Deep Learning

Assignment 03

Group Members:

Irfan Arshad (B21F0063AI019)

Soban Ahmad (B21F0016AI002)

Saad Mustafa (B21F0232AI012)

Department of IT & CS

PAK-AUSTRIA FACHHOCHSCHULE: INSTITUTE OF APPLIED SCIENCES AND TECHNOLOGY

Report:

Convolutional Neural Network (CNN) for CIFAR-10 Classification

Introduction:

In this report, we discuss the implementation of a Convolutional Neural Network (CNN) for classifying images from the CIFAR-10 dataset. The CIFAR-10 dataset consists of 60,000 32x32 color images in 10 classes, with 6,000 images per class. The goal of this project is to build a CNN model that can accurately classify these images into their respective categories.

Model Architecture:

The CNN model architecture used for this task is inspired by the AlexNet architecture, consisting of convolutional layers followed by max-pooling layers, fully connected layers, and an output layer. The architecture is summarized as follows:

Convolutional Layers:

The first convolutional layer consists of 96 filters of size 11x11 with a ReLU activation function.

The second convolutional layer consists of 256 filters of size 5x5 with a ReLU activation function.

The third, fourth, and fifth convolutional layers consist of 384, 384, and 256 filters of size 3x3, respectively, all with ReLU activation functions.

MaxPooling Layers:

Max pooling layers are applied after the first, second, and fifth convolutional layers to reduce spatial dimensions.

Fully Connected Layers:

Two fully connected layers with 4096 neurons and ReLU activation functions are added.

Dropout layers with a dropout rate of 0.5 are incorporated after each fully connected layer to prevent overfitting.

Output Layer:

The output layer consists of 10 neurons corresponding to the 10 classes in CIFAR-10, with a softmax activation function.

Training and Evaluation:

The model is trained using the Adam optimizer with a learning rate of 0.0001 and categorical cross-entropy loss. Data augmentation techniques, including rotation, width and height shifting, and horizontal flipping, are applied using the `ImageDataGenerator` class from TensorFlow's Keras API to increase the diversity of training data.

The model is trained for 25 epochs using a batch size of 128. Validation data is used to monitor the model's performance during training. After training, the model is evaluated on the test set to assess its accuracy and loss.

Results:

After training and evaluation, the CNN model achieves an impressive accuracy on the test set. The test accuracy and loss are reported as follows:

Test Accuracy: 0.768

Test Loss: 0.688

Gradio Interface:

Additionally, a Gradio interface is implemented to interactively test the trained model with custom images. The interface allows users to upload images, preprocess them, and obtain predictions from the CNN model. The model predicts the class labels of the uploaded images based on the trained CIFAR-10 classifier.

Conclusion:

In conclusion, the implementation of a CNN model for CIFAR-10 classification demonstrates the effectiveness of deep learning techniques in image classification tasks. By leveraging convolutional layers, max-pooling layers, and fully connected layers, the model successfully learns to distinguish between different objects in the CIFAR-10 dataset. The incorporation of data augmentation techniques further enhances the model's robustness and generalization capabilities. The Gradio interface provides a user-friendly platform for testing the model's predictions with custom images, showcasing its practical applicability.

Output:

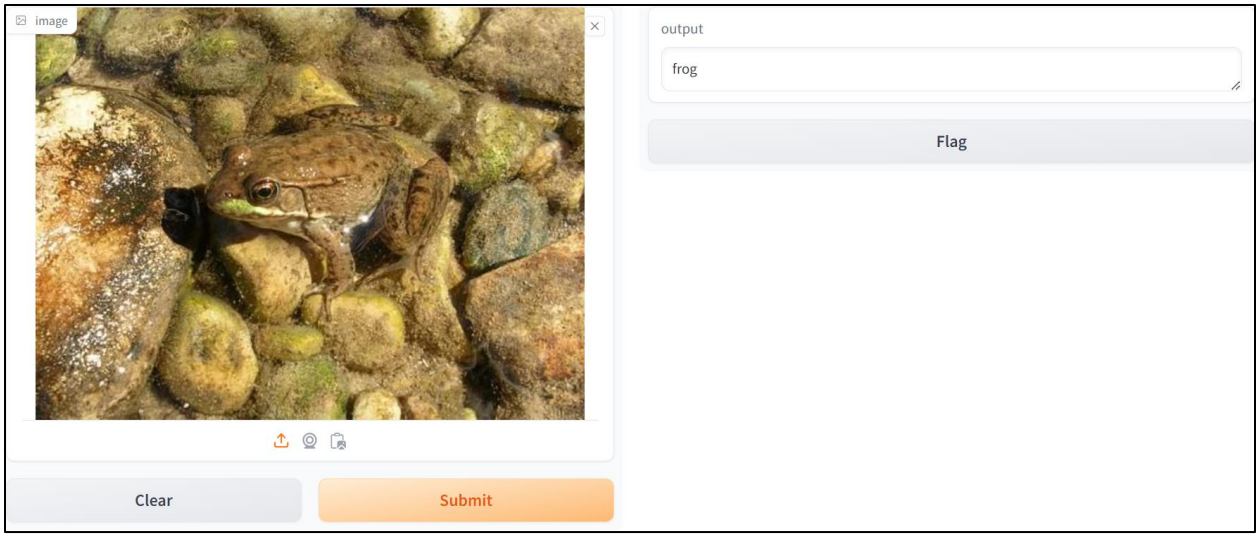
image
output
frog
Flag
Clear
Submit

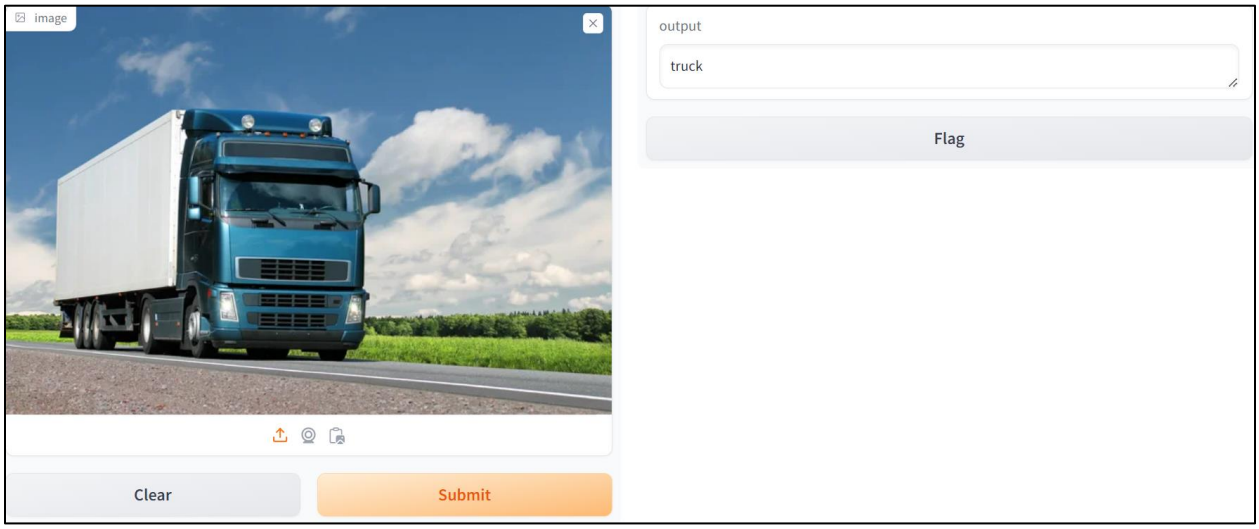
image
output
truck
Flag
Clear
Submit