2024 Second International Conference on Networks, Multimedia and Information Technology (NMITCON)

EmoSense: Deep Analysis of Real-Time Facial Expression for Multi-Class Emotions

Sharvesh S
Computing Technologies,
School of computing,
SRM Institute of Science and Technology,
Kattankulathur, Chennai.
ss9502@srmist.edu.in

Kaileshwar M
Computing Technologies,
School of computing,
SRM Institute of Science and Technology,
Kattankulathur, Chennai.
km0151@srmist.edu.in

Arthi B
Computing Technologies,
School of computing,
SRM Institute of Science and Technology,
Kattankulathur, Chennai.
arthib@srmist.edu.in

Jothikumar C
Computing Technologies,
School of computing,
SRM Institute of Science and Technology,
Kattankulathur, Chennai.
jothikuc@srmist.edu.in

***Abstract*— This study explores the efficacy of CNNs in classifying images into predefined categories, highlighting advancements in automated image recognition technology. The project aims to develop a robust CNN model that can accurately identify and classify images based on their content, which has significant implications for fields requiring automated image analysis. A sequential CNN architecture featuring layers of convolution, max pooling, and dropout, complemented by fully connected layers for feature learning and classification, developed and validated using TensorFlow and Keras. The model was trained and tested on a curated dataset comprising diverse images categorized into seven distinct classes, preprocessed to normalize image sizes and pixel values to enhance model training efficiency. 8L-CThe model used in the research includes 2L-CNN, 3L-CNN, and 4L-CNN architecture are used. In that, better accuracy of 62.2% for 3-layer architecture received. The CNN demonstrated high accuracy and generalizability in image classification, proving capable of handling complex pattern recognition tasks, underscoring the potential of CNNs in transforming image processing applications.**

Index Terms — Facial expression recognition, emotion recognition, computer vision, machine learning, deep learning, facial features.

I. Introduction

Automated Facial Expression Recognition presents a complex and compelling challenge in computer vision, driven by the intricate and diverse nature of human emotional expressions. This paper introduces a novel CNN-based architecture tailored for automated facial expression recognition, aiming to address the inherent variability in human expressions. The research delves into the impact of CNN parameters on model performance, providing valuable insights into optimal design strategies specifically tailored for this task.

Recent advances in CNN-based facial expression recognition are reviewed here, with a focus on exploring architectural variations and their implications on recognition accuracy. By identifying and addressing key bottlenecks, particularly the limitations of basic CNN architectures, novel approaches have achieved significant improvements in test accuracy, setting a new benchmark for the field.

While the adoption of advanced CNN architectures offers clear advantages in classification performance, it's important to acknowledge potential drawbacks such as increased computational complexity and resource requirements, underscoring the need for efficient model optimization and scalability in practical applications. Overall, this research contributes to the evolving landscape of automated facial expression recognition by advancing understanding of optimal CNN design strategies and pushing the boundaries of recognition accuracy in real world scenarios.This investigation underscores the importance of fine-tuning CNN architectures to handle the complexity and variability inherent in facial expressions, ultimately contributing to the development of more reliable and robust automated facial expression recognition systems.

This work includes, but is not limited to, the following contributions:

(i) The study demonstrates the efficacy of CNNs in accurately classifying images into predefined categories, showcasing advancements in automated image recognition.
(ii) A sequential CNN architecture with convolution, max pooling, and dropout layers was developed and validated using *TensorFlow* and *Keras*.
(iii) The model was prepared on an organized dataset of different pictures in seven classes, accomplishing an outstanding precision of 62.2% with a 3-layer engineering.
(iv) The CNN model showed high accuracy and generalizability, proving effective in handling complex pattern recognition tasks and highlighting its transformative potential in image processing.

The remainder of the paper is structured as follows: Section 2 contains the pertinent work. The model's architecture and construction techniques are described in detail in Section 3. The experimental setup, a description of the dataset, and measurements are presented in Section 4. Section 5 presents the research's conclusion and findings. Future work are presented in section 6.

979-8-3503-7289-2/24/$31.00 ©2024 IEEE

II. Related Work

Common methods in Facial Expression Recognition (FER) [1] systems include face detection, smoothing, PCA, Local Binary Patterns (LBP), Optical Flow (OF), and Gabor filters. Researchers use various databases with 2D or 3D images. Comparing FER results is challenging due to different databases, data splits, and methodologies, though studies with similar procedures allow meaningful comparisons. CNN [2] are widely used to address facial expression classification challenges. This research introduces a new CNN-based FER architecture, evaluated on public databases (CK+, MUG, RAFD). The outcomes demonstrate that the CNN method outperforms cutting-edge techniques in terms of recognition rates for visual expression recognition. Photos of faces in a single posture were used to train the model.

Authors	Methodologies	Limitations
Daniel Canedo (2019) [1]	Data Augmentation, Active Appearance Model	It struggles with variations in facial expressions.
Ali Douik et al. (2017) [2]	Artificial Perception Techniques, Computer Vision	It's susceptibility to errors due to variations in lighting, pose, and occlusions
Ying-Li Tian et al. (2010) [3]	Geometric feature-based methods and appearance-based methods	Sensitivity to variations in pose and occlusions, while appearance-based methods can struggle with changes in lighting and expression intensity.
Christopher Pramerdorfer et al. (2016) [4]	Ensemble Learning, Graphical Models for Sequence-Based FER	It requires significantly more computational resources and memory compared to single models.
Wafa Mellouka et al. (2020) [5]	Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs)	It can struggle with capturing long-term dependencies in facial sequences, limiting their effectiveness in understanding nuanced emotional expressions over extended durations.
Zi-Yu Huang (2010) [6]	Residual neural networks (ResNet), Residual block architectures	Can suffer from increased complexity and difficulty in training deeper networks due to challenges with gradient propagation and optimization.
Sabrina Begaj (2020) [7]	Deep Learning Approach, Traditional FER Approach	For training, it needs a significant amount of labelled data and powerful computing power.
Pawel Tarnowski et al. (2017) [8]	Six action units (AU), calculated by the Kinect device	Limitations in accurately capturing subtle and complex facial expressions due to the device's resolution and sensitivity to movement.

Table 1. Existing work limitations in facial emotion recognition.

Geometric feature-based and appearance-based methods are the two main approaches [3]. Face characteristics such as the nose, eyes, mouth, and brows are examples of geometric features. A feature vector is created by combining the forms and locations of these features. Databases should include individual action units and combinations, especially those with co-articulation effects, which may perform poorly. The study aims to clarify early vs. late integration benefits by analyzing current CNN-based FER techniques [4] and conducting a comparative study of CNN architectures in uniform environments. It identifies bottlenecks and paths for enhancing FER performance, considering factors like illumination, age, stance, expression intensity, and occlusion variability. Example photos from the FER2013 dataset demonstrate these variations, showing anger, disgust, fear, happiness, sadness, surprise, and neutral expressions.

Facial Expression Recognition methods from the past 20 years fall into two main categories [5]. Various datasets, each with unique characteristics, are used for Facial Emotion Recognition, requiring large datasets and input from psychologists. This study uses the squeeze-and-excitation module (SENet) [6] with ResNet-18 to create a lightweight FER model, reducing trainable parameters compared to larger models like ResNet-50. The approach is evaluated on AffectNet and RAF-DB datasets, which include diverse facial emotion data. Cross-database validation shows that transfer learning significantly improves prediction accuracy for smaller datasets. The neural network extracts fine lines in shallow layers and focuses on mouth and nose regions in deeper layers.

Automatic recognition of facial expressions is widely studied but challenging due to unique emotional expressions and obstacles like head pose variations, luminosity, age, gender, background, and occlusion from sunglasses, scarves, or skin conditions [7]. Traditional methods include geometric and texture features like local binary patterns (LBP), facial action units (FAC), local directional patterns (LDA), and Gabor wavelets. Facial expressions [8] are crucial for recognizing emotions and play a key role in non-verbal communication. Overall limitations of the existing work are listed in table 1. This research aims to recognize seven basic emotional states based on facial expressions.

III. Materials and Methods

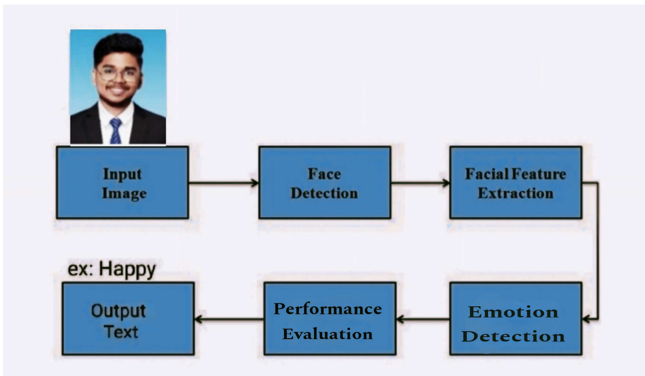


Fig. 1. Architecture diagram of facial emotion recognition.

The architecture diagram of facial emotion recognition is provided in the diagram 1. The steps involved in this system includes face detection, feature extraction, emotion detection, performance evaluation and output response. The pre-processing and 3 layer CNN is explained in this section.

A. Preprocessing

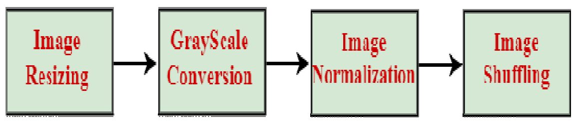


Fig. 2. Pre-processing steps of facial emotion recognition.

In the pre-processing [9] phase of the image classification project, a meticulous approach was taken to prepare the dataset for optimal CNN performance. The dataset's varying image sizes and qualities required key pre-processing steps to standardize the input data and enhance model training effectiveness. The various steps included in this process is given below:

Image Resizing: Initially, all images were resized to a uniform dimension of 48x48 pixels to ensure consistency across the input data and it ensures consistent image dimensions for the model, reducing computational complexity and memory usage during training.

Grayscale Conversion: Each image was converted to grayscale from RGB channel to a single intensity channel. This simplification decreases model complexity, speeds up training, and focuses on essential image features, helping to reduce overfitting.

Normalization: Scaling pixel values to a uniform range (e.g., 0 to 1) stabilizes training dynamics and accelerates convergence, enhancing model robustness.

Shuffling: Randomizing training sample order prevents the model from learning sequential biases, improving generalization and training stability.

These preprocessing techniques were vital for enhancing the robustness and generalization capability of the CNN [10], setting a strong foundation for effective training and accurate image classification.

B. 3L CNN Model

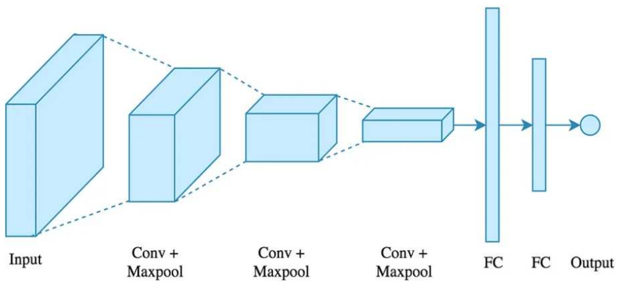


Fig. 3. Three Layer CNN model.

Fig 3. shows the proposed architecture 3 Layer CNN model. In this research developed and validated a robust CNN for the purpose of image classification, leveraging a tailored architecture to effectively recognize and classify images into one of seven categories. With numerous convolution layers and max pooling layers, the model architecture was painstakingly created to minimise dimensionality while preserving the most important properties. Dropout layers were added to each convolutional layer [11] to reduce the possibility of overfitting and guarantee that the model would perform well on untested data.

The model also incorporated dense layers towards the end to perform the final classification, with the entire network trained on a diverse dataset of labeled images [12] pre-processed to uniform dimensions of 48x48 pixels and normalized to enhance model training efficiency. The training process involved a batch size of 128 and extended over 50 epochs, demonstrating the model's learning capability through a validation process parallel to the training. The Adam optimizer was used to carry out the optimisation, and accuracy was the main metric and categorical cross-entropy the loss function to measure the model's efficacy. Upon completion of the training, the model achieved promising results, indicative of its capability to handle com- plex image classification tasks effectively.

C. Algorithm

In this section, the proposed algorithm is described which is used for classification of seven type of emotions using facial recognition. The following sections describe each of the layers of the model in more detail. Few parameters included in the algorithm [13] are n is the batch size, f_i are the filter sizes, d is the input depth, k_i are the number of filters, and p_i are the dropout rates.

<u>Step 1:</u> *Initialize Parameters*
Initialize the input tensor with a specific shape, typically including batch size, height, width, and depth (number of channels). The input layer X is given by $X \in R^{n\times 46\times 46\times 128}$.

<u>Step 2:</u> *Conv2D Layer 1*
Apply the first convolutional layer with a set of filters to extract features from the input. The weight W_1, output Z_1 and output shape is given by $W_1 \in R^{f_1\times f_1\times d\times k_1}$, $Z_1 = conv(X, W_1)$ and (n, 46, 46,128) respectively.

<u>Step 3 :</u> *Max Pooling Layer 1*
Apply max pooling with a specific filter size to reduce the spatial dimensions of the feature maps, retaining only the most significant features. The output Z_2 and the output shape is given by $Z_2 = maxpool(Z_1, f_2)$ and (n, 23, 23, 128) respectively.

<u>Step 4 :</u> *Dropout Layer 1*
Apply dropout to randomly set a fraction of input units to zero during training to prevent overfitting. The output Z_3 and the output shape is given by $Z_3 = dropout(Z_2, p_1)$ and (n, 23, 23, 128) respectively.

<u>Step 5 :</u> *Conv2D Layer 2*
Apply the second convolutional layer with another set of filters to further extract features from the output of the previous layer. The weight W_2, output Z_4 and output shape is given by $W_2 \in R^{f_3\times f_3\times k_1\times k_2}$, $Z_4 = conv(Z_3, W_2)$ and (n, 21, 21, 256) respectively.

<u>Step 6 :</u> *Max Pooling Layer 2*
Apply max pooling again to further reduce the spatial dimensions of the feature maps. The output Z_5 and the

output shape is given by $Z_5 = maxpool(Z_4, f_4)$ and (n, 10, 10, 256) respectively.

Step 7 : *Dropout Layer 2*
Apply another dropout layer to prevent overfitting during training. The output Z_3 and the output shape is given by $Z_6 = dropout(Z_5, p_2)$ and (n, 10, 10, 256) respectively.

Step 8 : *Conv2D Layer 3*
Apply the third convolutional layer with additional filters to extract even more complex features. The weight W_2, output Z_7 and output shape is given by $W_3 \in R^{f_5 \times f_5 \times k_2 \times k_3}$, $Z_7 = conv(Z_6, W_3)$ and (n, 8, 8, 512) respectively.

Step 9: *Max Pooling Layer 3*
Apply max pooling to further downsample the spatial dimensions of the feature maps. The output Z_8 and the output shape is given by $Z_8 = maxpool(Z_7, f_6)$ and (n, 4, 4, 512) respectively.

Step 10 : *Dropout Layer 3*
By randomly changing a portion of the input units to zero during training, a final dropout layer can be applied to lessen overfitting. The output Z_9 and the output shape is given by $Z_9 = dropout(Z_8, p_3)$ and (n, 4, 4, 512) respectively.

Step 11 : *Flatten the Output*
Flatten the output tensor from the previous layer to create a 2D tensor where each row represents a single sample. The output Z_{10} and the output shape is given by $Z_{10} = flatten(Z_9)$ and (n, 4×4×512) respectively.

Step 12 : *Fully Connected Layer*
Apply a dense layer with weights W_{fc} and biases b_{fc} to produce the final class scores for 7 categories. The Weight W_{fc}, bias b_{fc}, output Z_{10} and the output shape is given by $W_{fc} \in R^{(4\times4\times512)\times7}$, $b_{fc} \in R^7$, $Z_{11} = Z_{10} \cdot W_{fc} + b_{fc}$ and (n, 7) respectively.

Step 13 : *Softmax*
Apply the softmax activation function to convert the class scores into probabilities. The output Z_{10} and the output shape is given by $\hat{Y} = softmax(Z_{11})$ and (n, 7) respectively.

This step transforms the output of the convolutional layers into a 7-class classification output, with each element in the final output tensor representing the probability of the input belonging to one of the 7 predefined categories.

IV. Experiment & Results

A. Experimental Setup

For the image classification project using a convolutional neural network (CNN), the combination of *TensorFlow* and *Keras* was utilized for efficient computation and data processing. *TensorFlow* offers a flexible ecosystem of tools and libraries, while *Keras* simplifies tasks by acting as an interface for *TensorFlow*, emphasizing user-friendliness and modularity [14]. On the hardware side, GPUs were employed to accelerate training and testing. GPUs enable parallel processing of large data blocks, essential for high-dimensional data like images, significantly enhancing efficiency.

B. Metrics Evaluation

The metrics show how well the models work by indicating how accurate they are at predicting outcomes. The models were assessed using the following metrics: Accuracy, Weighted Average Precision (W.A.P.), Weighted Average Recall (W.A.R.), and Weighted Average F1 Score (W.A.F.). Equation (1) defines accuracy as the proportion of accurate predictions in three-layer CNN model makes.

$$Accuracy = \frac{TP + \mathrm{FP}}{TP + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \quad (1)$$

Where, TP= True Positives, TN = True Negatives, FP = False Positives and FN= False Negatives.

The output average in the weighted average technique is determined by taking the weight and the total number of occurrences in each class. By averaging the Precision for each class and accounting for the support of each class, the Weighted Average Precision is calculated. The proportion of examples in a given class to all instances in the dataset is known as support.

$$Wt.\ \mathrm{Avg.\ Precision} = \frac{\sum_{i=1}^{n} y_i \frac{TP_i}{TP_i + FP_i}}{Y} = \frac{\sum_{i=1}^{n} y_i\ \mathrm{P}_i}{Y} \quad (2)$$

Where, y_i represents the no. of data points in 'i'th class, TP_i represents True Positive of 'i'th class, FP_i represents False Positive of 'i'th class and Y represents total no. of data points in the dataset.

Similarly, WAR and WAF are calculated as shown in equations (3) and (4).

$$Wt.\ \mathrm{Avg.\ Recall} = \frac{\sum_{i=1}^{n} y_i \frac{TP_i}{TP_i + FN_i}}{Y} = \frac{\sum_{i=1}^{n} y_i\ \mathrm{R}_i}{Y} \quad (3)$$

$$Wt.\ \mathrm{Avg.\ F1} = \frac{\sum_{i=1}^{n} y_i\ \mathrm{F1}_i}{Y} \quad (4)$$

C. Dataset Description

The dataset used in the facial emotion recognition is a mutli-class dataset which includes classes like angry, disgust, fear, neutral, happy, sad and surprise [16]. The distribution of the dataset across the different classes is listed in the table. There are 27091 images in the dataset which is divided into 70% for training and 30% for testing.

After splitting the total number of images in the each category includes 18964 for training and 8127 for testing.

	Training	Testing
Angry	2685	1150
Disgust	390	167
Fear	2633	1128
Neutral	2085	893
Happy	5413	2319
Sad	3964	1698
Surprise	1797	769

Table 2. Distribution of instances in the 7 different emotion categories.

D. Results and Discussions

This study compares the effectiveness of many CNN designs for a particular task like context-dependent facial emotion recognition. Table 3 compares the performance of 2, 3, and 4 Layer CNN models for facial emotion recognition using Accuracy, Precision, Recall, and F1 Score. The 2 Layer CNN [15] has the lowest performance with an accuracy of 22.09%, precision of 21.31%, recall of 22.56%, and F1 score of 21.54%. The 4 Layer CNN shows moderate performance with an accuracy of 54.58%, precision of 40.62%, recall of 52.09%, and F1 score of 45.76%. The 3 Layer CNN [17] outperforms both, with the highest accuracy of 62.22%, precision of 66.26%, recall of 60.48%, and F1 score of 61.33%. This indicates that the 3 Layer CNN strikes the best balance between precision and recall, providing the most accurate and reliable performance among the three models. Increasing complexity from 2 to 3 layers significantly improves performance, while adding a fourth layer offers less improvement.

Model	Accuracy	W.A.P.	W.A.R.	W.A.F.
2 Layer CNN	22.09	21.31	22.56	21.54
4 Layer CNN	54.58	40.62	52.09	45.76
3 Layer CNN	**62.22**	**66.26**	**60.48**	**61.33**

Table. 3. Performance comparison of different models for facial emotion recognition.

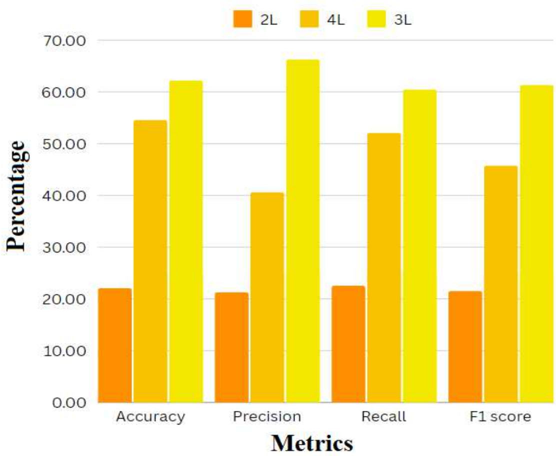
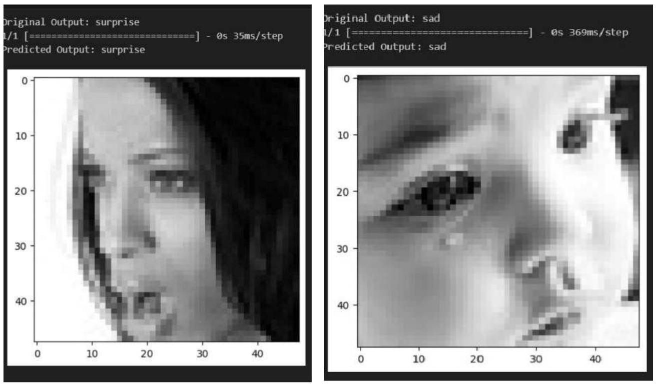


Fig. 4. Performance comparison of different models for facial emotion recognition.

Figure. 4 compare the performance metrics of three CNN architectures: 2 Layer CNN (2L), 3 Layer CNN (3L), and 4 Layer CNN (4L), across Accuracy, Precision, Recall, and F1 score. The horizontal axis represents these metrics, while the vertical axis shows the percentage values. Each metric is depicted with three bars: orange for 2L, yellow-orange for 4L, and yellow for 3L. The 3 Layer CNN (3L) consistently achieves the highest values across all metrics, followed by the 4 Layer CNN (4L) [18]. The 2 Layer CNN (2L) has the lowest performance in all metrics. This indicates that the 3 Layer CNN (3L) is the most effective architecture for the given image classification task.

E. User Interface

The figure 5. showcases two images processed by a facial emotion recognition system, with the left image labeled and predicted as "surprise" and the right image labeled and predicted as "sad." Both images are displayed in grayscale with their corresponding original and predicted outputs indicated above. The system accurately identifies the emotional expressions depicted in the images. Fig 6. shows a user-friendly graphical interface, enabling individuals to easily access and utilize the system's functionalities for emotion analysis in educational settings, psychological research, or human-computer interaction applications. The frontend design prioritizes usability and accessibility, ensuring users of all technical expertise levels can engage with the facial emotion recognition technology. The interface features intuitive navigation and straightforward controls, allowing users to upload images or video content for emotion analysis with just a few clicks. Thoughtful design includes clear prompts and visual cues to enhance usability.



(a) Surprise **(b) Sad**

Fig. 5. Surprise and Sad facial emotions recognition.

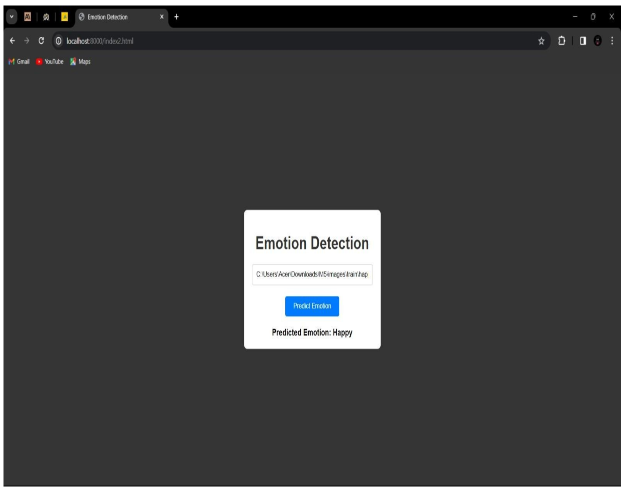


Fig. 6. User Interface of facial emotions recognition.

F. Research Discussion

Existing methods in image classification often struggle with large-scale, high-dimensional data and require extensive feature engineering. CNN models address these limitations by automatically extracting hierarchical features from raw image data. Their sequential architecture includes convolution, max pooling, and dropout layers, enhancing robustness. Utilizing curated datasets improves training efficiency and accuracy. Achieving 62.2% accuracy with a 3-layer CNN demonstrates superior effectiveness in complex pattern recognition tasks, transforming automated image analysis.

G. Limitations

The study's limitations include a relatively modest accuracy of 62.2%, indicating room for improvement in model performance. The evaluation is based on a curated dataset, which may not fully represent real-world variability. The focus on predefined categories limits the model's adaptability to new or unseen classes. Additionally, the reliance on a single posture for training may reduce the model's robustness in varied conditions. Further research is needed to enhance accuracy, generalizability, and adaptability to diverse and dynamic datasets.

VI. CONCLUSION

In conclusion, the study demonstrates the significant potential of CNN in the realm of image classification, particularly for the task of facial expression recognition. Through meticulous architectural design and comprehensive preprocessing, model achieved a commendable accuracy of 62.2% on the test set, showcasing its robustness and generalizability. This research underscores the importance of optimizing CNN parameters and pre-processing techniques to enhance model performance, contributing valuable insights to the field of automated facial emotion recognition.

VII. FUTURE SCOPE

Future improvements could include experimenting with more advanced architectures like deeper CNNs or integrating techniques such as transfer learning to boost accuracy. Enhancing the dataset to include more diverse and varied images can improve model robustness and generalizability. Incorporating data augmentation and exploring other preprocessing techniques may further optimize training efficiency. Additionally, addressing different postures and expressions can expand the model's applicability in real-world scenarios.

REFERENCS

[1]. Canedo D, Neves AJR. Facial Expression Recognition Using Computer Vision: A Systematic Review. Applied Sciences. 2019; 9(21):4678. https://doi.org/10.3390/app9214678.

[2]. A. Fathallah, L. Abdi and A. Douik, "Facial Expression Recognition via Deep Learning," 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), Hammamet, Tunisia, 2017, pp. 745-750, doi: 10.1109/AICCSA.2017.124.

[3]. Li, Ying & Kanade, Takeo & Cohn, Jeffrey & Li, Stan & Jain, Anil. (2005). Facial Expression Analysis. 10.1007/0-387-27257-7_12.

[4]. Pramerdorfer, Christopher and M. Kampel. “Facial Expression Recognition using Convolutional Neural Networks: State of the Art.” ArXiv abs/1612.02903 (2016).

[5]. Mellouk, Wafa & Wahida, Handouzi. (2020). Facial emotion recognition using deep learning: review and insights. Procedia Computer Science. 175. 689-694. 10.1016/j.procs.2020.07.101.

[6]. Huang, ZY., Chiang, CC., Chen, JH. et al. A study on computer vision for facial emotion recognition. Sci Rep 13, 8425 (2023). https://doi.org/10.1038/s41598-023-35446-4

[7]. Begaj, Sabrina & Topal, Ali & Ali, Maaruf. (2020). Emotion Recognition Based on Facial Expressions Using Convolutional Neural Network (CNN). 58-63. 10.1109/CoNTESA50436.2020.9302866.

[8]. Paweł Tarnowski, Marcin Kołodziej, Andrzej Majkowski, Remigiusz J. Rak, Emotion recognition using facial expressions, Procedia Computer Science, Volume 108, 2017, Pages 1175-1184, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2017.05.025.

[9]. K. He, X. Zhang, S. Ren and J. Sun, “Deep Residual Learning for Image Recognition”, 2016 Vegas.

[10]. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C., “SSD: Single shot multibox detector”, European conference on computer vision, ECCV, pp. 21–37, 2016.

[11]. K. Vikram and S. Padmavathi, “Facial parts detection using Viola Jones algorithm”, 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore,2017, pp. 1-4, doi: 10.1109/ICACCS.2017.8014636.

[12]. M.V. B. Martinez, “Advances, Challenges, and Opportunities in Automatic Facial Expression Recognition,”in Advances in Face Detection and Facial Image Analysis. Springer, 2016, pp. 63–100.

[13]. Hu, M.; Zheng, Y.; Yang, C.;Wang, X.; He, L.; Ren, F. Facial Expression Recognition Using Fusion Features Based on Center-Symmetric Local Octonary Pattern. IEEE Access 2019, 7, 29882–29890.

[14]. Zafeiriou, S., Zhang, C., & Zhang, Z. (2017). A survey on face detection in the wild: Past, present and future. Computer Vision and Image Understanding, 138, 1-24.

[15]. Happy, S. L., & Routray, A. (2015). Automatic facial expression recognition using features of salient facial patches. IEEE Transactions on Affective Computing, 6(1), 1-12.

[16]. S. Sivakumar, B. S. Vinay, A. Arulmurugan, U. M. Prakash, T. R. Kumar and S. Sridevi, "Secured Normalized Tagged Response for Patient Critical Care Monitoring System: A Unified approach," 2023 International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT), Faridabad, India, 2023, pp. 890-895, doi: 10.1109/ICAICCIT60255.2023.10465945.

[17]. M. S. Abd Rahim, F. Yakub, M. Omar, R. Abd Ghani, I. Dhamanti and S. Sivakumar, "Prediction of Influenza A Cases in Tropical Climate Country using Deep Learning Model," 2023 IEEE 2nd National Biomedical Engineering Conference (NBEC), Melaka, Malaysia, 2023, pp. 188-193, doi: 10.1109/NBEC58134.2023.10352612.

[18]. Soubraylu Sivakumar, S.S. Sridhar, Ratnavel Rajalakshmi, M. Pushpalatha, S. Shanmugan, G. Niranjana, “Intelligent and assisted medicine dispensing machine for elderly visual impaired people with deep neural network fingerprint authentication system”. Internet of Things. 23, ISSN 2542-6605, (2023). Doi: 10.1016/j.iot.2023.100821