MACHINE LEARNING

WEEK-14

CNN Image Classification using PyTorch

Name: Gavini Prithvi Ananya

Sec: C

SRN: PES2UG23CS198

Introduction

The objective of this lab was to design, implement, and train a Convolutional Neural Network (CNN) using PyTorch to classify images of hand gestures into three categories: rock, paper, and scissors. The dataset was loaded using `ImageFolder`, pre-processed with standard transforms, and split into training and testing sets. The goal was to build an effective CNN model, train it on the dataset, evaluate its performance on unseen images, and analyse the results.

Model Architecture:

1. Describe the CNN architecture you built.

2. Mention key parameters like kernel size, number of channels, and the use of Max Pooling.

3. Describe the fully connected classifier.

The CNN model consists of three convolutional blocks followed by a fully connected classification head.

Convolutional Layers

The architecture includes:

1.Conv Block 1

a. `Conv2d(3 → 16, kernel_size=3, padding=1)`
b. Activation: ReLU
c. Downsampling: MaxPool2d(2)

2. **Conv Block 2**
    a. `Conv2d(16 → 32, kernel_size=3, padding=1)`
    b. Activation: ReLU
    c. Downsampling: MaxPool2d(2)
3. **Conv Block 3**
    a. `Conv2d(32 → 64, kernel_size=3, padding=1)`
    b. Activation: ReLU
    c. Downsampling: MaxPool2d(2)

After three max-pool layers, the spatial dimensions reduce:

- $128 \rightarrow 64 \rightarrow 32 \rightarrow 16$
  So the feature map becomes **64 × 16 × 16**.

Training and Performance:

○ State the key hyperparameters used for training: optimizer, loss function, learning rate, and number of epochs.

○ Report the final Test Accuracy your model achieved.

During training, the CNN was optimized using the Adam optimizer with a learning rate of 0.001, a batch size of 32, and the CrossEntropyLoss function. The model was trained for 10 epochs, where each epoch involved forward propagation, loss calculation, backpropagation, and weight updates. Throughout the training process, the loss steadily decreased, indicating that the model was effectively learning the features required for classification. After training, the model was evaluated on the test dataset, achieving a final accuracy of 98.40%, demonstrating strong performance on the Rock-Paper-Scissors gesture recognition task.

Conclusion and Analysis:

○ Briefly discuss your results. Did the model perform well?

○ Were there any challenges you faced?

○ Suggest one or two ways you could potentially improve the model's accuracy in the future.

Performance Evaluation

The trained CNN performed well on the classification task, achieving high accuracy on the test set. This indicates that the model successfully learned discriminative features for rock, paper, and scissors gestures.

Challenges Faced

- Ensuring the correct preprocessing transforms (resize, normalize).
- Understanding spatial dimension changes after repeated max pooling.
- Balancing training time with accuracy within the given constraints.

Possible Improvements

1. **Add Data Augmentation**
   Techniques like random rotation, flipping, or color jitter could improve generalization.
2. **Use Transfer Learning**
   Models like ResNet-18 or MobileNetV2 could significantly boost accuracy with fewer epochs.
3. **Tune Hyperparameters**
   a. Higher epoch count
   b. Learning rate scheduling
   c. Larger fully connected layer