Convolutional Neural Network (CNN) for CIFAR-10 Classification

Introduction:

This document explains the implementation of a Convolutional Neural Network (CNN) for the classification of images in the CIFAR-10 dataset. The CIFAR-10 dataset consists of 60,000 32x32 color images in 10 different classes.

Code Overview:

1. Importing Libraries:

The code begins by importing necessary libraries, including TensorFlow (via Keras), NumPy for numerical operations, and Matplotlib for visualization.

2. Loading CIFAR-10 Dataset:

The CIFAR-10 dataset is loaded using Keras' dataset module, splitting it into training and testing sets. The dimensions of the datasets and the number of samples are printed for verification.

3. Data Preprocessing:

The image data is normalized to ensure pixel values lie in the range [0, 1]. Furthermore, one-hot encoding is applied to the class labels.

4. Model Architecture:

The CNN model is constructed using the Sequential API of Keras. It consists of convolutional layers with batch normalization, activation functions (ReLU), and max-pooling. Fully connected layers are included for classification, with a softmax activation function in the output layer.

5. Model Compilation:

The model is compiled using the Adam optimizer, categorical crossentropy loss function, and accuracy as the evaluation metric.

6. Model Training:

The model is trained using the training data. The training history (accuracy and loss) is stored in the history variable.

7. Model Evaluation:

The trained model is evaluated on the test set. The test loss and accuracy are printed to assess the model's performance.

```
Test loss :  1.4846692085266113
Test Accuracy :  0.7894999980926514
```

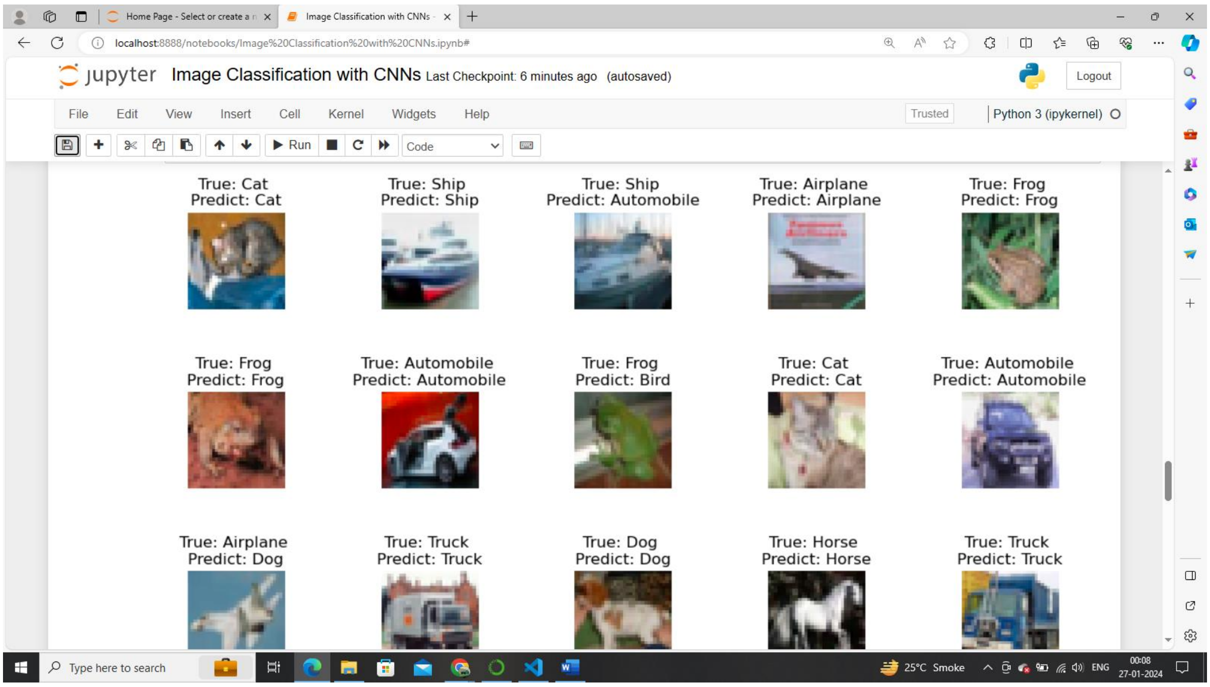
8. Classification Report:

A classification report is generated using scikit-learn's classification_report function. This report provides precision, recall, and F1-score for each class.

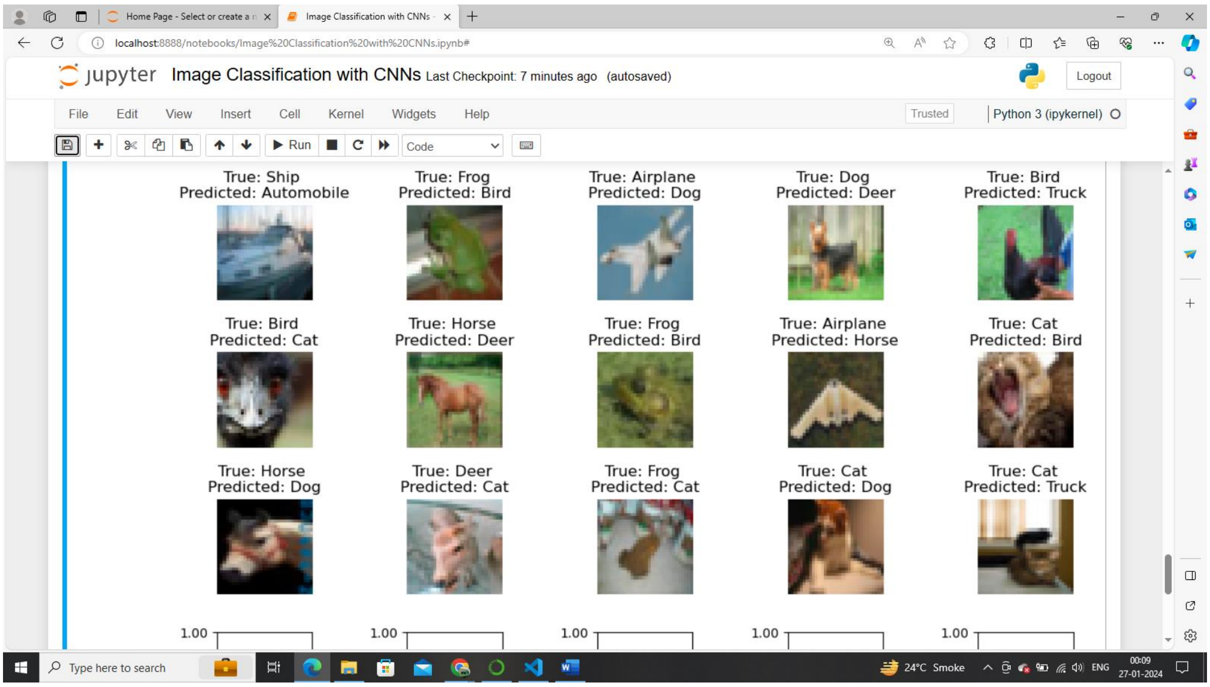
```
precision    recall  f1-score   support

        0       0.81      0.81      0.81      1000

        1       0.91      0.87      0.89      1000

        2       0.72      0.72      0.72      1000

        3       0.63      0.58      0.60      1000

        4       0.76      0.78      0.77      1000

        5       0.72      0.70      0.71      1000

        6       0.81      0.87      0.84      1000

        7       0.85      0.80      0.82      1000

        8       0.87      0.86      0.87      1000

        9       0.82      0.91      0.86      1000

   accuracy                         0.79     10000

  macro avg       0.79      0.79      0.79     10000

weighted avg       0.79      0.79      0.79     10000
```

9. Visualization:

Two sets of images are visualized:



A set of correctly classified images with their true and predicted labels.



A set of misclassified images with their true and predicted labels.

Results and Analysis:

The model achieves a satisfactory accuracy on the CIFAR-10 dataset. The classification report provides a detailed breakdown of the model's performance on individual classes. The visualization of correctly and misclassified images aids in understanding the model's strengths and areas for improvement.

Conclusion:

This implementation demonstrates the construction, training, and evaluation of a CNN for image classification using the CIFAR-10 dataset. Continuous refinement and experimentation with hyperparameters can further enhance the model's performance.