Optimizing Facial Expression Recognition through Effective Preprocessing Techniques

Lakshminarayanan Meena[1], Thambusamy Velmurugan[2]

[1]PG Department of Computer Applications, Dwaraka Doss Goverdhan Doss Vaishnav College, Chennai, India
[2]PG and Research Department of Computer Science, Dwaraka Doss Goverdhan Doss Vaishnav College, Chennai, India
Email: semeen31@gmail.com, velmurugan_dgvc@yahoo.co.in

Abstract

Analyzing human facial expressions using machine vision systems is indeed a challenging yet fascinating problem in the field of computer vision and artificial intelligence. Facial expressions are a primary means through which humans convey emotions, making their automated recognition valuable for various applications including man-computer interaction, affective computing, and psychological research. Pre-processing techniques are applied to every image with the aim of standardizing the images. Frequently used techniques include scaling, blurring, rotating, altering the contour of the image, changing the color to grayscale and normalization. Followed by feature extraction and then the traditional classifiers are applied to infer facial expressions. Increasing the performance of the system is difficult in the typical machine learning approach because feature extraction and classification phases are separate. But in Deep Neural Networks (DNN), the two phases are combined into a single phase. Therefore, the Convolutional Neural Network (CNN) models give better accuracy in Facial Expression Recognition than the traditional classifiers. But still the performance of CNN is hampered by noisy and deviated images in the dataset. This work utilized the preprocessing methods such as resizing, grayscale conversion and normalization. Also, this research work is motivated by these drawbacks to study the use of image pre-processing techniques to enhance the performance of deep learning methods to implement facial expression recognition. Also, this research aims to recognize emotions using deep learning and show the influences of data pre-processing for further processing of images. The accuracy of each pre-processing methods is compared, then combination between them is analysed and the appropriate preprocessing techniques are identified and implemented to see the variability of accuracies in predicting facial expressions.

Keywords

Facial Expression Recognition, Preprocessing Techniques, Normalization, Convolutional Neural Network (CNN), Deep Neural Networks (DNN)

1. Introduction

Facial Expression Recognition (FER) is a field of study that plays a crucial role in various human-computer interaction applications and emotional understanding. Recognizing and interpreting human emotions from facial expressions involves several components and challenges [1] [2].

Figure 1 explains the various components of a FER model as follows. It is essential to collect a large and diversified dataset as input. To make sure the model learns to recognize a broad range of expressions, it should cover a variety of emotions. The data gathered are typically unorganized and originate from several sources [3] [4]. Preprocessing of the data is mandatory. It must be standardized and cleaned up before given as input to the machine learning model. Preprocessing is often employed to decrease complexity of the applied algorithm and to increase accuracy of prediction [5] [6]. The preprocessing techniques proposed by this research work is a combination of resizing the raw input facial image, converting the resized image to grayscale, finally normalizing that image. The preprocessed image is then given as input to the CNN model, which classify them into specific expressions such as happy, sad, anger, disgust, surprise, fear and neutral.

There are many challenges in FER of which Intra-Class Variance and Inter-Class Variation are major challenges. Intra-Class Variance is minimizing the variation within the same class of expressions from different individuals is a challenge. People can express the same emotion in diverse ways, making it hard to standardize these expressions. Inter-Class Variation is maximizing the differences between expressions of different emotions across various individuals is also challenging. Different individuals might express different emotions in ways that are visually similar. FER techniques constantly evolve to address these challenges [7]. Deep learning methods, especially Convolutional Neural Networks (CNNs), have shown promising results in automatically learning features from facial images, improving accuracy in recognizing and categorizing expressions [8] [9]. These techniques can learn hierarchical representations, helping to deal with the complexities of inter-class and intra-class variations.

Despite the challenges, ongoing research and technological advancements continue to refine FER methodologies, making them more accurate and applicable in various domains, from security to emotional intelligence applications. The steps outlined provide a comprehensive guide for building a facial expression recognition system using machine learning. Each step mentioned below is crucial for creating an effective and accurate model for recognizing expressions.
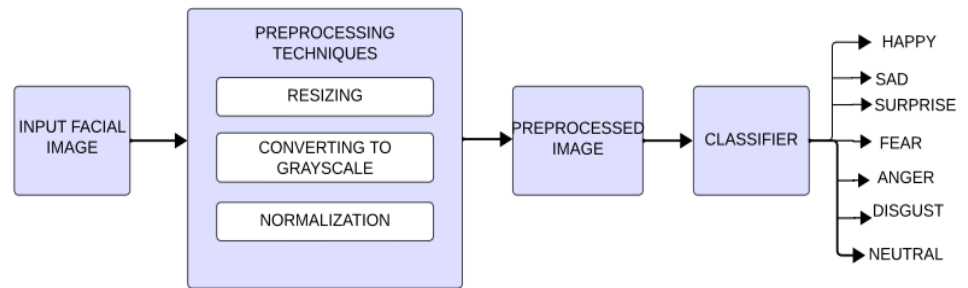
Figure 1. Components of Facial Expression Recognition (FER).

Step 1: Defining clearly the purpose and scope essential for setting the right objectives and understanding the context in which the model will be used.

Step 2: Gathering a diverse and sizable dataset is vital. It should cover various emotions to ensure the model learns to detect a wide range of expressions.

Step 3: Preparing the data involves cleaning and standardizing it. Each data type might require specific preprocessing techniques to make it suitable for machine learning algorithms.

Step 4: Correctly labelled data is fundamental for supervised learning. Ensuring accurate annotations of emotions is crucial for the model's learning process.

Step 5: Splitting the dataset into training, validation, and testing subsets helps in training the model, tuning hyper parameters, and evaluating its performance.

Step 6: Choosing an appropriate model or algorithm for the emotion detection task, considering the specific characteristics of the dataset and problem.

Step 7: Designing the architecture of the selected model, specifying layers, neurons, and activation functions, ensuring to suit the problem's complexity.

Step 8: Training the model on the training dataset involves adjusting the model's parameters and optimizing its performance.

Step 9: Assessing the model's performance using various evaluation metrics on the test dataset is crucial to determine its accuracy and generalization capabilities.

Step 10: Integrating the model into target application or system for real-world use.

Step 11: Models should be periodically retrained with new data to maintain accuracy and relevancy.

Step 12: Being mindful of ethical implications and ensuring the responsible and fair use of the model is essential.

By following the above-mentioned steps, we can create a robust and effective facial expression recognition system [10] [11]. It is important to note that machine learning is an iterative process, and refinement might be necessary at various stages to achieve the best results.

The organization of this paper is as follows. A review of some of the publications that are relevant to this topic is given in Section 2. The research resources and methods are described in Section 3. It includes the description of the dataset utilized for the research, the pre-processing techniques, and the outcome of using the pre-processing techniques on the input facial images. Furthermore, it

provides more information on the different steps involved in the preprocessing techniques to covert raw image to a form that enhances the prediction of facial expressions. The experimental results and their explanations are discussed in the Section 4. The last and the most important is Section 5 which presents the conclusions of this research work with some of its findings which will be essential for future research.

2. Literature Survey

A number of researches carried out by many peoples in the globe in the same field. Here, some of the papers which utilise this research oriented are discussed. But, a research work was done by Lopes *et al.* [12], in which a facial expression recognition system was proposed that uses a combination of standard method, like Convolutional Network and specific image pre-processing steps. Experiments showed that the combination of both Spatial and Intensity Normalization procedures with synthetic sample achieved an accuracy of 92% which was highest when compared to other preprocessing techniques thereby increasing the accuracy of the method significantly and additionally it takes less time to train the model. Five types of data input were tested, and the accuracy was compared. A paper presented by Shin *et al.* [13] compares the accuracy of facial expression recognition system on five different types of input data namely: raw, histogram equalization, isotropic smoothing, diffusion-based normalization, difference of Gaussian. Concluding that for the preprocessing method of the input image, the histogram equalization method showed the most reliable performance in all the network models.

The research work done by Yao Qian *et al.* [14] proposes a new preprocessing algorithm with local binary pattern for facial expression recognition. The skin color model is first established to extract the face region, and the complete face region is obtained by using the cumulative projection. Then all the images are normalized by rotating the inclined faces thereby removing the effect of light resulting in desired facial input samples, followed by facial features extraction with rotation invariance uniform local binary pattern. Finally, the facial expressions are classified by the support vector machine classifier using Matlab to implement the algorithm on Indian male facial expression database. The effectiveness of the proposed algorithm is evident from achieving an accuracy of 72.75% in predicting facial expressions.

A paper presented by Zhao *et al.* [15], employs the independent component analysis (ICA) technique to improve the performance of Locally Linear Embedding (LLE) algorithm in facial expression recognition. The preprocessing is initiated by representing the face images by some independent components and filtering the noise from them. The work also proposes a Supervised LLE (SLLE) algorithm to learn the hidden manifold such as pose variations, illumination conditions, and facial expressions. Based on the cluster information and Euclidean distances between the data, SLLE creates neighborhood graphs for the data. It uses the same embedding step as LLE. In the final phase, the research work

extracts an implicit nonlinear mapping from the ICA space to the embedded manifold using a Generalized Regression Neural Network (GRNN). Experiments performed on the JAFFE database yield encouraging findings.

3. Methods and Materials

This research work discusses about the methods and other required information needed to obtain desired accuracy in predicting the facial expressions. Preprocessing is standardization of raw input facial images. Two sets of facial images are required to train and test the model. To efficiently train the FER system a large number facial images are taken from available repositories and preprocessed. Later to test the model, facial images can be downloaded from social media or captured which is also preprocessed. The resultant facial image after preprocessing is fed to the CNN model to accurately predict the seven basic facial expressions namely happy, sad, fear, anger, surprise, disgust and neutral.

3.1. Preprocessing Methods

Preprocessing techniques play a vital role in improving the accuracy of facial expression recognition models. A comprehensive overview of various image enhancement techniques each of which plays an important role in improving image quality and making images more suitable for specific applications are as follows:

Gaussian Smoothing is a method that reduces noise by averaging neighboring pixels. Replacing each pixel's value with the median value of its neighborhood to reduce noise is done by Median Filtering. Wavelet Denoising is decomposing the image into different frequency bands to eliminate noise from specific bands. Histogram Equalization improves contrast by spreading out the most frequent intensity values [16] [17] [18].

Contrast Stretching widens the range of pixel intensity values to improve contrast. Normalization is sometimes called contrast stretching or histogram stretching. Normalization transforms an n-dimensional grayscale image with intensity values in the range, into a new image with intensity values in the range Nearest Neighbour Interpolation is simple but may result in pixelation. Bilinear Interpolation averages the nearest four pixels to determine a new pixel's value [19] [20] [21]. Bicubic Interpolation utilizes a more complex mathematical model to estimate pixel values during resizing.

Gray World Assumption assumes the average color in an image should be a shade of gray. White Balance adjusts color balance based on the color temperature of the light source. Color Transfer transfers color characteristics from a reference image to the target image. Thresholding divides an image into foreground and background based on a certain threshold. Edge Detection identifies edges and boundaries in an image [22] [23] [24]. Region Growth expands regions based on certain criteria to segment the image.

Each technique has its strengths and limitations, and their effectiveness often depends on the specific application and the quality of the input image. In many

cases, a combination of these techniques might be used to achieve the desired image enhancement or processing goals. Based on the selection of the preprocessing techniques the accuracy of prediction of facial expressions by the system also varies. Identifying the best preprocessing techniques is therefore essential to enhance the performance of the system.

3.2. Dataset for Preprocessing

In a FER system two types of datasets are mandatory, one for training the system and the other for testing or validating the system. The dataset for training the model must be large in number which we can be downloaded from the repositories that are available. In this research work the dataset for training is downloaded from FER2013 [25] [26] [27].

Table 1 shows the details of the image used here for testing the model and for preprocessing. An average of ten students in the age group of 9 - 10 was asked to pose with different facial expressions and their facial images were captured. The images were cropped to a size of 2 KB focusing on the facial region alone. Then these facial images were utilized as raw image input for the proposed preprocessing techniques.
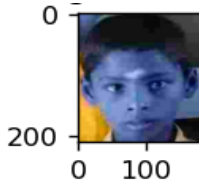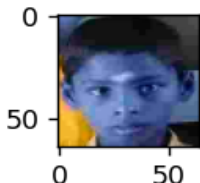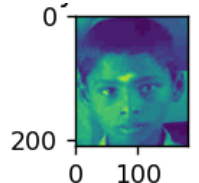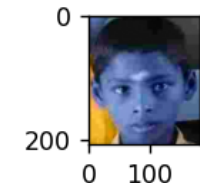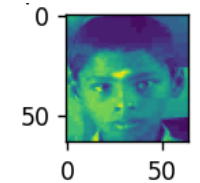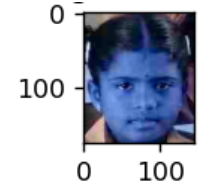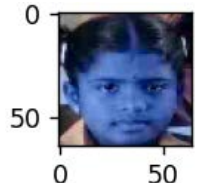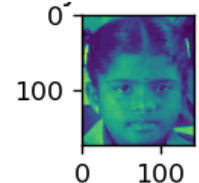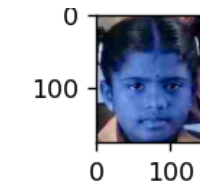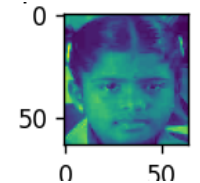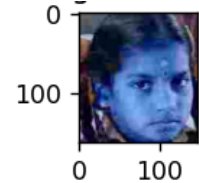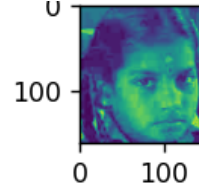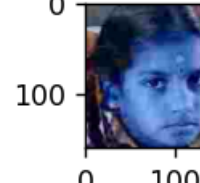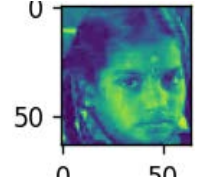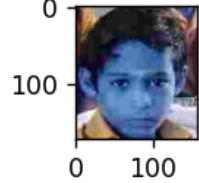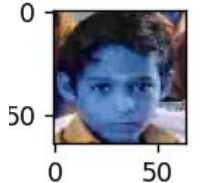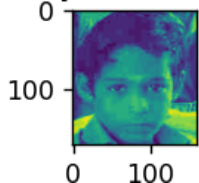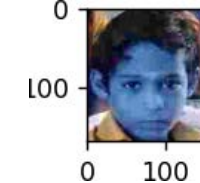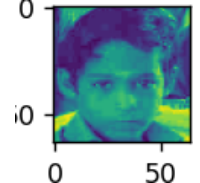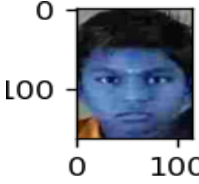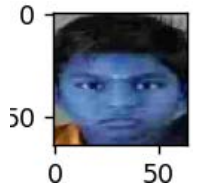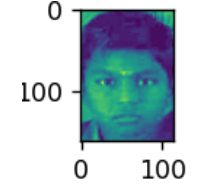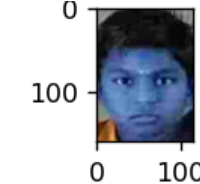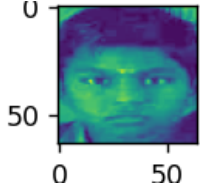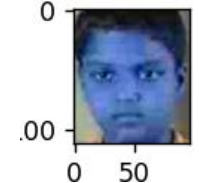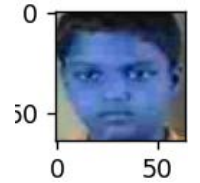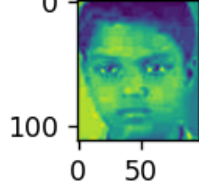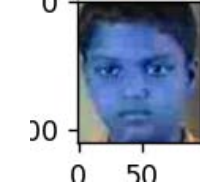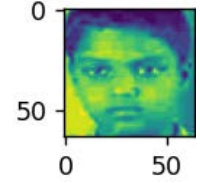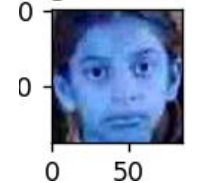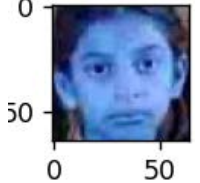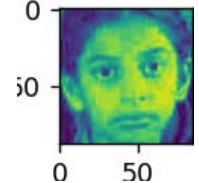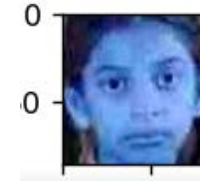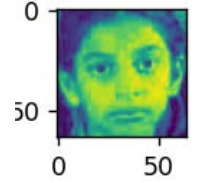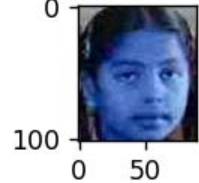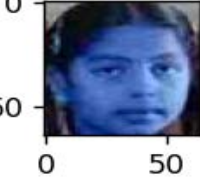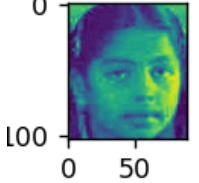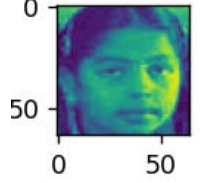
3.3. Proposed Preprocessing Technique

Technical features of this Facial Expression Recognition system include the usage of hardware resources, freely available in Google Colab. The dataset can be downloaded from Kaggle and the same is processed using the python package Tensorflow 2.0 to train the model. The dataset for testing the model can be created by capturing facial images from real world or can be downloaded from any social media. The preprocessing can be accomplished with Python using the following libraries: 1) Pandas and NumPy: It is used to manipulate data which is mandatory in python for all the Machine Learning tasks. 2) Matplotlib: Data and the performance of the model can be visualized using this. 3) OpenCV: It is a library of programming functions for real-time computer vision [28] [29]. The preprocessing procedure includes resizing, converting to grayscale followed by normalization. The result of the coding for preprocessing is shown in Table 2.

Table 1. Characteristics of dataset for preprocessing.

Dataset characteristics	Features of captured facial images
Number of subjects	10 (6 female and 4 male)
Age group	09 to 10 years
Region of origin of subjects	Indian
Image size	2 KB
Number of basic expressions	7 (Happy, Anger, Fear, Sad, Disgust, Surprise & Neutral)
Nature of expressions	Posed

Table 2. Stepwise preprocessing of facial images.

ORIGINAL IMAGE	RESIZED IMAGE	GRAYSCALE IMAGE	NORMALIZED IMAGE	FINAL IMAGE

Continued

The programming for preprocessing is done in Visual Studio Code using Python. The preprocessing begins with resizing the raw facial image in terms of the desired pixels followed by converting the image to grayscale. The purpose of converting the image to grayscale is that image processing operations works on one plane of image data at a time in grayscale images whereas in RGBA images the operations are applied on each of the four image planes and then the results are combined. Moreover, gray scale images need to process only 1/4 of the data compared to the color image. This data reduction allows the algorithm to run in a reasonable amount of time [30] [31]. The only drawback of grayscale images is the loss of color information which is not needed in recognition of facial expressions. After converting to gray scale, the facial image is normalized where each pixel value in the image is divided by 255. This will normalize the image values so that they range from 0 to 1, instead of 0 to 255. This is often useful for machine learning tasks, as it can make the images more comparable and easier to process. For example, if a happy facial image is 255 pixels wide and 255 pixels high, and each pixel has a value ranging from 0 to 255, then the code **img = img/255** will divide each pixel value by 255. This will result in an image that is still 255 pixels wide and 255 pixels high, but each pixel will now have a value ranging from 0 to 1. Therefore, when training a machine learning model to recognize happy facial images, it would be helpful if all the happy facial images had the same pixel values. Thus, normalization ensures that all the images have the same range of pixel values, which can make the training process easier for the machine learning model. Both the train and test data must be preprocessed to enhance the accuracy of predicting the emotions by the facial expression recognition model [32] [33].

4. Results and Discussions

In this research work, python libraries such OpenCV and TensorFlow on GPU NVIDIA version 375.74 from nvidia-375 is used to perform preprocessing and function the CNN model. In this work the preprocessing stage includes three steps namely, 1) resizing, 2) converting the image to grayscale and 3) finally

normalization of the facial image. The result of preprocessing in predicting facial expressions is discussed in this section

Figure 2 and **Figure 3** show the result of preprocessing. The raw facial image is given as input for preprocessing, where the image is first resized to a uniform pixel then it is converted to grayscale thereby saving storage space. As the size of data handled in FER is enormous. Finally, the grayscale image is normalized which enhances the performance of the FER system in accurately predicting the facial expressions.

Table 3 shows the effect of different preprocessing techniques on the accuracy of predicting the facial expressions by the model. The seven basic facial expressions considered for prediction by the model are anger, disgust, fear, happy, sad, surprise and contempt. The accuracy of predicting each class of emotion is very low when input facial images not preprocessed. When only cropping and resizing is done the accuracy of prediction fairly increases. After only converting the image to grayscale the accuracy rate increases furthermore. Performing only normalization as a preprocessing technique produces a slight increase in accuracy. But the best accuracy rate is achieved only when all the three stages namely, resizing, converting to grayscale and normalization are performed together in preprocessing stage. The average accuracy of prediction of all the seven classes of emotions is least when the images are not preprocessed, is medium when the images are only converted to grayscale and is the highest when the images are resized, converted to grayscale, and normalized.
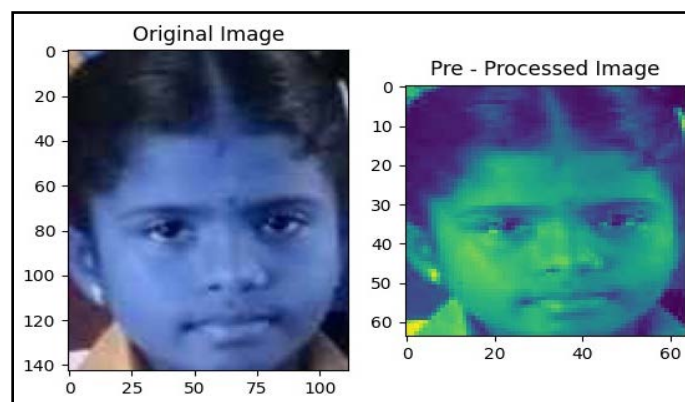
Figure 2. Facial-image-1 before and after preprocessing.

Table 3. Preprocessing techniques and accuracy.

Pre-processing Step	Anger %	Disgust %	Fear %	Happy %	Sad %	Surprise %	Neutral %	Average %
Raw image	89.01	52	44.23	67.45	30.45	84.5	60.68	61.19
Cropping & resizing	98.89	86.7	73.23	100	65.23	92.89	77.78	84.96
Converting to Grayscale	93.25	87.15	82.25	100	73.35	100	82.25	88.32
Normalization	100	91.56	83.18	100	65.45	65.86	84.23	84.33
Resizing + Grayscale + Normalization	100	94.43	89.12	100	87.56	100	88.76	94.27
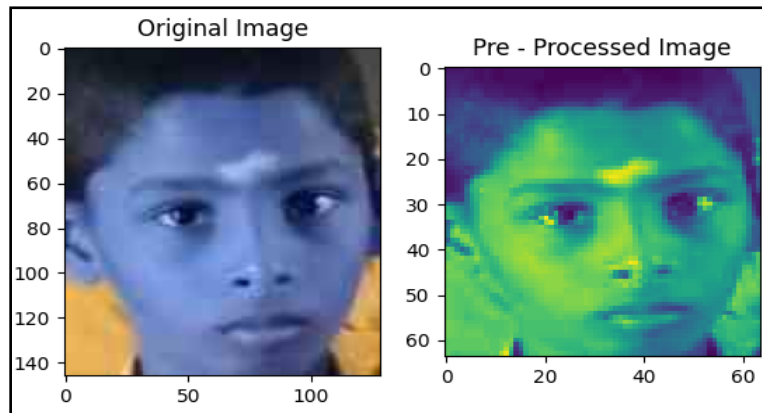
Figure 3. Facial-image-2 before and after preprocessing.

Table 4 illustrates the training and validation losses obtained at the end of every five epochs. Training loss and validation loss are metrics used in machine learning to evaluate how effectively a model fits both training and test data respectively. When all the training data has gone through the network once, it is called an epoch. How well a model fits the training data in each epoch is measured by training loss. How well a model fits new data at the end of each epoch is measured by validation loss. The difference between the training and validation loss curves shows how much the model overfits or underfits the data. A wide gap indicates that the model is overfitting. Also, it is observed that as the number of epochs increases both the training and validation losses decreases and the difference between the curves is negligible during epochs ranging from 55 to 65 when the model is neither overfitting nor underfitting. This state is achieved only when the input facial images both for training and testing the model is essentially pre-processed.

Figure 4 shows that after preprocessing the images, both validation and training losses are getting reduced. The difference between the training and validation loss curves shows how much the model overfits or underfits the data. This shows that the preprocessing techniques supports in resolving over fitting problem. This can be treated as a good baseline. When the epoch reaches 50 the difference between the training and validation loss curves have almost reduced therefore it can be concluded that the prediction model is accurate and efficient. However, there is still scope for advancement in terms of attempting to eliminate the difference between training and validation errors. A systematic strategy to further enhance the model would involve conducting a search for an optimized hyper-parameter across the entire set of potential hyperparameters. The learning rate, batch size, number of epochs, and other parameters may all be modified to obtain the best possible combination. The huge requirement of computer resources is a major compromise in the search for hyperparameter. The HParams dashboard in TensorBoard provides several tools to help with this process of identifying the best experiment or most promising sets of hyperparameters.
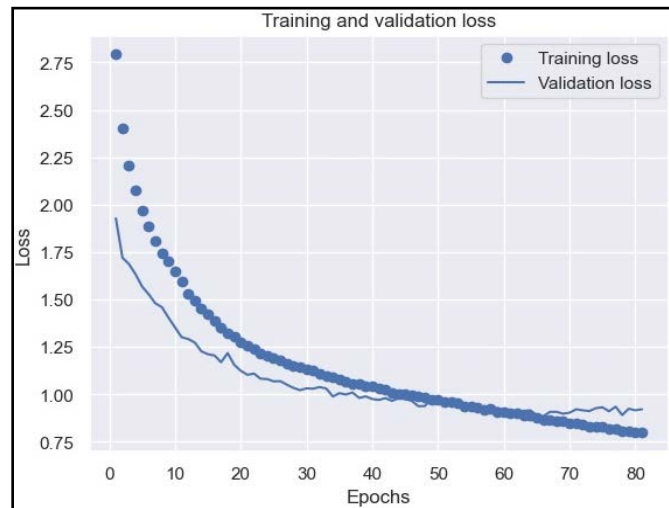
Figure 4. Training and validation loss across the epochs.

Table 4. Results of training and validation loss across epochs.

S. No	Epoch	Training Loss	Validation Loss
1	5	1.9401	1.5158
2	10	1.6078	1.3239
3	15	1.4065	1.1954
4	20	1.2691	1.1488
5	25	1.1864	1.1173
6	30	1.1274	1.0917
7	35	1.0848	1.0639
8	40	1.0481	1.0323
9	45	1.0173	1.0101
10	50	0.9795	0.9825
11	55	0.8657	0.8659
12	60	0.8462	0.8462
13	65	0.8243	0.8243
14	70	0.8065	0.8542
15	75	0.7723	0.8834
16	80	0.7501	0.9156

Table 5 illustrates the training and validation accuracy of the model at the end of every five epochs. The training accuracy of the model is a measure of how well the model performs on the training dataset. It is the ratio of the number of correctly predicted instances to the total number of instances in the training set. Validation accuracy is a measure of how well the model generalizes to new, unseen data. It is calculated using a separate dataset called the validation set, which the model has not seen during training. The table shows a clear increase in both the accuracies as the epochs progresses. At the end of 80th epoch, the accuracies are maximized with training accuracy increasing from 32.81 to 71.45. In a simi-

lar trend, we also observe the validation accuracy reaching a value of 68.15 at 80th Epoch from its initial value of 42.11. The preprocessing techniques plays a vital role in increasing the accuracy of prediction which is evident from the experimental results obtained.

From **Figure 5**, it is evident that the accuracy of both the training dataset and the test dataset increases to the best with the increase in the epochs. In the initial value of epoch both the training and validation accuracies are very low. When the accuracy reaches to an ultimate value the training can be stopped at that epoch value. From the experimental results it is evident that the accuracy rate is enhanced by the preprocessing techniques proposed in this work.
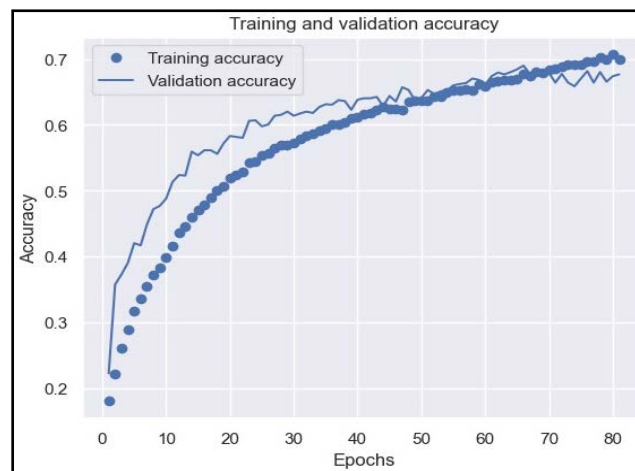
Figure 5. Training and validation accuracies across epochs.

Table 5. Training and validation accuracy across epochs.

S. No	Epoch	Training Accuracy	Validation Accuracy
1	5	32.81	42.11
2	10	40.78	49.79
3	15	47.34	55.17
4	20	52.16	56.22
5	25	54.89	59.01
6	30	57.82	60.61
7	35	59.42	62.57
8	40	60.57	63.69
9	45	62.82	64.72
10	50	64.87	65.18
11	55	66.43	66.54
12	60	67.56	67.58
13	65	68.53	68.61
14	70	69.92	69.01
15	75	70.51	68.81
16	80	71.45	68.15

5. Conclusion

This research work proposes a facial expression recognition system in which seven different facial expressions of different people from FER2013 dataset have been analyzed. The captured facial images are preprocessed from which the features are extracted and the CNN model detects the facial expressions. Accuracy is used as a metric to evaluate the performance of the system. Tests reveal that the accuracy of the model increases greatly when the normalization procedures are combined with rescaling and grayscale conversion. According to the results, this method provides a simpler solution and achieves higher accuracy when compared to the traditional classifiers that make use of the same facial expression database. Furthermore, training takes less time. In future the proposed work can be extended to be tested in other databases thereby achieving a cross database validation and the accuracy of the model can further be enhanced with more appropriate combinations of preprocessing techniques.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Vyas, A.S., Prajapati, H.B. and Dabhi, V.K. (2019) Survey on Face Expression Recognition Using CNN. 2019 5*th International Conference on Advanced Computing & Communication Systems* (*ICACCS*), Coimbatore, 15-16 March 2019, 102-106. https://doi.org/10.1109/ICACCS.2019.8728330

[2] Modi, S. and Bohara, M.H. (2021) Facial Emotion Recognition Using Convolution Neural Network. 2021 5*th International Conference on Intelligent Computing and Control Systems* (*ICICCS*), Madurai, 6-8 May 2021, 1339-1344. https://doi.org/10.1109/ICICCS51141.2021.9432156

[3] Srinivas, P.V.V.S. and Mishra, P. (2022) A Novel Framework for Facial Emotion Recognition with Noisy and de Noisy Techniques Applied in Data Pre-Processing. *International Journal of System Assurance Engineering and Management*, 1-11. https://doi.org/10.1007/s13198-022-01737-8

[4] Mishra, P. and Srinivas, P.V.V.S. (2021) Facial Emotion Recognition Using Deep Convolutional Neural Network and Smoothing, Mixture Filters Applied during Preprocessing Stage." *IAES International Journal of Artificial Intelligence*, **10**, 889. https://doi.org/10.11591/ijai.v10.i4.pp889-900

[5] Tsai, H.-H. and Chang, Y.-C. (2018) Facial Expression Recognition Using a Combination of Multiple Facial Features and Support Vector Machine. *Soft Computing*, **22**, 4389-4405. https://doi.org/10.1007/s00500-017-2634-3

[6] Prabhu, K., Sathish Kumar, S., Sivachitra, M., Dineshkumar, S. and Sathiyabama, P. (2022) Facial Expression Recognition Using Enhanced Convolution Neural Network with Attention Mechanism. *Computer Systems Science & Engineering*, **41**, 415-426. https://doi.org/10.32604/csse.2022.019749

[7] Bartlett, M.S., Littlewort, G., Frank, M., Lainscsek, C., Fasel, I. and Movellan, J. (2005) Recognizing Facial Expression: Machine Learning and Application to Spon-

taneous Behavior. *IEEE, Computer Society Conference on Computer Vision and Pattern Recognition* (*CVPR'*05), **2**, 568-573.

[8] Fathima, A. and Vaidehi, K. (2020) Review on Facial Expression Recognition System Using Machine Learning Techniques. In: Satapathy, S., Raju, K., Shyamala, K., Krishna, D. and Favorskaya, M., Eds., *Advances in Decision Sciences, Image Processing, Security and Computer Vision: International Conference on Emerging Trends in Engineering* (*ICETE*), Springer, Vol. 4, 608-618. https://doi.org/10.1007/978-3-030-24318-0_70

[9] Ojala, T., Pietikainen, M. and Maenpaa, T. (2002) Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**, 971-987. https://doi.org/10.1109/TPAMI.2002.1017623

[10] Shan, C.F., Gong, S.G. and McOwan, P.W. (2009) Facial Expression Recognition Based on Local Binary Patterns: A Comprehensive Study. *Image and Vision Computing*, **6**, 803-816. https://doi.org/10.1016/j.imavis.2008.08.005

[11] Mayya, V., Pai, R.M. and Pai, M.M.M. (2016) Automatic Facial Expression Recognition Using DCNN. *Procedia Computer Science*, **93**, 453-461. https://doi.org/10.1016/j.procs.2016.07.233

[12] Lopes, A.T., De Aguiar, E. and Oliveira-Santos, T. (2015) A Facial Expression Recognition System Using Convolutional Networks. 2015 28*th SIBGRAPI Conference on Graphics, Patterns and Images*, Salvador, 26-29 August 2015, 273-280. https://doi.org/10.1109/SIBGRAPI.2015.14

[13] Shin, M., Kim, M. and Kwon, D.-S. (2016) Baseline CNN Structure Analysis for Facial Expression Recognition. 2016 25*th IEEE International Symposium on Robot and Human Interactive Communication* (*RO-MAN*), New York, NY, 26-31 August 2016, 724-729. https://doi.org/10.1109/ROMAN.2016.7745199

[14] Qian, Y., Ren, F.J. and Quan, C.Q. (2013) A New Preprocessing Algorithm and Local Binary Pattern Based Facial Expression Recognition. *Proceedings of the* 2013 *IEEE/SICE International Symposium on System Integration*, Kobe, 15-17 December 2013, 239-244. https://doi.org/10.1109/SII.2013.6776757

[15] Zhao, Q.J., Zhang, D. and Lu, H.T. (2005) Supervised LLE in ICA Space for Facial Expression Recognition. *IEEE, International Conference on Neural Networks and Brain*, **3**, 1970-1975.

[16] Lopes, A.T., De Aguiar, E., De Souza, A.F. and Oliveira-Santos, T. (2017) Facial Expression Recognition with Convolutional Neural Networks: Coping with Few Data and the Training Sample Order. *Pattern Recognition*, **61**, 610-628. https://doi.org/10.1016/j.patcog.2016.07.026

[17] Jabid, T., Kabir, M.H. and Chae, O. (2010) Robust Facial Expression Recognition Based on Local Directional Pattern. *ETRI Journal*, **32**, 784-794. https://doi.org/10.4218/etrij.10.1510.0132

[18] Mollahosseini, A., Hasani, B., Salvador, M.J., Abdollahi, H., Chan, D. and Mahoor, M.H. (2016) Facial Expression Recognition from World Wild Web. 2016 *IEEE Conference on Computer Vision and Pattern Recognition Workshops* (*CVPRW*), Las Vegas, NV, 26 June-1 July 2016, 58-65. https://doi.org/10.1109/CVPRW.2016.188

[19] Abdulrahman, M. and Eleyan, A. (2015) Facial Expression Recognition Using Support Vector Machines. 2015 23*nd Signal Processing and Communications Applications Conference* (*SIU*), Malatya, 16-19 May 2015, 276-279. https://doi.org/10.1109/SIU.2015.7129813

[20] Connie, T., Al-Shabi, M., Cheah, W.P. and Goh, M. (2017) Facial Expression Recognition Using a Hybrid CNN–SIFT Aggregator. In: Phon-Amnuaisuk, S., Ang, S.P. and Lee, SY., Eds., *International Workshop on Multi-Disciplinary Trends in Artificial Intelligence*, Springer, 139-149.
https://doi.org/10.1007/978-3-319-69456-6_12

[21] Michel, P. and El Kaliouby, R. (2003) Real Time Facial Expression Recognition in Video Using Support Vector Machines. *Proceedings of the 5th International Conference on Multimodal Interfaces*, 258-264.
https://doi.org/10.1145/958432.958479

[22] Shafira, S.S., Ulfa, N. and ArifWibawa, H. (2019) Facial Expression Recognition Using Extreme Learning Machine. 2019 *3rd International Conference on Informatics and Computational Sciences* (*ICICoS*), Semarang, 29-30 October 2019, 1-6.
https://doi.org/10.1109/ICICoS48119.2019.8982443

[23] Ullah, S., Jan, A. and Khan, G.M. (2021) Facial Expression Recognition Using Machine Learning Techniques. 2021 *International Conference on Engineering and Emerging Technologies* (*ICEET*), Istanbul, 27-28 October 2021, 1-6.
https://doi.org/10.1109/ICEET53442.2021.9659631

[24] Feng, X., Hadid, A. and Pietikäinen, M. (2004) A Coarse-to Fine Classification Scheme for Facial Expression Recognition. In: Campilho, A. and Kamel, M., Eds., *Image Analysis and Recognition*, Springer, 668-675.
https://doi.org/10.1007/978-3-540-30126-4_81

[25] Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z. and Matthews, I. (2010) The Extended Cohn Kanade Dataset (CK+) A Complete Dataset for Action Unit and Emotion-Specified Expression. 2010 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition—Workshops*, San Francisco, CA, 13-18 June 2010, 94-101. https://doi.org/10.1109/CVPRW.2010.5543262

[26] Guo, G. and Dyer, C.R. (2005) Learning from Examples in the Small Sample Case: Face Expression Recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* (*Cybernetics*), **35**, 477-488. https://doi.org/10.1109/TSMCB.2005.846658

[27] Niu, Z. and Qiu, X. (2010) Facial Expression Recognition Based on Weighted Principal Component Analysis and Support Vector Machines. *IEEE, International Conference on Advanced Computer Theory and Engineering*, 174-178.

[28] Song, K.T. and Chen, Y.W. (2011) A Design for Integrated Face and Facial Expression Recognition. *IEEE, Annual conference on Industrial Electronics Society*, 4306-4311.

[29] Viola, P. and Jones, M. (2004) Robust Real-Time Face Detection. *International Journal of Computer Vision*, **57**, 137-154.
https://doi.org/10.1023/B:VISI.0000013087.49260.fb

[30] John, A., Abhishek, M.C., Ajayan, A.S., Sanoop, S. and Kumar, V.R. (2020) Real-Time Facial Emotion Recognition System with Improved Preprocessing and Feature Extraction. 2020 *Third International Conference on Smart Systems and Inventive Technology* (*ICSSIT*), Tirunelveli, 20-22 August 2020, 1328-1333.
https://doi.org/10.1109/ICSSIT48917.2020.9214207

[31] Li, J.N. and Lam, E.Y. (2015) Facial Expression Recognition Using Deep Neural Networks. 2015 *IEEE International Conference on Imaging Systems and Techniques* (*IST*), Macau, 16-18 September 2015, 1-6.
https://doi.org/10.1109/IST.2015.7294547

[32] Shan, K., Guo, J., You, W., Lu, D. and Bie, R. (2017) Automatic Facial Expression Recognition Based on a Deep Convolutional-Neural-Network Structure. 2017 *IEEE 15th International Conference on Software Engineering Research, Management and*

Applications (*SERA*), London, 7-9 June 2017, 123-128.
https://doi.org/10.1109/SERA.2017.7965717

[33] Hu, Y.Y., Zeng, X.P., Huang, Z.Y. and Dong, X. (2021) A Preprocessing Method of Facial Expression Image under Different Illumination. 2021 13*th International Conference on Communication Software and Networks* (*ICCSN*), Chongqing, 4-7 June 2021, 318-322. https://doi.org/10.1109/ICCSN52437.2021.9463605