ML Lab Week 14: CNN Image Classification

Name	SRN	Section
C PANSHUL REDDY	PES2UG23CS154	C

Date:19-11-2025

1. Introduction:

This lab focused on building, training, and evaluating a Convolutional Neural Network (CNN) to classify hand gesture images in the game of Rock, Paper, Scissors. The objective was to develop a deep learning model capable of accurately recognizing which gesture a hand is displaying from image data. The lab involved downloading and preprocessing a dataset of 2,188 images, designing a three-layer CNN architecture with convolutional blocks for feature extraction and fully connected layers for classification, training the model over 10 epochs using the Adam optimizer and cross-entropy loss, and evaluating its performance on unseen test data.

2. Model Architecture:

The CNN architecture consists of two main components: a convolutional feature extraction block and a fully connected classification block.
The convolutional block contains three sequential layers, each following the pattern of Conv2d → ReLU → MaxPool2d:

- **Layer 1:** Uses a 3×3 kernel to transform the 3 input RGB channels into 16 feature maps with padding=1 to preserve spatial dimensions, followed by ReLU activation and 2×2 max pooling that reduces the feature map size by half (128×128 → 64×64).
- **Layer 2:** Expands from 16 to 32 channels using the same 3×3 kernel size and 2×2 pooling configuration, further reducing dimensions (64×64 → 32×32).
- **Layer 3:** Increases to 64 channels with identical kernel and pooling parameters, producing final feature maps of size 16×16.
  After three max pooling operations, the spatial dimensions are reduced from 128×128 to 16×16, resulting in 64×16×16 = 16,384 flattened features.

The fully connected classifier transforms these features into class predictions through the following layers:

- **Flatten Layer:** Converts the 3D feature maps into a 1D vector of 16,384 features.
- **Linear Layer 1:** Reduces dimensionality from 16,384 to 256 neurons, followed by ReLU activation.
- **Dropout Layer:** Applies dropout with probability p=0.3 for regularization to prevent overfitting.
- **Linear Layer 2 (Output):** Maps the 256 features to 3 class scores representing rock, paper, and scissors.
  The model was trained using the Adam optimizer with a learning rate of 0.001 and cross-entropy loss function for multi-class classification.

3. Training and Performance:

The model was trained using carefully selected hyperparameters to optimize performance. The **Adam optimizer** was employed due to its adaptive learning rate capabilities and efficient convergence properties. The **cross-entropy loss function** was used as the optimization criterion, which is well-suited for multi-class classification problems. A **learning rate of 0.001** was set to ensure stable and consistent weight updates throughout training. The model was trained for **10 epochs** with a batch size of 32, processing the training dataset of 1,750 images split from the original 2,188 total images (80% training, 20% testing).

During training, the model demonstrated excellent convergence, with the loss decreasing from 0.6628 in the first epoch to 0.0040 by the tenth epoch. This significant reduction in training loss indicated that the model effectively learned the distinguishing features of rock, paper, and scissors hand gestures.

Upon evaluation on the unseen test set of 438 images, the model achieved a **final test accuracy of 98.17%**. This outstanding performance demonstrates the model's strong generalization capability and its ability to accurately classify new, previously unseen hand gesture images with high reliability.

4. Conclusion and Analysis:

The model performed exceptionally well, achieving a test accuracy of 98.17%, which indicates that it successfully learned to distinguish between rock, paper, and scissors hand gestures with high precision. The dramatic decrease in training loss from 0.6628 to 0.0040 over 10 epochs demonstrates effective learning and convergence. The model's strong performance on unseen test data confirms its ability to generalize beyond the training set, making it reliable for real-world classification tasks.

During the lab, several challenges were encountered.The computational intensity of training a CNN, even on a relatively small dataset, highlighted the importance of efficient architecture design and the potential benefits of GPU acceleration for larger-scale applications.

To further improve the model's accuracy, two key enhancements could be implemented. First, **data augmentation** techniques such as random rotations, flips, brightness adjustments, and minor translations could be applied during training to artificially expand the dataset and improve the model's robustness to variations in hand positioning and lighting conditions. Second, **transfer learning** could be leveraged by using a pre-trained model like ResNet or VGG as the feature extractor and fine-tuning it on the rock-paper-scissors dataset, which would allow the model to benefit from features learned on large-scale image datasets and potentially achieve even higher accuracy with fewer training epochs.