

# Classification of Street View House Numbers(SVHN) using VGG Neural Network

DC127235 Yang Zheyu

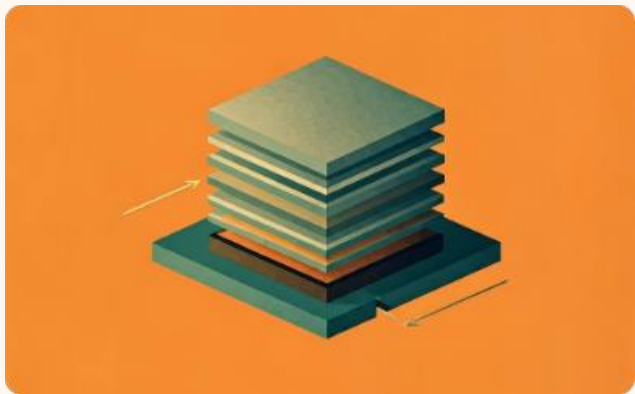
DC228002 Gu Shunshun



# Project Overview



Data Processing and Augmentation



Model Architecture



Training and Evaluation



Performance Metrics and Results

# Data Processing and Augmentation

1

## Custom Augmentations

Implemented rotation, random cropping, and aspect ratio changes using PyTorch and Albumentations. Enhanced dataset diversity.(Only for training dataset)

2

## Normalization

Standardized pixel values to improve model convergence. Ensured consistent input scaling across the dataset.

3

## Tensor Conversion & Other methods

Converted processed images to PyTorch tensors. Optimized for efficient GPU computation during training.

Gaussian blur, random occlusion, and random brightness



# Model Architecture

## Convolutional Layers

- **Block 1:** Conv(3→8) → GroupNorm → ReLU → Conv(8→16) → GroupNorm → ReLU → MaxPool → Dropout
- **Block 2:** Conv(16→32) → GroupNorm → ReLU → Conv(32→32) → GroupNorm → ReLU → MaxPool → Dropout
- **Block 3:** Conv(32→32) → GroupNorm → ReLU → Conv(32→32) → GroupNorm → ReLU → MaxPool → Dropout

## Fully Connected Layers

- **Layer 1:** Flatten → Linear(512→256) → GroupNorm → ReLU → Dropout
- **Output Layer:** Linear(256→10)

## Optimization

- **Loss:** CrossEntropy
- **Optimizer:** Adam (L2 Regularization, weight\_decay=1e-4)



# Training and Evaluation

1

## Training Loop

Implemented forward pass, loss calculation, and backpropagation. Optimized model parameters using stochastic gradient descent.

2

## Accuracy Tracking

Monitored model performance on validation set. Implemented early stopping to prevent overfitting.

3

## ROC, AUC and Confusion Matrix

Calculated Receiver Operating Characteristic curves and Area Under Curve. Evaluated model's discrimination ability across all classes. Used confusion matrices to evaluate classification performance by analyzing correct and incorrect predictions per class.

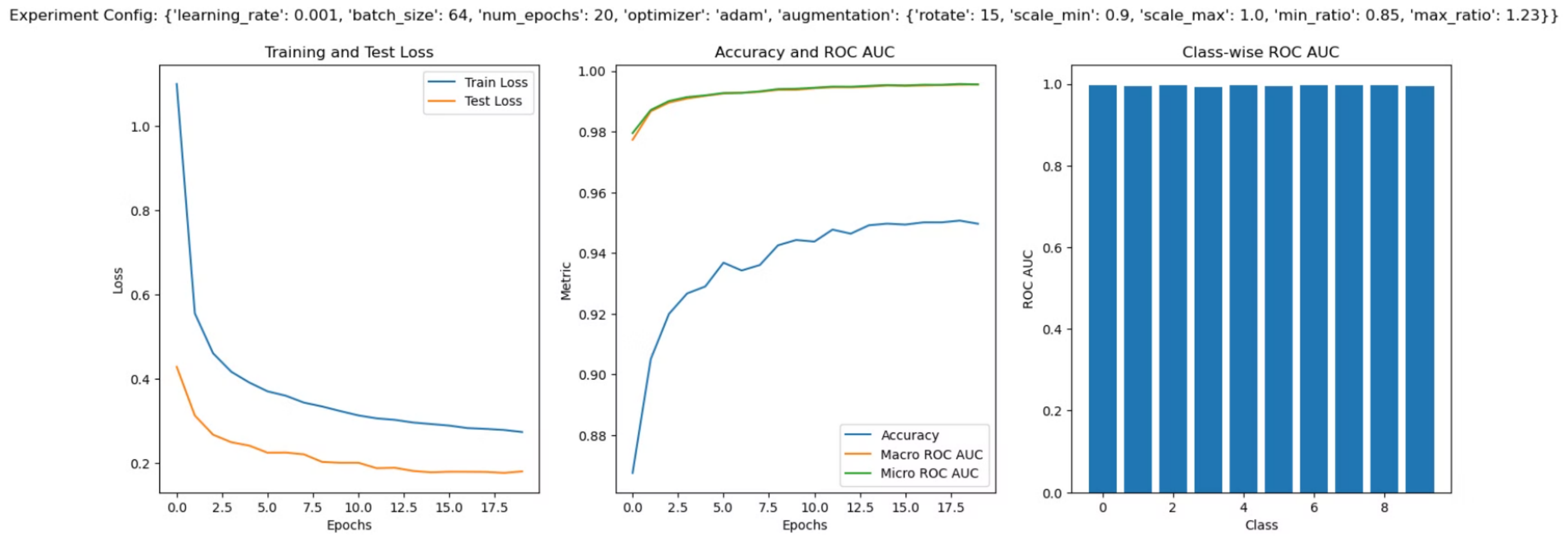
```
Epoch 3/10, Train Loss: 1.8074, Test Loss: 1.4028, Accuracy: 0.5216, Macro ROC AUC: 0.8583, Micro ROC AUC: 0.8771
Confusion Matrix: [[ 276  189  735   75  142   54    8  243    0   22]
 [ 149 3446  755   15  234   54   22  413    1   10]
 [  39  178 3157   94  266  151   29  191    1   43]
 [  67  112  605  692  251  665   61  161    0  268]
 [  52  171  234   23 1792   80   19   16    0  136]
 [   7   17   69  120   90 1877   72   25    0  107]
 [  10   87  165  171  147  924  397   48    1   27]
 [  37  209  116   15   11  157    9 1459    0    6]
 [  48   61  321  359  114  540   63   53    1  100]
 [  30   49  195  199  170  396   12   62    0  482]]
```



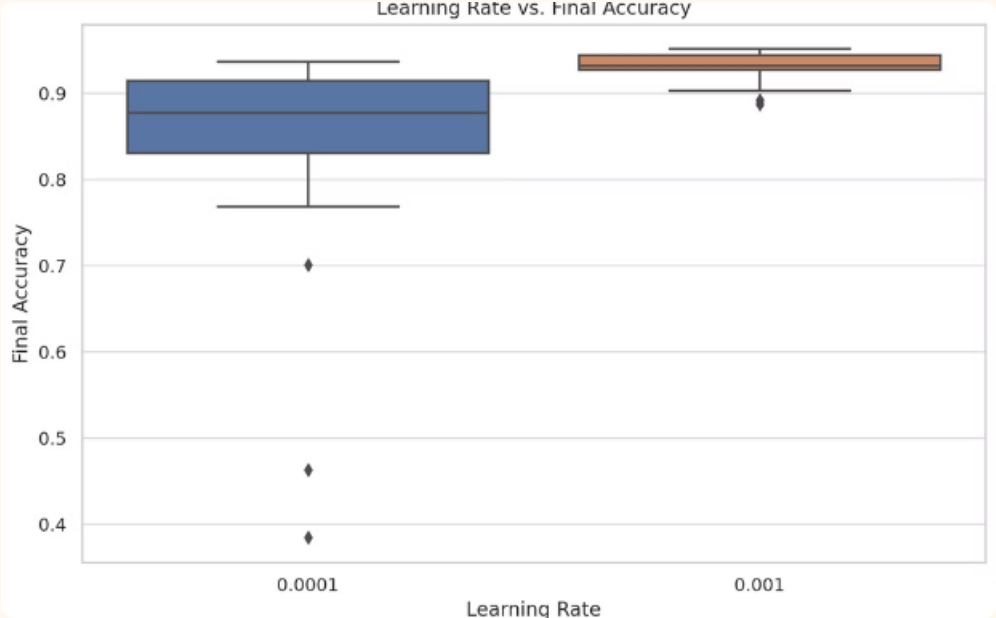
# Hyperparameter Tuning and Analysis

| Learning Rate | Batch Size | Epochs | Optimizer | Augmentations      |
|---------------|------------|--------|-----------|--------------------|
| 0.0001 - 0.01 | 32 - 64    | 5 - 20 | Adam, SGD | Rotate/Scale/Ratio |

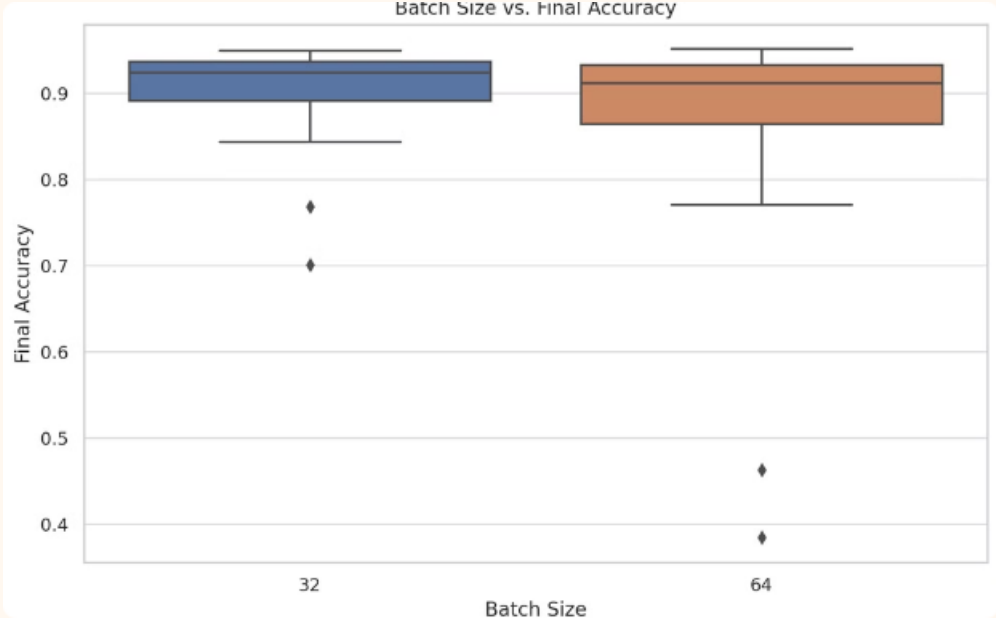
Experimented with various hyperparameters to optimize model performance. Analyzed impact on training speed, convergence, and generalization.



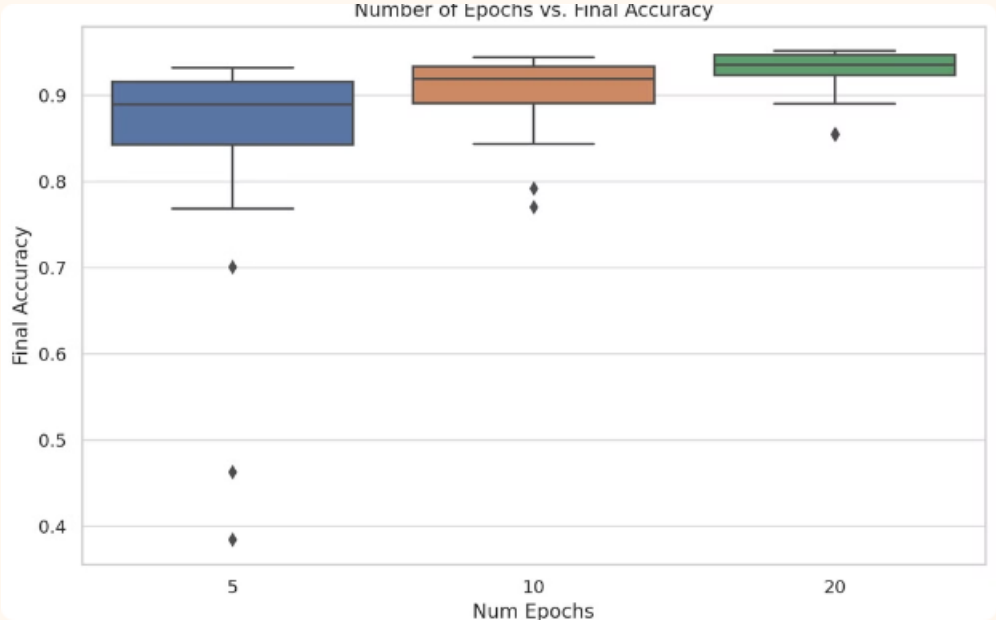
# Hyperparameter Tuning and Analysis



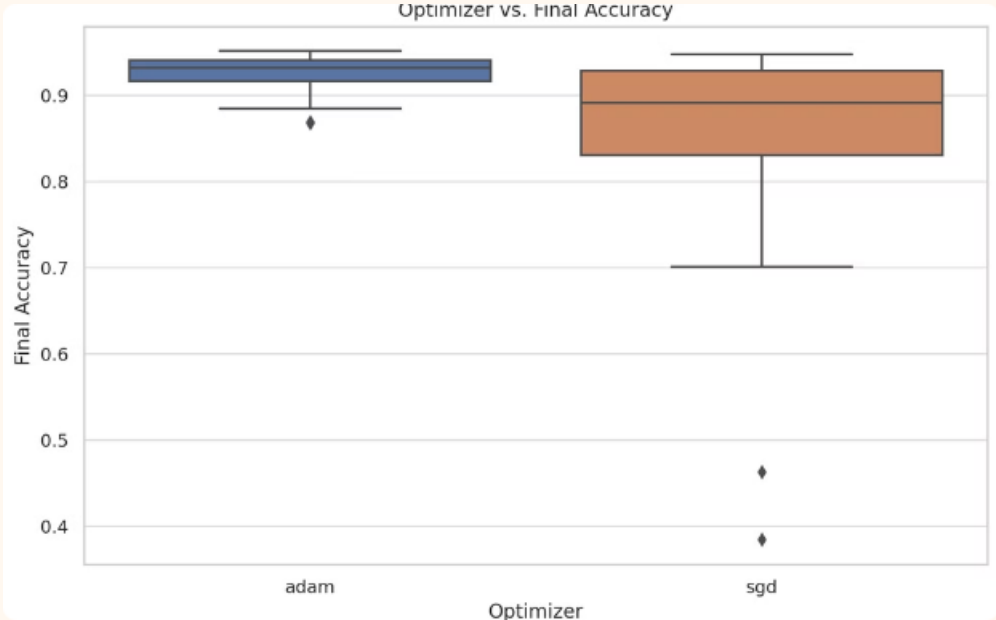
Learning Rate: 0.001



Batch Size: 64



Epochs: 20



Optimizer: Adam

# Conclusion and Future Work

## 1 Optimal Results

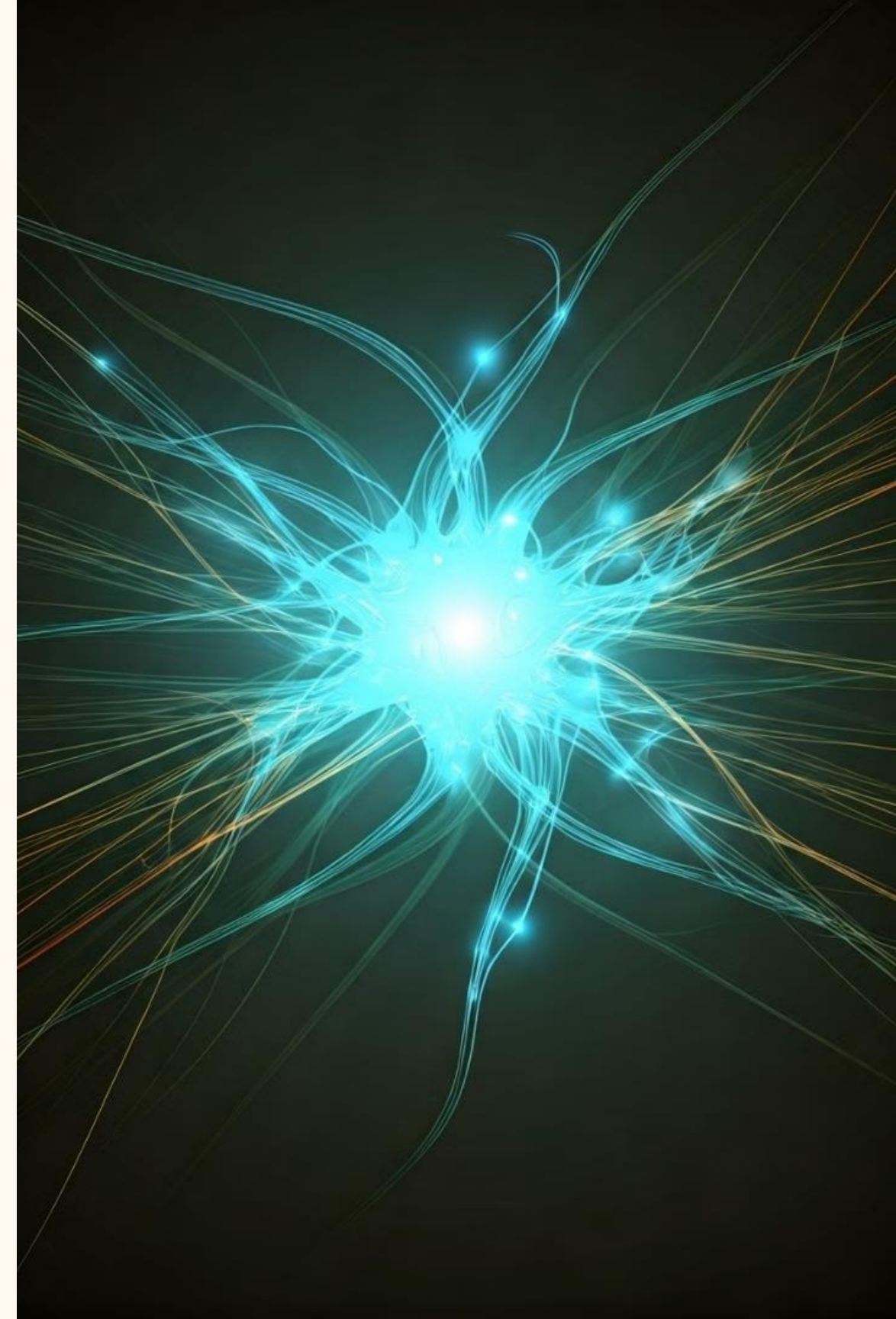
Achieved an accuracy of % with a combination of [learning rate 0.001, batch size 128, and 30 epochs].

## 2 Challenges

Balancing model complexity and training time. Addressing overfitting issues during training.

## 3 Future Improvements

Explore transfer learning from pretrained models. Implement attention mechanisms for better feature focus.





# Interactive Digit Recognition Using Gradio

Handwriting Recogniti...

