**Assignment 3:** MNIST CNN Visualization

**Name: Email:**

Statement:

For this assignment's preparation, the author(s) did not use any generative AI tools.

For this assignment's preparation, the author(s) have utilized [Generative AI Tool Name], a language model created by [Generative AI Tool Provider]. Within this assignment, the [Generative AI Tool Name] was used for purposes such as [e.g., brainstorming, grammatical correction, writing paraphrasing, citation, specific sections of the assignment].”

**Objective**: Train a Convolutional Neural Network (CNN) on the MNIST dataset and visualize the learned features at different layers.

**Tasks**:

1. **CNN Implementation**:
   * Load the MNIST dataset.
   * Preprocess the dataset: normalize the images, convert labels to one-hot vectors, etc.
   * Design and implement a CNN model using TensorFlow/Keras.
   * Train the model on the training data and validate its performance on the validation data.
   * Report the test accuracy and loss.
2. **Visualization**:
   * Visualize the filters/kernels from the initial convolutional layers. What kind of features (e.g., edges, textures) can you infer from these visualizations?
   * Use techniques like feature map visualization to see the activations produced by different convolutional and pooling layers for a given input image.
   * Advanced (optional): Implement DeepDream or style transfer for MNIST, or visualize higher layer activations using dimensionality reduction techniques.

**GitHub Resources**:

<https://github.com/ZZUTK/An-Example-of-CNN-on-MNIST-dataset#kernels>

* This repository provides an example of a CNN on the MNIST dataset, with visualizations of the learned kernels in the first and second convolutional layers.
* It showcases the effect of using a wider or deeper CNN for better classification accuracy​.
* You can leverage the Github code, but please describe what kind of work you have done based on it.

**Assessment Criteria**:

1. **Code Quality** (30%): Readability, modularity, and efficiency.
2. **Model Performance** (10%): Achieving a high accuracy on the test dataset.
3. **Visualization Quality & Analysis** (30%): Clarity and relevance of visualizations, and the depth of analysis on what the CNN might be learning.
4. **Report** (30%): Clear documentation on implementation decisions, model architecture, and analysis of visualizations.