Labwork 6: CNN

Phi Doan Minh Luong - 2440046

May 2025

1 Design

1.1 Convolutional Layers

- The architecture follows the VGG19 design, consisting of convolutional layers with increasing filter sizes (64, 128, 256, 512)
- Each convolutional layer uses a kernel size of 3x3, stride of 1, and padding of 1 to preserve spatial dimensions
- Batch normalization is applied after each convolution to stabilize training
- ReLU activation is used for non-linearity

1.2 Pooling Layers

- Max-pooling layers with a kernel size of 2x2 and stride of 2 are interspersed to reduce spatial dimensions

1.3 Fully Connected Layers

- After the convolutional layers, the feature maps are flattened and passed through three fully connected layers
- The first two fully connected layers have 4096 units each, followed by ReLU activation and dropout for regularization
- The final fully connected layer outputs predictions for the number of classes

1.4 Input and Output

- The network accepts input images of size 224x224x3 and outputs class probabilities for the specified number of classes

2 Implementation

2.1 Class Definition

- The VGG19 class is implemented using PyTorch's nn.Module

- The convolutional layers are dynamically created based on the VGG_architecture list

2.2 Forward pass

- The input is passed through the convolutional layers, flattened, and then processed by the fully connected layers

2.3 Training and Testing

- The network is trained using the CIFAR-10 dataset with cross-entropy loss and the Adam optimizer
- Accuracy is evaluated on the test set