ABBES Ayoub

I.A/python Project report

Introduction

This project aims to develop a convolutional neural network (CNN) model to classify images from the CIFAR-100 dataset. The CIFAR-100 dataset comprises 60,000 32x32 color images in 100 classes, with 600 images per class. The dataset is divided into 50,000 training images and 10,000 test images.

Data Preprocessing

The data was preprocessed to normalize the pixel values to the range [0, 1]. The training labels were one-hot encoded to match the output layer of the model, representing 100 distinct classes.

Model Architecture

The implemented CNN model consists of several layers designed to extract and learn features from the images:

Three blocks of Convolutional layers with Batch Normalization and ReLU activation, followed by MaxPooling and Dropout layers. The dropout rates were adjusted in each block to prevent overfitting. A Flatten layer to convert the 2D features into a 1D vector.

A Dense layer with 512 units, Batch Normalization, ReLU activation, followed by a Dropout layer.

The output Dense layer with 100 units and softmax activation to output the probability distribution across the 100 classes.

Training

The model was compiled with the Adam optimizer and categorical cross-entropy loss function, considering the multi-class nature of the problem. The learning rate was initially set to 0.0005 and adjusted according to a custom learning rate scheduler.

During training, the model used real-time data augmentation, including rotations, width and height shifts, zoom, and horizontal flip to improve generalization.

Callbacks :

LearningRateScheduler: To adjust the learning rate after 10 epochs.

EarlyStopping: To halt the training when the validation accuracy stopped improving, with a patience of 15 epochs.

ModelCheckpoint: To save the best model based on validation accuracy.

ReduceLROnPlateau: To reduce the learning rate when the validation accuracy plateaued, with a patience of 40 epochs.

Results :

The training process spanned over 300 epochs, although the EarlyStopping callback ensured the best model was retained before overfitting could occur. The best performance achieved on the test set was a loss of 1.0696 and accuracy of 72.74%. The evolution of accuracy and loss over epochs is visualized in the provided charts, showing a steady improvement in accuracy and a decrease in loss as training progressed.

Confusion Matrix Analysis:

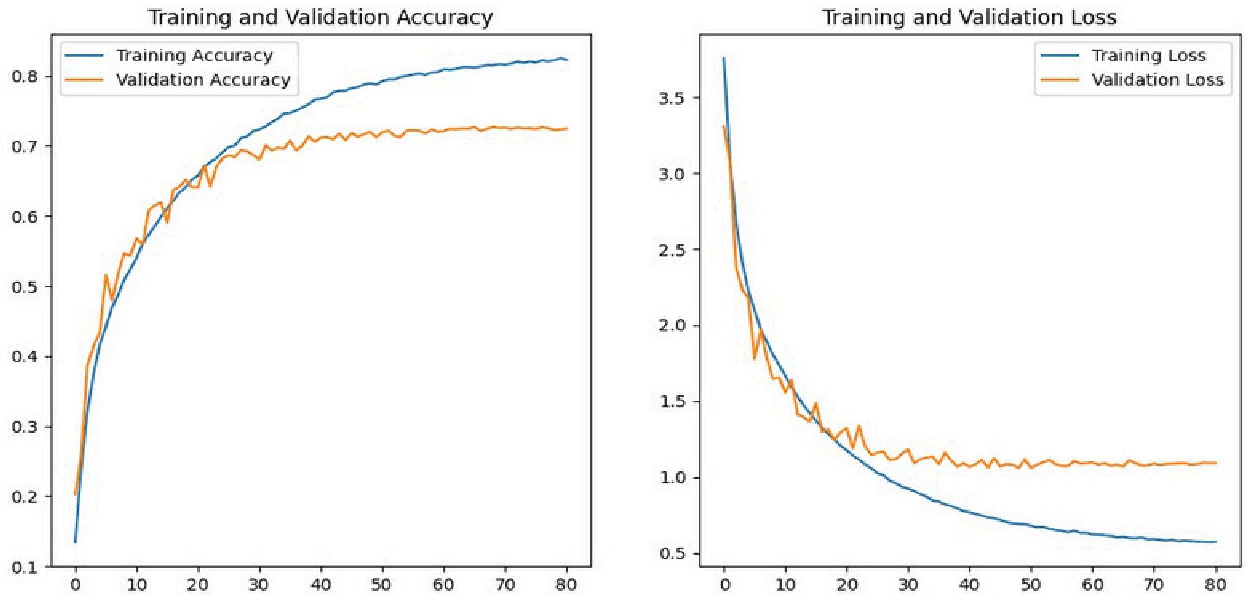
The confusion matrix shows the model's performance across all classes. The diagonal represents correctly classified instances, while off-diagonal elements represent misclassifications. The relatively even distribution along the diagonal suggests that the model has learned to classify most classes correctly. However, some confusion is observed, which is expected given the complexity of the dataset and the similarity between different classes.

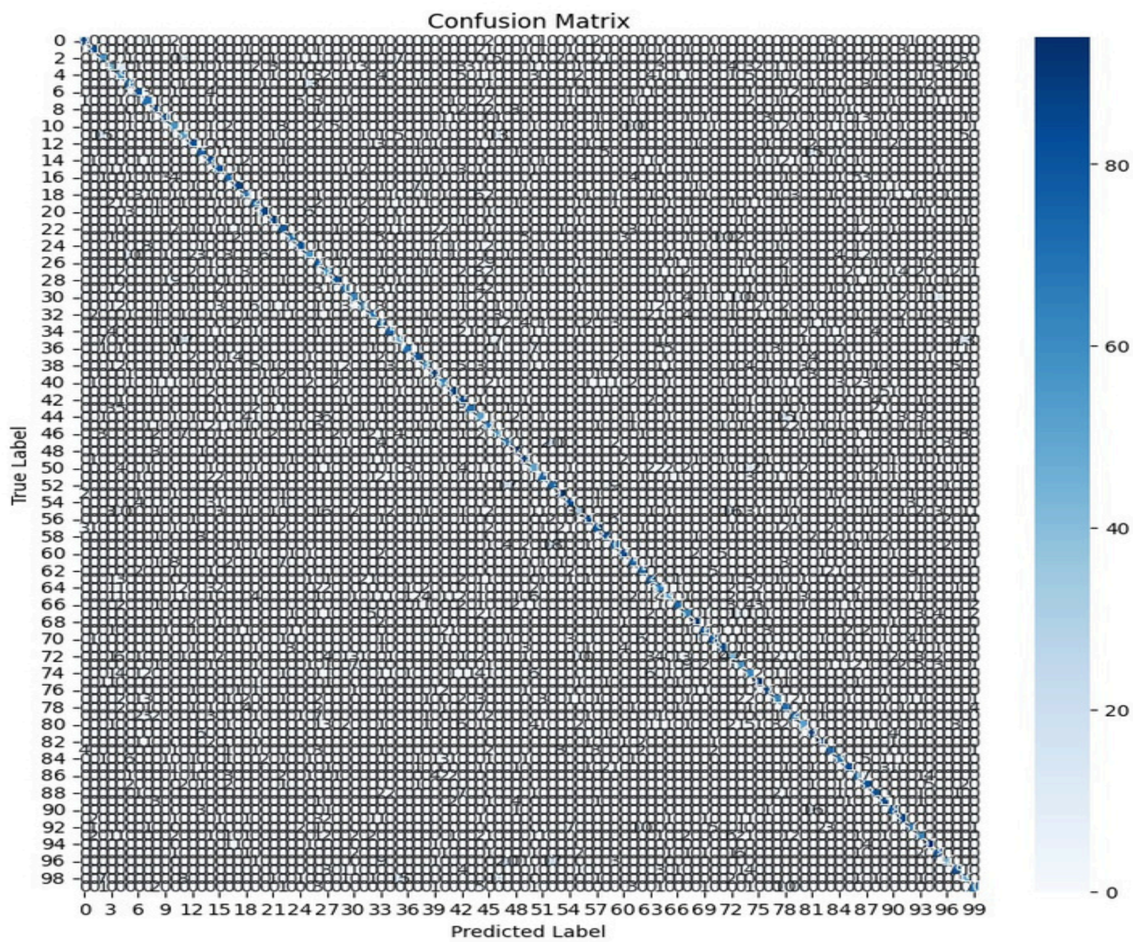
Discussion :

The model demonstrates good performance with a final test accuracy of over 72%, which is a strong result for the CIFAR-100 dataset. The training and validation accuracy curves reveal that the model continued to learn effectively throughout training, with no evident signs of overfitting thanks to the implemented regularization techniques and callbacks.

Conclusion

The project successfully developed a CNN that can classify images from the CIFAR-100 dataset with high accuracy. Future work could explore more sophisticated data augmentation techniques, deeper architectures, or alternative regularization methods to further enhance the model's performance.


Training and Validation Accuracy
Training Accuracy
Validation Accuracy
Training and Validation Loss
Training Loss
Validation Loss


Confusion Matrix
True Label
Predicted Label