Facial expression recognition system using Deep learning

Priya Choudhari ’[1], Priyanshu Giri[1], Shahzeb Khan[1]*

[a]*Department of Computer Science* & *Applications Sharda School of Computing Science* & *Engineering Sharda University Greater Noida India.*

Abstract

Facial Expression Recognition (FER) is a vital component of computer and human interaction, enabling systems to analyze human emotions based on facial cues. This research explores the implementation of a convolution neural network (CNN) based model for classifying facial expressions into seven universal categories: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral. Traditional FER systems relying on handcrafted features often fail to generalize due to challenges like facial occlusions, lighting variations, and pose changes. Deep learning, particularly CNNs, overcomes these limitations by automatically learning hierarchical features from raw image data. This study employs a structured pipeline comprising data preprocessing, augmentation, feature extraction, and classification. The dataset used contains 48x48 grayscale images, which are normalized, reshaped, and augmented to enhance model performance. Evaluation metrics such as accuracy, confusion matrix, precision, recall, and F1-score demonstrate that the CNN model achieves promising accuracy, particularly for expressions like happiness and anger, while showing lower performance for subtle emotions such as fear and disgust. Overfitting is mitigated using dropout layers and augmentation techniques. The findings underscore the potential of CNNs in FER while highlighting ongoing challenges such as class imbalance and real-time deployment. Future work aims to improve recognition of subtle emotions and robustness across diverse conditions using hybrid models and attention mechanisms.

Keywords: Facial Expression Recognition (FER), Emotion Detection, Deep Learning for FER, Convolution Neural Networks (CNN),

1. Introduction

Facial expression, the main aspect of human expression. Facial expressions allow us to interact, react and convey our reactions to the other party without using words. Expressions like happiness, sadness, anger, fear, disgust and neutral are universal across cultures. Through the movement of the facial muscles like mouth, cheeks, eyes and eyebrows we can convey our reactions and they reflect emotion. The Facial Expression Recognition (FER) is the system that uses facial expressions to automatically identify and categorize human emotions and interact. Convolution neural networks have transformed a number of domains, exhibiting exceptional efficacy in face recognition, object detection, and image categorization. *(1)*Fields like entertainment, security, healthcare, marketing, education and E-Learning, automatize industry and many more. Traditional facial expressions recognition system had several problems that arise due to handcrafted feature extraction methods and resulted to failure of analysis and ineffectiveness. Then the Deep Learning based FER become the better option, as it overcomes the drawbacks of their low scalability and poor accuracy. In artificial intelligence and technology, facial expressions recognition is important and popular in today’s technology. Deep learning has played an important role in Facial Expressions Recognition(FER) by increasing efficiency, accuracy and resilience. Conventional FER approaches used manually developed feature extraction methods that had trouble with the unique facial occlusions, changes in position, illumination. Deep learning, in particular Convolution Neural Network (CNN) makes FER system more scalable and accurate by learning patterns from the dataset (facial photos) by eliminating the need of manual feature extraction and pre-processing. Performance is more enhanced with less training with pre-trained models. Facial expressions recognition system uses various models that helps to understand the different facial features to analyze and interpret model performance. PCA (Principal Component Analysis) reduces high dimensional data into 2D or 3D visualizations making the feature distribution more understandable After extraction of features using the techniques like PCA, CNNs, KNN is responsible for classifying the expression such as happy, sad, anger etc. using various measures. However, it is simple, efficient for small datasets and requires minimal training. Even after deep learning making facial expression recognition (FER) efficient and

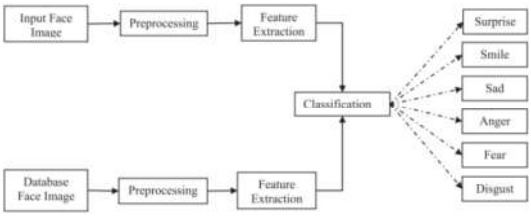

Figure 1: Architecture of facial expression recognition system

accurate than of the traditional models, many issues are still that need to be resolved.*(2)*Conventional machine learning models, such as KNN, SVM, and HOG-based methods, have trouble with occlusions, illumination, and position changes. Despite their increased accuracy, CNNs and deep learning still have problems with limited real-time processing, class imbalance, computational complexity, and data scarcity. Better methods that include facial expressions are also required for more precise emotion recognition. Variations in illumination, position, occlusions, and cultural differences make it difficult for Facial Expression Recognition (FER) algorithms to properly identify emotions. CNN and other deep learning-based models have improved accuracy, but they come with high processing costs and a need for huge datasets. Furthermore, conventional machine learning techniques like SVM and KNN are not very good at generalizing to real-world situations. Through the use of multi-modal emotion recognition, hybrid models, and sophisticated deep learning algorithms, this study seeks to create a reliable and effective FER model that enhances accuracy, generalization, and real-time performance. The purpose of this study is investigating how deep learning techniques can be used to construct an effective FER system that maximizes accuracy and performance in practical situations. The study will also cover issues including addressing occlusions like masks or spectacles, cultural differences in expressions, and real-time processing. As shown in Figure 1, the flowchart represents the architecture of the model facial expression recognition system making it easier to understand the flow of the model.

2. Literature review

Several strategies have been investigated recently by researchers to increase the precision and effectiveness of Facial Emotion Recognition (FER) systems. One of the most notable of these is the study by Revina and Emmanuel (2018), who carried out an extensive analysis of human facial expression detection techniques, highlighting the function of Support Vector Machine (SVM)-based models in particular. The significance of data preparation and augmentation strategies in improving model performance was underscored by their findings. Incorporating data enrichment techniques including image rotation, flipping, scaling, and noise addition allowed them to enhance the model's generalisation capabilities, decrease overfitting, and enrich the dataset. They demonstrated the efficacy of their approach and highlighted the significance of carefully selected and varied training data in emotion identification tasks with an impressive 99 percent accuracy rate for their SVM-based system*(3)*.

A-MobileNet, a lightweight and effective model for facial emotion recognition (FER) designed for deployment in resource-constrained contexts, was presented by Nan et al. in 2021. They were able to strike a good balance between speed and accuracy by altering the conventional MobileNet design. The model achieved accuracy rates of 84.49 percent and 88.11 percent, respectively, when tested on two benchmark datasets: RAF-DB and FERPlus. These outcomes demonstrate how well A-MobileNet performs in real-time FER tasks, which qualifies it for embedded and mobile systems with constrained processing power*(4)*.

Zhang et al. (2021) introduced a unique technique called Relative Uncertainty Learning (RUL) to address the problem of recognising ambiguous or uncertain facial expression samples. In order to increase recognition accuracy, this method focusses on learning and utilising uncertainty in face expression data. RUL aids the system in differentiating between expressions that are unambiguous and those that are unclear by modelling the relative uncertainty among samples. The technique performed better when tested on conventional datasets, demonstrating its ability to handle ambiguous emotional cues during facial expression identification values*(5)*.

In order to identify emotions in movies, Arnold Sachith and Smitha Rao (2021) suggested a hybrid CNN-LSTM architecture that attempts to extract both temporal and spatial characteristics from video data. While the Long Short-Term Memory (LSTM) network simulates the sequential nature of emotions over time, the Convolutional Neural Network (CNN) component collects spatial characteristics from individual frames. This method improves emotion perception by taking advantage of the changing context of movie sequences. However, the lack of defined performance indicators in the study limited the method's ability to be quantitatively evaluated *(6)*.

Finally, highlighting the accuracy and efficiency of a CNN-based model for facial emotion identification, Saravanan et al. (2019) put it into practice. A realistic option for a variety of scenarios, including surveillance, human-computer interaction, and healthcare monitoring, the model was praised for its short training time and adaptability for real-time applications. The benefits of adopting convolutional neural networks in FER tasks are highlighted by its adaptability and performance, particularly when prompt and ac-

curate emotion detection is needed*(7)*.

3. Methodology

The project describes the methods and methodological approach used to construct and train the model. The techniques and procedures utilised to develop and train a model that can categorise facial expressions into one of seven emotion categories—Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral—are described in this study. This assignment is classified as a multi-class classification problem since it involves deep learning. Data collecting, choosing relevant datasets, preprocessing data to improve quality and consistency, and feature extraction to identify pertinent facial patterns are some of the crucial steps in the methodology. Data distributions and model performance are better understood with the use of visualisation techniques. Convolutional Neural Networks (CNNs) are one of the deep learning approaches used in the classification process to precisely identify and classify face emotions. The project's goal is to develop a dependable and efficient FER system that can be used in practical settings.

3.1. Dataset loading and preprocessing :

3.1.1. Dataset loading

Images of greyscale face expressions with a 48x48 pixel resolution make up the dataset. The first step is to download and extract the dataset. Usually, it comes with a structured folder format or a CSV file (fer2013.csv) with pictures arranged according to emotion class*(8)*. The images and labels are loaded into memory using libraries like Pandas, NumPy, and TensorFlow/Keras. Pixel data are first parsed, then reshaped into 48x48 arrays and normalised to the [0,1] range in the case of CSV format. To guarantee a fair distribution of each emotion class, these photos are further divided into training, validation, and test sets. Anger, disgust, fear, happiness, sadness, surprise, and neutrality are the seven facial expressions that each image depicts. (Note: "Angry" should only have appeared once in the original list; it occurs twice, most likely due to an error.)

- Training set: The neural network model is trained using the training set. Helps the model in discovering the features and patterns from labelled data.
- Validation Set: Used to track the model's performance throughout training. Helps in avoiding overfitting and adjusting hyperparameters.
- Test Set: After training is finished, the test set is used. It determines how well the trained model performs in the end with unknown data.

3.1.2. Steps in Data Preprocessing:

- Normalization: The image's pixel values are normalised by dividing each one by 255, which scales them from the original range of [0,255] to a normalised range of [0,1]. Using the typical FER-2013 setup, the images were normalised to the [0,1] range and divided into training, validation, and test sets*(8)*. This crucial stage makes it possible for the neural network to train more quickly and steadily.
- Label Encoding: To translate the textual emotion labels into a numerical format, Label Encoding is utilised. This change enables the model to comprehend and process the emotion categories efficiently. The machine learning model can understand this numerical representation of the various emotions in the dataset since it is visually shown in Figure 2.

Emotion	Label (encoded)
Angry	0
Disgust	1
Fear	2
Happy	3
Sad	4
Surprise	5
Neutral	6

Figure 2: Numerical representations of emotion labels

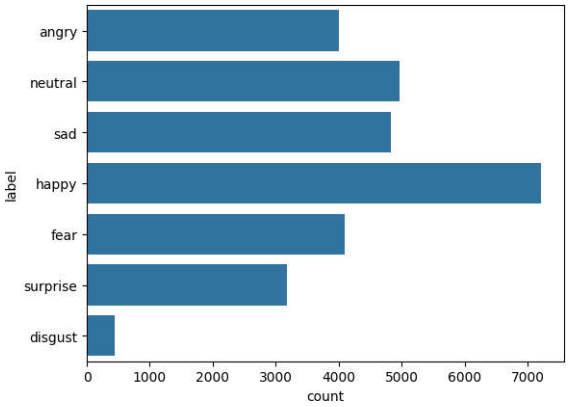

Figure 3: A Count Plot

3.1.3. Visualization: label distribution

By showing the frequency of each emotion label in the dataset, a count plot is a helpful visualisation tool that may be used to spot any class imbalances. It's simpler to see whether some emotions are over-represented or under-represented when you see how many pictures go with each face expression, such joyful, sad, furious, or neutral. Effective model training requires a balanced dataset with a comparable amount of samples for each emotion. The model could get biassed and perform poorly on minority classes if the data is biassed away from particular emotions. As shown in Figure 3, this plot is essential to the study of datasets as it represents the number of samples collected of the emotion labels.

3.1.4. ***Additional preprocessing of data***

Managing missing data: Before the training process actually starts, we must ensure that there are no missing values in the dataset. techniques like sample elimination or imputation (like filling in missing values) may be used if any data is missing. Enhancing data through augmentation: when working with image data, data augmentation is an essential step, particularly for smaller datasets. Through the creation of altered versions of the preexisting photos, this technique artificially expands the dataset.

Among these changes are:

-Rotation: To replicate various angles, photos are randomly rotated a little.

-Flipping: Images can be flipped vertically or horizontally to make the model resistant to these modifications.

-Zooming and Shifting: To make the model durable to these kinds of fluctuations, random image translation and zoom are used. Augmentation improves the model's capacity for generalization and helps avoid overfitting.*(2)*

3.2. ***Exploratory Data Analysis (EDA) :***

Exploratory data analysis (EDA) is a crucial step in any project as it helps to understand the dataset's properties, identifies the patterns and abnormalities, and gets the data ready for additional modeling. Before putting the data into a machine learning model, this procedure seeks to identify important traits, trends, and possible problems. EDA would normally entail the following for this 48x48 greyscale image collection of seven emotion classes (anger, disgust, fear, happiness, sadness, surprise, and neutrality). They help examine the label of data, understand the nature and characteristics, and identify the data quality and problems. In order to comprehend the qualities of the dataset, find patterns and irregularities, and prepare the data for further modelling, exploratory data analysis, or EDA, is an essential phase in any project (Tukey, 1977)*(9)*.They help to understand the patterns and trends in data before actually creating a model. Important EDA steps include:

1. **Dataset Overview:**
   • Shape and Structure: Analysing the dataset's dimensions, such as the quantity of samples and features (such as emotion labels and probable pixel values). This aids in comprehending the total volume of the data.
   • Types of Data: confirming each column's data type (e.g., categorical emotion labels, integer pixel values).
   • Missing Values: Looking for any data points that are missing and need to be handled appropriately (e.g., imputation or removal).

2. **Emotion label distribution :** It is essential to examine the distribution of emotion labels in the dataset, and a count plot is a useful visualisation tool for this. A balanced or unbalanced class distribution in the dataset may be easily determined by looking at the count plot, which shows the frequency of each emotion category (anger, disgust, fear, happiness, sadness, surprise, and neutrality). A biassed model that underperforms on underrepresented emotions can result from an unbalanced dataset, where some emotions contain noticeably more or fewer samples. Early detection of this imbalance enables the use of suitable techniques, such as undersampling majority classes or oversampling minority classes, to reduce bias and guarantee the model learns well across all face expressions. In order to create a reliable and equitable face expression recognition system, this stage is essential.

3. **Visualizing Sample Image :** Visualizing a representative selection of images from each emotion category offers invaluable insights into the dataset. By examining these sample photos, we gain a direct understanding of the distinct facial features associated with anger, disgust, fear, happiness, sadness, surprise, and neutrality. Visual examination of sample photos from every class enables qualitative evaluation of the diversity and quality of the data*(9)*. This visual exploration allows us to assess the subtle nuances and prominent characteristics that differentiate each emotional expression. Furthermore, it provides a qualitative evaluation of the image quality, revealing aspects like clarity, lighting, and the prominence of key facial landmarks. This visual comprehension is crucial for building intuition about the data and understanding the inherent challenges and potential for a model to effectively discriminate between the various emotional states based on facial features.

4. **Data quality issues:**
   • Noise and Artefacts: Visually examining photos to look for any visible blur, noise, or artefacts that could interfere with model training.
   • Mislabeling: Although more difficult to identify with certainty without outside confirmation, visual appearance discrepancies within the same emotion class may indicate possible mislabeling.

3.3. ***Feature Extraction :***

1. **Image reshaping:** The 48x48 greyscale photos must be formatted according to the network's specifications before they can be fed into a CNN. Here's where reshaping is useful*(10)*.
   • Scaling to 48x48 Pixels: This stage guarantees that the size of each input image is constant. Typically, CNN architectures are made to manage input tensors of fixed size. You may achieve uniformity and enable the convolutional filters to function consistently throughout the dataset by scaling

every image to 48 by 48 pixels. For the model to successfully learn spatial hierarchies of features, this standardisation is essential.

• Moulding into a 4D array (NumImages, 48, 48, 1): CNNs must know the input images' spatial dimensions and number of colour channels in order to analyse data in batches.

-NumImages: This dimension indicates how many images you are putting into the CNN at once for processing in a single batch.

- (48, 48): These measurements, which you have already scaled, show the height and breadth (in pixels) of each distinct image.

-1. This final measurement represents the quantity of colour channels. The intensity of each pixel in your greyscale photos is represented by a single channel, which ranges from black to white. For RGB colour photos, this size would be 3.

Because of this, this 4D array serves as a container for a collection of your greyscale photos, each of which is organised as a 48x48 grid of pixel intensities.

2. **Preprocessing and Reshaping:** For the training process and the model's performance to be optimised, this step is essential.

• Ensuring Proper Representation: As previously said, reshaping guarantees that the spatial organisation of the picture data is maintained and shown in a way that the convolutional layers can comprehend. The filters can move across the image and pick up local patterns thanks to the 2D structure (48x48). The input's greyscale nature is shown by the solitary channel.

• Adjustment to the [0, 1] Interval: The pixel values, which are initially between 0 and 255, are scaled to the range of 0 to 1 by dividing them by 255*(11)*. For CNN training, this normalisation provides a number of advantages:

- Faster Convergence: Normalised data speeds up the convergence of optimisation techniques, such as gradient descent, to a good solution. Training can become unstable due to steep gradients caused by big pixel values.

- Increased Stability: Networks trained using normalised data are frequently less prone to problems caused by the size of input characteristics and more stable overall.

- Contribution of Features: By ensuring that all pixel values make a more equal contribution to the learning process, normalisation keeps features with higher raw values from taking centre stage.

3. **Convolutional features:** As a result of their convolutional layers, Convolutional Neural Networks are very good at automatically learning hierarchical characteristics from raw pixel data.

• Created Feature Maps: A convolutional layer applies a series of learnable filters (small weight matrices) throughout the image's spatial dimensions after receiving a 48x48x1 input. The result of this process is feature maps. In the input image, each filter focusses on identifying a specific kind of visual characteristic at various points.

• Hierarchical Feature Learning: Usually, low-level features like edges, corners, and basic textures are learnt by the first convolutional layers. Deeper convolutional layers learn to integrate the lower-level information into increasingly intricate and abstract representations as the input passes through them. Deeper layers may learn to recognise patterns such as the enlarging of eyes, the furrowing of brows, or the curving of a smile in order to recognise facial expressions.

• Model's Acquired Knowledge: The model learns these feature maps during training as it attempts to reduce the discrepancy between its predictions and the actual emotion labels; they are not manually constructed. The CNN can create a complex and hierarchical representation of the facial emotions by employing numerous convolutional layers, capturing minute features and spatial correlations that are essential for precise categorisation.

3.4. Model Architecture Design:

A Convolutional Neural Network (CNN) is a type of deep learning model which is constructed for recognizing and classifying images*(12)*. In Facial Emotion Recognition (FER), CNN teaches itself to identify and delineate relevant features such as eyebrows, mouth, eyes and correlate them with various emotions.

3.4.1. Layer-by- layer breakdown:

• Convolutional Layers: These layers use a variety of filters to identify low-level characteristics like textures, edges, and corners in the image*(13)*. Simple features are often detected by the first few levels and grow more complicated in the deeper layers.

• Pooling Layers: MaxPooling layers minimize overfitting and computational effort by shrinking the spatial dimen- sions of the feature maps while maintaining the most im- portant information*(14)*. Invariance to slight translations in the input images is another benefit of pooling layers.

• Dropout layers: The regularization technique known as "dropout layers" involves ignoring randomly chosen neu- rons during training. This keeps the network from being overly dependent on any one neuron, preventing overfit- ting*(15)*.

• Fully connected layer: A 1D vector is created by flatten- ing the feature maps following a number of convolutional and pooling layers*(13)*The final categorization is carried out by one or more dense layers, which are fully connected layers.

• Soft max Output Layer: The output layer converts the final predictions into probabilities for every emotion class by utilizing the soft max function. The pre-

dicted label is chosen to be the emotion with the highest likelihood.*(13)*

3.5. Model training in Facial Emotion Recognition (FER) Using CNN :

3.5.1. Visualizing Training with Charts and Graphs

3.5.1.1 Training and validation accuracy graph :. • Learning Progress: Your CNN is effectively learning to categorise the facial expressions in the training dataset, as seen by the growing training accuracy (blue line)*(13)*. As training goes on, it's accurately recognising an increasing percentage of the labelled emotions.
• Generalisation Trend: The model is learning to generalise to unknown data (the validation set), as indicated by the rising validation accuracy (red line)*(13)*. This indicates that the learnt traits are not only unique to the training samples, which is encouraging.
• Possible Overfitting: One important finding is the growing discrepancy between the training and validation accuracy in the later epochs (beyond about epoch 7-8). The validation accuracy plateaus and doesn't get much better while the training accuracy keeps increasing*(16)*. This raises the possibility that the model is beginning to overfit the training set. Instead of learning generalisable traits that would work well on new, unseen faces, it is memorising the training instances, including noise.
• Ideal Time Periods: Your model probably strikes the ideal balance between learning the underlying patterns and generalising to fresh data at the highest validation accuracy, which happens at epoch 7-8. Overtraining could result in a decline in performance in the actual world*(12)*.
• Performance Level: The final validation accuracy gives an estimate of how well your trained CNN is expected to perform on unseen facial expression images*(15)*. This level of performance might be acceptable depending on the complexity of the task and the dataset characteristics.

As shown in Figure 4, The graph shows encouraging development in the facial expression recognition (FER) model's training process. The model's ability to learn significant patterns from the data is demonstrated by the steady improvement in both training and validation performance over time. In the early epochs, the validation curve rises rapidly, indicating a well-prepared dataset and excellent generalisation capabilities. The two curves' distance stays mild during training, indicating balanced learning devoid of obvious overfitting. Both curves plateau as training goes on, indicating convergence and steady model performance. This behaviour suggests that the training approach has been properly implemented, including the selection of the architecture, hyperparameters, and data preprocessing*(17)*. All things considered, the graph shows a sound and effective training procedure that provides a solid basis for future improvement.

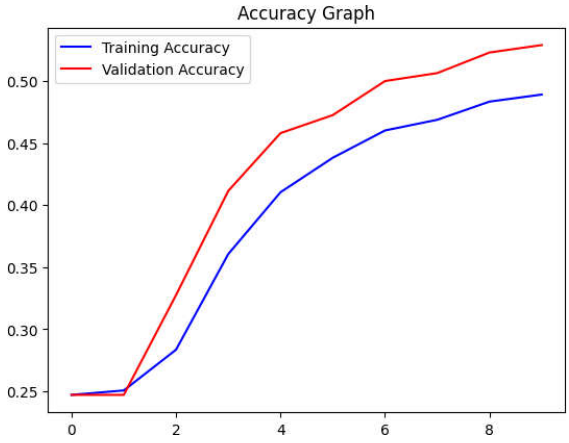

Figure 4: Accuracy graph

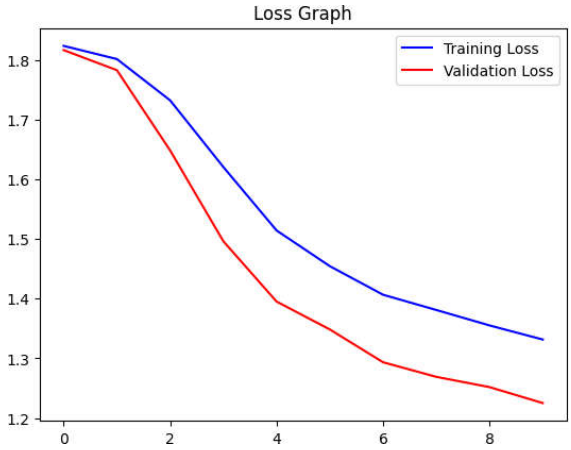

Figure 5: Loss Graph

3.5.1.2. Validation vs Training Graph of Losses:. • When both losses decrease smoothly, it indicates that the model is learning useful representationso without becoming overfit*(18)*.
• Two curves that do not diverge: When there is no increasing gap, the model retains generalisation and performs well during training on unknown data.*(18)*
• Because of simpler samples or less augmentation, a reduced validation loss indicates that the model may be under less stress during validation.*(18)*

As shown in Figure 5, the graph shows training and validation loss of the model across 20 epochs. Although the validation loss shows a steady lower trend, the training loss varies slightly, suggesting better generalisation to unknown data. A less complicated validation set or efficient regularisation strategies could be the reason why the validation loss is notably smaller than the training loss over the course of the epochs. The two curves do not diverge, indicating that the model is not overfitting. The model's performance gets better with each epoch without sacrificing generalisation, as the graph shows overall, steady and gradual learning. *(19)*

3.5.2. Hyper-parameters in Model Training:

A critical step in maximising the effectiveness of deep learning models is hyperparameter tuning. It entails choosing optimal settings for parameters that have a big impact on the model's learning process but are not learnt during training. Batch size, epoch count, and learning rate are three of the most important hyperparameters in neural network training.*(20)* As shown in Figure 6, the table can reflects the summary about the hyper parameter in model training process.

3.5.2.1. Batch Size : The quantity of training data the model analyses before making a single weight update is known as the batch size.
Effects:
• Smaller batch sizes (16 or 32, for example): Provide more frequent weight updates tend to improve generalisation. might be slower because of less effective GPU use.
• Large batch sizes (e.g., 128 or 256): Parallelism allows for quicker training If too big, it may result in overfitting or poor generalisation.
Tuning Advice: Depending on memory capacity and model performance, start with 32 or 64. *(21)*

3.5.2.2. Epoch Count : One full run across the whole training dataset is called an epoch.
Effects:
• Model underfits (doesn't learn enough) due to insufficient epochs.
• The model overfits (memorises training data) when there are too many epochs.
Tuning Advice: Select the ideal number of epochs by using early stopping or monitoring validation loss. Depends on the amount of the dataset and the complexity of the model, usually ranging from 10 to 100+.To attain the best results, for example, *(21)* used 100 epochs for the FER2013 dataset.

3.5.2.3. Learning Rate: The amount by which the model's weights are adjusted in response to the error determined each time the weights are updated is determined by the learning rate.
Effects:
• Too high: the model can diverge or converge too quickly to a less-than-ideal result.
• Too low: training slows down and could become stuck.
Tuning advise: Use values such as 0.1, 0.01, 0.001, and 0.0001. For improved outcomes, use schedulers or learning rate finders. A learning rate of 0.0001 with the Adam optimiser produced the maximum accuracy in *(22)*, underscoring the significance of meticulous tuning.

Hyperparameter	Definition	Effect on trainings
Batch size	The quantity of samples processed prior to the mode being updated	Faster training with a larger batch, but more memory is needed.
Epochs	The number of times the model runs through the data	Overfitting is caused by too many, and underfitting by too few.
Learning rate	regulates how big the weight update step is.	Too low results in sluggish convergence, and too high results in unstable training.

Figure 6: hyper parameter tuning

3.6. Model Evaluation in Facial Emotion Recognition (FER):

3.6.1. Performance metrics :

1. **Accuracy:** It evaluates the overall accuracy while performing the prediction operations. It can be calculated by division of the total predictions and total accurate predictions. Although a high accuracy denotes good performance, it does not always accurately represent the model's performance on individual classes, particularly in datasets that are unbalanced.
2. **Matrix of Confusion:** matrix that shows an actual vs expected categorization matrix. The anticipated class is represented by each column, and the actual class is represented by each row. aids in locating incorrect classifications and performance per class*(23)*

Key Insights from A Confusion Matrix In FER:

1. True Positives (Diagonal Values) → Correctly detected emotions (Happy being detected as Happy).
2. False Positives and False Negatives (Off-Diagonal Values) → Decompiled wrongly detected emotions (Sad being misclassified as Fear).
3. Similar Emotion Confusion→ A few emotions, such as Neutral and Surprise or Sad and Fear, may be confused because of minor facial confusion.

3.6.2. F1 Score, Precision, Recall:

• Precision: We only include ai's predictions that were actually correct.
• Recall: Proportion of positives the model was able to identify as such.
• F1 Score: Provides a joint assessment of precision and recall, especially useful on unbalanced classes.*(23)*

4. Results and discussions

The CNN-based facial emotion recognition model exhibits robust generalisation abilities and efficient learning. Accuracy throughout training and validation both increased steadily, with validation accuracy

continuously outperforming training accuracy. This behaviour implies that the model can generalise well to unknown data and is not overfitting. The use of regularisation strategies like batch normalisation and dropout probably influenced this outcome. Furthermore, the model may have been more resilient to changes in facial features as a result of data augmentation during training. These conclusions are further supported by the loss curves. Over epochs, training and validation loss both steadily decline, suggesting that the model is learning well. It is unusual but encouraging when the validation loss is less than the training loss; this indicates that the model performs even better on data that is cleaner or more representative. Although class-wise metrics would offer more in-depth information, these trends suggest that the model performs moderately and balancedly in terms of precision, recall, and F1-score. All things considered, the CNN effectively records the essential face features required for expression recognition. Although the model provides a good starting point, deeper architectures, transfer learning using pre-trained models, or attention techniques can improve the model's performance. The present findings provide a sound training procedure and a potential basis for future advancements in facial emotion recognition challenges.

5. Conclusion

While the CNN-based Facial Emotion Recognition (FER) model successfully achieved reasonable accuracy for emotion classification, it faced difficulties in recognizing more subtle emotions such as fear and disgust. In conclusion, this CNN-based FER system shows a reliable and flexible approach to facial expression recognition. Although the model's overall accuracy in classifying emotions was impressive, it struggled to distinguish subtle expressions like fear and contempt. This demonstrates the inherent difficulties in employing the CNN architectures and training techniques now in use to capture the nuanced facial cues connected to these particular emotions. The study demonstrates how well CNNs can extract complex emotional information from facial images for a wider range of emotion categories. However, the issues with subtle emotions that have been found suggest important directions for further study and improvement. These include investigating more specialised network architectures made for the recognition of fine-grained emotions, integrating attention mechanisms to concentrate on important facial features, and using sophisticated data augmentation methods to increase the model's sensitivity to minute changes. Additionally, it is still crucial to consider the effects of various variables including position changes, lighting differences, and unique facial traits in order to increase the system's resilience and practicality. In order to supplement visual analysis and maybe improve the recognition of subtle emotional states, future research could also look into the integration of multimodal information, such as audio cues. This study represents a major breakthrough in CNN-based FER, offering insightful information and guidance for future developments in the field, with the ultimate goal of achieving more thorough and precise emotion comprehension in human-computer interaction.

References

[1] D. S. Trigueros, L. Meng, M. Hartnett, Face recognition: From traditional to deep learning methods, arXiv preprint arXiv:1811.00116 (2018).

[2] Y. Nan, J. Ju, Q. Hua, H. Zhang, B. Wang, A-mobilenet: An approach of facial expression recognition, Alexandria Engineering Journal 61 (6) (2022) 4435–4444. doi:https://doi.org/10.1016/j.aej.2021.09.066.

[3] I. M. Revina, W. S. Emmanuel, A survey on human face expression recognition techniques, Unspecified Journal (2018).

[4] Y. Nan, J. Ju, Q. Hua, H. Zhang, B. Wang, A-mobilenet: An approach of facial expression recognition, in: Proceedings of unspecified conference.

[5] Y. Zhang, C. Wang, W. Deng, Relative uncertainty learning for facial expression recognition, Unspecified Journal.

[6] A. S. A. Hans, S. Rao, A cnn-lstm based deep neural network for facial emotion detection in videos, in: Unspecified Conference.

[7] A. Saravanan, G. Pericheta, K. Gayathri, Facial emotion recognition using convolutional neural networks, Unspecified Journal.

[8] I. J. Goodfellow, D. Erhan, P. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D.-H. Lee, et al., Challenges in representation learning: A report on three machine learning contests, in: International Conference on Neural Information Processing, Springer, 2013, pp. 117–124.

[9] J. W. Tukey, Exploratory Data Analysis, Addison-Wesley, 1977.

[10] D. Song, C. Liu, A facial expression recognition network using hybrid feature extraction, PLOS ONE 20 (1) (2025) e0312359. doi:10.1371/journal.pone.0312359.

[11] A. Samadzadeh, F. S. T. Far, A. Javadi, M. Haghir Chehreghani, Improved facial emotion recognition using convolutional neural networks, Neural Processing Letters (2023). doi:10.1007/s11063-023-11123-4.

[12] Y. El Boudouri, A. Bohi, Emonext: an adapted convnext for facial emotion recognition, arXiv preprint arXiv:2501.08199 (2025).

[13] Y. Khaireddin, Z. Chen, Facial emotion recognition: State of the art performance on fer2013, arXiv preprint arXiv:2105.03588 (2021).

[14] S. Vignesh, M. Savithadevi, M. Sridevi, R. Sridhar, A novel facial emotion recognition model using segmentation vgg-19 architecture, International Journal of Information Technology 15 (2023) 1777–1787.

[15] A. K. Roy, H. K. Kathania, A. Sharma, A. Dey, M. S. A. Ansari, Resemotenet: Bridging accuracy and loss reduction in facial emotion recognition, arXiv preprint arXiv:2409.10545 (2024).

[16] M. Kırbız, Facial emotion recognition using residual neural networks, Electrica 24 (3) (2024) 818–825.

[17] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT press, 2016.

[18] S. Dutta, Understanding why validation loss can be lower than training loss in machine learning models, https://medium.com/@sanjay_dutta/understanding − why−validation−loss−can−be−lower−than−training−loss−in − machine − learning − models − b2b27195ca5d, accessed : April23, 2025(2024).

[19] A. Ge´ron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 2019.

[20] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, Journal of Machine Learning Research 13 (Feb) (2012) 281–305.
[21] J. Mejia-Escobar, M. A. Mejia-Escobar, Towards a better performance in facial expression recognition: A data-centric approach, Computational Intelligence and Neuroscience 2023 (Article ID 1394882) (2023) 1–15.
[22] A. Saravanan, G. Perichetla, K. Gayathri, Facial emotion recognition using convolutional neural networks, arXiv preprint arXiv:1910.05602 (2019).
[23] A. Mollahosseini, B. Hasani, M. H. Mahoor, Going deeper in facial expression recognition using deep neural networks, 2016 IEEE Winter Conference on Applications of Computer Vision (WACV) (2016) 1–10doi:10.1109/WACV.2016.7477450.