Saugat Malla

Final Project Proposal

Project: Building a Convolutional Neural Network (CNN) from Scratch in C++

1. Introduction:

- In this project, I aim to build a CNN from scratch using C++, implementing all the necessary components including convolutional layers, pooling layers, activation functions, and training algorithms without using any packages and libraries. The goal is to gain a deeper understanding of CNNs by implementing them from scratch and testing them on a dataset.

2. Objectives:

- Implement a CNN framework in C++ from scratch.
- Develop classes for convolutional layers, pooling layers, activation functions, and training algorithms.
- Implement forward propagation, backpropagation, and training loop specifically tailored for CNNs.
- Train and test the CNN on a dataset to evaluate its performance.
- Experiment with different architectures, activation functions, and optimization algorithms to understand their effects on network performance.

3. Methodology:

- Define the architecture of the CNN, including convolutional layers, pooling layers, and fully connected layers.
- Implement classes for ConvolutionalLayer, PoolingLayer, and FullyConnectedLayer, each with appropriate operations.
- Implement common activation functions such as ReLU, sigmoid, and tanh.
- Implement forward propagation to compute the output of the network given input data, including convolution, pooling, and activation.
- Implement backpropagation algorithm tailored for CNNs to compute gradients of the loss function with respect to network parameters.
- Implement training loop to update network parameters using gradient descent or other optimization algorithms.
- Choose a suitable loss function based on the task (e.g., cross-entropy for classification).
- Evaluate the performance of the trained CNN on a separate test dataset.

4. Dataset:

- A suitable dataset for training and testing CNN. This could be a dataset for image classification or object detection, depending on the objectives.
- Ensure the dataset is properly preprocessed and split into training and testing sets.