International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V74I1P125

Volume 74 Issue 1, 317-332, January 2026
© 2026 Seventh Sense Research Group®

Original Article

Facial Expression Recognition System Using a Hybrid CNN and LSTM Model

Sunny Bagga[1], Hemant Makwana[2]

[1]*Department of Computer Science & Engineering, Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore (M.P.), India.*
[2]*Department of Information Technology, Institute of Engineering & Technology, Devi Ahilya Vishwavidyalaya, Indore, India.*

[1]*Corresponding Author : ssunnybagga@gmail.com*

Received: 19 June 2025 Revised: 31 December 2025 Accepted: 06 January 2026 Published: 14 January 2026

Abstract *- Facial expression analysis represents one of the most challenging and exciting problems in the fields of machine applications and human–computer interaction. The recognition of facial expressions has long been an important domain of academic interest. While various identification processes behind the recognition activities are mainly based on the identification of emotional attributes, intra-class variation can considerably inhibit the identification of the same. It is observed that static imagery does not capture the facial traits quite well. In the given paper, the authors present a novel architecture based on neutral terms for categorizing seven basic expressions: happiness, anger, neutrality, fear, disgust, sadness, and Surprise. FER2013 is used to present comprehensive experimental results. The proposed design is based on CNN and LSTM. For improving the accuracy of the FER system, a hybrid approach involving CNN and LSTM is presented. This hybrid approach involves two steps: first, CNN learns on the FER2013 dataset for the extraction of visual features, after which LSTM is used in modeling the temporal dependencies between sequences of images and their corresponding emotions. The outputs of the architecture are evaluated by using a confusion matrix and are compared with other relevant architectures. Experimental results on publicly available datasets indicate that the method proposed in this work outperforms modern techniques.*

Keywords *- Convolution Neural Network (CNN), Deep Learning, Facial Expression Recognition, Long Short-Term Memory (LSTM).*

1. Introduction

A central research area in computer vision is Facial Expression Recognition (FER), which aims to interpret the complex language of human emotions expressed through facial movements. Facial expression recognition systems are based on psychological investigations. A Facial Expression Recognition System (FER) identifies emotion types based on the facial expressions in an image. Facial expression recognition has emerged from the studies of psychology. In the year 1872, Darwin first proposed the development of human expressions from components of facial expressions in animals and explained the relationship between humans and animals [1]. Since then, the approach for expression recognition began to emerge and is still an area of research today. In the 1970s, Ekman and Friesen classified human facial expressions into six types: fear, sadness, Surprise, happiness, anger, and disgust. In the contemporary era, rapid advancements in Artificial Intelligence and the development of "Deep Learning" technologies have increased the attention of experts and academia towards facial expression identification. Faces have immense significance in expressing emotions, engaging with people, and imagining oneself. For example, facial expressions are among the most sought-after methods of depicting human emotions. It can easily judge with a single look at people's facial expressions whether they are smiling or crying, surprised or showing anger or fear. On average, the face is an essential tool for communication and an invaluable source of psychological and intellectual information. Facial expression recognition is one of the most important approaches in helping a machine understand human emotion. Emotion recognition plays an important role in various areas, such as online learning, mental health assessment, and human-computer interaction. Facial expression recognition, also referred to as FER, is a crucial feature in different applications in health, such as pain assessment, mental disorder recognition, and the design of help robots that need a close connection between machines and humans [2].

Facial expressions are a means of non-verbal communication that assists in communication between people and correspond to specific information. It is critical to assess basic expressions like angry, sad, happy, fearful, and disgusted expressions. The fact that people are also capable of feeling complex emotions makes it essential to identify these expressions correctly. The technique of decoding human

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

emotions by interpreting the facial expressions of people is termed facial emotion recognition. The task of facial expression analysis to interpret human emotions is a challenging feat to achieve using computational algorithms. The same is now attainable by using the latest innovation in the field of computer vision and machine learning, which allows for the detection of emotions in images. A significant amount of research has been conducted in recent years on FER algorithms using machine learning. Different machine learning techniques were used to test the effectiveness of machine learning algorithms on the FER2013 dataset. Three machine learning algorithms were adopted for recognizing facial expressions, including logistic regression, random forest, and AdaBoost. The optimum value of the RF algorithm attained the highest level of precision after achieving an accuracy rate of 61%. For identifying human feelings like sadness, fear, happiness, and so on, a kernel function-based support vector machine classifier is used. The experimental results obtained illustrate that the method developed in this study possesses superiority over conventional methods in terms of being able to classify a wide range of expressions and achieve improved recognition rates [4].

Wang et al. [5] emphasize deep learning-based facial expression identification utilizing CNNs for educational state surveillance, showing how crucial CNNs are to real-time recognition systems. In the context of human behavior analysis, Friji et al. [6] introduced a geometric Deep Neural Network approach named KShapenet for 2D and 3D landmark-based human movement evaluation, which involves face expression recognition. To overcome the difficulties associated with facial expression recognition in uncontrolled settings, Xiao et al. [7] proposed a technique for FER in the wild that relies on CNNs and Graph Convolutional Networks (GCNs).

Baddar et al. [8] presented a solution for spontaneous facial expression forecasting that increased the precision of forecasting for subtle expressions at the start of a sequence. Guo et al. [9] investigated the importance of spatial-temporal data with respect to facial expression-driven cold pain and evaluated the performance of both the customized and generalized models. As illustrated, the customized spatial-temporal architecture performs better in estimating cold pain intensity. Rajasimman et al. [10] proposed a robust facial expression recognition system utilizing an evolutionary algorithm coupled with a deep learning framework. The Teaching and Learning-Based Optimization (TLBO) paradigm, in concert with Long Short-Term Memory (LSTM), is used to recognize and identify expressions. Mahayossanunt et al. [11] developed a depression detection system based on the combination of relevant face features from interview videos, such as radial glance angles and action unit intensity. Its model uses LSTM combined with the attention mechanism. It tries to combine some elements through the intermediary fusion methodology. Nowadays, deep learning, which is a machine learning technique, has been widely used for complex visuals and complex operations. Convolutional Neural Networks are the most widely used Deep Learning Models designed for image processing because their results are more efficient and accurate compared to traditional methods.

Current methods depend on manually developed techniques for feature extraction, including Local Binary Patterns (LBP) and Histogram of Oriented Gradients (HOG). However, these algorithms frequently encounter difficulties in generalizing across various datasets due to differences in head positions, lighting conditions, and obstructions. Managing an uneven dataset is one of the most challenging parts of applying classification models in practical settings. Because the model is less capable of identifying minority classes, traditional methods of training classification models with these kinds of datasets lead to biased training. Many current methods, like CNNs with attention mechanisms or graph-based structures, focus mainly on extracting spatial features from still images of faces. The changes in facial expressions over time in video frames are not taken into account. But temporal learning models, like those based on LSTM, typically incorporate small datasets or low-resolution parameters, which makes it harder for them to find certain spatial features.

Based on the insights derived from the previous research, this research constructs a hybrid CNN+LSTM-based facial expression recognition system using the FER-2013 dataset. Random Oversampling Technique is applied in this work to balance the dataset for addressing class imbalance problems in the FER2013 dataset. The combined CNN-LSTM technique integrates the advantages of convolutional and recurrent neural networks to present a new model for facial expression identification capable of capturing both temporal and spatial aspects of facial emotions. While the LSTM component simulates the temporal dynamics of facial motions across video frames, the CNN component captures spatial data from each frame comprehensively, ranging from facial contours and textures down to local features. This research's primary contributions are as follows:

- A random oversample layer is suggested to synthesize complex features to eliminate class imbalance in the recognition of facial expressions.
- A hybrid CNN-LSTM model is introduced, which improves emotion recognition accuracy by effectively learning spatial-temporal dependencies.
- The hybrid model outperforms several previous methods with a superior accuracy of 85.11% on the FER2013 dataset.

2. Literature Review

Computers are an essential component of daily life. Better human-computer communication requires human-like relationships. Using a range of sources, emotional recognition

and categorization have captured the interest of numerous scholars. CNLSTM, a method that integrates machine learning and deep learning, was proposed to identify emotions in video expressions [12].

To analyze the video data and identify emotions, a hybrid technique is used. Once CNN features have been extracted from relevant frames, the LSTM is used to examine the video's temporal dynamics. Aff-wild2, JAFFE, and CK+ are the datasets used for verifying the suggested method. Using the Aff-wild2 dataset as a test dataset, researchers developed a neural network model for recognizing emotions that included a CNN for feature extraction and an RNN for investigating temporal dynamics. From the results found in the research, the proposed method can achieve maximum recognition accuracy on the angry mode of the CK+ dataset. Additionally, it is notable that the proposed model performs better on imbalanced data.

Artificial intelligence methods, such as Convolutional Networks, have been used for emotion identification. However, irrespective of this fact, from the perspective of Complexity and computational requirements, this approach is vastly expensive. A lightweight CNN-based model (CLCM), based on MobileNetV2 architecture that recognizes facial emotions, is suggested by M. C. Gursesli et al. [13] to mitigate this issue. Seven facial emotions were identified using four publicly available datasets: FER-2013, CK+, AffectNet, and RAF-DB, which were employed to assess the effectiveness. The renowned MobileNetV2 and ShuffleNetV2 architectures were contrasted with the CLCM model. The CLCM evaluation outcomes were on par with or superior to those of the more intricate models. In several datasets, CLCM outperformed models such as DenseNet-121, ResNet-50, EfficientNet-B0, and Inception V3, demonstrating higher precision than some of the models found in the research literature. With efficiency varying from 54% to 84%, the CLCM model outperformed ShuffleNetV2 and MobileNetV2 in several datasets.

J.V. Vardhan et al. [14] developed a method to recognize emotion by facial expressions and speech using deep learning. This investigation offered two methods for emotion recognition: one approach based on facial expressions and another that relies on speech, considering the mutually beneficial characteristics of emotional elements of speech and facial expressions. CNN is used for learning facial emotions, while LSTM is used for learning speech emotion features. This study covers the use of CNNs for image recognition and LSTM for sequence modeling tasks, emphasizing how well these models can capture complex interactions and structures. The features of an image in a CNN are obtained by moving through the series of layers intended to analyze the input image. The RAVDESS dataset, which includes audio data, is used to train the LSTM model. The model is employed to extract the important characteristics from it. They identify the emotion by using distinct speech and facial expression characteristics. Y. Luo et al. [15] suggest an enhanced CNN model that includes the following main processes: initial processing of face images, retrieval of features from them, training of input test samples, collection of test sample attributes, categorization of face image characteristics, regeneration of images depicting facial expressions, and classification outcomes. An enhanced image feature classification approach was used to perform competitive tests, using the FGD Fractional Gradient descent algorithm, SGD stochastic gradient descent algorithm, and MGD mini-batch gradient descent algorithm. This research examines the architecture of CNN models, the importance of neurons, the excitation function of the convolution layer, and the loss function of the classification layer to propose a more effective CNN (Convolutional Neural Network) Model.

N. T. Singh et al. [16] suggested a comparative analysis of traditional machine learning and deep learning techniques for facial expression recognition. This investigation offers a comparison of the two techniques and a comprehensive analysis of FER methodologies. The Author categorized these techniques into two broad groups: deep learning-based techniques and traditional machine learning-based techniques. Traditional machine learning algorithms consist of three steps: face identification, feature extraction, and expression classification using these features. Deep learning-based techniques, comparable to conventional machine learning tactics, require additional processing power and storage space to train and validate the assumptions. Therefore, when employing deep learning algorithms for prediction, it is imperative to minimize computation time. A collection of Deep Neural Network models for identifying emotions from visual input was given by L.N. Do et al. [17].

The study of the facial regions extracted from the video sequence was the primary objective of the proposed models. Based on CNN, three facial renderings were created. The first model utilized a 3DCNN to analyze sequential data, after extracting facial characteristics from each video frame using a multilayer CNN. The VGG-FACE and LSTM modules were combined to create the second model. The third model empirically integrated the features and fine-tuned the Xception network. To integrate these three models, they examined four fusion techniques: weight fusion techniques, average scoring, maximal voting, and feature fusion. Usually, the weight fusion approach performed the best.

The primary finding of this research is the development and training of deep neural networks that excel at identifying emotions from visual data collected in two scenarios: unconstrained, where the data are collected in natural settings, and constrained, where the data are collected in a controlled environment, such as an indoor space. Using facial expression images as the research object, this article proposes a facial

expression detection approach based on a Long Short-Term Memory (LSTM) network using the Facial Expression Recognition (FER) 2013 dataset [18]. This work suggests a revolutionary CNN and LSTM-based facial emotion recognition technique. First, higher-quality data collection is achieved through preprocessing techniques such as data cleaning, data regularization, feature extraction, data splitting, data augmentation, label encoding, and batch processing.

In terms of procedures and organization, this study uses a feature pyramid design to combine feature maps at various levels to determine the face's location, which enhances the expressive capability of the model. LSTM and CNN networks are utilized concurrently to extract information from feature maps, which improves the model's performance. Experiments show that the approach in this work performs well on the FER2013 data set, particularly in recognizing the happy and sad faces with an outstanding recognition rate, but not the neutral, disgust, or surprise expressions. M. M.

Kabir et al. [19] proposed a system for facial expression recognition using a CNN-LSTM approach. For automatic facial expression identification from staged and spontaneous photos, the study provides a CNN-LSTM architecture called the PNFE dataset. The LSTM networks are arranged after the convolutional layers in the suggested architecture. Six well-known FER datasets are also used to test the suggested architecture: EmotioNet, CK+, AffectNet, Multi-PIE, CIFE, and ExpW.

The PNFE dataset is used to evaluate the outcomes that were achieved. The results of each test show that, when employing the regular and candid image collection, the suggested architecture outperforms the state-of-the-art techniques for face emotion recognition. The research utilizes assessment units constructed around confusion matrices.

The suggested architecture is evaluated and compared using precision, accuracy, and recall. S. Das et al. [20] designed a framework structure using IoT for automatic detection of human emotion. Based on this, the current study proposes a development on emotion detection using ECG and GSR sensors as two efficient mechanisms for reliable real-time identification of the emotional state of a person. The key objective here is to design a framework that would enable professionals to help people understand and control their emotions and simultaneously assist specialists in many fields of activity, including mental health.

The system has been proposed using the Internet of Medical Things paradigm, employing machine learning classification methods. Random Forest and K-Nearest Neighbor performed very well, reaching the highest values in recall, accuracy, precision, and F1-score, while Gaussian Naive Bayes and Support Vector Machine showed competitive but comparatively worse performances.

A. Hindu and B. Bhowmik [21] have created a deep learning framework powered by IoT that can identify different stress levels as well as emotions felt by an individual. The structure evaluates people's mental states efficiently, considering their conditions for various reasons. Additionally, the model can efficiently identify stress at an early stage and keep the individual from depressive symptoms. An IoT-enabled, discreet, real-time surveillance system is created to analyze recordings of a person's facial expressions to determine their emotional states. The suggested method determines the distinct emotions present in every video frame and determines the degree of stress at the sequence stage.

3. Methodology

3.1. FER2013 Dataset

Figure 1 represents the sample images of the FER2013 dataset. A popular standard dataset for Facial Expression Recognition (FER) studies is FER2013. It is made up of images of faces that have been assigned to various emotion categories. The 35,887 grayscale images in the FER2013 collection have a resolution of 48 by 48 pixels [22]. Three subsets of the dataset are separated: the training, private, and public test sets. The seven types of emotions that are classified on the facial images in the FER2013 dataset are: anger, disgust, fear, happiness, neutral, sad, and Surprise. The FER2013 dataset has a slightly even image distribution throughout the emotion classes. There is a tiny percentage of samples in the dataset that are linked to the "disgust" category, while roughly 25.7% of samples are related to the "happy" category [23].

The FER2013 dataset has certain restrictions, despite offering a sizable and varied selection of facial expressions. The grayscale and low resolution (48x48 pixels) of the images might make it more difficult for the model to catch subtle facial characteristics. Disgust is one of the emotions that is underestimated in the dataset in comparison to other emotions, which could cause problems with disparities in class. This dataset contains a variety of faces that are automatically registered, guaranteeing the same space in every image [24]. The primary classification of the images in this dataset is emotion. The images in FER vary more than those in any other datasets, with features like poor contrast, spectacles, and facial occlusion present [25].

FER2013 was collected under real-world conditions, exhibiting a high degree of variation in lighting conditions, pose, occlusion, distortion, and intensity of expression. The diversity of expressions, lighting conditions, and postures makes it an extremely challenging yet realistic database for training Deep Learning Models. Upon analyzing the FER13 database, the first observation is the diversity of expressions it contains. This diversity is an added advantage while modeling machines that can identify expressions from facial features. The FER13 database is a highly valuable resource for anyone working in the field of facial expression recognition.

Fig. 1 Examples of images extracted from the FER2013 dataset

3.2. Data Preprocessing

The images fed to the FER model are noisy and could experience variations in illumination, scale, and color. Before feeding the raw images to the deep learning model, specific preprocessing methods are incorporated, generally termed facial image recognition. Data preprocessing aims to improve the quality of the data and extract the required information essential for the effective recognition of facial images [26]. During the preparation of the data, the dataset used is optimized with a focus on the general algorithms to ensure effective results [27]. The preprocessing methods incorporated in this study include the conversion of the images to greyscale, normalization of the images, and scaling of the images [28]. The preprocessing methods used are effective in improving the input images and the extraction of the required information essential for the growth of accuracy in facial recognition systems. The preprocessing methods used ensure the reliability of facial recognition systems, which could experience variations in illumination and obstruction.

3.2.1. Resize Image

Image resizing aims to reduce the size of the data, thereby increasing the speed of processing. The resizing scale changes randomly between 0.1 and 0.9, thus making the dimensions of each image different. Resizing is performed to convert all input images to a 48 × 48 dimension. Redundant elements are cropped out of the image. This reduces the amount of memory used, thereby speeding up computation [29].

3.2.2. Image Normalization

By transforming data to a similar magnitude, data normalization improves consensus and model accuracy. By restricting any feature from controlling the mode, it ensures that every input characteristic contributes effectively to the learning procedure. The pixel values of the face images should be standardized when training the model to ensure consistency and enhance convergence. It is used to ensure that the values of the pixels in images are kept within a specific range. The aim is to speed up the process of modelling during training and provide a way for improvements.

3.2.3. Data Augmentation

Data augmentation method increases the basic data by producing more diverse samples, thereby causing a tremendous increase in the accuracy of facial expression recognition. Data augmentation involves an artificial increase in the size of the training dataset using various transformations of the basic images, such as resizing, flipping, rotating, and splitting them.

3.2.4. Oversampling

In machine learning or deep learning, oversampling is a strategy used to deal with datasets with discrepancies, which include unbalanced classes in comparison to other classes. By enhancing the number of cases in the minority classes, oversampling attempts to equalize the class composition. To attain the appropriate balance, samples from the minority class are randomly duplicated using random oversampling. In complicated class situations, oversampling can aid in improving the classifier's decision limits and help it discriminate between various classes more accurately.

3.3. Convolution Neural Network (CNN)

Convolutional Neural Networks are among the best deep learning architectures for dealing with spatial data and image analysis tasks. Convolutional Neural Networks (CNNs) leverage the spatial hierarchy characteristic of images through convolutional layers to identify patterns, boundaries, and textures, as opposed to regular neural networks, which analyze data uniformly. Normally, Convolutional Neural Networks (CNNs) automatically extract significant features through a hierarchical modeling process. In this type of neural network, the convolution layer is paired with the pooling layer. Although the convolution layer compresses the data into a smaller space to identify distinct characteristics, the pooling layer chooses the most pertinent details within a limited region [30]. In contrast to other image classification methods, CNNs do not require extensive preprocessing to learn different filters and properties.

Deep learning technique in expression recognition is mainly dependent on Convolutional Neural Networks (CNN). CNN is a feed-forward neural network [1]. Traditional CNN-based expression recognition methods have two essential steps: feature extraction, where the CNN learns image features through various layer approaches, and classification, where the obtained characteristics are input into the fully connected layer, resulting in the final expression label through the output layer. There is only one operation needed, the convolution. Therefore, it is known as a "Convolutional Neural Network" [31]. Convolution is distinct in that it produces identifiable features inside an image [32]. The primary benefit of CNN over its predecessors is that it can extract relevant features autonomously without requiring any human intervention. The main advantage of CNN over previous networks is its ability to detect relevant features automatically. Figure 2 represents the basic architecture of a CNN. The primary layer attributes and functionalities of CNN are listed below.

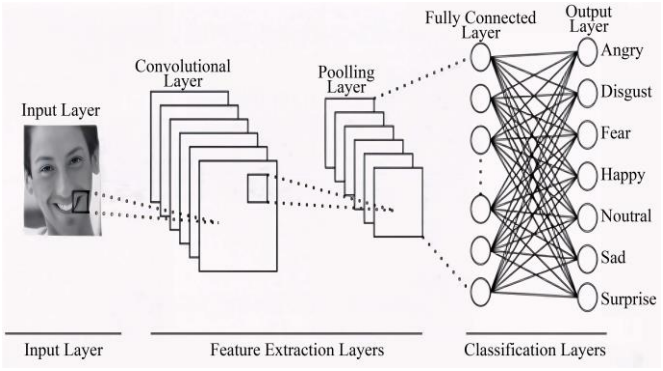

Fig. 2 Basic architecture of CNN [33]

3.3.1. Convolutional Layers

One of the fundamental layers of a CNN is the convolutional layer, which is used to extract features from images. These layers are made up of learnable filters, also known as kernels, which conduct convolution processes on the input picture by sliding over it. Every convolutional computation produces an output known as a feature map, which shows where a specific feature exists in the input image at various spatial positions. As the outcome values are linear, an activation function is used to generate a non-linear expression from the linear data. Convolutional layers have the following operation formula [1]:

$$G_j = f\,(G_{j-1} * \omega_j + b_j) \quad (1)$$

$G_{j:}$ symbolizes the j^{th} convolution layer; f: symbolizes the activation function after convolution; $\omega_{j:}$ symbolizes the weight attributes of the j^{th} convolution layer; $b_{j:}$ symbolizes the bias term.

3.3.2. Activation Function

Following each convolution process, each feature map component is activated nonlinearly by the activation function. Activation function gives the network nonlinearity, which helps it recognize complicated structures and connections in the data. The Rectified Linear Unit (ReLU) activation function is used in this research. The formula of the ReLU activation function is given below.

$$f(x) = \max(0, x) \quad (2)$$

3.3.3. Pooling Layers

Reduction of feature maps is achieved by pooling layers, preserving key features but decreasing their spatial dimensions. A static-size window is usually employed to move across the feature map in the pooling layer. A pooling function is then applied to every window to produce an entirely novel downsampled feature map. Average and maximum pooling are two types of pooling operations. Using max pooling, the maximum value of each window is calculated by keeping the most noticeable features. The outcome of average pooling is the average value of the attributes in each window, which can flatten out the feature map and lessen the likelihood of overfitting. The subsequent formula represents the max pooling function used in this work:

$$G_j \text{ MaxPooling } (G_{j-1}) \quad (3)$$

The j^{th} convolutional layer is symbolized by G_j, while the maximum pooling function is symbolized by MaxPooling.

3.3.4. Fully Connected Layers

One or more fully connected layers are used to route the feature maps from the final pooling or convolutional layer. A fully connected layer enables the network to learn complex decision boundaries, as each neuron is connected to all the others. Backpropagation will be utilized in this layer to lower the percentage of errors. In the fully connected layer, each node has a weight and a bias term. The weighted sum of the input features and output nodes can be computed by using the weight, and then this result is forwarded to the activation function via the bias. The fully connected layer's functioning formula is as outlined below:

$$F(x) = f(x * \omega + b) \quad (4)$$

F(x): fully connected layer; f: activation function; ω: weight parameters; b: bias

3.3.5. Output Layers

Neurons belonging to various classifications or groups make up the output layer. In the output layer, the SoftMax activation function is frequently employed to translate the initial ratings into probabilities that indicate the possibility of each category.

3.4. Long Short-Term Memory (LSTM)

RNN is a backpropagation network better suited for linear data estimation with random lengths since it collects time-based data. A variant of the conventional RNN called Long-Short Term Memory (LSTM) was developed by Hochreiter & Schmidhuber [34] to solve the gradient vanishing and explosion issues that arise frequently when training RNNs. Three gates manage and govern the state of the cell in LSTMs: an input gate that permits or prohibits the cell state to be altered by an input signal, an output gate that permits or prohibits the cell state to influence another neuron, and a forget gate that modifies the cell's self-recurrent association to either gather or forget its prior state. First, the input is used to specify the data that will be erased. In the forget gate, these are completed. Second, the sigmoid is used as an activation mechanism to modify the data in the input gate.

The tanh function is then used to produce the new data [35]. To reduce the discrepancy between the real training values and the LSTM outcomes, the procedure is repeated iteratively. An input, forget, and output gate, as well as a cell activation element, are all part of an LSTM cell. These devices use specific multipliers to control cell activations after receiving activation signals from various sources [36].

Figure 3 represents the architecture of the LSTM block, which includes the input signal, gates, activation function, output, and peephole connection [37]. The following are the components of the LSTM block.

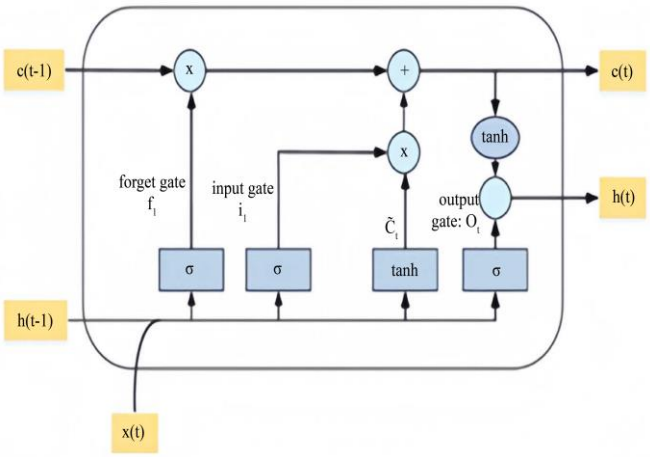

Fig. 3 Architecture of LSTM Block [37]

3.4.1. Input Gate (i)

The input gate manages information entering the cell. After applying a sigmoid activation function with the input attributes and the prior hidden state (h(t-1)), it generates a gate output that modifies the input data.

3.4.2. Forget gate (f)

Data from the previous state of the cell (C(t-1)) is retained or discarded based on the forget gate. Through sigmoid activation, it generates gate outputs that modify the prior state of the cell after passing through the input parameter set and the prior hidden state (h(t-1)).

3.4.3. Cell State (c)

An LSTM cell's memory is represented by its state. It is modified by the output of the input gate, the output of the forget gate, and a new candidate value computed using the input characteristics and the prior hidden state.

3.4.4. Output Gate (o)

It is responsible for deciding which portion of the cell state is to be output. It uses a combination of sigmoid and tanh activation functions to control information expansion. The output gate determines the next hidden state based on the regulation of the cell state. It ensures only the most relevant information is passed on through the next step to help make accurate predictions based on the current environment.

3.4.5. Output (h(t))

The final output layer h(t) represents the information produced by the LSTM cell in all successive steps after calculating the input pattern. It is a refined representation of the inner cell state c, modulated by the output gate o.

3.5. Combining CNN and LSTM

This study developed a hybrid technique for the automatic detection of facial expressions using the JAFFE and FER2013 datasets. This architecture integrates CNN and LSTM networks, employing CNN for complicated feature extraction from images and LSTM for classification purposes. This is achieved by integrating the practical conceptual vision illustrations acquired through CNNs with the effectiveness of LSTM for varying input lengths and their outcomes. The LSTM component records variations in time and correlations in expression order; the CNN component enables the algorithm to extract discriminatory spatial characteristics from face images. The integrated architecture can withstand changes in illumination, obstructions, facial expressions, and positions, making it suitable for real-world scenarios where cluttered or challenging visual information may be present.

4. Experiment and Evaluation

4.1. Dataset

The most widely used dataset for facial emotion detection applications is the JAFFE dataset, also known as FER2013. FER 2013 consists of 35887 48 by 48-pixel grayscale images. There are 28709 and 7178 photos in the training set and testing set. Anger, disgust, fear, happiness, neutral, sadness, and Surprise are the seven emotions in the dataset.

Table 1 shows the distribution of labels in the training and testing data. The unequal category distribution of this dataset, where certain emotions (like happiness) are more commonly reflected than others (like disgust), is one of the main problems. This imbalance problem for the FER-2013 dataset can be addressed by utilizing the Random Over Sampler. The model can become more proficient at identifying all emotions instead of being biased toward the more common ones by evenly allocating the quantity of data for each emotion. Improved accuracy of models can result from a balanced dataset.

Table 1. Distribution of labels in train and test data

7 Class	Angry	Disgust	Fear	Happy	Neutral	Sad	Surpise
Train	3995	436	4097	7215	4965	4830	3171
Test	958	111	1024	1774	1233	1247	831

4.2. Model Evaluation

The evaluation measures used were accuracy, recall, F1-score, and precision. Based on these evaluation measures, this research will investigate the CNN and LSTM algorithm outcomes in more detail [38].

4.3. Simulation Environment

Python is utilized for the assessment process, model construction, and data preparation. CNN and LSTM layers are implemented using Keras and TensorFlow. NumPy is used to carry out simple mathematical computations. The model was trained and validated using Google Colab.

4.4. Experimental Setup

4.4.1. Experiment Setup 1(CNN Model1)

Table 2 represents a summary of the proposed CNN model with 1 layer. The model takes as input grayscale images of size 48 × 48 × 1 and processes these through a series of Conv2D layers that extract features at an increasing level of depth. First, the convolution uses 32 filters, kernel size 3 × 3, stride (1,1), and valid padding, followed by batch normalization (axis = 3) and a ReLU activation. The second convolution expands to 64 filters, kernel size 3 × 3, and same padding; again, this is followed by normalization and ReLU and succeeded by a 2 × 2 max-pooling to downsample the feature maps in space. Subsequently, a convolution with 64 filters and a kernel size of 3 × 3 uses valid padding to support mid-level feature learning further, followed by a deeper layer of 128 filters using identical padding to find higher-level patterns.

A second 2 × 2 pooling layer downsamples the feature maps. At the same time, the final 128-filter convolution uses a 3 × 3 kernel and valid padding to capture the most abstract spatial relationships. This is followed by normalization, ReLU activation, and a third 2 × 2 pooling layer.

These feature maps are flattened and passed to a dense layer with 200 neurons, each activated via ReLU to allow complex interactions among features. This is accompanied by a dropout layer that has a rate of 0.6, which prevents overfitting by randomly discarding 60% of neurons during training. Finally, an output layer with 7 units, softmax-activated, generates for each of layer with 7 units, softmax-activated, generates for each of the seven classes a probability estimate.

Table 2. Summary of proposed CNN model 1 layers

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 46, 46, 16)	160
batch_normalization (BatchNormalization)	(None, 46, 46, 16)	64
activation (Activation)	(None, 46, 46, 16)	0
conv2d_1 (Conv2D)	(None, 46, 46, 32)	4,640
batch_normalization_1 (BatchNormalization)	(None, 46, 46, 32)	128
activation_1 (Activation)	(None, 46, 46, 32)	0
max_pooling2d (MaxPooling2D)	(None, 23, 23, 32)	0
conv2d_2 (Conv2D)	(None, 21, 21, 32)	9,248
batch_normalization_2 (BatchNormalization)	(None, 21, 21, 32)	128
activation_2 (Activation)	(None, 21, 21, 32)	0
max_pooling2d_1 (MaxPooling2D)	(None, 10, 10, 32)	0
flatten (Flatten)	(None, 3200)	0
dense (Dense)	(None, 32)	102,432
dropout (Dropout)	(None, 32)	0
dense_1 (Dense)	(None, 7)	231
Total params	117031	
Trainable params	116,871	
Non-trainable params	160	

4.4.2. Experiment Setup 2(CNN + LSTM model2)

Table 3. Summary of proposed CNN+LSTM Model 2 layers

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 46, 46, 32)	320
batch_normalization (BatchNormalization)	(None, 46, 46, 32)	128
activation (Activation)	(None, 46, 46, 32)	0
conv2d_1 (Conv2D)	(None, 46, 46, 64)	18,496
batch_normalization_1 (BatchNormalization)	(None, 46, 46, 64)	256
activation_1 (Activation)	(None, 46, 46, 64)	0
max_pooling2d (MaxPooling2D)	(None, 23, 23, 64)	0
conv2d_2 (Conv2D)	(None, 21, 21, 64)	36,928
batch_normalization_2 (BatchNormalization)	(None, 21, 21, 64)	256
activation_2 (Activation)	(None, 21, 21, 64)	0
conv2d_3 (Conv2D)	(None, 21, 21, 128)	73,856
batch_normalization_3 (BatchNormalization)	(None, 21, 21, 128)	512
activation_3 (Activation)	(None, 21, 21, 128)	0
max_pooling2d_1 (MaxPooling2D)	(None, 10, 10, 128)	0
conv2d_4 (Conv2D)	(None, 8, 8, 128)	147,584
batch_normalization_4 (BatchNormalization)	(None, 8, 8, 128)	512
activation_4 (Activation)	(None, 8, 8, 128)	0
max_pooling2d_2 (MaxPooling2D)	(None, 4, 4, 128)	0
reshape (Reshape)	(None, 64, 32)	0
lstm (LSTM)	(None, 32)	8,320
reshape_1 (Reshape)	(None, 1, 32)	0
lstm_1 (LSTM)	(None, 32)	8,320
dense (Dense)	(None, 200)	6,600
dropout (Dropout)	(None, 200)	0
dense_1 (Dense)	(None, 7)	1,407
Total parameters	303495	
Trainable parameters	302,663	
Non-trainable parameters	832	

Table 3 presents the proposed design of the hybrid CNN+LSTM Model 2. The hybrid design model is composed of a total of 12 layers: 5 convolutional, 3 pooling, two fully connected, two LSTM, and one output layer. The hybrid design uses the first convolutional layer, which takes a single grayscale input channel, using 32 filters of size 3x3. The resulting output layer has a dimension of 46 x 46 x 32, resulting in a total of 320 output parameters. The number of filters is increased to 128. Each layer is followed by batch normalization. The rectified linear unit activation function is applied. The MaxPooling layer, also referred to as the MaxPooling2D layer, is used here for reducing the spatial size of the feature map. The size of each max-pooling layer is defined by a pool size of 2x2, along with a stride value of 2. Each max-pooling layer is connected by a reshape layer, where the output is reshaped by reshape (none,64,32), acting as input for the first LSTM layer, composed of 32 memory cells, generating an output parameter of 8320. The next layer is also reshaped by reshape (none,1,32), serving as input for the second LSTM layer, also composed of 32 memory cells, generating an output parameter of 8320. The design also uses a 200 neuron ReLU dense layer with a 0.6 dropout for high-level feature learning, followed by a 7-unit SoftMax layer for emotion recognition. Training is done using Adam Optimizer with 0.001 learning rate and Categorical cross entropy loss with instances per minibatch of 32/64, and the process continues for 30-100 epochs.

4.4.3. Experiment Setup 3(CNN + LSTM model3)

Table 4. Summary of proposed CNN+LSTM model 3 layers

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 46, 46, 32)	320
batch_normalization (BatchNormalization)	(None, 46, 46, 32)	128
activation (Activation)	(None, 46, 46, 32)	0
conv2d_1 (Conv2D)	(None, 46, 46, 64)	18,496
batch_normalization_1 (BatchNormalization)	(None, 46, 46, 64)	256
activation_1 (Activation)	(None, 46, 46, 64)	0
max_pooling2d (MaxPooling2D)	(None, 23, 23, 64)	0
conv2d_2 (Conv2D)	(None, 21, 21, 64)	36,928

batch_normalization_2 (BatchNormalization)	(None, 21, 21, 64)	256
activation_2 (Activation)	(None, 21, 21, 64)	0
conv2d_3 (Conv2D)	(None, 21, 21, 128)	73,856
batch_normalization_3 (BatchNormalization)	(None, 21, 21, 128)	512
activation_3 (Activation)	(None, 21, 21, 128)	0
max_pooling2d_1 (MaxPooling2D)	(None, 10, 10, 128)	0
conv2d_4 (Conv2D)	(None, 8, 8, 128)	147,584
batch_normalization_4 (BatchNormalization)	(None, 8, 8, 128)	512
activation_4 (Activation)	(None, 8, 8, 128)	0
max_pooling2d_2 (MaxPooling2D)	(None, 4, 4, 128)	0
reshape (Reshape)	(None, 16, 128)	0
lstm (LSTM)	(None, 128)	131,584
reshape_1 (Reshape)	(None, 2, 64)	0
lstm_1 (LSTM)	(None, 64)	33,024
dense (Dense)	(None, 200)	13,000
dropout (Dropout)	(None, 200)	0
dense_1 (Dense)	(None, 7)	1,407
Total params	457,863	
Trainable params	457,031	
Non-trainable params	832	

Table 4 presents the proposed hybrid CNN+LSTM Model 3 architecture. The proposed model adopts a hybrid CNN–LSTM architecture that is designed to learn spatial and temporal features from 48×48 grayscale facial images. The convolutional component consists of multiple convolutional and pooling layers with filter depths gradually increasing from 32 to 128 while learning low-level edges to high-level abstract facial features. Batch normalization with ReLU activation is carried out in each layer for improved training stability and introducing nonlinearity, while max-pooling operations reduce spatial dimensions to retain dominant patterns. Following the extraction of spatial features, the resultant features are reshaped and fed into two LSTM layers containing 128 and 64 units, respectively, to let the model capture temporal dependencies and further refine sequential information.

Later, the integrated spatial–temporal representation is fed to a dense layer comprising 200 neurons that is followed by a dropout layer to avoid overfitting. Finally, a softmax output layer assigns the input to one of seven categories of facial expressions. For training, the proposed model uses the Adam optimizer with a learning rate of 0.001 to adaptively and efficiently update the weights. Multi-class classification makes use of categorical cross-entropy as the loss function, while the training is done in batches of 32 or 64 samples through 30 to 100 epochs, enabling the model to learn from the dataset through multiple iterative passes effectively.

5. Experiment Result and Analysis

This work utilizes the CNN model 1, the CNN+LSTM model 2, and the CNN+LSTM model 3 on the JAFFE dataset. For evaluation purposes, these models use accuracy, precision, f1-score, and support evaluation metrices.

5.1. Experiment using CNN Model 1

The metrics used to evaluate the effectiveness of CNN Model 1, developed for this study, are listed in Table 5. The proposed model achieves an average prediction accuracy of 74.65%, an average precision of 75%, an average recall of 74.86%, and an average F1-score of 74.57%.

Table 5. Evaluation metrics for the proposed CNN model 1

Class	Precision	Recall	F1-score	Support
0	0.68	0.73	0.7	935
1	0.99	1	0.99	895
2	0.69	0.65	0.67	800
3	0.76	0.71	0.73	900
4	0.56	0.67	0.61	888
5	0.86	0.92	0.89	869
6	0.71	0.56	0.63	920
Accuracy	—	—	**0.75**	6,293
Macro Avg	0.75	0.75	0.75	6,293
Weighted Avg	0.75	0.75	0.75	6,293

The CNN model's learning curve, which displays the model's efficiency in learning across epochs, is displayed in Figures 4 and 5. The model achieved an accuracy of 74.65% on test data and a loss of 1.1 after 100 epochs. Based on Figure 3, it is evident that model accuracy increases with the number of epochs. Similarly, test accuracy increases with regard to the number of epochs. As shown in Figure 4, the train and test loss curves depend on the number of epochs. Loss value will tend to decline in relation to epochs.

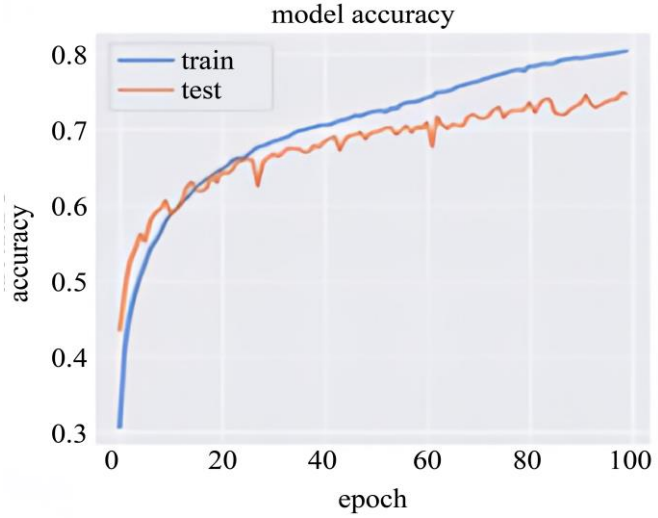

Fig. 4 CNN model 1 accuracy

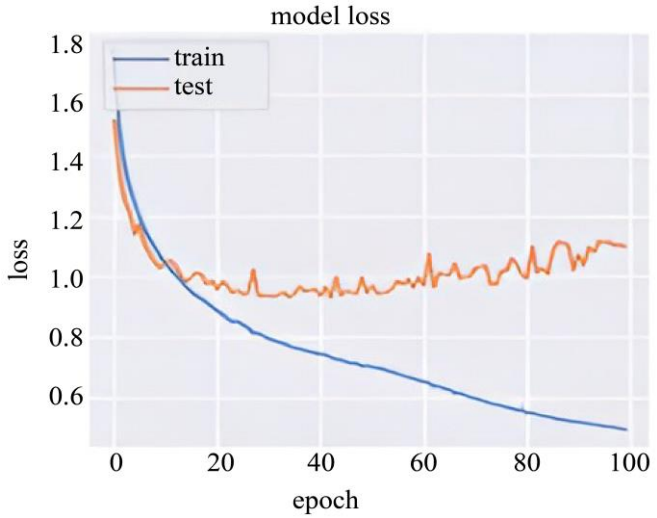

Fig. 5 CNN model 1 loss

Figure 6 represents the confusion matrix of CNN Model 1. For the CNN model 1, the strong diagonal values and good accuracy are associated with emotions, including sadness, happiness, disgust, and anger. Because of the face feature similarity, the model often groups together the Fear, Neutral, and Surprise; these are the leading causes of ambiguity. Neutral expression has the biggest dispersion of incorrect classifications across multiple categories.

Although on one level, "Fear" has similarities with "Neutral" (109) and "Surprise" (42), "Angry" is often mistaken for "Neutral" (116). The reason for this could be similar cues: "surprise is sometimes mistakenly labelled as 'Neutral' (159) and 'Happy' (69)," and "Neutral" Is Also "Angry" (89).

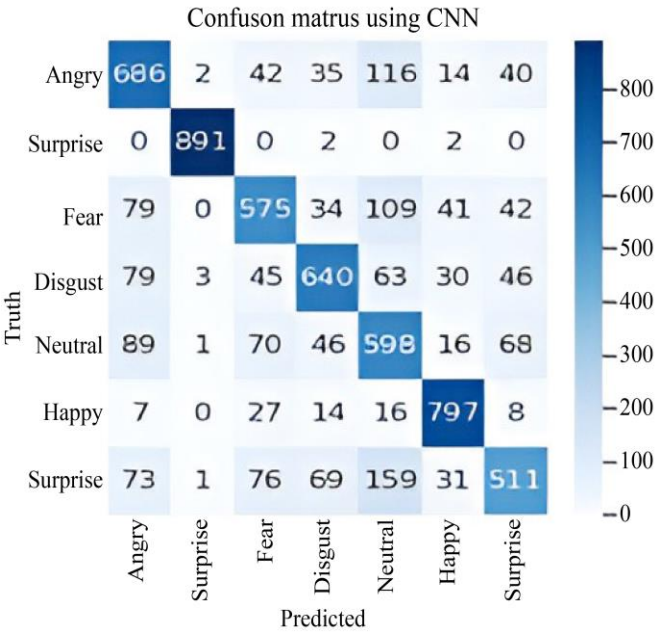

Fig. 6 Confusion matrix of the CNN model 1

5.2. Experiment using CNN+LSTM Model 2

Table 6. Evaluation metrics for proposed hybrid CNN+LSTM model 2

Class	Precision	Recall	F1-score	Support
0	0.86	0.73	0.79	935
1	0.99	0.97	0.98	895
2	0.72	0.8	0.76	800
3	0.65	0.85	0.74	900
4	0.72	0.67	0.69	888
5	0.89	0.94	0.91	869
6	0.82	0.63	0.71	920
Accuracy	—	—	0.8	6,293
Macro Avg	0.81	0.8	0.8	6,293
Weighted Avg	0.81	0.8	0.8	6,293

Table 6 represents the metrics used to evaluate the performance of the hybrid CNN+LSTM model 2 developed for this research. The proposed model has an average prediction accuracy of 79.79%, average precision of 80.71%, average recall of 79.86%, and average f1-score of 79.71%. The model's accuracy and loss are displayed in Figures 7 and 8.

Figure 9 represents the confusion matrix of the hybrid CNN+LSTM Model2. For the CNN+LSTM model 2, the strong diagonal values and good accuracy are associated with emotions, including fear, happiness, disgust, and Surprise. With a small number of misclassifications, sadness and happiness continue to be the most reliably forecasted classes. Straightforward cases where classification goes wrong are between emotions that look very similar.

This is why there is often an error between Surprise and both Happy (143) and Neutral (92), between Fear, which can often be mistaken for Surprise (28) or Happy (58), or between Ambiguous cases represented by Neutral, which is often mistaken for Fear (110) or Happy (91).

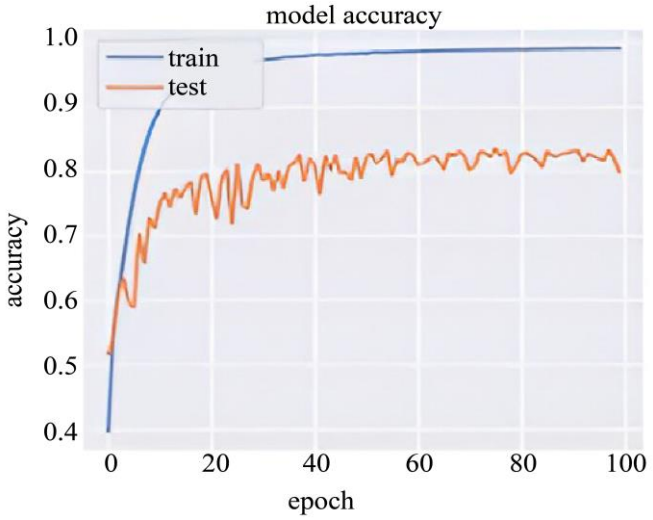

Fig. 7 Hybrid CNN+LSTM model 2 accuracy

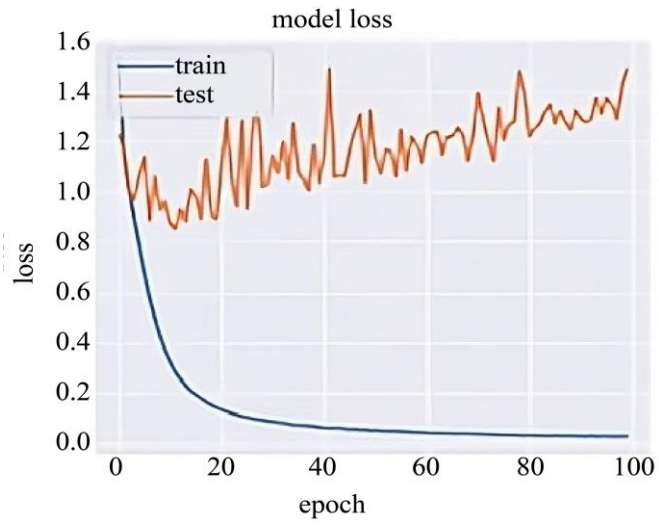

Fig. 8 Hybrid CNN+LSTM model 2 loss

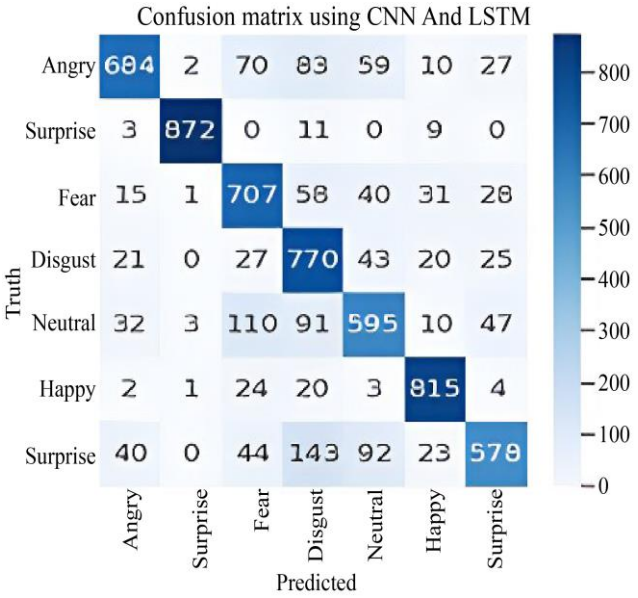

Fig. 9 Confusion matrix of the CNN+LSTM model 2

5.3. Experiment using CNN+LSTM Model3

Table 7. Evaluation metrics for the proposed hybrid CNN+LSTM model 3

Class	Precision	Recall	F1-score	Support
0	0.87	0.82	0.85	935
1	0.99	1	0.99	895
2	0.83	0.82	0.83	880
3	0.84	0.83	0.83	906
4	0.79	0.72	0.75	888
5	0.93	0.96	0.94	869
6	0.72	0.81	0.76	920
Accuracy	—	—	**0.85**	6,293
Macro Avg	0.85	0.85	0.85	6,293
Weighted Avg	0.85	0.85	0.85	6,293

Table 7 represents the metrics used to evaluate the performance of the hybrid CNN+LSTM model 3 developed for this research. The proposed model has an average prediction accuracy of 85.11%, average precision of 85.28%, average recall of 85.14%, and average f1-score of 85%.

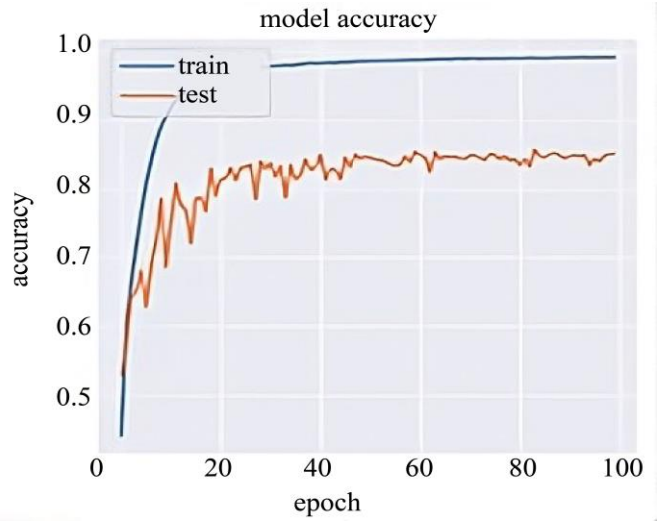

Fig. 10 CNN+LSTM model 3 accuracy

The model's accuracy and loss are displayed in Figures 10 and 11. The model achieved an accuracy of 85.11% on test data and a loss of 1.2 after 100 epochs.

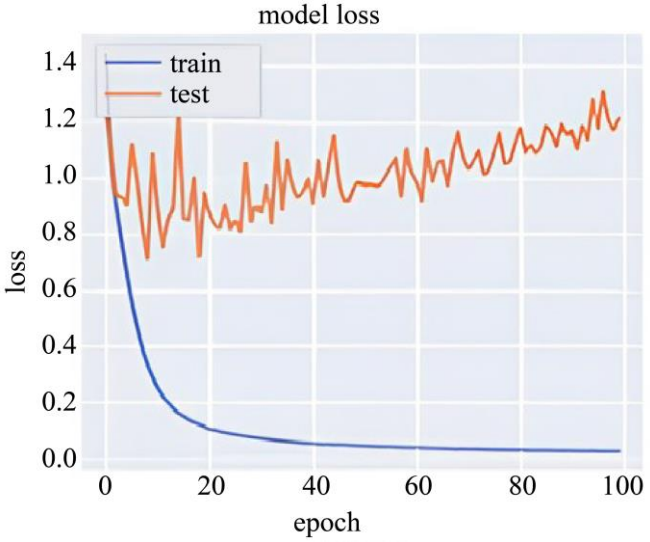

Fig. 11 CNN+LSTM model 3 loss

Figure 12 represents the confusion matrix of the hybrid CNN+LSTM Model3. Fear has a significant level of similarity to both Neutral (125) and Surprise (5), although the most frequent incorrect labelling for Angry is Neutral (43). The closely related facial cue of Surprise is often wrongly classified as either Neutral (78) or Happy (24), and the characteristics of the neutral face are associated with fear (53) and Angry (32), reflecting ambiguity for the latter two.

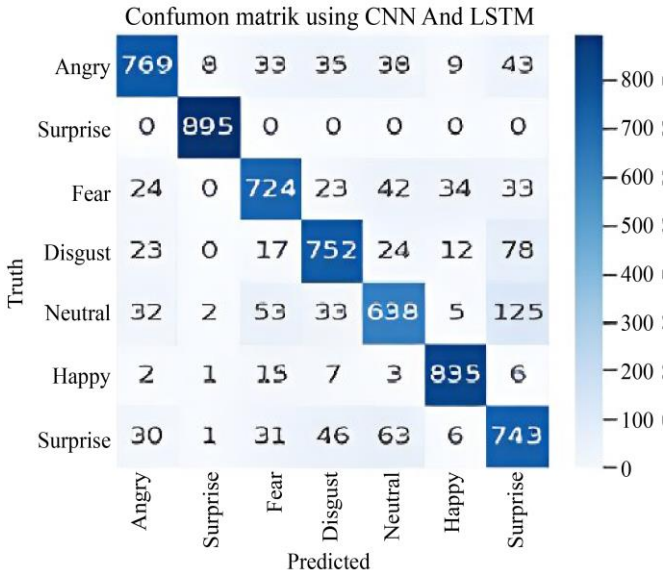

Fig. 12 Confusion matrix of the CNN+LSTM model 3

5.4. Relative Performance of Proposed Methods

The suggested CNN and Hybrid CNN+LSTM's relative performance on FERs is shown in Figure 13. The table represents the performance comparison of the suggested methods. The hybrid CNN+LSTM model 2 had an accuracy of 85.11%, a precision of 85.28%, a recall of 85.14%, and an F1-score of 85.00 %. The hybrid CNN+LSTM model 2 performance is better than the other suggested CNN and the hybrid CNN+LSTM model 1.

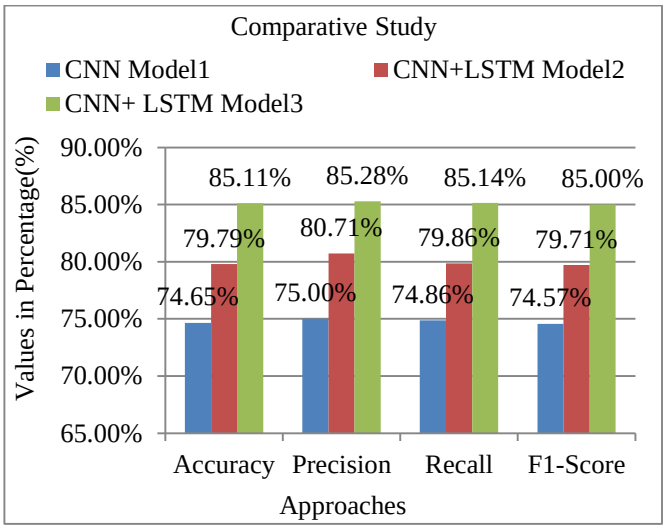

Fig. 13 Performance comparison of suggested methods

Table 8. Performance comparison of proposed methods

Metrics	Proposed Methods		
	CNN Model1	CNN+LSTM Model2	CNN+ LSTM Model3
Accuracy	74.65%	79.79%	85.11%
Precision	75.00%	80.71%	85.28%
Recall	74.86%	79.86%	85.14%
F1-Score	74.57%	79.71%	85.00%

5.5. Comparison with Contemporary Method

The proposed method was compared with the current ones in [22, 39-48], and to illustrate the importance of the outcomes. The comparison between the suggested model and alternative models on the Fer2013 Dataset is summarized in Table 9. The accuracy levels for traditional architectures like VGG16 and these simple CNN-based methods varied from 67% to approximately 72%. More complex architectures like EC-Net family, LHC-Net family, and ResNet demonstrated steady improvement towards reaching 74% to 76% accuracy. By utilizing focus methods and ensembles, as done effectively in CBAM+ResNet, VGG19, and Ensemble Residual Masking Network, accuracy levels were significantly uplifted towards 77%. The best models for the research, ResEmoteNet, DCRAN, and ROS+VGG16, offered the best modulated accuracy, ranging from 79.79% to 82%. The comparative result for the suggested models includes 74.65% for CNN Model 1, equivalent accuracy as ResEmoteNet (79.79%) for CNN+LSTM Model 2, and 85.11%, which is among the very best among all the compared systems, as CNN+LSTM Model 3 attained maximum accuracy at 85.11%. Therefore, it can be realized that the CNN-LSTM combination-based system is trustworthy and efficient.

Table 9. Comparison of the proposed model with other models on the Fer2013 dataset

Refrence	Model Used	Accuracy %
41	VGG16	67%
42	CNN+SVM	71.27%
43	EC-Net	72.36%
44	LHC-Net	74.42%
45	ResNet CNN	75.47%
46	CBAM+ RESNET	75.77%
47	VGG19	75.97%
48	Ensemble Residual Masking Network CNN	76.82%
49	ResEmoteNet	79.79%
50	DCRAN	81.10%
51	ROS+VGG16	82.00%
Proposed	CNN Model1	74.65%
	CNN+LSTM Model2	79.79%
	CNN+LSTM Model3	**85.11%**

6. Discusion

On the FER-2013 dataset, the hybrid model of CNN+LSTM provides a validation accuracy of 85.11%. The model performs very well in detecting distinctive emotions such as "happy" and "sadness". However, due to class disparity and a likeness in facial patterns, it performs relatively poorly in detecting subtle facial expressions such as "neutral" and "surprise". Future research can also be done to explore

other modes, such as audio and facial expressions, to develop more accurate emotion detection systems. It can be helpful in daily life applications, such as marketing, medical treatment, fraud detection, and customer service. Facial expression recognition systems raise several important issues that should be considered while performing research and development in FER technology. Facial expression recognition involves gathering and analyzing facial data; privacy issues arise with this form of system, especially when those facial images are taken without proper informed consent and when stored in an insecure fashion. Based upon a lack of diversity in demographics that leads to biased estimates of performance that fail across all ages, genders, and ethnic groups, this form of system has issues with objectivity. Emotion itself is a complex human experience that involves circumstances, which means that the FER system has issues with being misleading with emotion interpretation, in turn providing a distorted view of proper emotional characterization, especially in areas of concern such as healthcare, education, and surveillance, which are particularly challenging.

7. Conclusion

For this study, facial expression recognition was analyzed using a hybrid model CNN-LSTM, as well as Convolutional Neural Networks (CNNs), on FER2013. The research work was focused on alleviating the issues associated with accurately detecting and classifying facial expressions.

To identify facial expressions, facial data must be collected and analyzed. This raises issues of privacy, especially when these images are captured without the mandatory need for informed authorization and stored in a harmful manner. In this study, a random oversample layer is employed to mitigate class discrepancies in facial expression identification. The layering effect involving several convolutional and pooling layers was implemented for the capability of the CNNs to detect complex expressions or variations within face expressions. By combining Long Short-Term Memory Networks with CNNs, two types of hybrid CNN-LSTM models were created, where facial expression sequences are analyzed over a specific duration using LSTMs. Using the hybrid approach, it is possible to generate spatial features using CNNs or learn the classification sequence by employing LSTMs. The experimental work was carried out using the CNN model 1, the CNN+LSTM model 2, and the CNN+LSTM model 3. According to the findings, the hybrid CNN-LSTM model is more effective than the CNN approach for identifying facial expressions. The LSTM component is also able to understand the expression changeover aspect, which increases the recognition accuracy, making it efficient. The accuracy for CNN model1 is 74.65%, CNN+LSTM model1 is 79.79%, while the accuracy for CNN+LSTM model2 is 85.11%. Future studies will focus on techniques for combining a multi-modal approach, such as facial expression analysis, voice, and physical characteristics.

References

[1] Chengxu Liang et al., "Facial Expression Recognition using LBP and CNN Networks Integrating Attention Mechanism," *2023 Asia Symposium on Image Processing (ASIP)*, Tianjin, China, pp. 1-6, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[2] Farshad Safavi, Kulin Patel, and Ramana Kumar Vinjamuri, "Towards Efficient Deep Learning Models for Facial Expression Recognition using Transformers," *2023 IEEE 19th International Conference on Body Sensor Networks (BSN)*, Boston, MA, USA, pp. 1-4, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Hadeel Mohammed, Mohammed Nasser Hussain, and Faiz Al Alawy, "Facial Expression Recognition: Machine Learning Algorithms and Feature Extraction Techniques," *Al-Iraqia Journal of Scientific Engineering Research*, vol. 2, no. 2, pp. 23-28, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] M. Priya et al., "Automatic Emotion Detection using SVM-Based Optimal Kernel Function," *2024 11th International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, pp. 1607-1611, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[5] Qisheng Wang, Xiaofei Yan, and Yanqiu Wang, "Research on Deep Learning-based Facial Expression Recognition and its Application in Online Learning State Monitoring," *3rd International Conference on Advanced Algorithms and Signal Image Processing (AASIP 2023)*, Kuala Lumpur, Malaysia, vol. 12799, pp. 459-464, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Rasha Friji et al., "Geometric Deep Neural Network using Rigid and Non-Rigid Transformations for Landmark-Based Human Behavior Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 11, pp. 13314-13327, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Huimin Xiao, Xinghao Wang, and Qiang Xing, "Facial Expression Recognition in the Wild based on Convolutional Neural Network and Graph Convolutional Network," *Fourth International Conference on Signal Processing and Computer Science (SPCS 2023)*, Guilin, China, vol. 12970, pp. 976-981, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Wissam J. Baddar, Sangmin Lee, and Yong Man Ro, "On-the-Fly Facial Expression Prediction using LSTM Encoded Appearance-Suppressed Dynamics," *IEEE Transactions on Affective Computing*, vol. 13, no. 1, pp. 159-174, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] Yikang Guo et al., "A Personalized Spatial-Temporal Cold Pain Intensity Estimation Model based on Facial Expression," *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 9, pp. 1-8, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] Mayuri Arul Vinayakam Rajasimman et al., "Robust Facial Expression Recognition using an Evolutionary Algorithm with a Deep Learning Model," *Applied Sciences*, vol. 13, no. 1, pp. 1-20, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[11] Yanisa Mahayossanunt et al., "Explainable Depression Detection based on Facial Expression using LSTM on Attentional Intermediate Feature Fusion with Label Smoothing," *Sensors*, vol. 23, no. 23, pp. 1-16, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[12] Rupali Gill, Aditi Moudgil, and Palak Bajaj, "Hybrid Approach for Emotion Recognition using CNLSTM in Video Expressions," *2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, Noida, India, pp. 1-5, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] Mustafa Can Gursesli et al., "Facial Emotion Recognition (FER) through Custom Lightweight CNN Model: Performance Evaluation in Public Datasets," *IEEE Access*, vol. 12, pp. 45543-45559, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[14] Jasthi Vivek Vardhan, Yelavarti Kalyan Chakravarti, and Annam Jitin Chand, "Emotion Recognition by Facial Expressions and Speech using Deep Learning," *2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, Bhilai, India, pp. 1-7, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Yue Luo et al., "Design of Facial Expression Recognition Algorithm based on CNN Model," *2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA)*, Shenyang, China, pp. 580-583, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[16] Nongmeikapam Thoiba Singh et al., "Comparative Analysis of Traditional Machine Learning and Deep Learning Techniques for Facial Expression Recognition," *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Delhi, India, pp. 1-7, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[17] Luu-Ngoc Do et al., "Deep Neural Network-Based Fusion Model for Emotion Recognition using Visual Data," *The Journal of Supercomputing*, vol. 77, no. 10, pp. 10773-10790, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18] Li Muyao, and Ding Weili, "Facial Expression Recognition based on FPN and LSTM," *2023 IEEE 5th International Conference on Power, Intelligent Computing and Systems (ICPICS)*, Shenyang, China, pp. 762-767, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[19] Md. Mohsin Kabir et al., "Facial Expression Recognition using CNN-LSTM Approach," *2021 International Conference on Science & Contemporary Technologies (ICSCT)*, Dhaka, Bangladesh, pp. 1-6, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[20] Saikat Das et al., "IoT based Framework Design for Automated Human Emotion Recognition," *2023 2nd International Conference on Ambient Intelligence in Health Care (ICAIHC)*, Bhubaneswar, India, pp. 1-6, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[21] Angalakuditi Hindu, and Biswajit Bhowmik, "An IoT-enabled Stress Detection Scheme using Facial Expression," *2022 IEEE 19th India Council International Conference (INDICON)*, Kochi, India, pp. 1-6, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[22] Jinyu Luo et al., "Facial Expression Recognition using Machine Learning Models in FER2013," *2021 IEEE 3rd International Conference on Frontiers Technology of Information and Computer (ICFTIC)*, Greenville, SC, USA, pp. 231-235, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[23] Uma Chinta, and Adham Atyabi, "A Framework Pipeline to Address Imbalanced Class Distribution Problem in Real-World Datasets," *2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, USA, pp. 746-753, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[24] Deepak Raj, and Md. Abdul Wassay, "Facial Emotional Recognition using Convolutional Neural Network," *2023 2nd International Conference for Innovation in Technology (INOCON)*, Bangalore, India, pp. 1-5, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[25] Özay Ezerceli, and M. Taner Eskil, "Convolutional Neural Network (CNN) Algorithm based Facial Emotion Recognition (FER) System for FER-2013 Dataset," *2022 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, Maldives, pp. 1-6, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[26] K. Kousalya et al., "Group Emotion Detection using Convolutional Neural Network," *2022 OPJU International Technology Conference on Emerging Technologies for Sustainable Development (OTCON)*, Raigarh, Chhattisgarh, India, pp. 1-6, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[27] Aicha Nouisser, Ramzi Zouari, and Monji Kherallah, "Enhanced MobileNet and Transfer Learning for Facial Emotion Recognition," *2022 International Arab Conference on Information Technology (ACIT)*, Abu Dhabi, United Arab Emirates, pp. 1-5, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[28] Dorfell Parra, and Carlos Camargo, "Design Methodology for Single-Channel CNN-Based FER Systems," *2023 6th International Conference on Information and Computer Technologies (ICICT)*, Raleigh, NC, USA, pp. 89-94, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[29] Tinuk Agustin, Moch Hari Purwidiantoro, and Mochammad Luthfi Rahmadi, "Enhancing Facial Expression Recognition through Ensemble Deep Learning," *2023 5th International Conference on Cybernetics and Intelligent System (ICORIS)*, Pangkalpinang, Indonesia, pp. 1-6, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[30] Mj Alben Richards et al., "Facial Expression Recognition using Convolutional Neural Network," *2023 International Conference on Bio Signals, Images, and Instrumentation (ICBSII)*, Chennai, India, pp. 1-5, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[31] Ankit S. Vyas, Harshadkumar B. Prajapati, and Vipul K. Dabhi, "Survey on Face Expression Recognition using CNN," *2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS)*, Coimbatore, India, pp. 102-106, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[32] Xinpeng Zhang, "Face Expression Recognition based on Convolutional Neural Networks," *2022 International Conference on Cloud Computing, Big Data Applications and Software Engineering (CBASE)*, Suzhou, China, pp. 254-258, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[33] Christian Białek, Andrzej Matiolański, and Michał Grega, "An Efficient Approach to Face Emotion Recognition with Convolutional Neural Networks," *Electronics*, vol. 12, no. 12, pp. 1-21, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[34] Shan Li, and Weihong Deng, "Deep Facial Expression Recognition: A Survey," *IEEE Transactions on Affective Computing*, vol. 13, no. 3, pp. 1195-1215, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[35] Ceren Özkara, and Pınar Oğuz Ekim, "Real-Time Facial Emotion Recognition for Visualization Systems," *2022 Innovations in Intelligent Systems and Applications Conference (ASYU)*, Antalya, Turkey, pp. 1-5, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[36] M.N. Kavitha, and A. RajivKannan, "Hybrid Convolutional Neural Network and Long Short-Term Memory Approach for Facial Expression Recognition," *Intelligent Automation & Soft Computing*, vol. 35, no. 1, pp. 689-704, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[37] Greg Van Houdt, Carlos Mosquera, and Gonzalo Nápoles, "A Review on the Long Short-Term Memory Model," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5929-5955, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[38] Harya Widiputra, Adele Mailangkay, and Elliana Gautama, "Multivariate CNN-LSTM Model for Multiple Parallel Financial Time-Series Prediction," *Complexity*, vol. 2021, no. 1, pp. 1-14, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[39] Chen Jia, Chu Li Li, and Zhou Ying, "Facial Expression Recognition based on the Ensemble Learning of CNNs," *2020 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)*, Macau, China, pp. 1-5, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[40] Muhamad Dwisnanto Putro et al., "Multi-View Facial Emotion Recognition using Attention-Based Convolutional Network for Human-Machine Interaction," *2025 International Conference on Information Management and Technology (ICIMTech)*, Bandung, Jawa Barat, Indonesia, pp. 512-517, 2025. [CrossRef] [Publisher Link]

[41] Roberto Pecoraro, Valerio Basile, and Viviana Bono, "Local Multi-Head Channel Self-Attention for Facial Expression Recognition," *Information*, vol. 13, no. 9, pp. 1-17, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[42] Sunil S. Harakannanavar et al., "Enhanced Facial Emotion Recognition using Deep Learning Techniques: A Comprehensive Analysis and Implementation," *2024 International Conference on Distributed Systems, Computer Networks and Cybersecurity (ICDSCNC)*, Bengaluru, India, pp. 1-6, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[43] Wenqian Yan et al., "Optimizing Facial Expression Recognition: A One-Class Classification Approach using ResNet18 and CBAM," *2024 3rd International Conference on Computer Technologies (ICCTech)*, Bali, Indonesia, pp. 1-5, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[44] S. Vignesh et al., "A Novel Facial Emotion Recognition Model using Segmentation VGG-19 Architecture," *International Journal of Information Technology*, vol. 15, no. 4, pp. 1777-1787, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[45] Luan Pham et al., "Facial Expression Recognition using Residual Masking Network," *2020 25th International Conference on Pattern Recognition (ICPR)*, Milan, Italy, pp. 4513-4519, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[46] Arnab Kumar Roy et al., "ResEmoteNet: Bridging Accuracy and Loss Reduction in Facial Emotion Recognition," *IEEE Signal Processing Letters*, vol. 32, pp. 491-495, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[47] Neha Gahlan, and Divyashikha Sethia, "Emotion Recognition from Facial Expressions using Deep Recurrent Attention Network," *2024 16th International Conference on Communication Systems & Networks (COMSNETS)*, Bengaluru, India, pp. 86-91, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[48] Rifky, Panca Dewi Pamungkasari, and Endah Tri Esti Handayani, "Performance Analysis of Facial Expression Recognition by Using Geometry Augmentation and Random Oversampling for CNN Model," *2024 10th International Conference on Smart Computing and Communication (ICSCC)*, Bali, Indonesia, pp. 614-618, 2024. [CrossRef] [Google Scholar] [Publisher Link]