EEP 596 Computer Vision - Assignment 4 Report

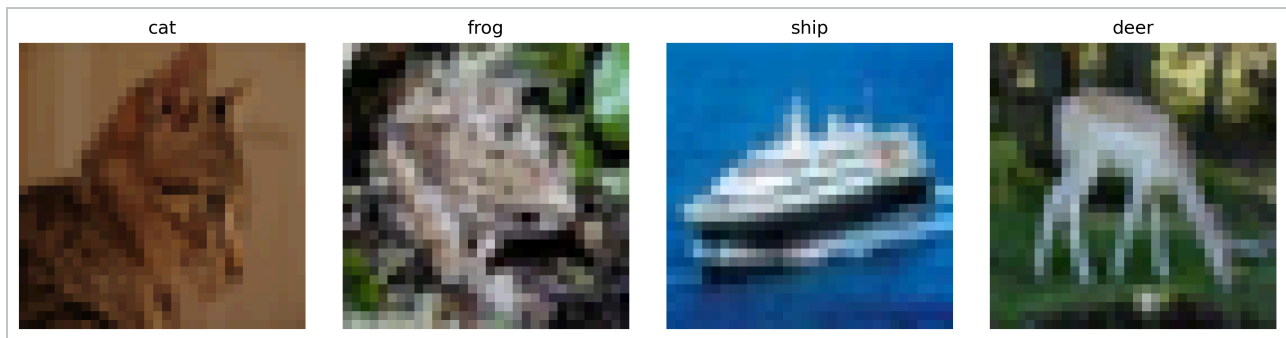## Task 1: CIFAR-10 Dataset

Dataset Statistics

Parameter	Value
num_train_batches	5
num_test_batches	1
num_img_per_batch	10000
num_train_img	50000
num_test_img	10000
size_batch_bytes (KB)	30000 KB (~29.3 MB)
size_image_bytes (KB)	3.072 KB (3072 bytes for 32x32x3)
size_batchimage_bytes (10k images, KB)	30720 KB (~30 MB)

1a. Sample Mini-batch (4 random images)

cat | frog | ship | deer

Image tensor shape: torch.Size([4, 3, 32, 32])

Labels tensor shape: torch.Size([4])

Task 2: Train Classifier

Network Architecture

```
Net(
  (conv1): Conv2d(3, 6, kernel_size=(5, 5), stride=(1, 1))
  (pool): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (conv2): Conv2d(6, 16, kernel_size=(5, 5), stride=(1, 1))
  (fc1): Linear(in_features=400, out_features=120, bias=True)
  (fc2): Linear(in_features=120, out_features=84, bias=True)
  (fc3): Linear(in_features=84, out_features=10, bias=True)
)
```

Test Accuracy

Test Accuracy on 10,000 images: 52.42%

Model weights loaded from: `cifar_net_2epoch.pth`

Task 3: Visualize Weights

3a. First Layer (conv1) Weights

> **Shape:** [6, 3, 5, 5]
>
> **Description:** 6 filters, 3 input channels, 5x5 kernel size

3b. Second Layer (conv2) Weights

> **Shape:** [16, 6, 5, 5]
>
> **Description:** 16 filters, 6 input channels, 5x5 kernel size

Task 4: Hyperparameter Sweep

4c. Training with Different Learning Rates

The network was trained for 2 epochs using three different learning rates: 0.01, 0.001, and 0.0001. Training loss and error rates were recorded every 2000 iterations.
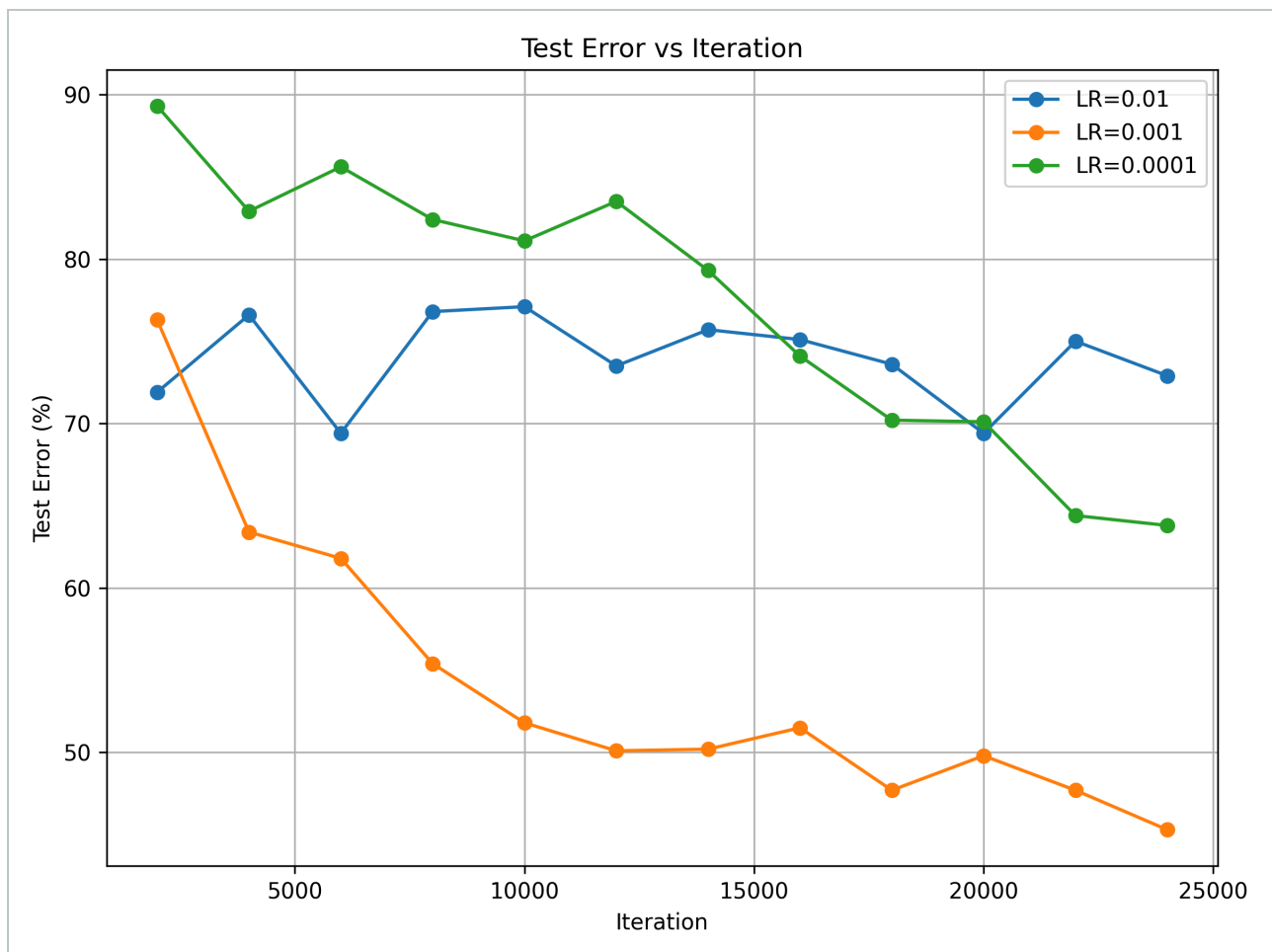
i. Training Loss vs Iteration

Training Loss vs Iteration

ii. Training Error vs Iteration

Training Error vs Iteration

iii. **Test Error vs Iteration**

Test Error vs Iteration

4d. Results Description

Observations:

- **Learning Rate = 0.01 (High):** Higher learning rate didn't bring faster convergence. Shows severe oscillation throughout training. In final iterations, it even jumps out of the current valley, making the result worse than the other two learning rates.
- **Learning Rate = 0.001 (Medium):** Shows steady, consistent decrease in both training loss and error. This learning rate provides a good balance between convergence speed and stability.
- **Learning Rate = 0.0001 (Low):** Training progresses slowly with gradual decrease in loss and error. The convergence is more stable but requires more iterations to reach comparable performance.
- **Test Error:** The test error patterns generally follow the training error trends, indicating the models generalize reasonably well without severe overfitting. The medium learning rate (0.001) typically achieves the best test performance within 2 epochs.

- **Conclusion:** The learning rate of 0.001 appears to be optimal for this network and dataset, providing stable convergence and good test performance within the limited training time.

End of Report