Assignment: Image Classification Using CNNs

Objective

This assignment aims to help you understand the basics of Convolutional Neural Networks (CNNs), their implementation, and evaluation in image classification tasks.

Instructions

Complete the tasks below. Each task specifies the marks assigned. Submit your code, outputs, and a brief explanation for each step. Use the CIFAR-10 dataset for this assignment.

Steps to Load CIFAR-10 Dataset:

1. CIFAR-10 is readily available in libraries such as TensorFlow. Use the following instructions to import the data:

from tensorflow.keras.datasets import cifar10

Load the dataset

(x_train, y_train), (x_test, y_test) = cifar10.load_data()

Verify the shapes

print("Training data shape:", x_train.shape)

print("Test data shape:", x_test.shape)

Tasks

Task 1: Data Exploration and Preparation (2 Marks)

Instructions:

1. Display 5 sample images along with their corresponding labels after loading.

2. Print the shape of the dataset and the count of unique labels.

3. Normalize the image pixel values to the range [0, 1].

4. Split the dataset into training, and test sets (80%, 20%).

Expected Output:

- Sample images with labels.

- Dataset shape and label distribution.

Task 2: Build and Train a CNN Model (5 Marks)

Instructions:

1. Design a simple CNN model .

   o Example architecture:

     ▪ Conv2D → ReLU → MaxPooling → Dropout (repeat 2–3 times).

      ▪   Flatten → Fully Connected Layers → Output Layer with Softmax.

2. Compile the model using appropriate loss and optimizer.

3. Train the model on the training set for 10–20 epochs.

4. Plot the training and validation loss and accuracy curves.

Expected Output:

● CNN architecture summary.

● Training/validation loss and accuracy plots.

● Comment briefly on overfitting or underfitting based on the plots.

Task 3: Evaluate the Model (2 Marks)

Instructions:

1. Evaluate the model on the test set and calculate accuracy.

2. Generate a confusion matrix and classification report.

Expected Output:

● Test set accuracy and classification report.

● Confusion matrix visualization.

● Examples of correctly and incorrectly classified images.

Task 4: Experimentation with Model Improvements (1 Marks)

Instructions:

1. Introduce following techniques to improve model performance:

      o   Experiment with different optimizers (e.g., SGD, RMSProp).

Expected Output:

● Brief explanation of the changes applied.

● Performance comparison table (original vs. improved model).

Submission Requirements

1. Submit your code and output in a Jupyter Notebook or script file.

2. Use markdown cells or comments for explanations of each task.

3. Include plots, metrics, and examples as required.