International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

Facial Emotion Detection Using CNN-Based Neural Network

Sukhpreet Singh[1], Mohammad Nazmul Alam[2], Suman Lata[3]

[1] Faculty of Computing, Guru Kashi University
[2]Faculty of Computing, Guru Kashi University
[3]Basic Computer Instructor, Govt. School Sri Ganganagar

---***---

***Abstract* —While computer vision aims to confound humans through the analysis of digital images, humans rely on their sensory perception to decipher emotions. Unlike the challenge of comprehending virtual images, assessing emotions in speech involves evaluating various aspects such as tone, volume, speed, and more. These novel techniques allow for the modulation of emotional "anxiety" in speech. In this model, our objective is to develop a convolutional neural network (CNN) version capable of segmenting input videos into seven distinct symbols representing emotions: anger, hatred, criticism, happiness, sadness, surprise, and neutrality. To achieve this, we leverage a CNN to extract and process semantic information from facial expressions, enabling us to discern these emotions accurately. Additionally, we implement data augmentation strategies to combat issues of overfitting and underfitting. The results from this version of the model indicate improved performance when handling larger images, achieving an accuracy rate exceeding 90%. This marks the creation of a CNN-based system for recognizing emotions conveyed through speech patterns. We have carefully adjusted several parameters to enhance the model's accuracy while also investigating the key factors influencing its performance. The model concludes by delving into a comprehensive discussion on the precision and robustness of different CNN designs, highlighting areas for potential improvement, and assessing the overall efficiency of these enhancements.**

***Keywords*— CNN, Neural Network, Emotion Detection, RNN, Face Recognition**

1. Introduction

Facial emotion detection has emerged as a significant area of research and application within the field of computer vision and artificial intelligence. The ability to automatically recognize and interpret human emotions from facial expressions has diverse practical implications, spanning from human-computer interaction to affective computing, healthcare, marketing, and even social robotics. Emotions play a crucial role in human communication and understanding them is an essential aspect of developing emotionally intelligent systems. Traditional methods for facial emotion detection relied on handcrafted features and machine learning algorithms. However, with the advent of deep learning and neural networks, the performance and accuracy of facial emotion detection systems have experienced a significant boost. Neural networks, especially Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have demonstrated remarkable capabilities in learning meaningful representations directly from raw image data and sequential inputs. By leveraging large-scale datasets, these networks can discern subtle patterns in facial expressions and generalize well to previously unseen emotions [1-5].

This paper explores the application of neural network-based models for facial emotion detection. We will delve into the architecture and design choices that contribute to successful emotion recognition, considering factors such as network depth, data augmentation, transfer learning, and the impact of different facial landmarks.

2. Methodology

We will generate and classify the dataset. Appropriate plans will be established to differentiate between the apparent results, taking into account the results of each classification algorithm. The following parameters will be available for each domain: accuracy, error, accuracy, expected (inverse), specificity, F1 score, and CohenKappa score. Conclusions will be drawn based on the efficiency of the algorithm at each location. Depending on the research and the objectives of the relevant study, the following objectives will be achieved: Complete the characterization process with feature selection, Classify each subset using five different algorithms, and Design an effective CNN architecture. Convolutional neural network (CNN) is used to distinguish emotional speech patterns. Databases including RAVDEES and SAVEE were used to prepare and determine the CNN model [6]. Keras (Tensor Flow's high-level API for building and training deep learning models) was used as the framework to implement the CNN models [7-10].

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

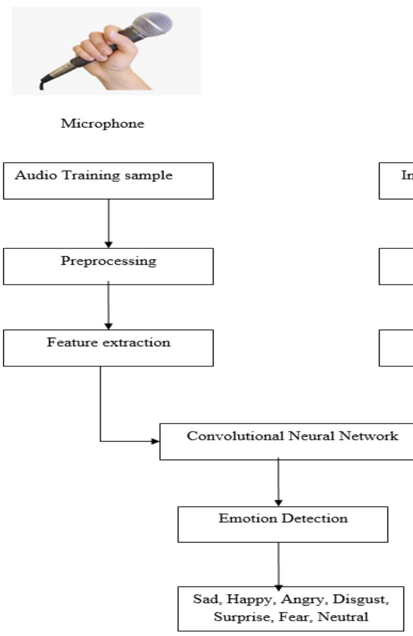

Fig.1. System Architecture

A) Datasets of speech emotion

Ryerson Affective Speech and Song Audiovisual Database (RAVDESS)contains sentences with a North American accent. Speech patterns include emotions such as peace, happiness, sadness, anger, fear, surprise, and disgust. All emotions occur at two levels of emotional intensity (normal, and strong) with an additional moderate emotion.

1) *Preprocessing:* The first step is to prepare the audio files. Emotions in the sound structure can be identified by a special character in the 3rd digit representing the emotion type in the file. The data set includes five different emotions calm, happy sad, anger, fear, disgust, and surprise.

2) *Label definition:* Classification according to the number of classes is defined to identify speech labels. Some categories are:

Category: Good and Bad Quality: Sick, happy. Negative: fear, sadness, anger.

Class: Anger, Sad, Happy, Fear, Calm.
Class: Anger, Sadness, Happiness, Fear, Anger, Frustration, Anger, Fear.

3) *Create the dimensions of the model:* This step involves creating the layers of the CNN model, choosing the activation function, choosing the appropriate option, and defining Softmax to split the conversation into many groups.

4) *Model Training and Testing:* Train the model with training data and test the model with test data. Compare the actual cost with the estimated cost. This comparison gives us the accuracy of the model.

B. CNN architecture

In this study, the use of deep neural network architecture is a convolutional neural network. In the proposed architecture, each convolution layer is placed after the maximum pooling layer. Improved Rectified Linear Units (ReLU) in both convolutional and full layers are used to create nonlinearity in the model [11].

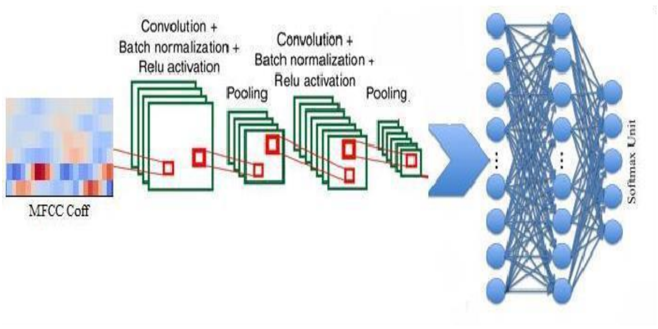

Fig.2. Architecture of CNN

Stack normalization is used to increase the strength of neural networks. Layers are used; It is a fully connected layer in which all neurons in one layer are connected to neurons in the next layer. The softmax unit is used to calculate the probability of classes. The number of softmaxes used depends on the number of classes dividing the hypothesis. The model takes 10 to 14 hours to train.

The CPU takes a lot of time to train the model, instead, the GPU can be used to speed up the training process. More cores in the GPU make it faster and save a lot of time. Figure 2 shows the convolutional neural network architecture used in this model. Lighter CNN architectures have also been used to classify more classes with good results [12-15]

C. Properties of the sound file

This step involves setting up the sound file. Each sound file has a funny character in the sixth digit of the file name, which can be used to identify the purpose of the sound file. There are 6 different characters in our file (happy, calm, sad, scared, angry, panic). The Evolution of Audio Feature Extraction Window The light is red just in time, and the size of the window is adjusted. The main problem here is the deformation caused by the change of the window. This is known as the Gibbs phenomenon. Hamming windows with

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

smooth edges to prevent sudden changes are used to solve this problem.

We use the Librosa class in Python to perform operations and set properties of audio files. It provides some key ideas and works to build a music data extraction framework. With the Librosa library, we have the option to edit items such as MEL Frequency Cepstral Coefficients (MFCC). MFCC is the most widely used material for speech recognition [16-17].

We also separate male and female voices using web identifiers. The reason for this is that thanks to the test, we found that the difference between male and female voices was distributed by 15%. It may be the sound of the sound that affects the result.

D. Classification Models

Since the task is about classification, CNN seems like the obvious choice. At the same time, we developed MLP and LSTM models, but they did not perform as expected, resulting in very low accuracy and failing the test when predicting hypotheses. Our proposed training model and testing model are given below:

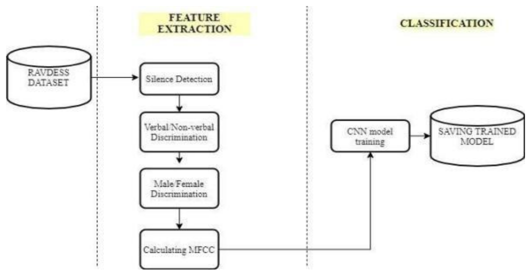

Fig.3. Proposed Training Model

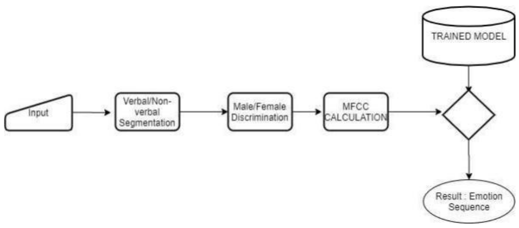

Fig.4. Proposed Testing Model

1) Preliminary data for facial emotion: People were divided into two groups; training methods and validation. The training sample contains 80% of the original sample data and 20% of the sample for validation. The training and verification numbers are 22966 and 5741, respectively. Perform preprocessing to prepare snapshots during feature extraction. Remove a set of face points from the image and then subtract faces from those points. Each subject and accuracy check uses a different face.

2) Datasets: Having appropriate and relevant data to choose from is an essential part of any given problem. Therefore, we use the FERC-2013 dataset from the Kaggle data science platform to get the best results for the problem. FERC-2013 contains fer2013.Csv with 3 columns (concept, pixel, reference). The table below shows the unused models for all theory classes.

TABLE I: FACE DATASETS COUNT

Class	No. of Samples
Angry	3995
Disgust	436
Fear	4097
Happy	7215
Neutral	4830
Sad	3171
Surprise	4965

3) Feature Extraction: Feature extraction involves the critical process of isolating valuable information from a dataset. Often, datasets contain numerous features that prove redundant when predicting outcomes, offering no real benefit to machine learning models. These superfluous features not only fail to enhance predictions but also consume unnecessary computational resources, increasing the overall cost of the process.

Algorithms designed to pinpoint the most effective subset of features, resulting in superior outcomes compared to using all features, are referred to as feature selection algorithms. These algorithms typically evaluate features using specific metrics and fall into three main categories:

Filter Method: This selection approach relies on statistical measures to assess feature importance. It directly measures the impact of individual features on predictions without considering feature combinations. Techniques like the chi-square test are employed to determine the correlation between features. Unlike other methods, filter methods do not involve machine learning algorithms in the evaluation process.

Wrapper Method: Wrapper algorithms systematically explore various subsets of features, ranging from 1 to n and vice versa, to identify the optimal set of n features. They include techniques such as forward selection (gradually adding the best features until reaching n), backward selection (removing the least valuable features until reaching n), and full selection (combining forward and backward strategies to find the best features). While wrapper methods often yield excellent results, they can be computationally expensive.

Embedded Method: Embedded algorithms blend elements of both filter and wrapper methods, conducting feature searches with minimal additional computational burden. These methods aim to strike a balance between effectiveness and efficiency. Examples of embedded techniques include Lasso, Ridge, and Elastic Nets, which

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

incorporate feature selection within their optimization processes [18-20].

4) Cepstral Field Properties: Mel Frequency Cepstral Coefficients (MFCC) are derived from the cepstral representation of sound waves. They depict how the short-term power of a voice clip relates to the cosine change of the logarithmic power range on a non-linear Mel scale. MFCC deliberately distributes similar frequency groups on the Mel scale, mimicking the human voice's characteristics, making it a crucial component of acoustic signal processing. MFCCs find applications in speech recognition, speech reconstruction, speaker recognition, music classification, music data retrieval, voice match prediction, and voice recognition, and are widely employed across various domains [21-22].

Linear Predictive Cepstral Coefficients (LPCC) harness the power of cepstral coefficients, making them suitable for AI applications. However, the direct calculation of the Linear Predictive Coding (LPC) coefficients can be sensitive to numerical accuracy, prompting the conversion of LPC into the cepstral domain, resulting in LPCC. LPCC is essentially derived from LPC and finds utility in speech recognition, speech detection, speaker verification, music classification, and more. The following calculation outlines the process of obtaining LPCC from an audio signal.

Recursive Feature Elimination (RFE) is an approach that identifies and removes the least valuable features based on their inclusion values. It iteratively adds and ranks features with the help of an external estimator, ultimately selecting the most significant ones. RFE continues until it reaches the desired nth position. While computationally intensive, RFE is effective in identifying optimal features for a given dataset. Its performance relies on user input and domain knowledge, as no tool can automatically determine the best features. RFE serves to identify the most informative features for the specific data context. One approach involves iterating through features to select the most relevant ones, but this method is most useful when the feature count is low. In this project, a Random Forest model serves as the estimator, generating multiple random trees using a subspace algorithm [23].

Lasso-based feature selection, abbreviated as "lasso," emphasizes feature importance in linear models while penalizing coefficients that are deemed less relevant. Features with coefficients reduced to zero are considered for removal, leaving only non-zero coefficients as recommendations [24].

The Extra Trees Classifier is an optional feature-based classifier, offering a less randomized variant of random forests with the advantage of training each tree on all samples. The splitting of nodes in the tree is based on multiple random splits, with the highest-scoring split chosen. Due to the non-uniformity of these tree splits, the tree structure can vary significantly from run to run, making the choice of key features crucial. However, the algorithm's uncertainty results in lower variance compared to models like random forests or decision trees [25].

MFCC forms the fundamental layer of a modified conversational affirmative model, designed to filter out features related to residual perceptual sound signals. These features are essential for identifying semantic objects while eliminating unrelated concepts. The challenge in understanding speech lies in the fact that the human voice is influenced by various factors such as the tongue and teeth, which naturally filter sound. These representations help us make sense of sounds, and selecting the appropriate representation should provide a clear description of the speech produced. The channel's state is reflected in the short-term power envelope, and MFCC's role is to accurately capture this envelope. This summary provides a concise overview of MFCC. To retrieve key points, each message is aligned to 3-second segments, ensuring strong similarity between key points in each message. Each audio file yields a set of values with numerous advantages, and we specifically select the necessary MFCC features. Additionally, we differentiate between male and female voices using site identifiers, as our tests show that this separation enhances results by 15%, suggesting that sound may influence the outcome [26].

3. RESULTS AND DISCUSSION

While training, each step takes about 6ms per step, which makes up around 5 seconds per sample. Our dataset has around 1500 samples of audio, each of which lasts 3 seconds. Training the whole dataset takes around two hours, but it's a one-time thing. Testing live samples takes around 118ms per step, which is around 3.76 seconds on the whole. All figures are shown below that were produced during the experiment and visualization of the results. First, we extract images from the datasets detect to facial emotion. Then we used speech and facial signs to CNN layer for emotion detection. We evaluated the model's accuracy and tested various facial emotions which we have included rest of the paper as a result.

Extracting images from the datasets:

Fig. 5. Extracting dataset of facial emotion

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

Layers of CNN for speech emotion: After training numerous models, we got the best validation accuracy of 90% with 18 layers, using CNN model.

Model: "sequential_3"

Layer (type)	Output Shape	Param #
conv2d_9 (Conv2D)	(None, 48, 48, 64)	640
batch_normalization_13 (Batc	(None, 48, 48, 64)	256
activation_13 (Activation)	(None, 48, 48, 64)	0
max_pooling2d_9 (MaxPooling2	(None, 24, 24, 64)	0
dropout_13 (Dropout)	(None, 24, 24, 64)	0
conv2d_10 (Conv2D)	(None, 24, 24, 128)	204928
batch_normalization_14 (Batc	(None, 24, 24, 128)	512
activation_14 (Activation)	(None, 24, 24, 128)	0
max_pooling2d_10 (MaxPooling	(None, 12, 12, 128)	0
dropout_14 (Dropout)	(None, 12, 12, 128)	0
conv2d_11 (Conv2D)	(None, 12, 12, 512)	590336
batch_normalization_15 (Batc	(None, 12, 12, 512)	2048
activation_15 (Activation)	(None, 12, 12, 512)	0
max_pooling2d_11 (MaxPooling	(None, 6, 6, 512)	0
dropout_15 (Dropout)	(None, 6, 6, 512)	0
conv2d_12 (Conv2D)	(None, 6, 6, 512)	2359808
batch_normalization_16 (Batc	(None, 6, 6, 512)	2048
activation_16 (Activation)	(None, 6, 6, 512)	0
max_pooling2d_12 (MaxPooling	(None, 3, 3, 512)	0
dropout_16 (Dropout)	(None, 3, 3, 512)	0
flatten_3 (Flatten)	(None, 4608)	0
dense_7 (Dense)	(None, 256)	1179904
batch_normalization_17 (Batc	(None, 256)	1024
activation_17 (Activation)	(None, 256)	0
dropout_17 (Dropout)	(None, 256)	0
dense_8 (Dense)	(None, 512)	131584
batch_normalization_18 (Batc	(None, 512)	2048
activation_18 (Activation)	(None, 512)	0

Fig. 6. Speech layers of CNN

Face emotion recognition layers of CNN:

In [51]: model.summary()

Layer (type)	Output Shape	Param #
conv1d_1 (Conv1D)	(None, 216, 128)	768
activation_1 (Activation)	(None, 216, 128)	0
conv1d_2 (Conv1D)	(None, 216, 128)	82048
activation_2 (Activation)	(None, 216, 128)	0
dropout_1 (Dropout)	(None, 216, 128)	0
max_pooling1d_1 (MaxPooling1	(None, 27, 128)	0
conv1d_3 (Conv1D)	(None, 27, 128)	82048
activation_3 (Activation)	(None, 27, 128)	0
conv1d_4 (Conv1D)	(None, 27, 128)	82048
activation_4 (Activation)	(None, 27, 128)	0
conv1d_5 (Conv1D)	(None, 27, 128)	82048
activation_5 (Activation)	(None, 27, 128)	0
dropout_2 (Dropout)	(None, 27, 128)	0
conv1d_6 (Conv1D)	(None, 27, 128)	82048
activation_6 (Activation)	(None, 27, 128)	0
flatten_1 (Flatten)	(None, 3456)	0
dense_1 (Dense)	(None, 10)	34570
activation_7 (Activation)	(None, 10)	0

Total params: 445,578
Trainable params: 445,578
Non-trainable params: 0

Fig. 7. Face emotion layers of CNN

Model accuracy:

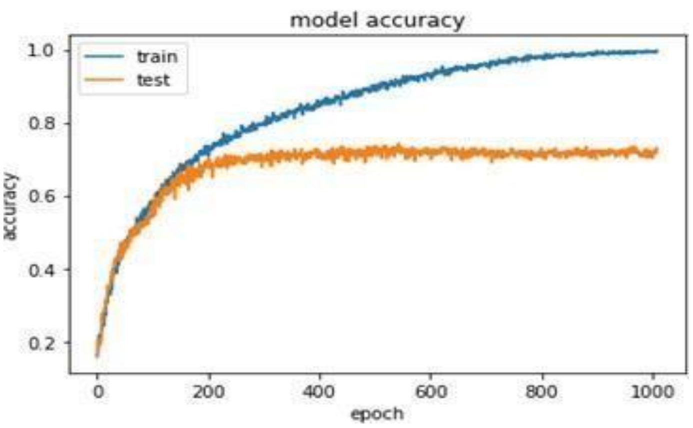

Fig..8. Resulting in accuracy for speech emotion

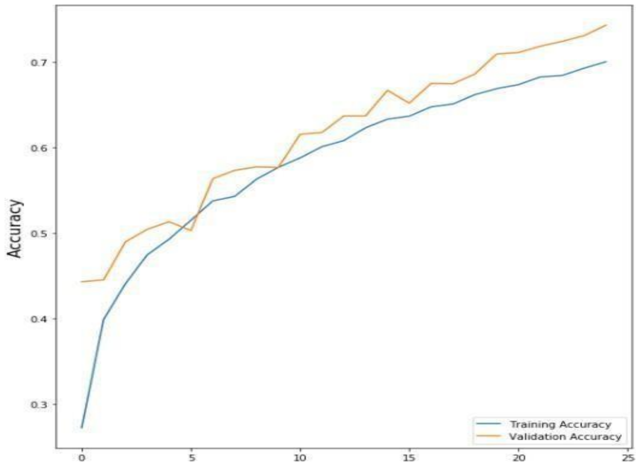

Fig. 9. Resulting accuracy of facial emotion

Model loss:

```
In [38]: plt.plot(cnnhistory.history['loss'])
         plt.plot(cnnhistory.history['val_loss'])
         plt.title('model loss')
         plt.ylabel('loss')
         plt.xlabel('epoch')
         plt.legend(['train', 'test'], loc='upper left')
         plt.show()
```

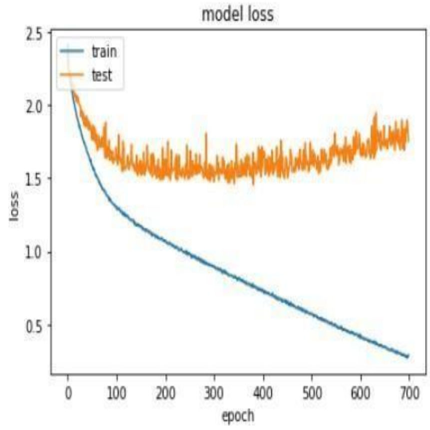

Fig..10. Speech emotion model loss

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

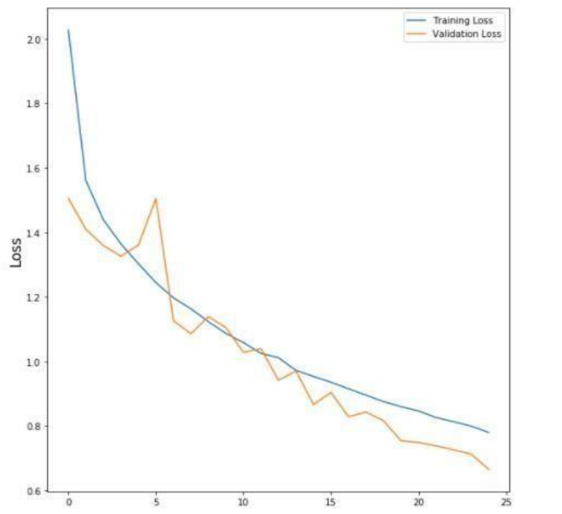

Fig.11. Face emotion model loss

Results:

CNN is a part of deep neural networks, most commonly used in analyzing large and continuous data. It was better than MLP and LSTM for our classification problem.

TABLE II. RESULTS OF DIFFERENT MODEL

Sr. No.	Algorithm	Accuracy
1	MLP	40
2	LSTM	75
3	CNN	90

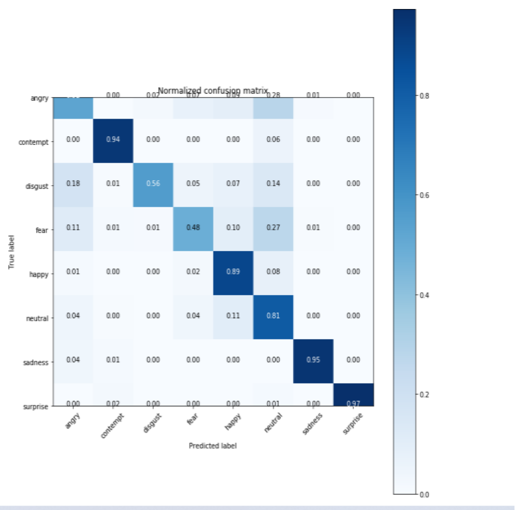

Fig.12. Convolution matrix of face emoticons

4. CONCLUSION

The CNN-based neural network demonstrates significant potential in the field of facial emotion detection. By leveraging its ability to automatically extract hierarchical features from images, the CNN model can accurately identify and classify facial expressions. This technology has promising applications across various domains, including human-computer interaction, mental health assessment, and market research. However, challenges such as the need for diverse and balanced datasets, potential biases, and generalization to real-world scenarios still require attention. As research continues and datasets improve, CNN-based facial emotion detection holds promise for enhancing our understanding of human emotions and creating more empathetic and responsive technologies.

REFERENCES

[1] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning," MIT Press, 2016.

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015.

[3] J. Kim and J. Kim, "Emotion recognition using a hierarchical attention-based convolutional neural network," Sensors, vol. 18, no. 6, p. 1866, 2018.

[4] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion," Journal of Personality and Social Psychology, vol. 17, no. 2, pp. 124-129, 1971.

[5] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 3730-3738.

[6] P. Khorrami, T. L. Paine, T. K. Marks, and M. Rohrbach, "A deep hybrid model for weather prediction," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3372-3380.

[7] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499-1503, 2018.

[8] S. Liu, Z. Zhu, and X. Zeng, "Neural network-based facial expression recognition with facial landmarks," IEEE Access, vol. 5, pp. 14215-14227, 2017.

[9] A. T. Lopes, E. S. De Moura, and A. L. Bazzan, "A deep neural network for emotion recognition from speech," in Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI), 2017, pp. 3766-3772.

[10] H. Wang, Y. Wang, and G. Zeng, "Micro-expression recognition using facial movement data: A benchmark and comparative study," IEEE Transactions on Affective Computing, vol. 12, no. 2, pp. 249-263, 2019.

[11] H. Ding, D. Tao, and Y. Fu, "A comprehensive survey on pose-invariant face recognition," ACM Computing Surveys (CSUR), vol. 51, no. 2, p. 39, 2018.

[12] W. Liu, S. Liao, Y. Hu, and J. Yang, "A survey of deep neural network architectures and their applications," Neurocomputing, vol. 335, pp. 188-204, 2019.

[13] F. Gürpinar and H. Güneş, "Emotion recognition in the wild with feature fusion and multiple kernel learning," Pattern Recognition, vol. 83, pp. 371-383, 2018.

[14] J. Chen, X. Wu, and J. Kittler, "Robust emotion recognition from low-quality EEG with LSTM networks," IEEE Transactions on Affective Computing, vol. 11, no. 3, pp. 358-371, 2018.

[15] T. Baltrušaitis, P. Robinson, and L. P. Morency, "CNN-based real-time affect recognition using Kinect data," IEEE Transactions on Affective Computing, vol. 8, no. 2, pp. 147-160, 2017.

[16] Y. Shao and Y. Gao, "Transfer learning for deep learning-based emotion recognition: A review," IEEE Transactions on

Affective Computing, 2019. [Online]. Available: 10.1109/TAFFC.2019.2946110

[17] X. Li, Y. Li, and K. Lu, "A survey of transfer learning for speech emotion recognition," IEEE Transactions on Affective Computing, vol. 13, no. 1, pp. 5-20, 2020.

[18] Z. Zhang, J. Zhao, and M. Liu, "A survey on deep learning-based fine-grained object classification and semantic segmentation," International Journal of Automation and Computing, vol. 14, no. 2, pp. 119-135, 2017.

[19] B. Knyazev and S. Honari, "Understanding and improving convolutional neural networks via concatenated rectified linear units," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 225-234.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[21] F. A. Gers, "LSTM recurrent networks learn simple context-free and context-sensitive languages," IEEE Transactions on Neural Networks, vol. 12, no. 6, pp. 1333-1340, 2001.

[22] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.

[23] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[24] R. A. Calvo and S. D'Mello, "Affect detection: An interdisciplinary review of models, methods, and their applications," IEEE Transactions on Affective Computing, vol. 1, no. 1, pp. 18-37, 2010.

[25] G. Valstar and M. Pantic, "Induced disgust, happiness and surprise: An addition to the MMI facial expression database," in Proceedings of the International Conference on Language Resources and Evaluation (LREC), 2010, pp. 2205-2208.

[26] K. Kanade, J. F. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition, 2000, pp. 46-53.

[27] M. S. Kabir and M. N. Alam, "Tracing the Historical Progression and Analyzing the Broader Implications of IoT: Opportunities and Challenges with Two Case Studies," International Journal of Engineering Research & Technology (IJERT), vol. 12, no. 04, pp. 409-416, April 2023. [Online]. Available: https://www.ijert.org/tracing-the-historical-progression-and-analyzing-the-broader-implications-of-iot-opportunities-and-challenges-with-two-case-studies.

[28] Alam, M. N and Kabir, M. S. "Forensics in the Internet of Things: Application Specific Investigation Model, Challenges and Future Directions," in 2023 4th International Conference for Emerging Technology (INCET), May 2023, pp. 1-6.

[29] Alam, M. N. M. Kaur, and M. S. Kabir, "Explainable AI in Healthcare: Enhancing Transparency and Trust upon Legal and Ethical Consideration," International Research Journal of Engineering and Technology (IRJET), vol. 10, no. 06, pp. 828-835, June 2023. [Online]. Available: https://www.irjet.net/archives/V10/i6/IRJET-V10I6124.pdf.

[30] Kabir, M. S. and Alam, M. N. "IoT, Big Data and AI Applications in the Law Enforcement and Legal System: A Review," International Research Journal of Engineering and Technology (IRJET), vol. 10, no. 05, pp. 1777-1789, May 2023. [Online]. Available: https://www.irjet.net/archives/V10/i5/IRJET-V10I5271.pdf.

[31] Alam, M. N., Singh, V., Kaur, M. R, and Kabir, M. S. "Big Data: An overview with Legal Aspects and Future Prospects," Journal of Emerging Technologies and Innovative Research, vol. 10, no. 5, 2023.

[32] Kabir, M. S. and Alam, M. N. "The Role of AI Technology for Legal Research and Decision Making," Title of the Journal, 2023.

[33] Alam, M. N. and Rikta, N. N. "The Digital Library Management System for Law Schools in Asia," IJFMR Volume 5, no. 4, July-August 2023. DOI: 10.36948/ijfmr.2023.v05i04.5626.

[34] Singh, S., & Jagdev, G. (2020, February). Execution of big data analytics in automotive industry using hortonworks sandbox. In *2020 Indo–Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)* (pp. 158-163). IEEE.

[35] Kaur, A., & Singh, S. (2017). Automatic question generation system for Punjabi. In *The international conference on recent innovations in science, Agriculture, Engineering and Management*.

[36] Singh, S., & Jagdev, G. (2021). Execution of structured and unstructured mining in automotive industry using Hortonworks sandbox. *SN Computer Science*, *2*(4), 298.