

AlexNet Training Report

Code Implementation

I implemented AlexNet following the original architecture specifications with some modern adaptations. The key components include:

First layer (Conv1): 11x11 kernels, 64 filters, stride 4

Classifier Layers:

The training loop includes learning rate scheduling: - Epochs 1-15: $lr = 0.01$ - Epochs 16-25: $lr = 0.001$ - Epochs 26-30: $lr = 0.0001$

Training Analysis

Training Trends

Loss: The training and test loss show rapid decrease in the first few epochs, indicating efficient learning of features. The convergence pattern suggests the model is learning without significant overfitting.

Accuracy: The test accuracy shows steady improvement, particularly in the early epochs, reaching 22% by the end of training.

Circular Variance: The kernel circular variance plots indicate that directional selectivity begins to emerge around 15 epochs into training. This suggests that the network starts learning orientation-specific features relatively early in the training process.

Kernel Visualization Analysis

The first layer kernels show several interesting patterns: - Multiple Gabor-like filters that detect edges at various orientations - Color-opponent kernels that respond to specific color transitions - Some high-frequency texture detectors - [Add any other notable patterns you observe]

Individual Filter Analysis