MACHINE LEARNING-LAB-WEEK-14

NAME:DRISHTI GOLCHHA

SRN:- PES2UG23CS185

Introduction

The objective of this lab was to design 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 sourced from Kaggle and contained labeled images organized into class-specific folders. The task involved completing the boilerplate notebook, implementing missing components such as data preprocessing, model architecture, training loops, and evaluation metrics, and finally generating predictions. This lab helped reinforce concepts of deep learning, image preprocessing, convolutional layers, and performance evaluation of vision models.

Model Architecture

The CNN consists of three convolutional layers, each using a 3×3 kernel with padding 1, followed by ReLU activation and MaxPooling to reduce the image size. The layers progressively increase the number of channels from 16 to 32 to 64. After feature extraction, the output is flattened and passed into a fully connected classifier with a 256-unit layer, ReLU activation, dropout for regularization, and a final linear layer that produces three class outputs (rock, paper, scissors).

Training And Performance

```
Classes: ['paper', 'rock', 'scissors']
Total images: 2188
Training images: 1750
Test images: 438
```

```
RPS_CNN(
  (conv_block): Sequential(
    (0): Conv2d(3, 16, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): ReLU()
    (2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (3): Conv2d(16, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (4): ReLU()
    (5): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (6): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (7): ReLU()
    (8): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  )
  (fc): Sequential(
    (0): Flatten(start_dim=1, end_dim=-1)
    (1): Linear(in_features=16384, out_features=256, bias=True)
    (2): ReLU()
    (3): Dropout(p=0.3, inplace=False)
    (4): Linear(in_features=256, out_features=3, bias=True)
  )
)
```

```
Epoch 1/10, Loss = 0.6509
Epoch 2/10, Loss = 0.2407
Epoch 3/10, Loss = 0.1069
Epoch 4/10, Loss = 0.0420
Epoch 5/10, Loss = 0.0237
Epoch 6/10, Loss = 0.0186
Epoch 7/10, Loss = 0.0182
Epoch 8/10, Loss = 0.0068
Epoch 9/10, Loss = 0.0049
Epoch 10/10, Loss = 0.0026
Training complete!
```

```
Test Accuracy: 98.63%
```

```
Randomly selected images:
Image 1: /content/dataset/paper/UJdZdj6nyFRHVFEg.png
Image 2: /content/dataset/paper/BoXJ7o2SeSRRfuAH.png

Player 1 shows: paper
Player 2 shows: paper

RESULT: Draw
```

Conclusion:-

The model performed well and achieved good accuracy in classifying rock, paper, and scissors images. The main challenge was preventing overfitting due to limited dataset size. Accuracy could be improved further by using data augmentation or enhancing the architecture with techniques like batch normalization or deeper layers.