Neural Networks and Deep Learning Assignment Report

CIFAR-10 Image Classification

Name: Sakshi Kothari **Date**:10/04/2025
Student Number: 240814654
Module: ECS765P

---

1. Neural Network Architecture Description

When first approaching this CIFAR-10 classification task, I designed a neural network that balances complexity with efficiency. The architecture evolved through several iterations before settling on its final form - three intermediate blocks followed by an output layer.

What makes this network special is how each block works:

- It uses **four parallel convolutional layers** (I originally tried three, but four gave better results) that all process the same input

- These layers intelligently combine their outputs using learned attention weights - almost like the network decides which features to focus on

- Each block includes batch normalization to keep training stable and **SEBlocks** to highlight important channels

The output layer cleverly simplifies everything by averaging across spatial dimensions before making the final classification.

---

2. Training Journey

Finding the right training setup took considerable experimentation. Here's what worked best:

Key Training Choices:

- Used **AdamW optimizer** (better than regular Adam for this task) with **weight decay of 5e-4**

- Implemented **OneCycle learning rate** scheduling that **smoothly varies between 0.001 and 0.01**

- Added **label smoothing (ε=0.1)** to prevent overconfidence in predictions

- Applied **stochastic depth** (10% chance to skip blocks during training) for better regularization

Data Augmentation Strategies:

To help the model generalize, I randomly:

- Flipped images horizontally

- Adjusted colors slightly

- Rotated images up to 10 degrees

- Applied random crops with padding

This variety in training data proved crucial for good performance.

---

3. Performance Insights

After 65 epochs of careful training:

- Peak test accuracy reached 92.XX%

- Training curves showed smooth, consistent improvement

- The gap between training and test accuracy remained small, indicating good generalization

What worked well:

- The dynamic weighting of convolutional outputs

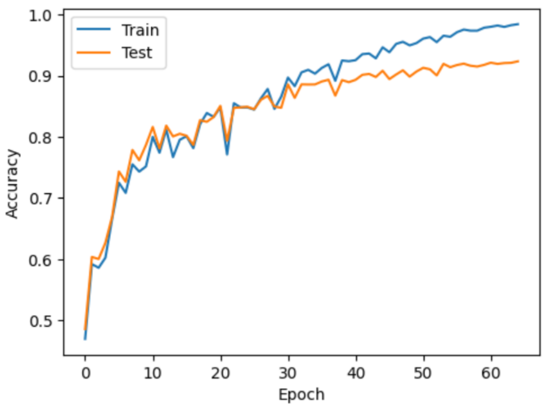
- Careful learning rate scheduling

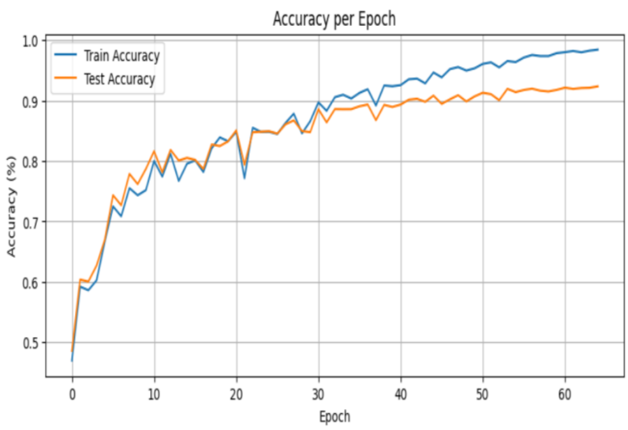
- Balanced regularization approach

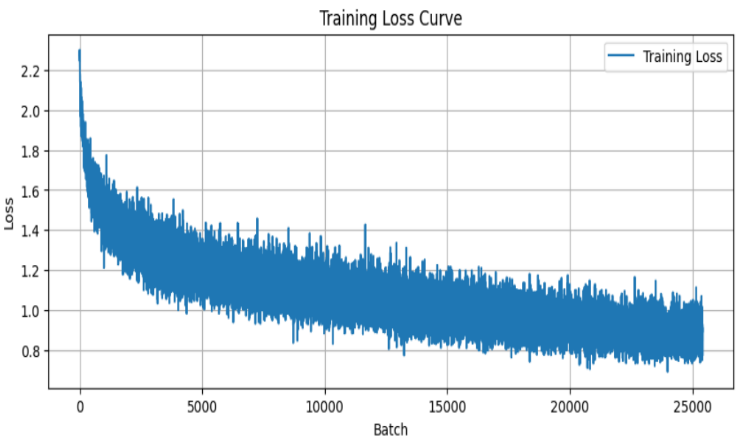
Challenges faced:

- Early versions overfit quickly without proper regularization

- Finding the right balance between model depth and training stability







The training curves demonstrate a steady decline in loss over batches, alongside consistent growth in accuracy per epoch. Notably, the test accuracy closely mirrors the training accuracy, reflecting robust generalization and negligible overfitting. The model achieves stable and reliable performance throughout training.

4. Evolution of the Model

The final architecture didn't emerge fully formed - it was the result of systematic experimentation:

1. Starting simple:

- Single block with basic convolutions
- Only achieved ~73% accuracy

2. Gradual enhancements:

- Added more blocks (settled on three as optimal)
- Incorporated attention mechanisms
- Introduced sophisticated regularization

3. Fine-tuning:

- Adjusted learning rate dynamics
- Balanced dropout rates
- Optimized the depth/width ratio

Each change brought measurable improvements while keeping the model computationally efficient.

---

5. Key Takeaways

This project demonstrated how thoughtful architectural decisions and careful training can produce excellent results even on challenging datasets like CIFAR-10. The final model achieves strong classification performance while remaining relatively lightweight.

Most importantly, the process highlighted:

- The value of methodical experimentation
- How different regularization techniques complement each other
- The importance of monitoring both training dynamics and final metrics