Assignment 4

Souvik Roy

Task – 1

Architectures of the model

Two Convolutional Neural Network Architectures are built to predict the emotion from images. The architectures of the model are given below:

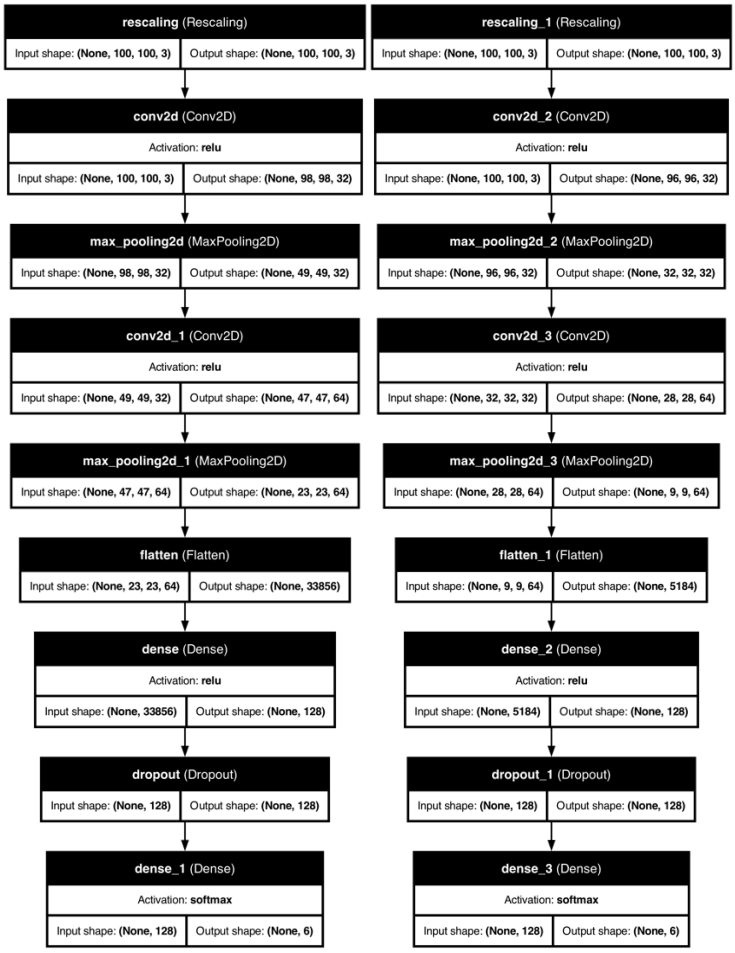

Figure: Model Architectures (Left: CNN-1, Right: CNN-2)

The first CNN model starts with a rescaling layer, followed by two convolutional layers (Conv2D) with ReLU activations, each paired with a max pooling layer. The output is flattened and passed through a dense layer with 128 units, followed by a dropout layer, and ends with a dense output layer using softmax activation for classification into six categories. Its intermediate shapes suggest a slightly deeper structure, resulting in a feature map size of 23x23 before flattening.

The second CNN model also begins with a rescaling layer and employs a similar structure of convolutional and max pooling layers. However, it has three Conv2D layers, each followed by max pooling, resulting in smaller feature maps (9x9) before flattening. The final dense layers are similar, with a dense layer of 128 units, dropout, and a softmax output layer.

The key difference is the number of convolutional layers and the progressive reduction in spatial dimensions. The first model emphasizes wider feature maps at the cost of fewer convolutional layers, while the second model extracts more compact features using additional convolutions and pooling steps, potentially allowing for better feature extraction for more complex data.

Training Accuracy over Epoch

Epoch	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
CNN-1	0.206	0.287	0.346	0.44	0.573	0.649	0.77	0.839	0.885	0.91	0.964	0.95	0.985	0.984	0.971	0.989	0.981	0.988	0.993	0.989
CNN-2	0.176	0.21	0.246	0.308	0.369	0.413	0.464	0.567	0.616	0.69	0.759	0.81	0.856	0.91	0.942	0.945	0.962	0.964	0.968	0.981

Test Accuracy of Different Models

Model	Test Accuracy
CNN – 1	0.365297
CNN – 2	0.324201
Pre-trained	0.497717

Discussion:

The comparison between CNN-1 and CNN-2 reveals distinct patterns in training and testing performance. CNN-1 achieves significantly higher training accuracy, quickly surpassing 90% by the 10th epoch and stabilizing near 99% by the 20th epoch. In contrast, CNN-2 progresses more gradually, achieving 98.1% training accuracy by the final epoch. Despite its superior training performance, CNN-1 slightly outperforms CNN-2 in test accuracy (36.5% versus 32.4%), but both models display a considerable gap between training and test accuracies, indicating potential overfitting. CNN-1's higher complexity, reflected in its larger feature map size and fewer pooling operations, might contribute to its better training results but potentially limit its generalization. Conversely, CNN-2’s additional convolution and pooling layers likely enhance its ability to extract features progressively, but this design seems less effective in adapting to unseen data in this case.

Task – 2

Description of the Pretrained Model

EfficientNetB0 has been modified to complete the task. The model is of 29 megabytes and tested on comparatively less amount of data with a very good accuracy. Since we have scarcity of training data, the model can be of a very good use. A Global Average Pooling layer and a fully connected layer of 1024 nodes with ReLu activation function has been added to the model. The pre-trained architecture is very complex. The architecture starts with a stem layer (a simple convolution), followed by a series of MBConv blocks arranged in stages, each with a specific number of layers, filter sizes, and strides. These blocks reduce computational cost while preserving accuracy. A key feature is the SE block in each MBConv, which applies attention mechanisms by reweighting channel-wise feature importance. The network ends with a global average pooling layer, a fully connected dense layer, and a softmax output for classification tasks.

Training Accuracy over Epoch

Epoch	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Training Accuracy	0.44	0.753	0.883	0.924	0.956	0.974	0.977	0.982	0.981	0.986	0.985	0.991	0.99	0.989	0.993	0.992	0.99	0.994	0.993	0.992

Test Accuracy of Different Models

Model	Test Accuracy
CNN – 1	0.365297
Pre-trained	0.497717

Discussion

The comparison between the pre-trained model and CNN-1 highlights significant differences in training and test performance. The pre-trained model shows consistently higher training accuracy across all epochs, reaching above 99% by epoch 15, indicating strong convergence and effective learning. In contrast, CNN-1 exhibits a slower start, with lower training accuracy in the initial epochs but catches up to a comparable level by epoch 20. However, the test accuracy reveals that the pre-trained model outperforms CNN-1, achieving 49.77% compared to CNN-1's 36.53%. This suggests that the pre-trained model generalizes better to unseen data, likely due to its prior knowledge, whereas CNN-1, despite achieving high training accuracy, may be prone to overfitting or insufficient capacity for this specific task.

Task – 3

Prediction using Manually Taken Images

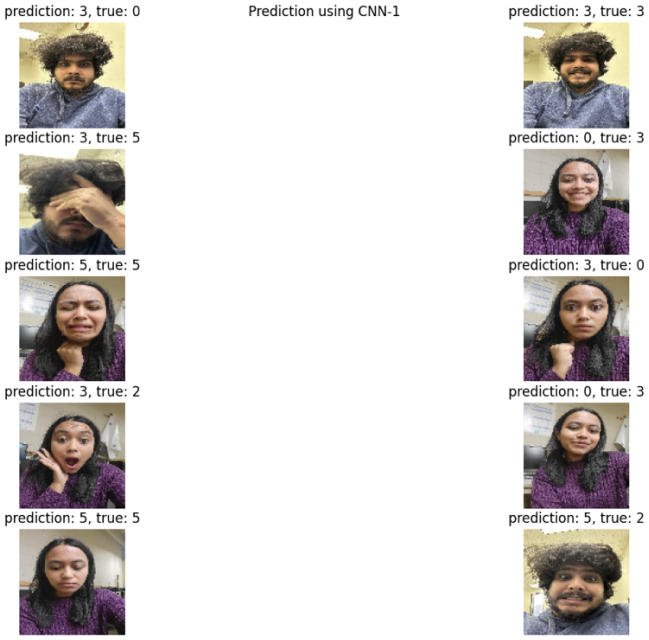

Figure: Prediction on 10 manually taken images using CNN-1

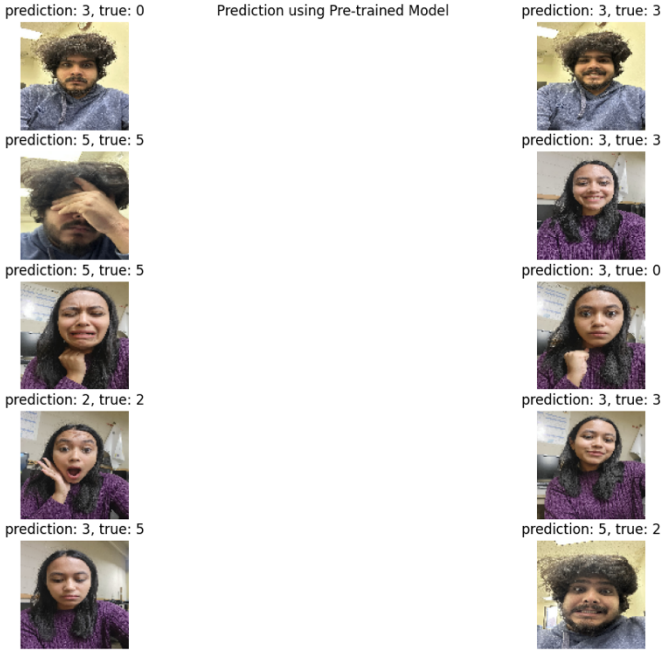

Figure: Prediction on 10 manually taken images using pre-trained model

Discussion

The evaluation of manually captured images demonstrates that the fine-tuned model outperforms CNN-1 in terms of accuracy. Specifically, CNN-1 achieves an accuracy of 30% on 10 manually taken images, whereas the fine-tuned model achieves 60% accuracy.

A primary factor contributing to errors in both models is the limited availability of training data. Increasing the volume of training data could significantly enhance the models' predictive performance. Additionally, the training dataset is imbalanced, with a dominance of two classes, ‘Happy’ (label 3) and ‘Sad’ (label 5). This class imbalance likely results in misclassifications favoring these majority classes, particularly class 'Happy,' which appears most frequently in incorrect predictions.

While the fine-tuned model demonstrates strong overall performance, it struggles to accurately predict the 'Fear' class (label 2). This challenge may stem from insufficient and less diverse data to distinguish between 'Sad' and 'Fear' effectively. Furthermore, when the fine-tuned model encounters ambiguity, it tends to predict 'Happy,' likely due to the class's dominance in the dataset. Addressing these issues through balanced and diverse training data could further improve the model's performance.

Acknowledgement

ChatGPT has been used for sentence refurbishment and grammatical correction.