CNN for CIFAR-10 Classification

Anas Bin Rashid[1]

Department of Computer Science
National University of Computer and Emerging Sciences,
Islamabad, Pakistan
i220907@nu.edu.pk

Abstract. This report presents the implementation and evaluation of a Convolutional Neural Network (CNN) for CIFAR-10 image classification. We conducted comprehensive ablation studies examining learning rate, batch size, number of filters, and network depth. The optimized CNN achieved 85.34% accuracy on the CIFAR-10 test set, demonstrating effective spatial feature learning through hierarchical convolutional layers.

Keywords: Convolutional Neural Networks · CIFAR-10 · Image Classification · Ablation Study

1 Introduction

Convolutional Neural Networks have become the standard approach for image classification tasks due to their ability to learn hierarchical spatial features. This work focuses on implementing and optimizing a CNN architecture for the CIFAR-10 dataset, which consists of 60,000 32×32 color images across 10 classes.

The primary objectives are: (1) to design and implement an effective CNN architecture, (2) to conduct systematic ablation studies on key hyperparameters, and (3) to analyze the learned feature representations through visualization.

2 Methodology

2.1 Architecture Design

We implemented a multi-layer CNN architecture consisting of alternating convolutional and pooling layers, followed by fully connected layers for classification.

Network Components:

- **Convolutional Layers:** Extract spatial features using learnable filters with ReLU activation
- **Batch Normalization:** Stabilize training by normalizing layer inputs
- **Max Pooling:** Reduce spatial dimensions and provide translation invariance
- **Dropout:** Prevent overfitting through random neuron deactivation during training

– **Fully Connected Layers:** Perform final classification based on extracted features

Base Architecture: The network follows the pattern: Conv → BatchNorm → ReLU → MaxPool, repeated multiple times, followed by fully connected layers with dropout and a softmax output layer for 10-class classification.

2.2 Data Preprocessing

Normalization: CIFAR-10 images were normalized using the dataset's computed statistics:

– Mean: $(0.4914, 0.4822, 0.4465)$
– Standard Deviation: $(0.2023, 0.1994, 0.2010)$

Data Augmentation: To improve generalization and prevent overfitting, we applied the following augmentations to training data:

– Random horizontal flips (probability = 0.5)
– Random crops with padding (padding = 4 pixels)

3 Experimental Setup

3.1 Ablation Study Design

We conducted a systematic ablation study examining four critical hyperparameters:

1. **Learning Rate:** $\{0.001, 0.01, 0.1\}$
   – Controls the step size for weight updates
   – Too high: unstable training; too low: slow convergence
2. **Batch Size:** $\{16, 32, 64\}$
   – Affects gradient estimation quality and training speed
   – Smaller batches: noisier gradients; larger batches: smoother but memory-intensive
3. **Number of Convolutional Filters:** $\{16, 32, 64\}$
   – Determines model capacity and feature representation richness
   – More filters: higher capacity but increased computation
4. **Number of Layers:** $\{3, 5, 7\}$
   – Controls network depth and feature abstraction levels
   – Deeper networks: more complex features but harder to train

3.2 Training Configuration

Optimizer: Adam optimizer with default parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$)
Loss Function: Cross-entropy loss for multi-class classification
Training Duration: 100 epochs with early stopping (patience = 10 epochs)
Hardware: Training conducted on GPU (NVIDIA Tesla V100)

4 Results and Analysis

4.1 Overall Performance

The final optimized model achieved strong performance on the CIFAR-10 test set:

Table 1. CNN Performance Metrics on CIFAR-10 Test Set

Model	Accuracy	Precision	Recall	F1-Score
Optimized CNN	0.8534	0.8521	0.8534	0.8527

The model demonstrates balanced performance across all metrics, indicating effective learning without significant class bias.

4.2 Ablation Study Results

Learning Rate Impact:

- **0.001:** Best convergence with stable training (selected)
- **0.01:** Faster initial convergence but less stable
- **0.1:** Training instability and poor final performance

Batch Size Impact:

- **16:** Noisy gradients, slower training
- **32:** Optimal balance of stability and efficiency (selected)
- **64:** Smoother training but diminishing returns

Number of Filters Impact:

- **16:** Insufficient capacity for complex features
- **32:** Best accuracy-complexity trade-off (selected)
- **64:** Marginal accuracy improvement with doubled computation

Network Depth Impact:

- **3 layers:** Underfitting, insufficient feature extraction
- **5 layers:** Optimal depth without overfitting (selected)
- **7 layers:** Signs of overfitting, diminishing test performance

4.3 Feature Map Visualization Analysis

Visualization of intermediate feature maps revealed progressive feature extraction:

Layer 1 (Early Layers):

- Detected low-level features: edges, corners, color gradients
- Feature maps showed high activation on object boundaries
- Resembled Gabor-like filters for edge detection

Layer 3 (Middle Layers):

- Combined simple features into more complex patterns
- Activation for textures, simple shapes, and local structures
- Beginning of object-part detection

Layer 5 (Deep Layers):

- Captured high-level semantic representations
- Class-specific activations (e.g., wheels for cars, wings for planes)
- Abstract feature representations less interpretable but highly discriminative

4.4 Per-Class Performance

Analysis of confusion matrix revealed:

- Strong performance on: automobiles (91%), ships (89%), trucks (88%)
- Challenging classes: cats vs. dogs confusion (78% each)
- Birds showed confusion with airplanes due to similar sky backgrounds

5 Discussion

5.1 Key Findings

The ablation study revealed several important insights:

Moderate Architectures Excel: For CIFAR-10's relatively small 32×32 images, moderate architectures (5 layers, 32 filters) outperformed both shallow and very deep networks. This suggests that excessive depth can lead to overfitting on smaller datasets.

Learning Rate Sensitivity: The learning rate proved to be the most critical hyperparameter, with optimal value of 0.001 providing stable convergence. Higher rates caused training instability, while lower rates would require significantly more training time.

Batch Size Trade-offs: Batch size of 32 provided the best balance between gradient estimation quality and computational efficiency. Smaller batches introduced excessive noise, while larger batches showed diminishing returns.

5.2 Limitations

- Limited architectural exploration (no skip connections, residual blocks)
- Single run per configuration (no statistical significance testing)
- Memory constraints limited maximum batch size and model size
- No learning rate scheduling implemented

6 Conclusion

This work successfully implemented and evaluated a CNN for CIFAR-10 classification, achieving 85.34% test accuracy through systematic hyperparameter optimization. The ablation study demonstrated that moderate architectures with careful hyperparameter tuning can achieve strong performance on small-scale image classification tasks.

Key contributions include:

1. Comprehensive ablation study identifying optimal hyperparameters
2. Feature map analysis revealing hierarchical feature learning
3. Performance metrics demonstrating effective classification across 10 classes

6.1 Future Work

Potential improvements include:

- Implementing advanced architectures (ResNet, DenseNet)
- Adding learning rate scheduling for improved convergence
- Exploring more sophisticated data augmentation techniques
- Implementing ensemble methods for improved accuracy
- Conducting statistical significance testing across multiple runs

References

1. Krizhevsky, A.: Learning multiple layers of features from tiny images. Technical Report, University of Toronto (2009)
2. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
3. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
4. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning, pp. 448–456 (2015)