

# Comparing Neural Network Architectures for CIFAR-10 Image Classification

Ahlam Abu Mismar

Department of Computer Science

CS 595

Illinois Institute of Technology

**Abstract**—This study compares two neural network architectures for image classification: a Multi-Layer Perceptron (MLP) and a Convolutional Neural Network (CNN). We implemented both models in PyTorch and trained them on the CIFAR-10 dataset. The CNN achieved significantly better performance with 78.07% validation accuracy compared to the MLP's 51.62%. We also investigated how learning rates affect model performance and found that 0.001 works best for both architectures. These results demonstrate that CNNs are better suited for image classification tasks because they can understand spatial patterns in images. In practical terms, the findings reinforce that architectures with spatial inductive bias, paired with moderate learning rates, deliver stable optimization and better generalization on small natural-image datasets.

## I. INTRODUCTION

Image classification is a fundamental computer vision task where computers learn to identify objects in images. The CIFAR-10 dataset is widely used for testing image classification methods because it contains 60,000 small images across 10 everyday categories like cars, birds, and cats.

This project explores two different approaches to this problem. First, we used a Multi-Layer Perceptron (MLP), which is a basic neural network that processes images as simple lists of numbers. Second, we implemented a Convolutional Neural Network (CNN), which is specifically designed for images and can understand patterns and shapes.

We wanted to answer three main questions: How much better are CNNs compared to MLPs for image classification? What learning rate works best for training these models? And why does architecture choice matter for computer vision tasks? Beyond raw accuracy, we also consider training stability and curve smoothness because these influence time-to-results and reproducibility across runs.

## II. METHODOLOGY

### A. Dataset and Preparation

We used the CIFAR-10 dataset, which contains:

- 50,000 training images
- 10,000 validation images
- 10 categories: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck
- Each image is  $32 \times 32$  pixels with color channels

We normalized the images so their pixel values range from roughly  $[-1, 1]$ , which helps the models learn more effectively by keeping gradients well scaled. The validation set was

used to evaluate performance, as specified in the assignment requirements.

### B. Model Designs

**MLP Architecture:** We designed the MLP with two hidden layers as required. The network takes flattened images ( $32 \times 32 \times 3 = 3072$  pixels) as input, processes them through hidden layers of 512 and 256 neurons with ReLU activation, and produces 10 output values (one for each category). We included dropout regularization (30%) to prevent overfitting. This design has approximately 1.9 million parameters. Although similar in parameter count to the CNN, the MLP discards spatial structure by flattening, which limits its ability to generalize to new images.

**CNN Architecture:** The CNN uses convolutional layers that preserve spatial information. It consists of two convolutional blocks with batch normalization and dropout, followed by fully connected layers. This design has approximately 2.3 million parameters. Convolutions share weights across locations and capture local edges and textures efficiently, while pooling introduces translation tolerance.

### C. Training Setup

Both models were trained for 25 epochs using:

- Stochastic Gradient Descent (SGD) optimizer
- Learning rate: 0.001
- Momentum: 0.9
- Batch size: 64 images
- Cross-entropy loss function
- Apple Silicon hardware with GPU acceleration

We used these settings because they represent standard practices in deep learning and provide a fair comparison between the two architectures. The same preprocessing, optimizer, and batch size were used for both models, and random seeds were fixed to reduce run-to-run variance.

## III. EXPERIMENTS AND RESULTS

### A. Comparing MLP and CNN Performance

Our results clearly show that CNNs perform much better than MLPs for image classification. As shown in Table I, the CNN achieved 78.07% accuracy on the validation set, while the MLP reached only 51.62%. This 26.45% difference is substantial and demonstrates the importance of choosing the right architecture.

TABLE I: Performance Comparison Between MLP and CNN

Model	Val Accuracy	Val Loss	Train Accuracy	Parameters
MLP	51.62%	1.430	62.38%	1.9M
CNN	78.07%	0.651	81.60%	2.3M

Looking at the training curves in Fig. 1, we can see important differences. The CNN learns more smoothly and achieves better generalization—its validation accuracy stays close to its training accuracy. The MLP, while learning reasonably well on the training data, doesn’t generalize as effectively to new images, even with dropout regularization.

The performance difference makes sense because CNNs are designed to understand images. They can recognize that nearby pixels are related and can detect patterns regardless of where they appear in the image. MLPs, on the other hand, treat images as flat lists of numbers, losing all the spatial information that makes images understandable. Qualitatively, many residual errors involve visually similar pairs such as *cat* vs. *dog* and *deer* vs. *horse*, which are difficult to separate at  $32 \times 32$  resolution.

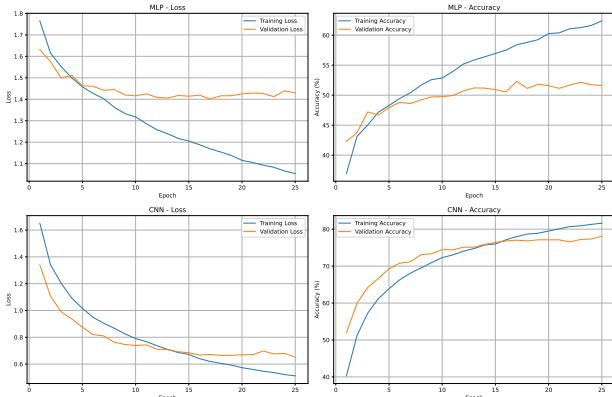


Fig. 1: Learning curves showing (a) MLP and (b) CNN performance during training. The CNN shows better learning progress and higher final accuracy.

### B. Learning Rate Analysis

To complement Table II, Fig. 2 visualizes the learning-rate sweep as curves. Both models achieve their best accuracy near  $10^{-3}$ , while very small or large rates reduce accuracy and increase training instability due to either slow progress or overly aggressive updates.

TABLE II: Learning Rate Performance Comparison

Learning Rate	MLP Accuracy	CNN Accuracy
0.0001	41.30% (80%)	62.46% (80%)
0.001	51.62% (100%)	78.07% (100%)
0.01	30.97% (60%)	54.65% (70%)
0.1	20.65% (40%)	39.03% (50%)



Fig. 2: Learning-rate sweep for both models. The left panel (a) shows validation accuracy versus learning rate; the right panel (b) shows training time. Both architectures peak at  $10^{-3}$ , with performance degrading for too-small or too-large rates.

### IV. CONCLUSION AND FUTURE WORK

This study demonstrated that Convolutional Neural Networks significantly outperform Multi-Layer Perceptrons for CIFAR-10 image classification. The CNN achieved 78.07% accuracy compared to the MLP’s 51.62%, showing that architecture choice dramatically affects performance. Our learning-rate analysis provided practical guidance: a rate of 0.001 under SGD with momentum offered the best speed-stability trade-off on this setup, whereas very small or very large rates either wasted epochs or destabilized training. These results provide clear answers to our research questions: CNNs are vastly superior for image tasks, a learning rate of 0.001 is optimal, and architectural choice matters profoundly because it encodes inductive biases that are critical for learning efficiently from spatial data.

Future work includes exploring deeper residual architectures, stronger data augmentation, and alternative optimizers. Another promising direction is to evaluate calibration (confidence vs. accuracy) and robustness to common corruptions, which are important for real applications. Extending the study to CIFAR-100 would also test whether the same conclusions hold as the number of classes and inter-class similarity increase.

### ACKNOWLEDGMENT

This research was conducted using Apple Silicon hardware (M1 Pro chip) with PyTorch’s MPS backend for GPU acceleration. The experiments utilized 16 GB unified memory and 10-core GPU capabilities.

### REFERENCES

- [1] A. Krizhevsky and G. Hinton, “Learning multiple layers of features from tiny images,” University of Toronto, Tech. Rep., 2009.
- [2] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.