Title: Facial Expression Recognition from Live Video Stream using CNN

Names of group members (in alphabetical order as it appears in iLearn)**:**
1. Md Kaykobad Reza (mreza025)
2. Shahriar M Sakib (ssaki004)

Group number (in iLearn)**:** 08

Abstract: The project aims to recognize the emotional state of a person through facial expressions captured from a live video stream. The project has diverse applications such as enhancing human-computer interactions, aiding communication disorders, improving security and surveillance systems, analyzing depression, and detecting customer satisfaction. The project addresses the challenges of recognizing emotions from facial expressions in real-time scenarios by developing a deep learning model based on Convolutional Neural Networks (CNN). The model is trained and evaluated using the Facial Expression Recognition 2013 dataset which consists of 48x48 pixel images depicting seven emotion categories (Angry, Disgust, Fear, Happy, Sad, Surprise, Neutral). The project explores different architectures including VGG16 and a customized 3-layer CNN model. Augmentation techniques are also considered to enhance the model's performance. The models are trained on a training set of 28,709 examples and evaluated on separate test sets consisting of 3,589 examples each. The goal is to identify the best performing model. Once the optimal model is determined, it has been used to detect emotions from live video streams.

Introduction: Facial expressions serve as a powerful form of nonverbal communication, allowing individuals to convey their emotions without the need for verbal interaction. These expressions, both involuntary and voluntary, rely on the movement of facial muscles, which are influenced by our state of mind. Just as humans instinctively recognize and interpret emotions through facial cues, machines can be trained to replicate this ability using Convolutional Neural Networks (CNNs), a deep learning method primarily utilized for image classification. By analyzing and detecting key features in facial expressions, CNNs can accurately classify emotions [1].

The recognition of emotions has a significant impact on our lives, shaping how we perceive and understand the world around us. While facial expressions are a common means of conveying emotions, emotions can also be interpreted through other mediums such as voice and text. Physiological signals, like electroencephalographic (EEG) signals, provide additional insights into emotional states, although their extraction and processing require extensive preprocessing [2]. Moreover, text-based emotion recognition traditionally relies on keyword analysis, but an alternative technique called knowledge-based Artificial Neural Networks (ANNs) utilizes the meaning of words in an ontology to interpret emotions [3].

Automatic facial expression recognition systems find wide-ranging applications in various fields. These systems can detect depression, assess health-related aspects, aid visually impaired individuals in understanding surrounding emotions, evaluate customer satisfaction, and provide valuable insights in fields like psychology, patient monitoring, advertising, and movie reviews [4,5,6]. By designing a CNN model trained on facial expression recognition datasets, it becomes

possible to classify images and detect real-time expressions from live video feeds. Such a system has the potential to revolutionize surveys, patient monitoring, customer feedback, and audience reaction analysis, among others.

Overall, the utilization of CNNs for facial expression recognition enables machines to replicate the natural way in which humans identify and classify emotions. This technology holds immense promise in various domains and offers opportunities for advancements in artificial intelligence and emotional analysis.

Dataset Description: The dataset used for this project consists of grayscale images of faces, each measuring 48x48 pixels. These images have undergone automatic registration to ensure that the faces are approximately centered and occupy a similar amount of space in each image. There are seven emotion categories used for classification: Angry (0), Disgust (1), Fear (2), Happy (3), Sad (4), Surprise (5), and Neutral (6). The quantity of images allocated for each emotion can be found in Table 1. The training set comprises a total of 28,709 examples, while the public test set contains 3,589 examples. These examples serve as the basis for training and evaluating the facial expression classification models in this project**.**

Emotion No	Emotion Type	Total Images
0	Angry	4953
1	Disgust	547
2	Fear	5121
3	Happy	8989
4	Sad	6077
5	Surprise	4022
6	Neutral	6198

Table 1: The quantity of images allocated for each emotion.

Related work: The recent advances in machine learning have led to significant improvements in facial expression classification (FEC). These methods have allowed for the development of classifiers to differentiate between various types of facial expressions captured from diverse individuals. Khaireddin et. al. [7] Discusses the challenges associated with achieving accurate and reliable facial emotion recognition (FER) through computer models. These challenges are attributed to the heterogeneity of human faces and variations in images. The authors proposed a deep learning model based on the VGGNet architecture that achieves a state-of-the-art single-network accuracy on FER2013 [13] without the use of additional training data. Roberto et. al. [8], in their paper, proposed a new self-attention module, LHC, designed for computer vision that is based on channel-wise application and a local approach. LHC-Net achieves a new state-of-the-art on the FER2013 dataset with lower complexity and computational cost compared to the previous state-of-the-art. The "Deep-Emotion" paper by Shervin et. al. [9] proposes a deep learning-based approach for facial expression recognition using an attention convolutional neural network. This network can achieve significant performance improvements by focusing on relevant parts of the face. The paper by Vignesh et. al. [10] discusses the challenges in achieving accurate and robust Facial Emotion Recognition (FER) pipeline due to factors like pose variation, illumination, and occlusion. The authors propose a novel CNN architecture by interfacing U-Net segmentation

layers in-between VGG layers to overcome these problems and achieve state-of-the-art (SOTA) single network accuracy on the FER-2013 dataset. On the other hand, the "Ad-Corre" paper Fard et. al. [11] proposes an Adaptive Correlation (Ad-Core) Loss method for improving the discriminative power of the learned embedded features in automated facial expression recognition. Ad-Corre consists of 3 components: Feature Discriminator, Mean Discriminator, and Embedding Discriminator, and is used in combination with the Xception network to achieve promising recognition accuracy on AffectNet, RAF-DB, and FER-2013 datasets.

Problem formulation: The project involves recognizing emotions from facial expressions in real-time scenarios. The project aims to address the challenges of recognizing emotions from facial expressions in real-time scenarios by developing a deep learning model based on Convolutional Neural Networks (CNN). The model is trained and evaluated using the Facial Expression Recognition 2013 dataset which consists of 48x48 pixel images depicting seven emotion categories. The project explores different architectures including VGG16, and a customized 3-layer CNN model. Descriptions of these models have been discussed later in this project. Augmentation techniques are also considered to enhance the model's performance. The goal is to identify the best performing model. Once the optimal model is determined, it has been used to detect emotions from live video streams using OpenCV and a webcam. The workflow for the project is given in figure 1.

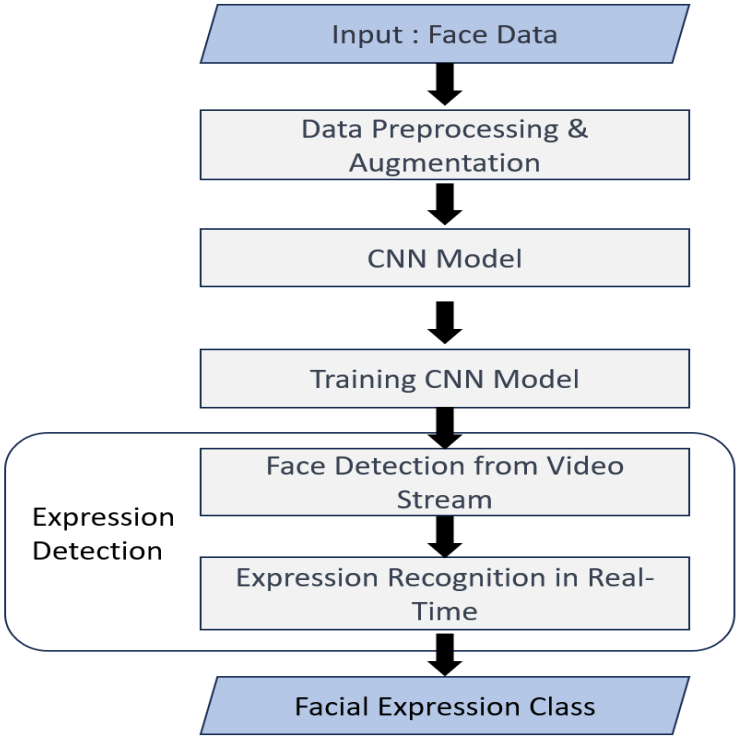

Figure 1: The flow chart for the project

Data Preprocessing & Augmentation: To ensure consistent and standardized input, the face data was normalized between the range of 0 to 1. This normalization process helps in balancing the pixel values across the dataset. Augmentation techniques were employed to

introduce variations in the images, allowing the model to generalize better and handle different facial expressions effectively. The following augmentation strategies were applied:

a. **Horizontal and Vertical Shifts:** Random horizontal and vertical shifts were applied to the images. This technique helps simulate slight changes in facial pose or alignment.

b. **Horizontal and Vertical Flips:** Images were horizontally and vertically flipped, creating mirror images. This augmentation helps in increasing the dataset's size and adding diversity in facial orientation.

c. **Rotation:** Random rotation within a certain degree range was applied to the images. This augmentation emulates variations in head position or tilts.

By applying data preprocessing techniques such as normalization and incorporating various augmentation strategies, the face data was enriched with a wider range of facial expressions and variations.

CNN Models: The project explores 2 different architectures named VGG16 and a customized 3-layer CNN model**.** Now, let's examine each model individually.

i.VGG16 [14] is a powerful Convolutional Neural Network (CNN) architecture commonly used for computer vision tasks, including image classification. With its 16 weight layers and small 3x3 convolution filters, it has become a benchmark model in the field. In my project, VGG16 was employed to recognize emotions from facial expressions in real-time scenarios. By training and evaluating the model using the Facial Expression Recognition 2013 dataset, the goal was to identify the best-performing model. The architecture of VGG16 includes convolutional layers with varying filter sizes and fully connected layers for classification. Its consistent arrangement of convolution and max pooling layers allows for effective feature extraction. Figure 2 summarizes the VGG-16 architecture [12].

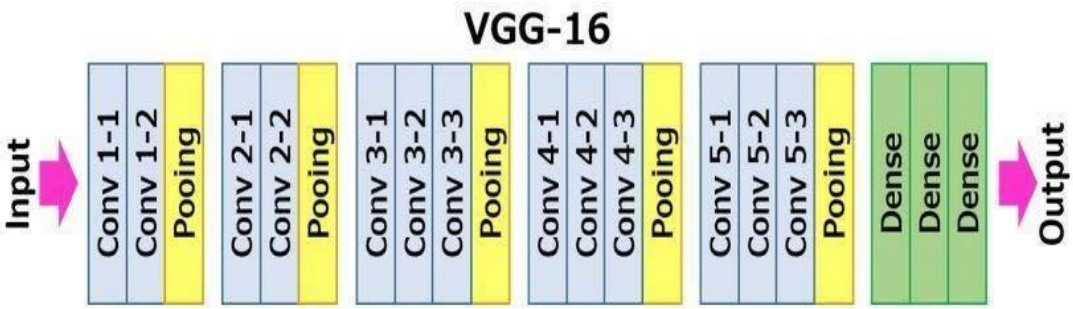

Figure 2 VGG-16 architecture

ii. Customized 3-Layer CNN Model: The customized 3-layer CNN model used in this project consists of multiple convolutional layers, pooling layers, and dense layers. The model architecture includes three Conv2D layers with kernel sizes of (3, 3) along with activation function ReLU and a total of four pooling layers using MaxPooling2D with pool sizes of (2, 2). The model's input shape is (48, 48, 1), indicating grayscale images of size 48x48 pixels. It also incorporates dropout layers with a dropout rate of 0.25 and 0.5 to mitigate overfitting. The final dense layer employs the SoftMax activation function to

classify the input into one of the seven output classes representing different emotions. Figure 3 summarizes the Customized 3-Layer CNN Model architecture.

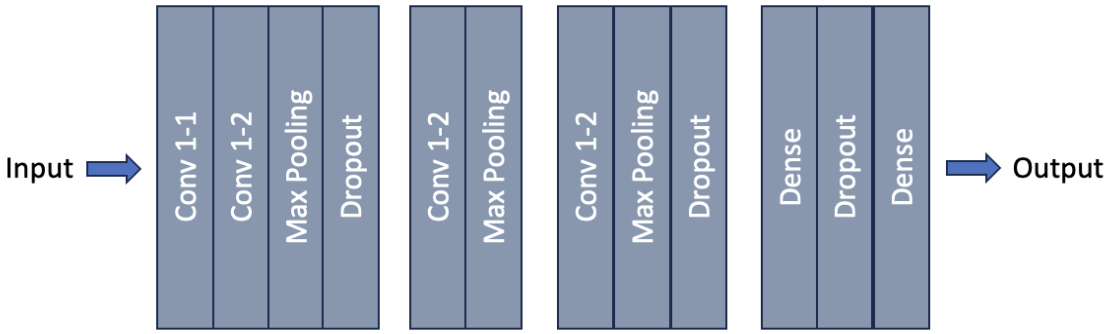

Figure 3: Customized 3-Layer CNN Model

Experimental Results: We conducted experiments to compare the performance of two models, VGG-16 and a Customized 3-Layer CNN Model. The experimental results are presented below, showcasing the training accuracy, training loss, and test accuracy for both models with and without data augmentation.

I. Training Accuracy:

We first examined the training accuracy for the VGG-16 model. The results indicated that the model's performance improved significantly when data augmentation was applied. The training accuracy plot revealed that the model achieved better performance with augmentation compared to without augmentation. Next, we analyzed the training accuracy for the Customized 3-Layer CNN Model. Surprisingly, the model displayed higher training accuracy without data augmentation compared to when augmentation was applied. Overall VGG16 performs better in terms of accuracy and faster convergence.

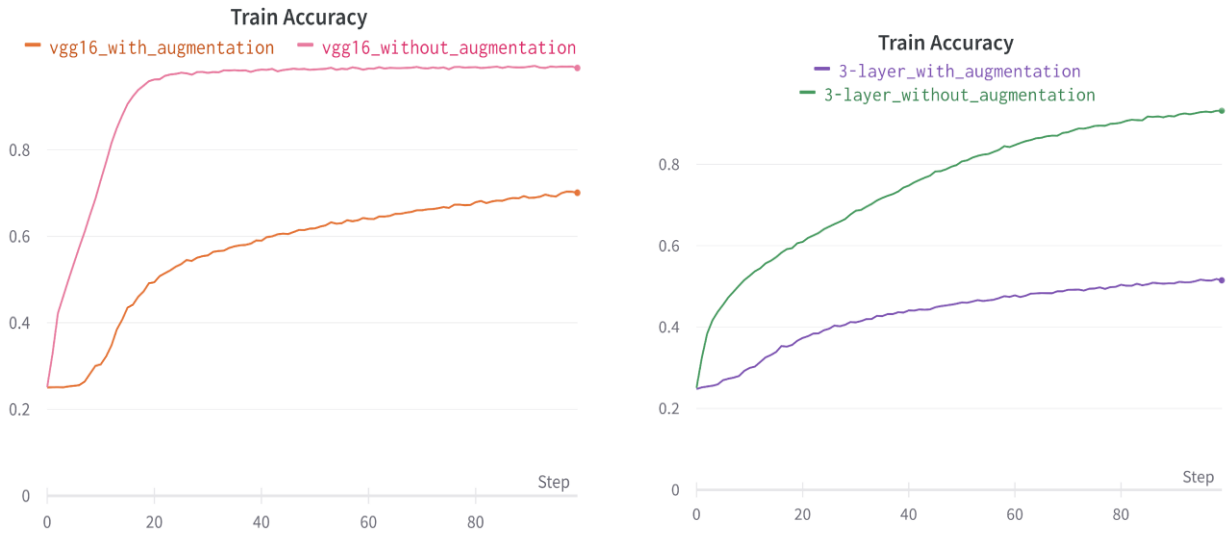

Figure 4: Training Accuracy for both VGG-16 and the Customized 3-Layer CNN Model

ii. Training Loss:

To further evaluate the models, we investigated the training loss for both VGG-16 and the Customized 3-Layer CNN Model. The training loss plot for VGG-16 & 3-Layer CNN Model indicated the models achieved lower training loss without data augmentation. Also, VGG16 converges faster and achieves lower training loss than the custom 3-Layer model.

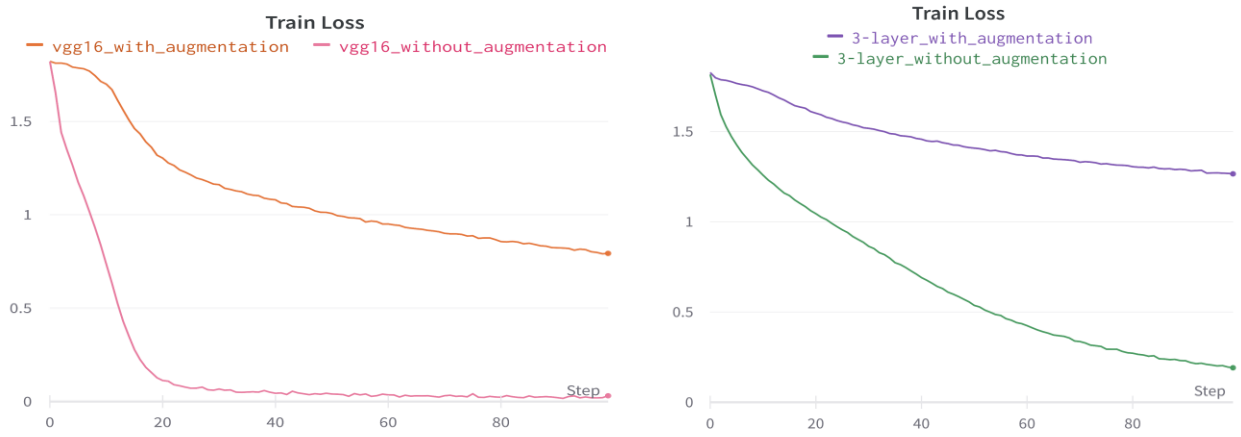

Figure 5: Training loss for both VGG-16 and the Customized 3-Layer CNN Model

iii. Test Accuracy:

Moving on to the test accuracy, we observed the performance of both models with and without data augmentation. The test accuracy results for the Customized 3-Layer CNN Model revealed that without augmentation, the model achieved an accuracy of 62.25%. However, when augmentation was applied, the accuracy dropped to 57.20%. This might be due to the simple structure of the model which is incapable of achieving greater generalization. In contrast, the test accuracy of VGG-16 without augmentation reached 60.97%. With the application of augmentation, the model's accuracy improved significantly to 65.74%. Overall, the experimental results demonstrate that data augmentation played a crucial role in improving the training accuracy of the VGG-16 model. However, for the Customized 3-Layer CNN Model, the absence of augmentation led to better training & test accuracy.

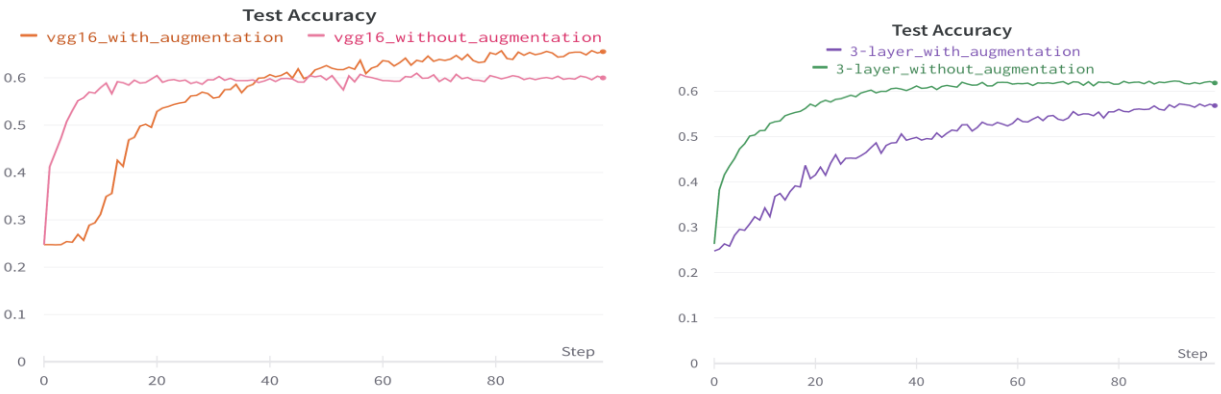

Figure 6: Test Accuracy for both VGG-16 and the Customized 3-Layer CNN Model

Model Name	Test Accuracy (Without Augmentation, %)	Test Accuracy (With Augmentation, %)
3-Layer CNN Model	62.25	57.20
VGG-16	60.97	**65.74**

Table 2: Shows the test accuracies of the two models.

Expression Recognition in Real-Time: To detect facial expressions in real time, we first trained our model on the FER2013 dataset. The best performing model was VGG16. Then we take the pretrained model for real time prediction. First, we captured a live video stream from a webcam. We used OpenCV and Python for capturing live camera streams. For each captured frame, we detect the face first. Then we resize the face portion, convert it to grayscale and resize it to 48x48 as the model was trained on 48x48 grayscale images. The model predicts a number from 0 to 6 corresponding to the 7 expressions. We map the number to proper expression and show a related emoji along with the detected expression name. Figure 7 shows 4 examples of real-time expression detection.

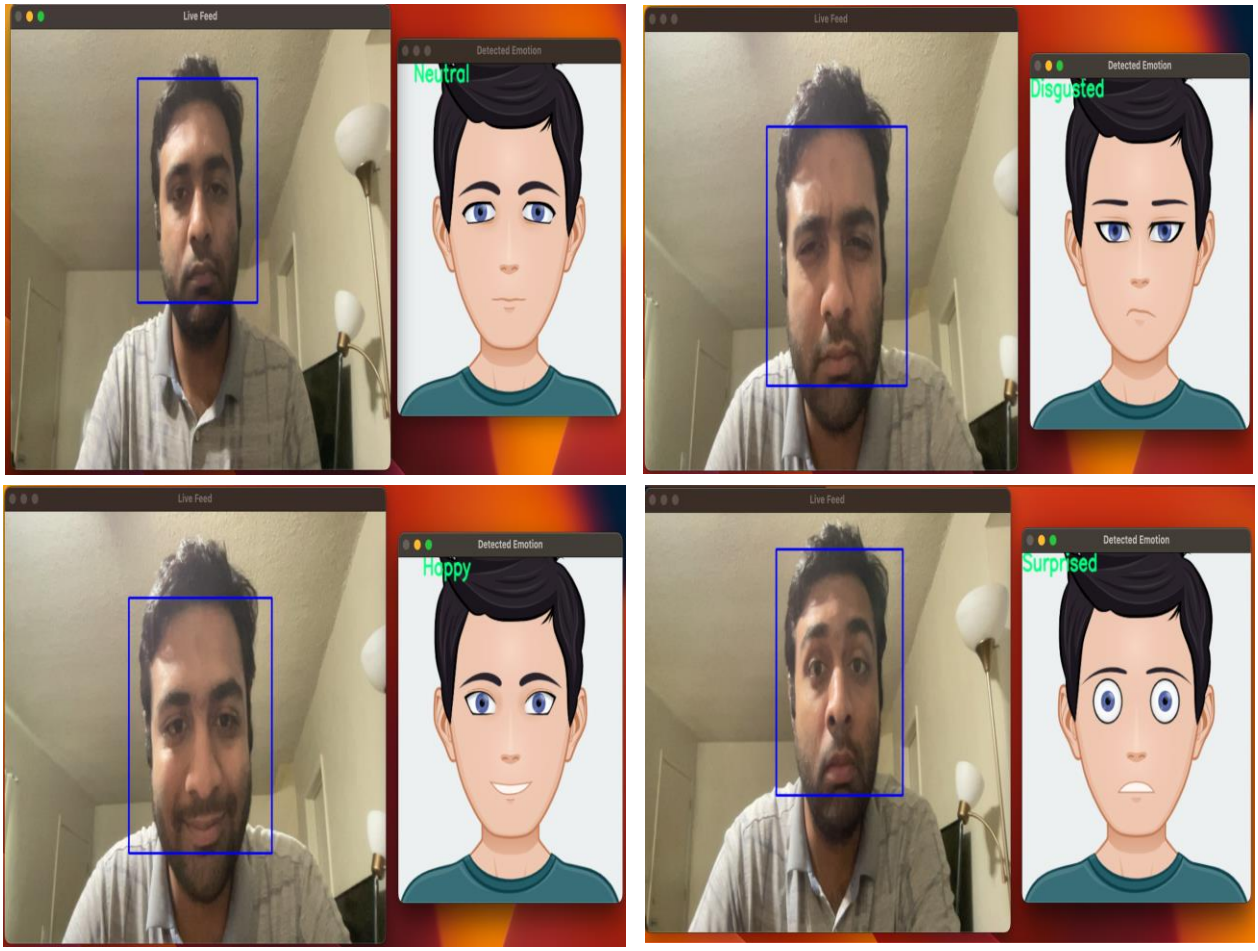

Figure 7: Expression Detection in Real-time Examples

Contributions: Briefly, our contributions are as follows:

- Train and compare the performance of VGG16 architecture with a basic 3-Layer CNN on facial expression recognition.
- Compare the effect of data augmentation on model performance.
- Build a realtime facial expression detection system using our pretrained model.

Limitations and Future Directions:

- Due to the limitation of computational resources, we could not train our models for a larger number of epochs. Given enough time, the model performance would have improved further.
- Our main goal was to develop a real-time system, so our focus wasn't on achieving state-of-the-art performance. However, we could have further improved the system's performance by training on multiple datasets and incorporating multiple models through ensemble learning.
- Training on larger datasets, more powerful models, different ensemble techniques while maintaining low resource utilization can be a good direction of future work.
- Future works can also involve evaluating these kinds of systems in real world scenarios.

Acknowledgements: We acknowledge the use of github, stackflow, and different blog websites for solving different issues that arose during the implementation process. All the images, xml files for face detection including the sample emojis are downloaded from the internet via google search. We took help from python, opencv and tensorflow documentations while preprocessing data and implementing different parts of the project. The dataset was downloaded from Kaggle: https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data

References:

1. Ramprasath, Muthukrishnan, M. Vijay Anand, and Shanmugasundaram Hariharan. "Image classification using convolutional neural networks." International Journal of Pure and Applied Mathematics 119.17 (2018): 1307-1319.
2. Raheel, Aasim, Muhammad Majid, and Syed Muhammad Anwar. "Facial expression Raheel, Aasim, Muhammad Majid, and Syed Muhammad Anwar. "DEAR-MULSEMEDIA: Dataset for emotion analysis and recognition in response to multiple sensorial media." Information Fusion 65 (2021): 37-49.
3. Seol, Yong-Soo, Dong-Joo Kim, and Han-Woo Kim. "Emotion recognition from text using knowledge-based ANN." ITC-CSCC: International Technical Conference on Circuits Systems, Computers and Communications. 2008.
4. Ko, Yen Huei, et al. "Customer retention prediction with CNN." Data Mining and Big Data: 4th International Conference, DMBD 2019, Chiang Mai, Thailand, July 26–30, 2019, Proceedings 4. Springer Singapore, 2019.
5. Seng, Kah Phooi, and Li-Minn Ang. "Video analytics for customer emotion and satisfaction at contact centers." IEEE Transactions on Human-Machine Systems 48.3 (2017): 266-278.
6. Saxena, Anvita, Ashish Khanna, and Deepak Gupta. "Emotion recognition and detection methods: A comprehensive survey." Journal of Artificial Intelligence and Systems 2.1 (2020): 53-79.

7. Yousif Khaireddin, Zhuofa Chen (2021). “Facial Emotion Recognition: State of the Art Performance on FER2013”, arXiv preprint - arXiv:2105.03588.
8. Roberto Pecoraro, Valerio Basile, Viviana Bono, Sara Gallo (2021). “Local Multi-Head Channel Self-Attention for Facial Expression Recognition”, arXiv preprint - arXiv:2111.07224
9. Shervin Minaee, Amirali Abdolrashidi (2019). "Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network", arXiv preprint - arXiv:1902.01019.
10. S. Vignesh, M. Savithadevi, M. Sridevi & Rajeswari Sridhar (2023). “A novel facial emotion recognition model using segmentation VGG-19 architecture”, Int. j. inf. tecnol. (2023). https://doi.org/10.1007/s41870-023-01184-z
11. A. P. Fard and M. H. Mahoor (2022). "Ad-Corre: Adaptive Correlation-Based Loss for Facial Expression Recognition in the Wild," in IEEE Access, vol. 10, pp. 26756-26768, 2022, doi: 10.1109/ACCESS.2022.3156598.
12. https://medium.com/@mygreatlearning/everything-you-need-to-know-about-vgg16-7315defb5918
13. https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data
14. Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).