (80%)			
Anger	98.91	82.35	99.72	87.50	87.12
Contempt	98.51	46.67	99.58	56.00	56.45
Disgust	99.05	94.23	99.42	93.33	92.83
Fear	98.78	63.64	99.86	75.68	76.52

Table 2. *Cont.*

Classes	$Accu_y$	$Sens_y$	$Spec_y$	F_{Score}	MCC
Happy	99.46	98.15	99.56	96.36	96.09
Neutral	97.42	99.36	94.01	98.00	94.43
Sad	99.59	91.30	99.86	93.33	93.15
Surprise	99.05	95.52	99.40	94.81	94.29
Average	98.85	83.90	98.93	86.88	86.36
TS Phase (20%)					
Anger	98.37	72.73	100.00	84.21	84.55
Contempt	100.00	100.00	100.00	100.00	100.00
Disgust	98.91	100.00	98.87	87.50	87.69
Fear	99.46	66.67	100.00	80.00	81.43
Happy	98.91	93.33	99.41	93.33	92.74
Neutral	98.91	100.00	96.67	99.20	97.54
Sad	100.00	100.00	100.00	100.00	100.00
Surprise	97.83	87.50	98.81	87.50	86.31
Average	99.05	90.03	99.22	91.47	91.28

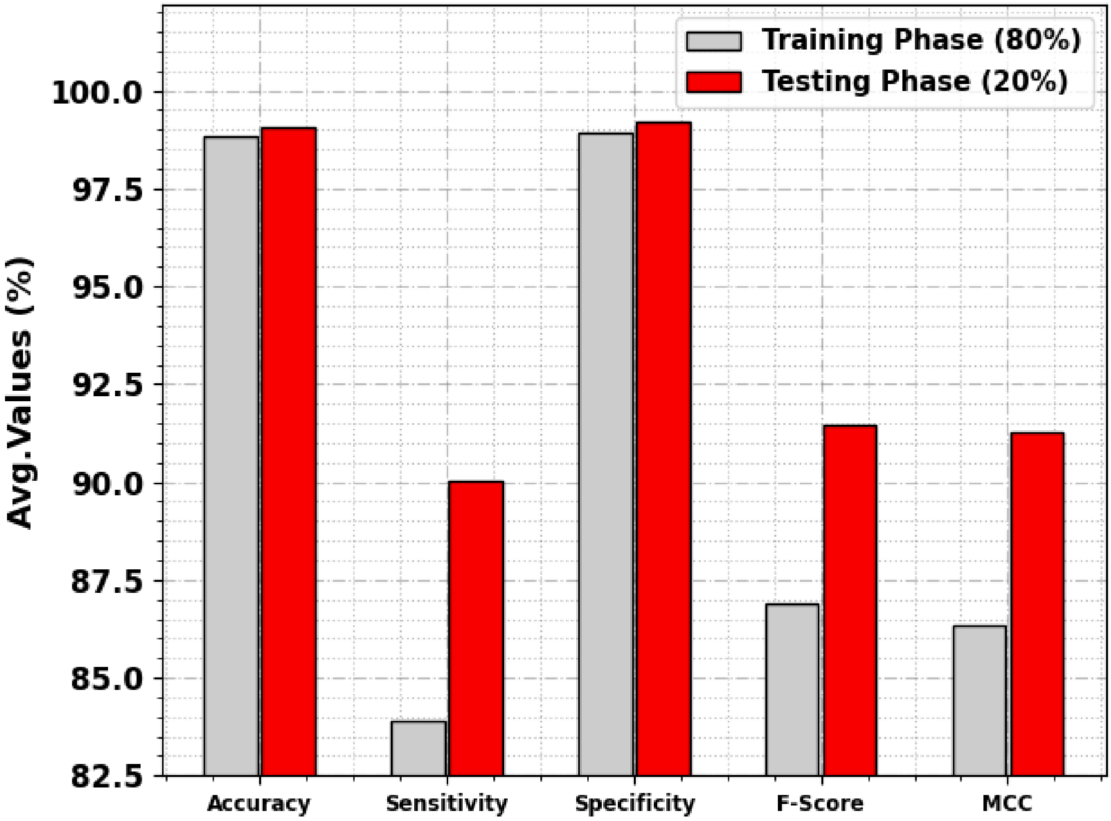


Figure 4. Average of the AFER-POADCNN algorithm at 80:20 of the TR phase/TS phase.

Table 3. FER outcome of AFER-POADCNN algorithm at 70:30 of the TR phase/TS phase.

Classes	$Accu_y$	$Sens_y$	$Spec_y$	F_{Score}	**MCC**
			TR Phase (70%)		
Anger	99.22	96.88	99.35	92.54	92.23
Contempt	99.38	62.50	99.84	71.43	71.87
Disgust	98.91	87.18	99.67	90.67	90.17
Fear	99.07	68.42	100.00	81.25	82.32
Happy	99.38	95.83	99.66	95.83	95.50
Neutral	97.67	99.76	93.67	98.25	94.86
Sad	98.91	64.71	99.84	75.86	76.52
Surprise	98.76	93.10	99.32	93.10	92.42
Average	98.91	83.55	98.92	87.37	86.99
			TS Phase (30%)		
Anger	99.28	100.00	99.24	92.86	92.74
Contempt	99.28	80.00	100.00	88.89	89.11
Disgust	98.91	90.00	99.61	92.31	91.76
Fear	99.64	83.33	100.00	90.91	91.12
Happy	98.91	95.24	99.22	93.02	92.46
Neutral	97.46	98.82	95.28	97.96	94.64
Sad	98.91	81.82	99.62	85.71	85.26
Surprise	98.19	88.00	99.20	89.80	88.82
Average	98.82	89.65	99.02	91.43	90.74

To estimate the performance of the AFER-POADCNN algorithm at 80:20 of the TR phase/TS phase, TR and TS $accu_y$ curves are defined, as illustrated in Figure 6. The TR and TS $accu_y$ curves demonstrate the outcome of the AFER-POADCNN algorithm on various epochs. The figure offers meaningful details regarding the learning task and generalization capabilities of the AFER-POADCNN approach. With an enhancement in the epoch count, it is observed that the TR and TS $accu_y$ curves are enhanced. It is still experimental that the AFER-POADCNN algorithm attains higher testing accuracy, which has the capability in identifying the patterns in the TR and TS data.

Figure 7 reveals the overall TR and TS loss values of the AFER-POADCNN algorithm at 80:20 of the TR phase/TS phase over epochs. The TR loss displays that the model loss is lesser over epochs. Primarily, the loss values are reduced as the model modifies the weight to minimize the prediction error on the TR and TS data. The loss curves demonstrate the extent to which the model fits the training data. It is detected that the TR and TS loss is steadily decreased, and represents that the AFER-POADCNN approach effectually learns the patterns displayed in the TR and TS data. It is also noticed that the AFER-POADCNN algorithm fine-tunes the parameters for decreasing the discrepancy between the prediction and the original training label.

The precision–recall (PR) outcome of the AFER-POADCNN approach at 80:20 of the TR phase/TS phase is represented by plotting the precision against the recall, as defined in Figure 8. The outcomes confirm that the AFER-POADCNN algorithm reaches higher PR performances under all classes. The outcome exhibits that the model learns to identify distinct classes. The AFER-POADCNN algorithm reaches improved solutions in the recognition of positive instances, with minimal false positives.

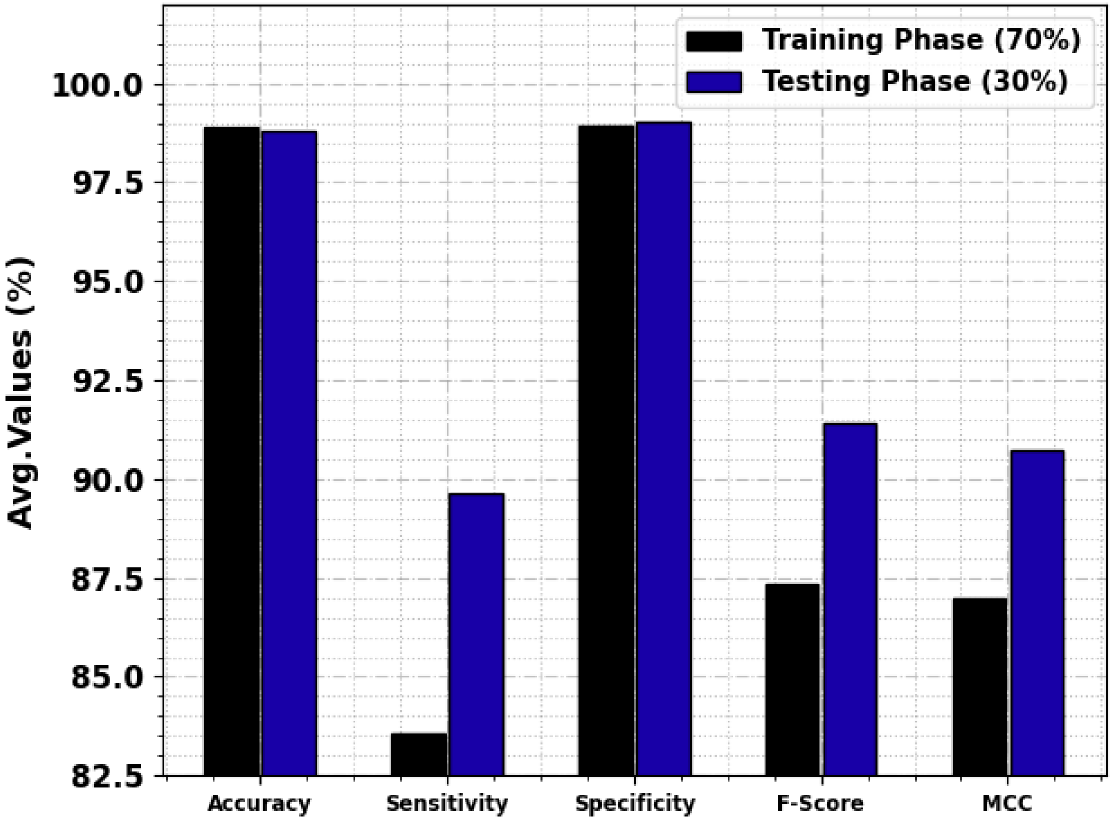


Figure 5. Average of the AFER-POADCNN algorithm at 70:30 of the TR phase/TS phase.

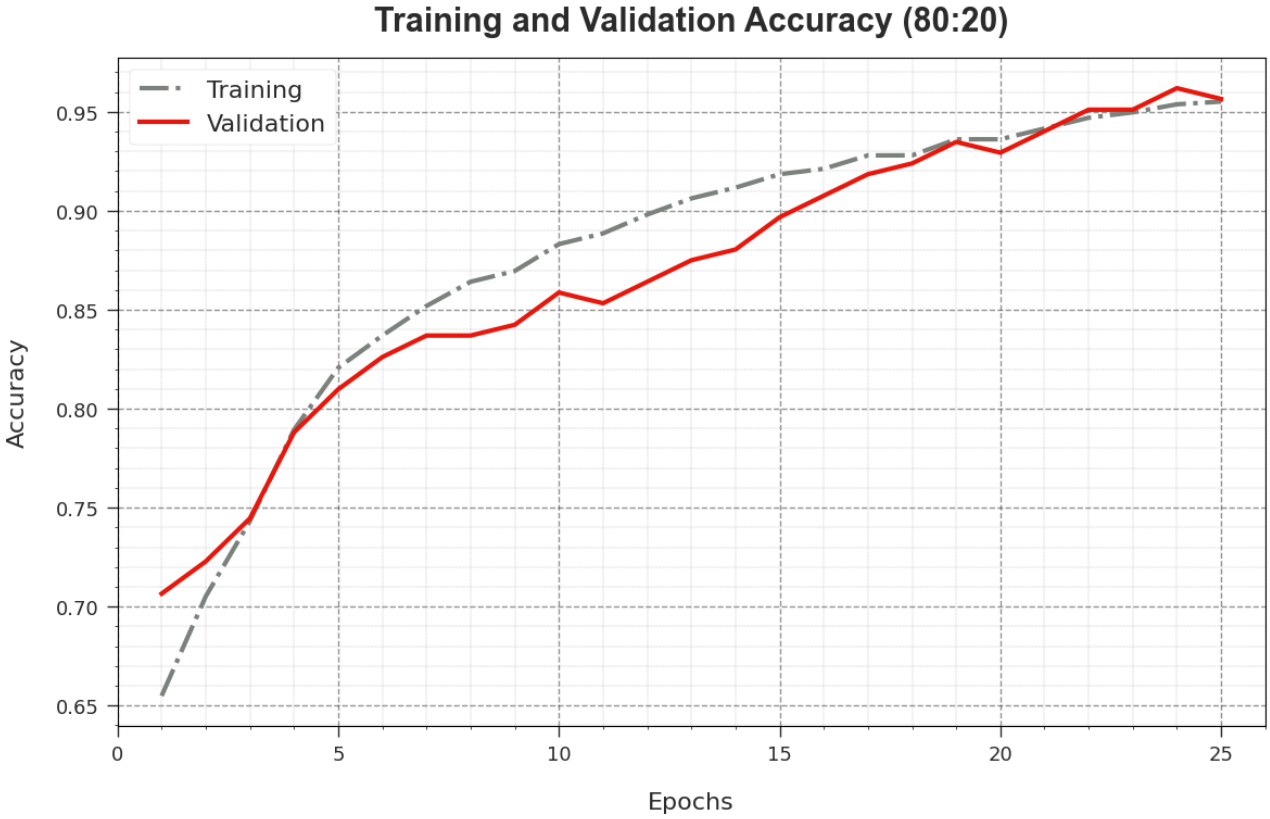


Figure 6. $Accu_y$ curve of the AFER-POADCNN algorithm at 80:20 of the TR phase/TS phase.

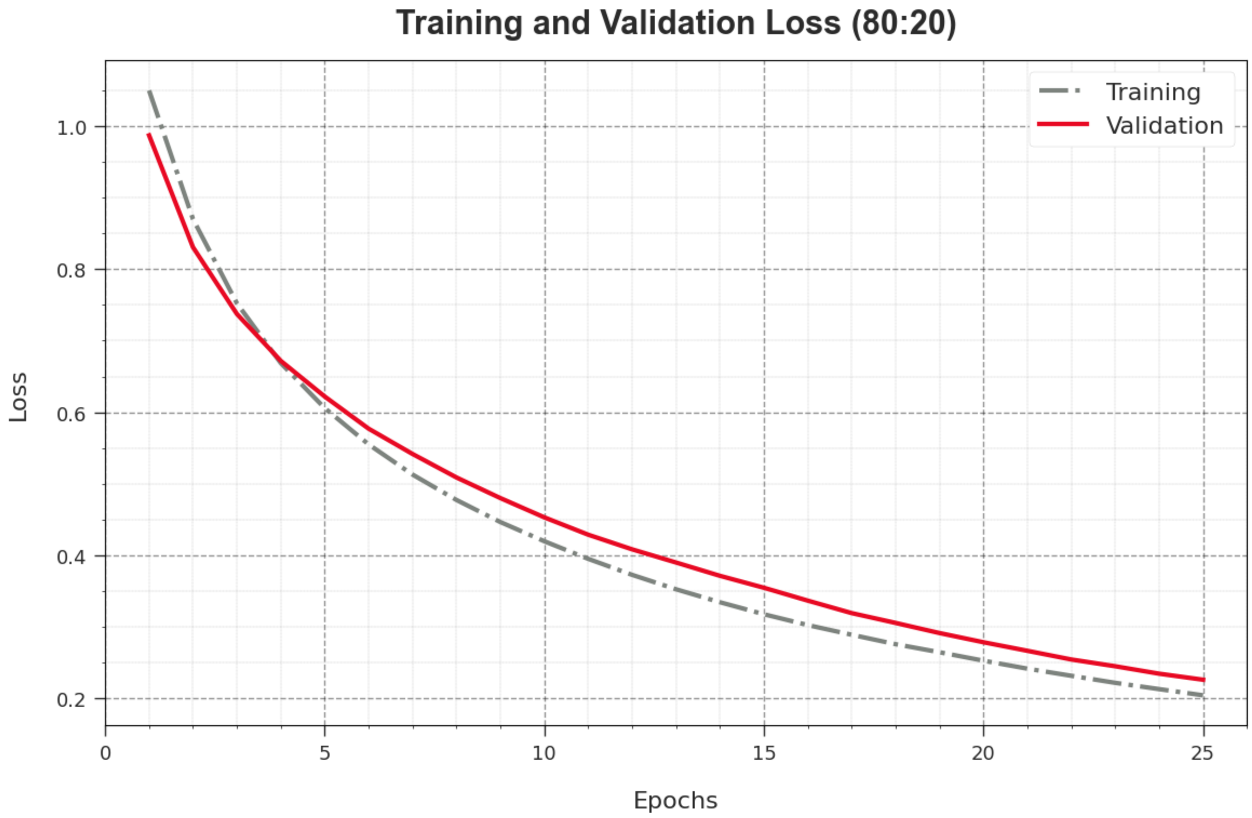


Figure 7. Loss curve of the AFER-POADCNN algorithm at 80:20 of the TR phase/TS phase.

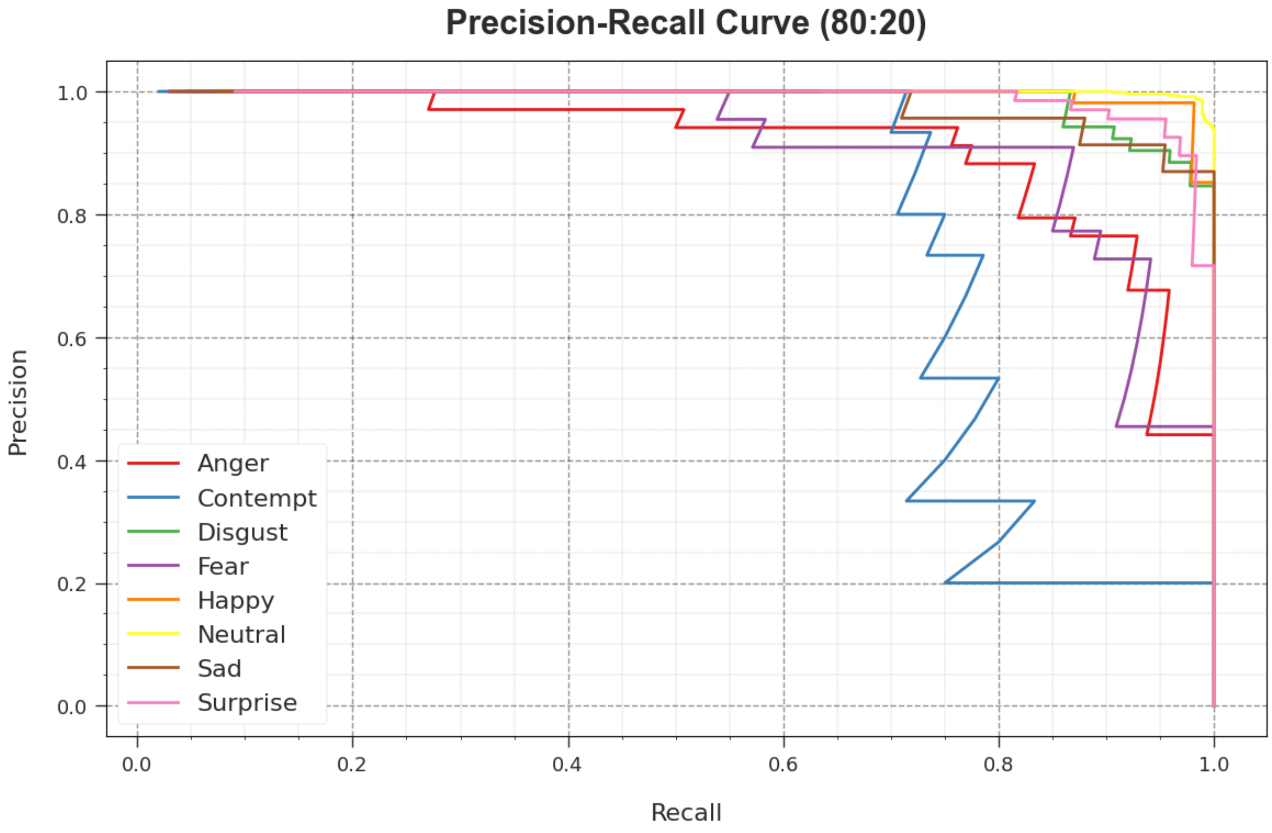


Figure 8. PR curve of the AFER-POADCNN algorithm at 80:20 of the TR phase/TS phase.

The ROC curves offered by the AFER-POADCNN algorithm at 80:20 of the TR phase/TS phase are illustrated in Figure 9, which has the capability of discriminating the class labels. The outcome implies appreciated insights into the trade-offs among the TPR and FPR rates, with various classifier thresholds and distinct counts of epochs. It defines the correct predictive outcome of the AFER-POADCNN approach on the classifier of various classes.

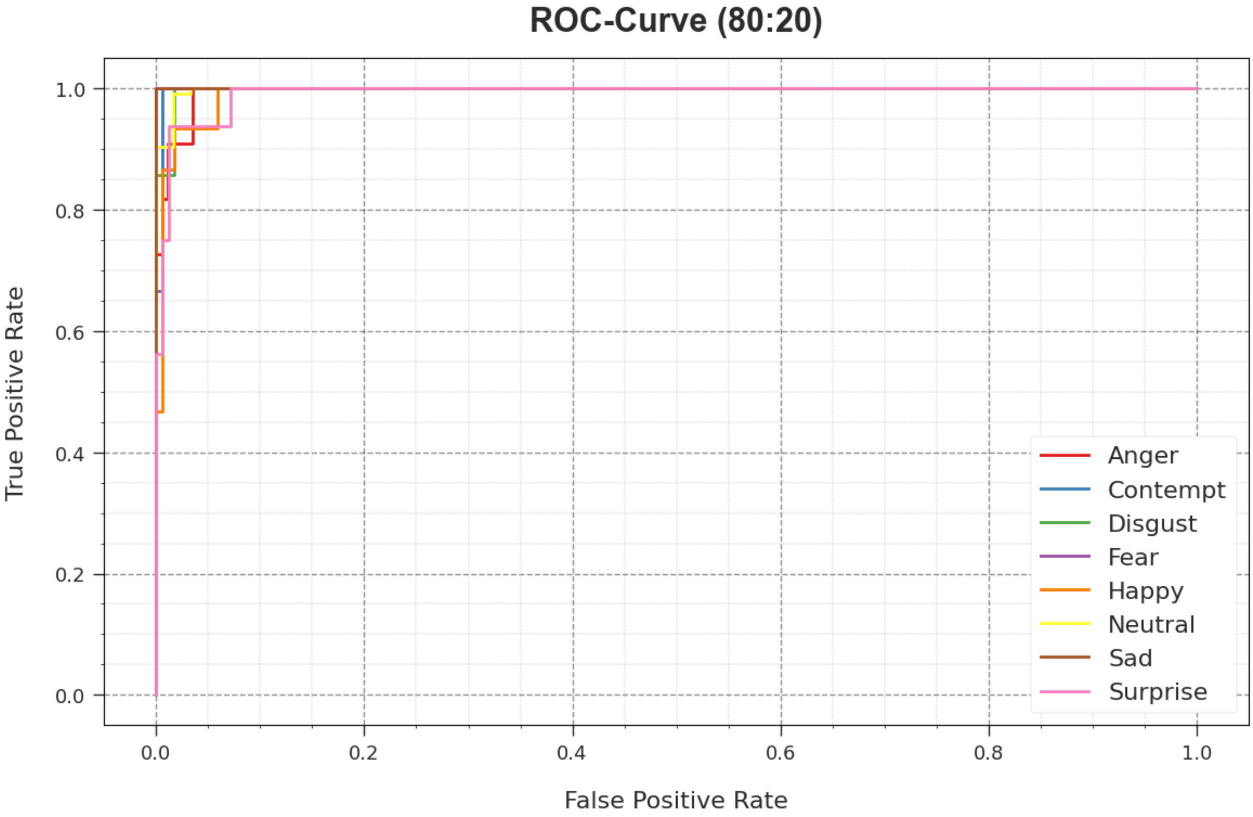


Figure 9. ROC curve of the AFER-POADCNN algorithm at 80:20 of the sTR phase/TS phase.

The comparison study of the AFER-POADCNN technique with recent systems is given in Table 4 [31]. Figure 10 represents a comparative $accu_y$ and F_{score} examination of the AFER-POADCNN technique with recent models. The results imply that the AFER-POADCNN approach gains better results. Based on the $accu_y$, the AFER-POADCNN technique offers a higher $accu_y$ of 99.05%, while the HGSO-DLFER, ResNet50, SVM, MobileNet, Inception-v3, and CNN-VGG19 models obtain reduced $accu_y$ values of 98.45%, 88.54%, 91.64%, 92.32%, 93.74%, and 94.035%, respectively. Additionally, based on the F_{score}, the AFER-POADCNN system achieves an enhanced F_{score} of 91.47%, while the HGSO-DLFER, ResNet50, SVM, MobileNet, Inception-v3, and CNN-VGG19 approaches reach decreased F_{score} values of 87.78%, 85.99%, 87.55%, 86.52%, 73.82%, and 81.75%, correspondingly.

Table 4. Comparative outcome of the AFER-POADCNN system with recent methods [31].

Methods	$Sens_y$	$Spec_y$	$Accu_y$	F_{score}
AFER-POADCNN	90.03	99.22	99.05	91.47
HGSO-DLFER	84.99	98.65	98.45	87.78
ResNet50	83.96	83.65	88.54	85.99
SVM	83.17	82.18	91.64	87.55

Table 4. *Cont.*

Methods	$Sens_y$	$Spec_y$	$Accu_y$	F_{score}
MobileNet	83.74	83.81	92.32	86.52
Inception-V3	80.23	84.06	93.74	73.82
CNN-VGG19	82.95	81.59	94.03	81.75

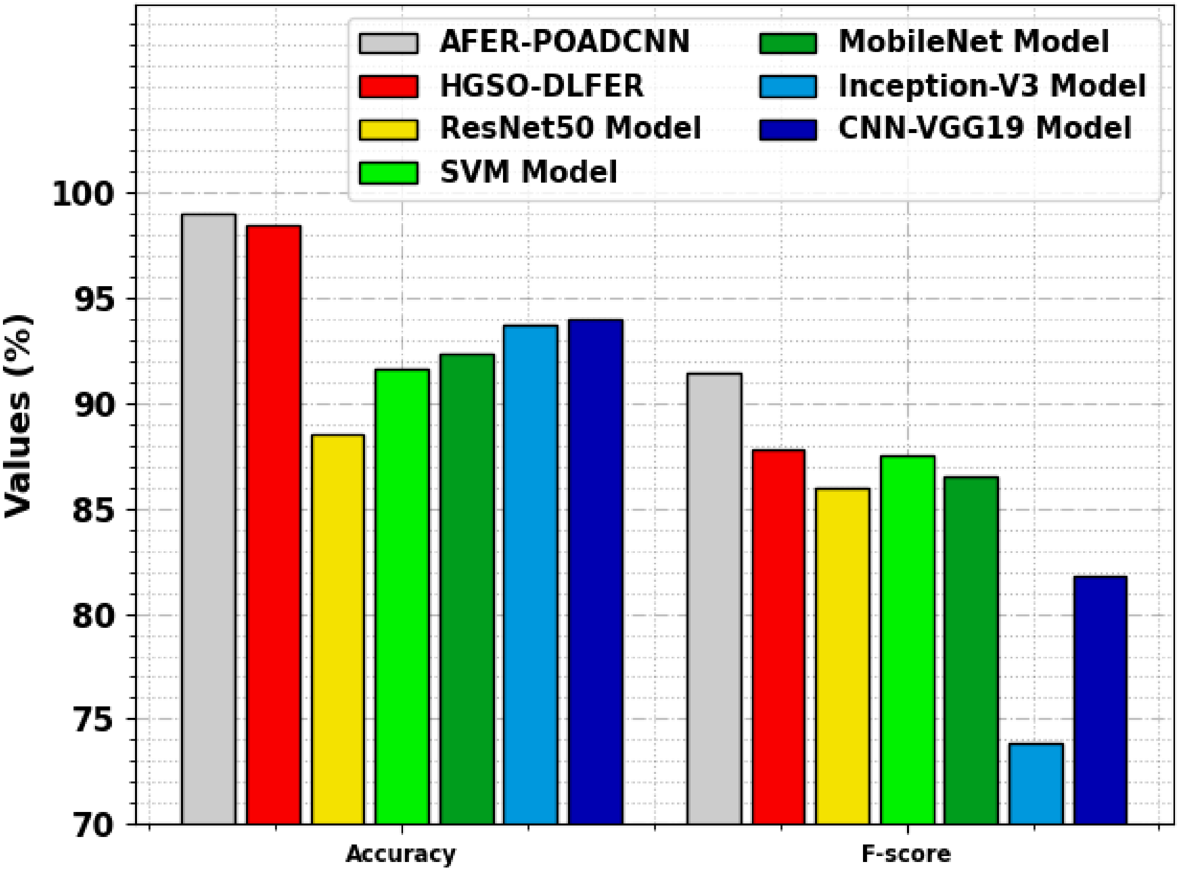


Figure 10. $Accu_y$ and F_{score} outcomes of the AFER-POADCNN approach with recent methods.

Figure 11 signifies a comparative $sens_y$ and $spec_y$ inspection of the AFER-POADCNN system with recent algorithms. The outcomes inferred that the AFER-POADCNN algorithm attains optimum solutions. Based on $sens_y$, the AFER-POADCNN algorithm achieves a superior $sens_y$ of 90.03%, while the HGSO-DLFER, ResNet50, SVM, MobileNet, Inception-v3, and CNN-VGG19 algorithms achieve lesser $sens_y$ values of 84.99%, 83.96%, 83.17%, 83.74%, 80.23%, and 82.95%, correspondingly. Furthermore, based on $spec_y$, the AFER-POADCNN system offers an enhanced $spec_y$ of 99.22%, while the HGSO-DLFER, ResNet50, SVM, MobileNet, Inception-v3, and CNN-VGG19 systems gain minimal $spec_y$ values of 98.65%, 83.65%, 82.18%, 83.81%, 84.06%, and 81.59%, correspondingly. These performances confirmed the higher outcome of the AFER-POADCNN algorithm.

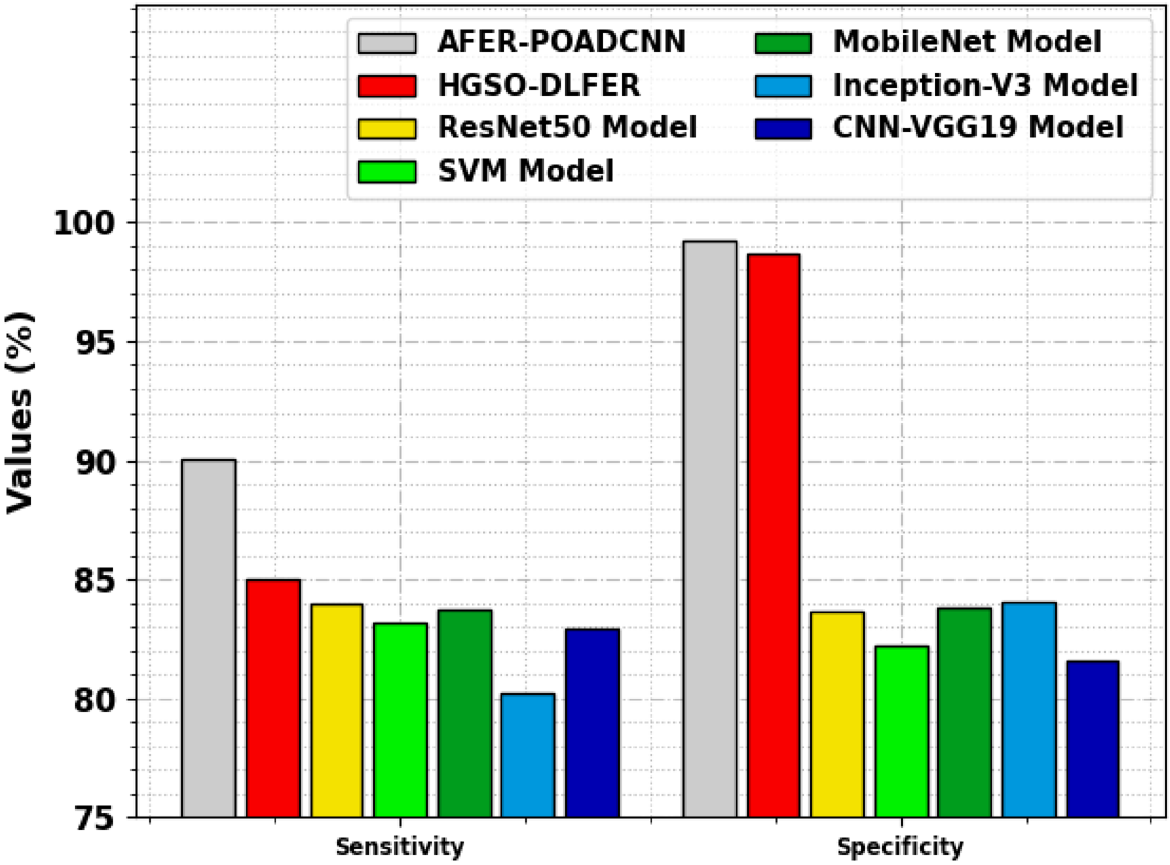


Figure 11. $Sens_y$ and $Spec_y$ outcomes of the AFER-POADCNN approach with recent methods.

5. Conclusions

In this study, we have designed a novel AFER-POADCNN model for automated and accurate FER. The primary objective of the AFER-POADCNN model lies in the automatic recognition and classification of facial emotions. To accomplish this, the AFER-POADCNN method comprises MF-based preprocessing, a CapsNet feature extractor, POA-based hyperparameter tuning, and BiLSTM-based classification for FER. Finally, the detection and classification of different kinds of facial emotions take place using the BiLSTM network. The simulation analysis of the AFER-POADCNN algorithm can be tested on benchmark FER databases. The stimulation values demonstrated the better performance of the AFER-POADCNN model over existing techniques, with a maximum accuracy of 99.05%. Further enhancements can include expanding the model's robustness to varying lighting conditions, facial expressions, and head poses. Additionally, exploring real-time applications and the integration of multimodal data, such as audio and text analysis, can pave the way for a more comprehensive understanding of human emotions in a wide range of contexts. Future work can investigate the computational efficiency of the proposed model.

Author Contributions: Conceptualization, M.A. (Mohammed Alonazi) and H.J.A.; Methodology, H.J.A., F.A.A., M.M. and A.S.; Software, M.M.; Validation, F.A.A. and M.M.; Resources, F.A.A.; Data curation, M.A. (Mohammed Alonazi); Writing—original draft, M.A. (Mohammed Alonazi), H.J.A., M.A. (Mohammed Alghamdi) and A.S.; Writing—review & editing, M.A. (Mohammed Alonazi), H.J.A., F.A.A., M.M., M.A. (Mohammed Alghamdi) and A.S.; Visualization, F.A.A. and M.M.; Project administration, M.A. (Mohammed Alonazi); Funding acquisition, M.A. (Mohammed Alonazi). All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through large group Research Project under grant number (RGP2/235/44). Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R281), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. Research Supporting Project number (RSPD2023R838), King Saud University, Riyadh, Saudi Arabia. This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2023/R/1444). This study is partially funded by the Future University in Egypt (FUE).

Data Availability Statement: Data sharing is not applicable to this article, as no datasets were generated during the current study.

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

References

1. Mukhiddinov, M.; Djuraev, O.; Akhmedov, F.; Mukhamadiyev, A.; Cho, J. Masked Face Emotion Recognition Based on Facial Landmarks and Deep Learning Approaches for Visually Impaired People. *Sensors* **2023**, *23*, 1080. [CrossRef]
2. Gupta, S.; Kumar, P.; Tekchandani, R.K. Facial emotion recognition based real-time learner engagement detection system in online learning context using deep learning models. *Multimed. Tools Appl.* **2023**, *82*, 11365–11394. [CrossRef]
3. Poulose, A.; Reddy, C.S.; Kim, J.H.; Han, D.S. Foreground Extraction Based Facial Emotion Recognition Using Deep Learning Xception Model. In *2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN), Jeju Island, Republic of Korea, 17–20 August 2021*; IEEE: New York, NY, USA, 2021; pp. 356–360.
4. Gaddam, D.K.R.; Ansari, M.D.; Vuppala, S.; Gunjan, V.K.; Sati, M.M. Human facial emotion detection using deep learning. In *Lecture Notes in Electrical Engineering, Proceedings of the ICDSMLA 2020: 2nd International Conference on Data Science, Machine Learning and Applications, Pune, India, 21–22 November 2020*; Springer: Singapore, 2021; pp. 1417–1427.
5. Hossain, S.; Umer, S.; Rout, R.K.; Tanveer, M. Fine-grained image analysis for facial expression recognition using deep convolutional neural networks with bilinear pooling. *Appl. Soft Comput.* **2023**, *134*, 109997. [CrossRef]
6. Chaudhari, A.; Bhatt, C.; Krishna, A.; Travieso-González, C.M. Facial emotion recognition with inter-modality-attention-transformer-based self-supervised learning. *Electronics* **2023**, *12*, 288. [CrossRef]
7. Sabour, S.; Frosst, N.; Hinton, G.E. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems 30, Proceedings of the Annual Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017*; Neural Information Processing Systems Foundation, Inc. (NeurIPS): San Diego, CA, USA, 2017; pp. 3856–3866.
8. Chen, H.; Wang, T.; Chen, T.; Deng, W. Hyperspectral image classification based on fusing S3-PCA, 2D-SSA and random patch network. *Remote Sens.* **2023**, *15*, 3402. [CrossRef]
9. Duan, Z.; Song, P.; Yang, C.; Deng, L.; Jiang, Y.; Deng, F.; Jiang, X.; Chen, Y.; Yang, G.; Ma, Y.; et al. The impact of hyperglycaemic crisis episodes on long-term outcomes for inpatients presenting with acute organ injury: A prospective, multicentre follow-up study. *Front. Endocrinol.* **2022**, *13*, 1057089. [CrossRef]
10. Bharti, S.K.; Varadhaganapathy, S.; Gupta, R.K.; Shukla, P.K.; Bouye, M.; Hingaa, S.K.; Mahmoud, A. Text-Based Emotion Recognition Using Deep Learning Approach. *Comput. Intell. Neurosci.* **2022**, *2022*, 2645381. [CrossRef]
11. Lasri, I.; Riadsolh, A.; Elbelkacemi, M. Facial emotion recognition of deaf and hard-of-hearing students for engagement detection using deep learning. *Educ. Inf. Technol.* **2023**, *28*, 4069–4092. [CrossRef]
12. Khattak, A.; Asghar, M.Z.; Ali, M.; Batool, U. An efficient deep learning technique for facial emotion recognition. *Multimed. Tools Appl.* **2022**, *81*, 1649–1683. [CrossRef]
13. Durga, B.K.; Rajesh, V.; Jagannadham, S.; Kumar, P.S.; Rashed, A.N.Z.; Saikumar, K. Deep Learning-Based Micro Facial Expression Recognition Using an Adaptive Tiefes FCNN Model. *Trait. Signal* **2023**, *40*, 1035–1043. [CrossRef]
14. Arora, T.K.; Chaubey, P.K.; Raman, M.S.; Kumar, B.; Nagesh, Y.; Anjani, P.K.; Ahmed, H.M.S.; Hashmi, A.; Balamuralitharan, S.; Debtera, B. Optimal facial feature-based emotional recognition using a deep learning algorithm. *Comput. Intell. Neurosci.* **2022**, *2022*, 8379202.
15. Sarvakar, K.; Senkamalavalli, R.; Raghavendra, S.; Kumar, J.S.; Manjunath, R.; Jaiswal, S. Facial emotion recognition using convolutional neural networks. *Mater. Today Proc.* **2023**, *80*, 3560–3564. [CrossRef]
16. Said, Y.; Barr, M. Human emotion recognition based on facial expressions via deep learning on high-resolution images. *Multimed. Tools Appl.* **2021**, *80*, 25241–25253. [CrossRef]
17. Umer, S.; Rout, R.K.; Pero, C.; Nappi, M. Facial expression recognition with trade-offs between data augmentation and deep learning features. *J. Ambient Intell. Humaniz. Comput.* **2022**, *13*, 721–735. [CrossRef]
18. Talaat, F.M. Real-time facial emotion recognition system among children with autism based on deep learning and IoT. *Neural Comput. Appl.* **2023**, *35*, 12717–12728. [CrossRef]
19. Chowdary, M.K.; Nguyen, T.N.; Hemanth, D.J. Deep learning-based facial emotion recognition for human–computer interaction applications. *Neural Comput. Appl.* **2023**, *35*, 23311–23328. [CrossRef]
20. Saeed, S.; Shah, A.A.; Ehsan, M.K.; Amirzada, M.R.; Mahmood, A.; Mezgebo, T. Automated facial expression recognition framework using deep learning. *J. Healthc. Eng.* **2022**, *2022*, 5707930. [CrossRef]

21. Sikkandar, H.; Thiyagarajan, R. Deep learning-based facial expression recognition using improved Cat Swarm Optimization. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 3037–3053. [CrossRef]
22. Helaly, R.; Messaoud, S.; Bouaafia, S.; Hajjaji, M.A.; Mtibaa, A. DTL-I-ResNet18: Facial emotion recognition based on deep transfer learning and improved ResNet18. *Signal Image Video Process.* **2023**, *17*, 2731–2744. [CrossRef]
23. Thuseethan, S.; Rajasegarar, S.; Yearwood, J. Deep3DCANN: A Deep 3DCNN-ANN framework for spontaneous micro-expression recognition. *Inf. Sci.* **2023**, *630*, 341–355. [CrossRef]
24. Kansizoglou, I.; Bampis, L.; Gasteratos, A. An active learning paradigm for online audio-visual emotion recognition. *IEEE Trans. Affect. Comput.* **2019**, *13*, 756–768. [CrossRef]
25. Li, Y.; Gao, Y.; Chen, B.; Zhang, Z.; Lu, G.; Zhang, D. Self-supervised exclusive-inclusive interactive learning for multi-label facial expression recognition in the wild. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 3190–3202. [CrossRef]
26. Kansizoglou, I.; Misirlis, E.; Tsintotas, K.; Gasteratos, A. Continuous emotion recognition for long-term behaviour modelling through recurrent neural networks. *Technologies* **2022**, *10*, 59. [CrossRef]
27. Kumar, A.; Patel, V.K. Classification and identification of disease in potato leaf using hierarchical-based deep learning convolutional neural network. *Multimed. Tools Appl.* **2023**, *82*, 31101–31127. [CrossRef]
28. Guo, X.; Ghadimi, N. Optimal Design of the Proton-Exchange Membrane Fuel Cell Connected to the Network Utilizing an Improved Version of the Metaheuristic Algorithm. *Sustainability* **2023**, *15*, 13877. [CrossRef]
29. Nie, Q.; Wan, D.; Wang, R. CNN-BiLSTM water level prediction method with an attention mechanism. *J. Phys. Conf. Ser.* **2021**, *2078*, 012032. [CrossRef]
30. Available online: http://www.jeffcohn.net/Resources/ (accessed on 14 July 2023).
31. AlEisa, H.N.; Alrowais, F.; Negm, N.; Almalki, N.; Khalid, M.; Marzouk, R.; Alnfiai, M.M.; Mohammed, G.P.; Alneil, A.A. Henry Gas Solubility Optimization with Deep Learning Based Facial Emotion Recognition for Human-Computer Interface. *IEEE Access* **2023**, *11*, 62233–62241. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.