Classification of Facial Images Based on Emotions Using CNNs.

1. Problem Statement:

The project endeavors to construct a Convolutional Neural Network (CNN)-based model aimed at recognizing emotions from facial expressions. Utilizing the CNN architecture, the objective is to discern various emotional states. Emotion recognition from facial images is a challenging yet pivotal task in various fields such as human-computer interaction, virtual reality, and affective computing. The project seeks to utilize transfer learning principles by incorporating pre-trained models like VGG16 and Resnet50 to extract generalized features pertinent to facial expression recognition. The model will be trained and evaluated using standard benchmark datasets for facial expression recognition, such as FER2013.

2. AI Principles:

2.1. Understanding the principles of AI:

Understanding theoretical foundations of artificial intelligence, including machine learning algorithms, deep learning architectures, and computer vision techniques. By applying these principles to the problem of facial emotion recognition, the aim is to explore how AI can be leveraged to interpret and respond to human emotions from facial expressions.

2.2. Ethical Considerations:

An important aspect of this project would be to explore the ethical considerations and societal implications associated with the development and deployment of AI systems, especially those related to facial emotion recognition. This includes considerations of privacy, bias, fairness, and accountability in the design and implementation of the system.

2.3. Practical implementation and experimentation:

Through the implementation of facial emotion recognition algorithms, the objective is to gain practical experience in applying AI principles to real-world datasets and scenarios. This involves data preprocessing, model training, evaluation, and fine-tuning to achieve optimal performance.

3. Dataset Information:

The FER2013 dataset, sourced from Kaggle, comprises an extensive repository of grayscale facial images capturing individuals from diverse ethnicities, age groups, and cultural backgrounds. Each image is meticulously represented as a matrix of pixel values, offering intricate insights into the facial features and expressions of the subjects. Furthermore, the dataset has annotations denoting the corresponding emotion depicted in each image. These annotations serve as invaluable labels for conducting supervised learning tasks like classification and regression. To facilitate robust model evaluation and validation, the dataset is meticulously partitioned into distinct subsets earmarked for training and testing purposes. This systematic partitioning ensures the rigorous assessment of machine learning models and their performance in accurately recognizing and interpreting facial emotions.

4. Flow-Chart:

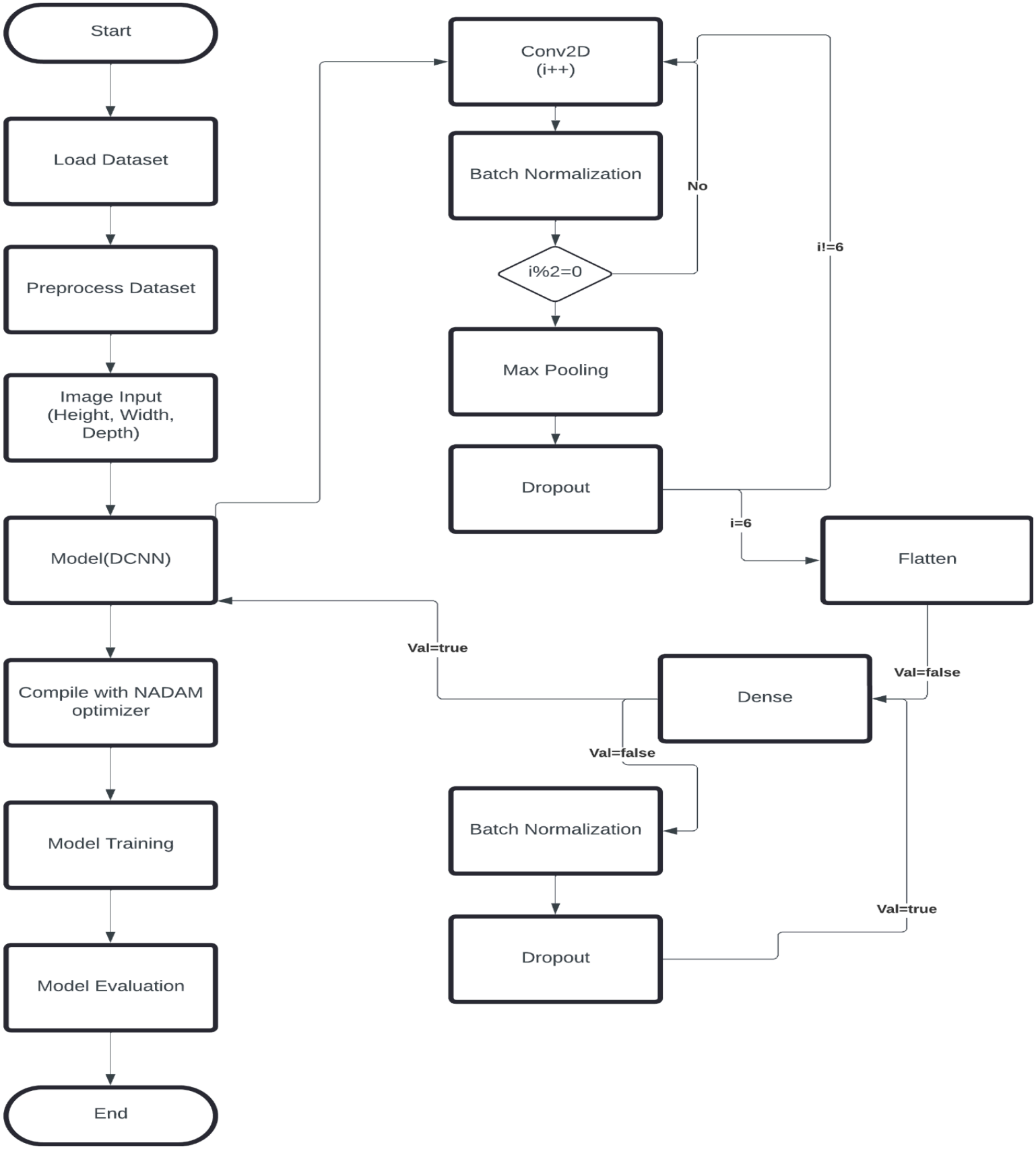


Figure 1: Flow-Chart

5. Architecture:

5.1. Model Architecture:

Crafted a deep convolutional neural network (CNN) model tailored for facial emotion recognition. The architecture encompasses a series of layers, including convolutional layers, batch normalization layers, max-pooling layers, dropout layers, and dense layers. The CNN model is built using the Keras Sequential API.

5.1.1 Pre-trained Models tested:

- Resnet50: ResNet50 is a deep convolutional neural network architecture consisting of 50 layers, introduced by Microsoft Research.

- VGG16: VGG16 is a deep convolutional neural network architecture consisting of 16 Convolutional layers followed by fully connected layers.

5.2. Callback Functions:

Two callback functions are employed during model training to enhance performance and prevent overfitting. The EarlyStopping callback is utilized to monitor the validation accuracy and halt training when the accuracy stops improving, thus preventing overfitting. The ReduceLROnPlateau callback dynamically adjusts the learning rate during training based on the validation accuracy, allowing the model to converge faster and achieve better performance.

5.3. Hyperparameter Tuning:

Fine-tuned crucial hyperparameters such as batch size, number of epochs, and optimization algorithms to achieve peak model performance. The optimal batch size of 32 is determined through experimentation, as it yields the best performance.

5.4. Training:

Executed model training using augmented training data, navigating through 100 epochs of iterative learning. Each epoch is imbued with batches of augmented images, fostering model comprehension and adaptability. To mitigate training time, multiprocessing is harnessed, Utilizing the system's parallel processing capabilities to expedite model convergence.

5.5. Evaluation and Results:

Once training is complete, the trained model is evaluated on the validation dataset to assess its performance in terms of accuracy and other metrics.

6. Methodology:

- Sequential API: The model architecture is built using the sequential API, facilitating the linear stacking of layers for streamlined construction.
- Convolutional Layers: A series of convolutional layers (Conv2D) are employed to extract relevant features from the input images. These layers utilize filter sizes ranging from 5x5 to 3x3, enabling the detection of spatial patterns.
- Batch Normalization: Following each convolutional layer, a batch normalization layer is added to standardize and accelerate the training process by stabilizing the input distributions.
- MaxPooling2D: MaxPooling2D layers are inserted after the convolutional layers to downsample feature maps, reducing computational complexity and focusing on the most salient features.
- Dropout Layers: Dropout layers are strategically placed after the max pooling layers to mitigate overfitting by randomly deactivating a fraction of neurons during training, thus enhancing model generalization.
- Output Layer: The final layer employs softmax activation to generate probability distributions across the different emotion categories, enabling the classification of input images.
- Loss Function and Optimizer: The model is compiled using categorical cross-entropy loss, suitable for multi-class classification tasks. The NADAM optimizer is chosen to minimize the loss function during training, offering efficient convergence and robust performance.
- Performance Metrics: During training, the model tracks accuracy metrics to evaluate its performance on the training and validation datasets, providing insights into its effectiveness in learning the underlying patterns.
- Weight Initialization: Convolutional and dense layers' weights are initialized using the He normal initializer, ensuring stable and efficient training by setting appropriate initial values for the network parameters.

7. Challenges and problems:

7.1. Coarse and ambiguous dataset:

The dataset poses challenges due to the inherent difficulty in discerning between emotions based solely on facial images. Each individual may express emotions differently, and many images within target classes such as anger, disgust, and fear exhibit similarities, making classification challenging.

7.2. Imbalanced class distribution:

Significant class imbalances exist within the dataset, with certain emotions being overrepresented while others are underrepresented.

7.3. Noisy or irrelevant features:

Facial images often contain irrelevant features or noise that may not contribute to emotion recognition but could confuse the model.

7.4. Computational complexity and resource constraints:

Deep CNN models, especially when trained on large datasets, require significant computational resources, including high-performance GPUs or TPUs. (for e.g. The model took over 40 minutes running on gpu(rtx2060) and over 1.5 hours running on CPU(Ryzen 7) and about 35 minutes on colab(T4-gpu))

7.5. Open Source Dataset:

Despite the importance of robust datasets for training deep learning models, I was unable to find a freely available open-source dataset larger than the one utilized in this project.

8. Results:

8.1. VGG Model:

The VGG16 model achieved an accuracy of 61.13% when trained on a dataset comprising images of all emotions. When trained on a subset of the dataset containing only the emotions happy, sad, and neutral, the accuracy improved to 73%.

8.2. Resnet50 Model:

Similarly, the ResNet50 model attained an accuracy of 53.7% on the complete dataset and demonstrated an accuracy of 68.4% on the subset containing happy, sad, and neutral emotions.

8.3. The DCNN Model:

Outperforming both VGG16 and ResNet50, the Deep Convolutional Neural Network (DCNN) model exhibited superior performance. When trained on the entire dataset, the DCNN model achieved an accuracy of 62%. However, its accuracy significantly improved to 83% when trained on a subset comprising only the emotions happy, sad, and neutral.

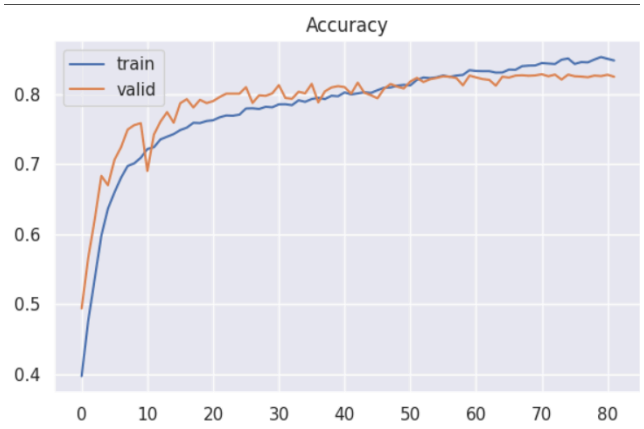
8.4. Performance Metrics of the DCNN Model:

The DCNN model, recognized for its superior performance, achieved good precision, recall, and F1 scores across different emotions, as illustrated in the table below:

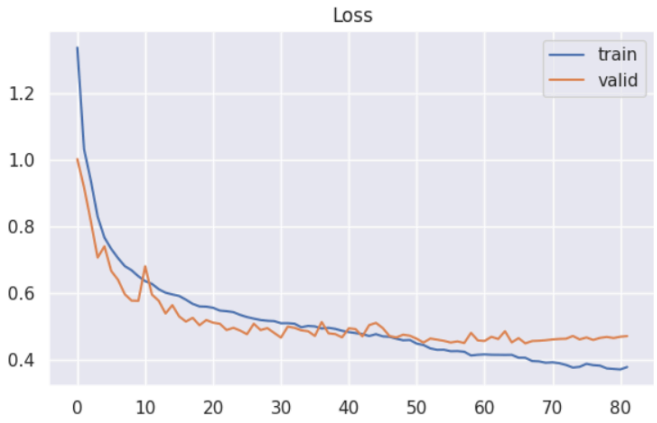
Emotion/Score	**Precision**	**Recall**	**F1 Score**
Happiness	0.92	0.92	0.92
Sadness	0.79	0.74	0.76
Neutral	0.73	0.78	0.75

Table 1: Performance Scores

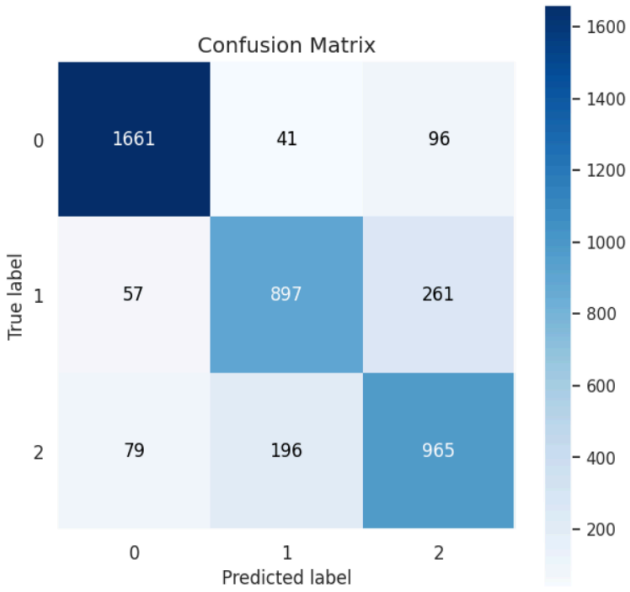
8.5. Accuracy vs. Epochs:



8.6. Loss vs. Epochs:



8.7. Confusion Matrix:



9. Conclusion:

This project aimed to develop an efficient facial emotion recognition system using deep learning techniques. Through the implementation and evaluation of various models, including VGG16, ResNet50, and a custom Deep Convolutional Neural Network (DCNN), we explored the effectiveness of different architectures in recognizing and classifying facial emotions. While the VGG16 and ResNet50 models demonstrated moderate accuracies, the DCNN model emerged as the most promising, achieving an accuracy of 62% on the complete dataset and 83% on a subset containing only the emotions happy, sad, and neutral. Furthermore, a detailed analysis of the DCNN model's performance metrics, including precision, recall, and F1 score, revealed its proficiency in distinguishing between different emotional states. These findings underscore the potential of deep learning models in capturing complex facial expressions and highlight avenues for further research and optimization.
However, despite the efforts, I was unable to achieve higher accuracies due to the unavailability of better datasets and the noise in the utilized dataset. The dataset's coarse and ambiguous nature posed challenges in accurately discerning between emotions, as individual expressions varied significantly. Moving forward, addressing these limitations through the acquisition of more extensive and cleaner datasets, as well as refining model architectures and training strategies, could lead to improved performance and greater accuracy in facial emotion recognition systems.

10. Future Directions:

10.1. Enhanced Modeling Techniques:

Integrating attention mechanisms or spatial-temporal modeling techniques can enhance the model's capacity to capture intricate spatial relationships and temporal dynamics in facial expressions. This approach holds promise for improving the recognition of subtle emotional cues, thereby enhancing overall performance.

10.2. Dataset Expansion and Diversity:

Expanding experimentation with larger and more diverse datasets, encompassing real-world scenarios and demographic variations, is crucial. Access to a broader range of facial expressions and demographic diversity can augment the model's robustness and generalization capabilities, leading to more efficient and reliable emotion recognition.

10.3. Multimodal Fusion Strategies:

Exploring multimodal fusion strategies that combine facial features with other modalities like audio or text presents an exciting avenue for research. Integrating complementary information from multiple sources can enrich the model's understanding of emotional cues and potentially boost accuracy and richness in emotion recognition tasks.

11. References:

11.1. Kaggle Dataset

11.2. Khorrami, P., Paine, T., Huang, T., & Kriesel, D. (2017). Do deep neural networks learn facial action units when doing expression recognition? Proceedings of the 2017 ACM on Multimedia Conference, 607-615.

11.3. Mollahosseini, A., Chan, D., & Mahoor, M. H. (2017). Going deeper in facial expression recognition using deep neural networks. Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 1-10.

11.4. Dhall, A., Goecke, R., Joshi, J., & Wagner, M. (2016). Emotion recognition in the wild challenge 2016. Proceedings of the 18th ACM International Conference on Multimodal Interaction, 503-510.

11.5. Liu, M., Li, S., & Shan, S. (2019). Deep learning for facial expression analysis: A survey. IEEE Transactions on Affective Computing, 10(4), 556-576.