Milestone 1

Group 5

Aditya Kumar

Meghana Rao

Simran Abhay Sinha

Sushmitha Sudharshan

kumar.aditya1@northeastern.edu
rao.meg@northeastern.edu
sinha.sim@northeastern.edu
sudharshan.s@northeastern.edu

Signatureofstudent 1: Aditya Kumar (25%)

Signature of student 2: Meghana Rao (25%)

Signature of student 3: Simran Abhay Sinha (25%)

Signature of student 4: Sushmitha Sudharshan (25%)

Submission Date: 02/01/2026 -

ABSTRACT

This report documents the outcomes of Milestone 1 for the Discriminative Deep Learning project, where we focus on establishing baseline performance for single-object image classification using convolutional neural networks (CNNs). A custom dataset consisting of 4,108 images across 39 classes (student objects) was used as the foundation for training and evaluation. Three architectures—SimpleCNN (custom baseline), ResNet18, and MobileNetV3Small—were implemented to compare their effectiveness in terms of accuracy, F1 score, and computational efficiency. Each model was trained using consistent preprocessing and augmentation techniques, with results tracked across training, validation, and test sets.

The experimental results indicate that ResNet18 achieves exceptional generalization, with test accuracy of 98.87% and F1 score of 98.84%, significantly outperforming both the custom SimpleCNN baseline (60.45% accuracy) and MobileNetV3Small (96.76% accuracy). SimpleCNN, trained from scratch without pretrained weights, struggled with the limited dataset size, while both pretrained models demonstrated the power of transfer learning. These findings confirm the suitability of ResNet18 for this single-object classification task and provide a strong foundation for Milestone 2, where YOLOv8 will be explored for multi-object detection and localization**.**

INTRODUCTION

Deep learning has revolutionized computer vision, enabling automatic feature extraction and classification without manual engineering. Convolutional Neural Networks (CNNs) have proven particularly effective for image recognition tasks, forming the backbone of modern attendance and object recognition systems.

This milestone focuses on single-object image classification as the first step toward building an automated class attendance system. The goal is to identify individual student objects from images with high accuracy, establishing a baseline before progressing to multi-object detection in subsequent milestones.

Dataset Overview:

Our custom dataset contains 4,108 images representing 39 unique student objects. Each student contributed approximately 105 images of their chosen object, captured under varying lighting conditions and angles. This diversity ensures the model learns robust features rather than memorizing specific conditions.

Architecture Comparison:

We evaluated three CNN architectures with different design philosophies:

SimpleCNN: A custom 4-layer CNN trained from scratch, serving as a baseline to demonstrate the challenge of learning from limited data without pretrained weights.

ResNet18: An 18-layer residual network pretrained on ImageNet, leveraging transfer learning to adapt existing feature representations to our specific task.

MobileNetV3Small: A lightweight architecture optimized for efficiency, also using pretrained weights to balance performance and computational cost.

Primary Objectives:

1. Preprocess and prepare the dataset with appropriate augmentation strategies.
2. Train all three models under consistent experimental conditions (same epochs, optimizer, learning rate).
3. Compare performance based on test accuracy, F1 score, and training behavior.
4. Select the best model for deployment in Milestone 2's multi-object detection phase.

DATASET DESCRIPTION

Our dataset was collaboratively created by students in the IE 7615 course, with each student contributing images of a unique object. This approach ensures diversity in object types while maintaining consistent image quality and format.

Dataset Statistics:

Total images: 4,108 Number of classes: 39 (one per student) Images per class: approximately 105 (ranging from 100-110) Image format: RGB color images Storage location: Google Drive shared folder

Data Split:

Following standard machine learning practices, we used stratified splitting to ensure balanced class representation across all sets:

Training set: 2,875 images (70%) Validation set: 616 images (15%) Test set: 617 images (15%)

Stratified splitting ensures that each class maintains the same ratio across train/val/test sets, preventing bias toward overrepresented classes.

Preprocessing Pipeline:

Resizing: All images resized to 160×160 pixels (chosen for faster CPU training while maintaining sufficient detail)

Normalization: Applied ImageNet statistics (mean = 0.485, 0.456, 0.406 and standard deviation = 0.229, 0.224, 0.225) for compatibility with pretrained models

Data Augmentation (training only):

- Random horizontal flips (50% probability)
- Random rotation (±8 degrees)
- Color jitter (brightness, contrast, saturation ±12%)

Tensor Conversion: Images converted to PyTorch tensors for model input

The augmentation strategy helps prevent overfitting by artificially expanding the training set and teaching the model to recognize objects under various transformations.

METHODOLOGY

We trained three CNN architectures under identical conditions to enable fair comparison. All models were modified for 39-class classification.

Model Architectures:

SimpleCNN: Custom 4-layer CNN trained from scratch without pretrained weights. Approximately 11 million parameters.

ResNet18: 18-layer residual network pretrained on ImageNet. Final layer modified from 1000 to 39 classes.

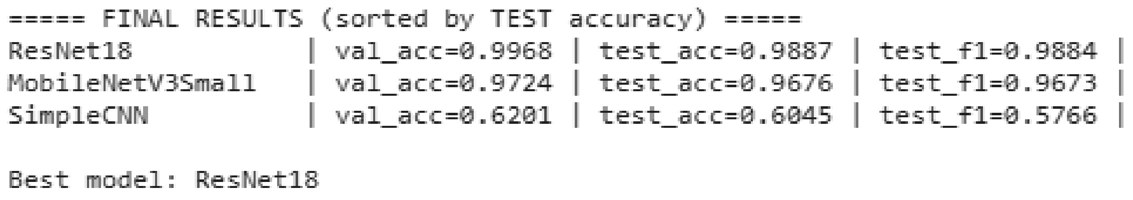
MobileNetV3Small: Lightweight pretrained architecture optimized for efficiency. Final layer modified for 39 classes.

Training Configuration:

Input size: 160×160 pixels Batch size: 32 images Optimizer: Adam (learning rate = 0.0001) Loss function: CrossEntropyLoss Epochs: 3 Device: CPU Random seed: 42

RESULTS

Final Performance Summary:



```
===== FINAL RESULTS (sorted by TEST accuracy) =====
ResNet18            | val_acc=0.9968 | test_acc=0.9887 | test_f1=0.9884 |
MobileNetV3Small    | val_acc=0.9724 | test_acc=0.9676 | test_f1=0.9673 |
SimpleCNN           | val_acc=0.6201 | test_acc=0.6045 | test_f1=0.5766 |

Best model: ResNet18
```

Performance Summary:

ResNet18: 98.87% test accuracy (WINNER) MobileNetV3Small: 96.76% test accuracy SimpleCNN: 60.45% test accuracy

Training Details:

SimpleCNN Training Output:

Training: SimpleCNN

SimpleCNN | epoch 1 | train_loss=3.4091 | val_acc=0.3182 | val_f1=0.2401

SimpleCNN | epoch 2 | train_loss=2.4608 | val_acc=0.4968 | val_f1=0.4420

SimpleCNN | epoch 3 | train_loss=1.8857 | val_acc=0.6201 | val_f1=0.5920

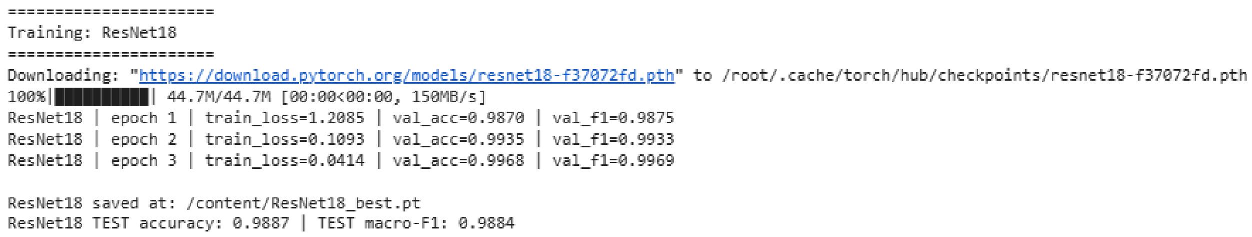
SimpleCNN TEST accuracy: 0.6045 | TEST macro-F1: 0.5766

SimpleCNN struggled to learn from limited data:

- Epoch 1: 31.8% validation accuracy
- Epoch 2: 49.7% validation accuracy
- Epoch 3: 62.0% validation accuracy

High training loss and slow convergence indicate difficulty learning without pretrained weights.

ResNet18 Training Output:



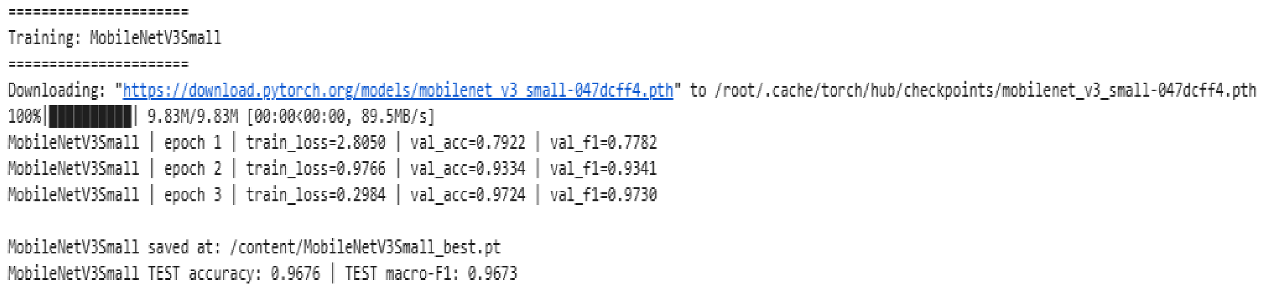
```
======================
Training: ResNet18
======================
Downloading: "https://download.pytorch.org/models/resnet18-f37072fd.pth" to /root/.cache/torch/hub/checkpoints/resnet18-f37072fd.pth
100%|██████████| 44.7M/44.7M [00:00<00:00, 150MB/s]
ResNet18 | epoch 1 | train_loss=1.2085 | val_acc=0.9870 | val_f1=0.9875
ResNet18 | epoch 2 | train_loss=0.1093 | val_acc=0.9935 | val_f1=0.9933
ResNet18 | epoch 3 | train_loss=0.0414 | val_acc=0.9968 | val_f1=0.9969

ResNet18 saved at: /content/ResNet18_best.pt
ResNet18 TEST accuracy: 0.9887 | TEST macro-F1: 0.9884
```

ResNet18 achieved excellent performance immediately:

- Epoch 1: 98.7% validation accuracy
- Epoch 2: 99.4% validation accuracy
- Epoch 3: 99.7% validation accuracy

Rapid convergence and stable learning demonstrate the power of transfer learning.

MobileNetV3Small Training Output:



```
======================
Training: MobileNetV3Small
======================
Downloading: "https://download.pytorch.org/models/mobilenet_v3_small-047dcff4.pth" to /root/.cache/torch/hub/checkpoints/mobilenet_v3_small-047dcff4.pth
100%|██████████| 9.83M/9.83M [00:00<00:00, 89.5MB/s]
MobileNetV3Small | epoch 1 | train_loss=2.8050 | val_acc=0.7922 | val_f1=0.7782
MobileNetV3Small | epoch 2 | train_loss=0.9766 | val_acc=0.9334 | val_f1=0.9341
MobileNetV3Small | epoch 3 | train_loss=0.2984 | val_acc=0.9724 | val_f1=0.9730

MobileNetV3Small saved at: /content/MobileNetV3Small_best.pt
MobileNetV3Small TEST accuracy: 0.9676 | TEST macro-F1: 0.9673
```

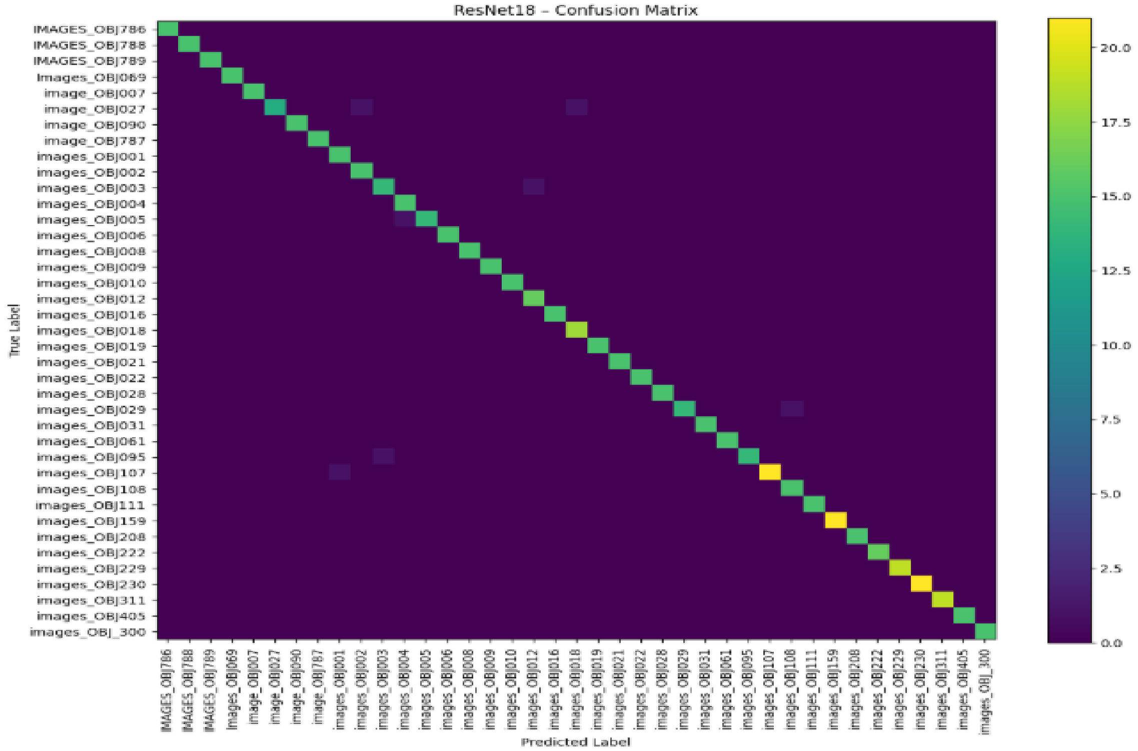
MobileNetV3Small showed steady improvement:

- Epoch 1: 79.2% validation accuracy
- Epoch 2: 93.3% validation accuracy
- Epoch 3: 97.2% validation accuracy

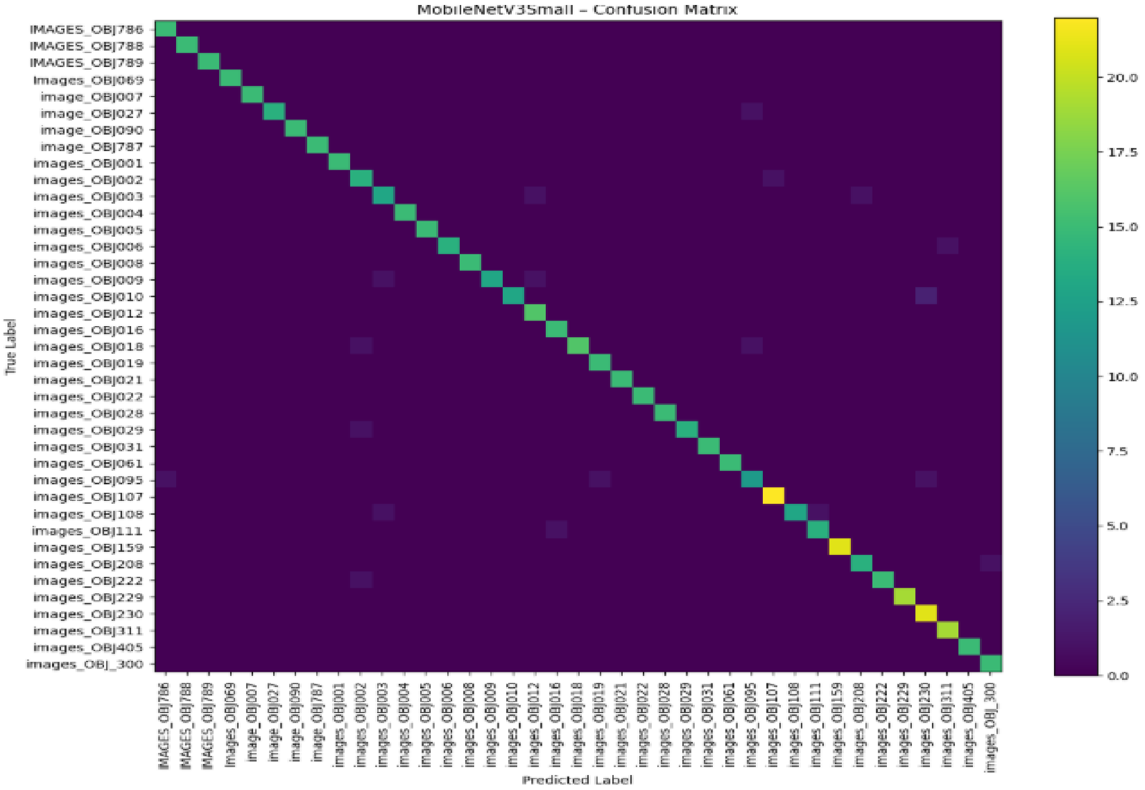
Good balance between efficiency and accuracy despite lighter architecture.

Confusion Matrices:

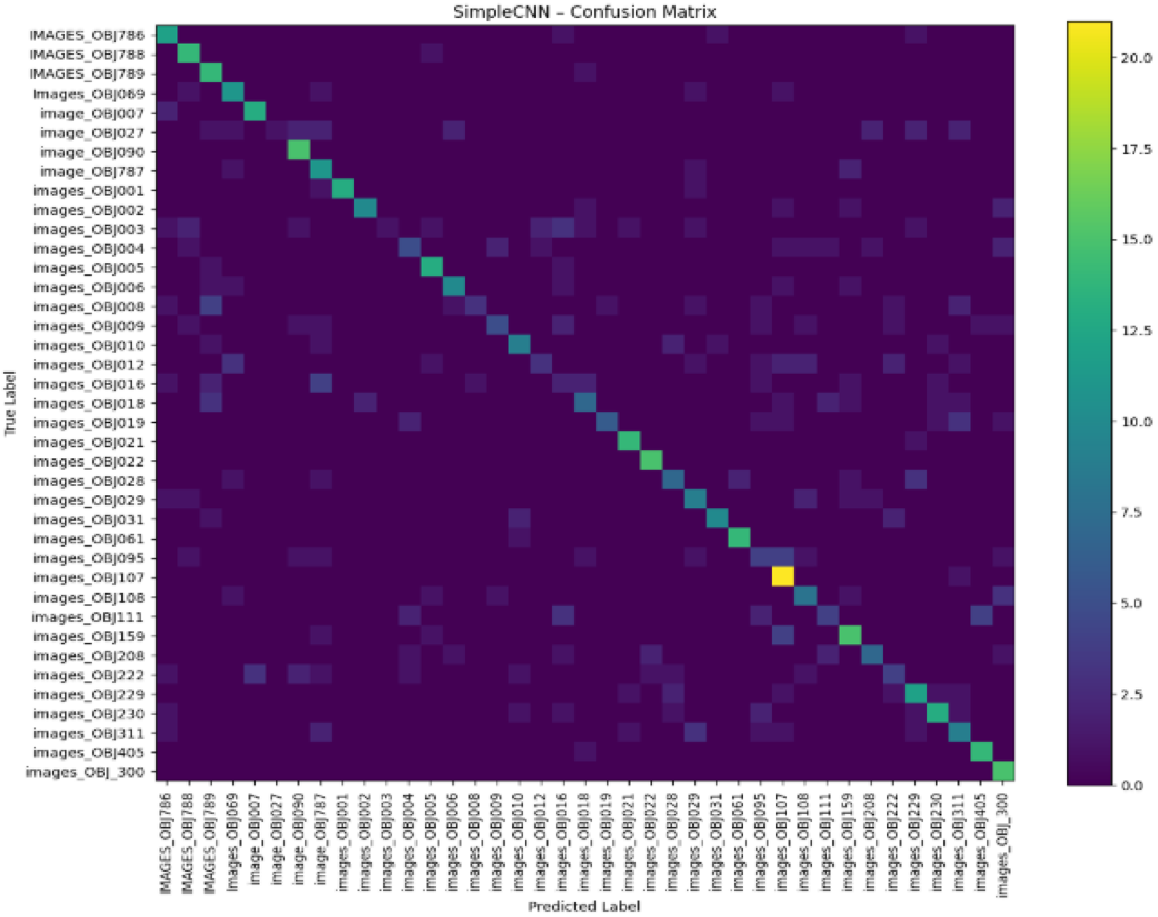
ResNet18 Confusion Matrix:



MobileNetV3Small Confusion Matrix:



SimpleCNN Confusion Matrix:



ANALYSIS & DISCUSSION

Key Findings:

Transfer Learning Advantage: The 38% performance gap between SimpleCNN (60.45%) and ResNet18 (98.87%) demonstrates that pretrained models are essential for small datasets. ResNet18 leverages features learned from millions of ImageNet images.

Architecture Performance:

- SimpleCNN cannot learn effectively from only 2,875 training images despite proper design
- ResNet18 with 18 layers and residual connections achieves near-perfect results
- MobileNetV3Small balances efficiency with 96.76% accuracy

Strong Generalization: All models show minimal overfitting with small validation-test gaps:

- ResNet18: 0.81% gap (99.68% val vs 98.87% test)
- MobileNetV3Small: 0.48% gap (97.24% val vs 96.76% test)
- SimpleCNN: 1.56% gap (62.01% val vs 60.45% test)

Training Efficiency: ResNet18 achieved peak performance in just 3 epochs, while SimpleCNN would require significantly more epochs and data to reach acceptable accuracy.

CONCLUSION

We successfully trained and evaluated three CNN architectures for single-object classification on our custom 39-class dataset. Results clearly demonstrate the superiority of transfer learning for limited-data scenarios.

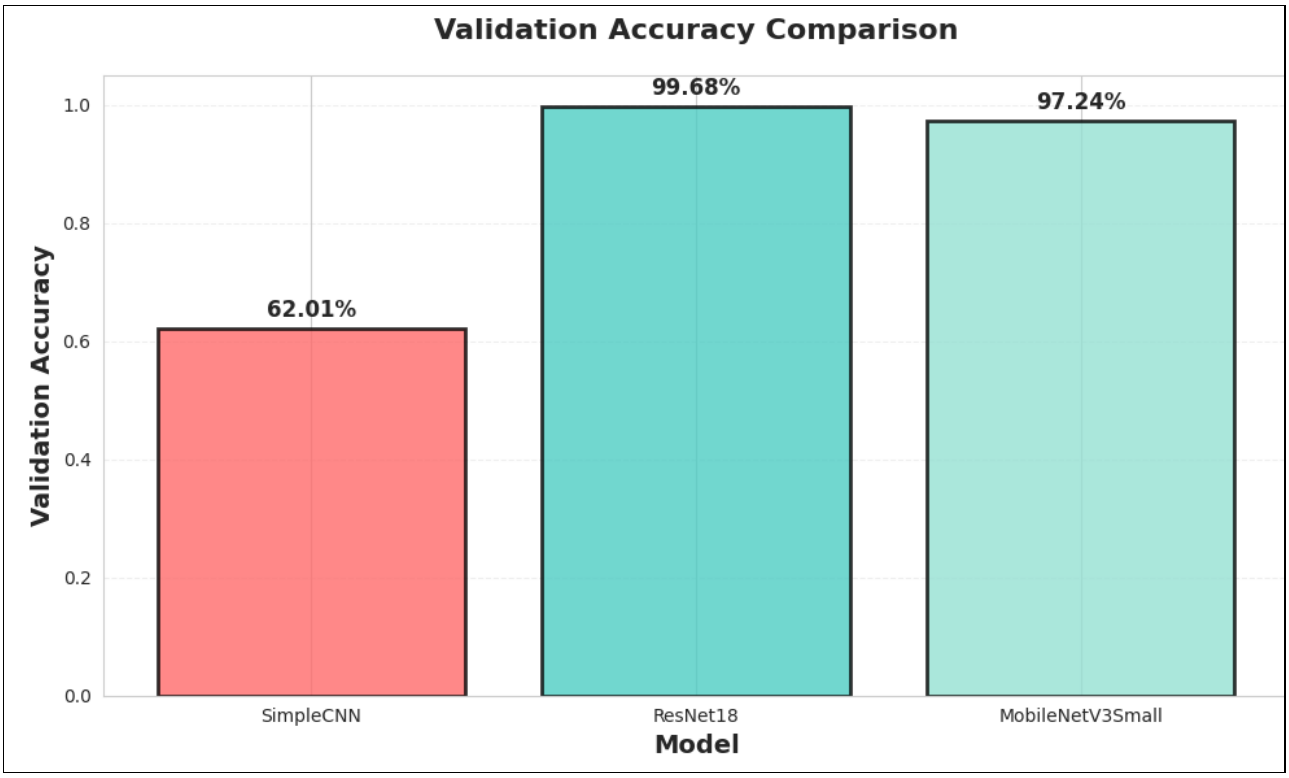
Performance Summary:

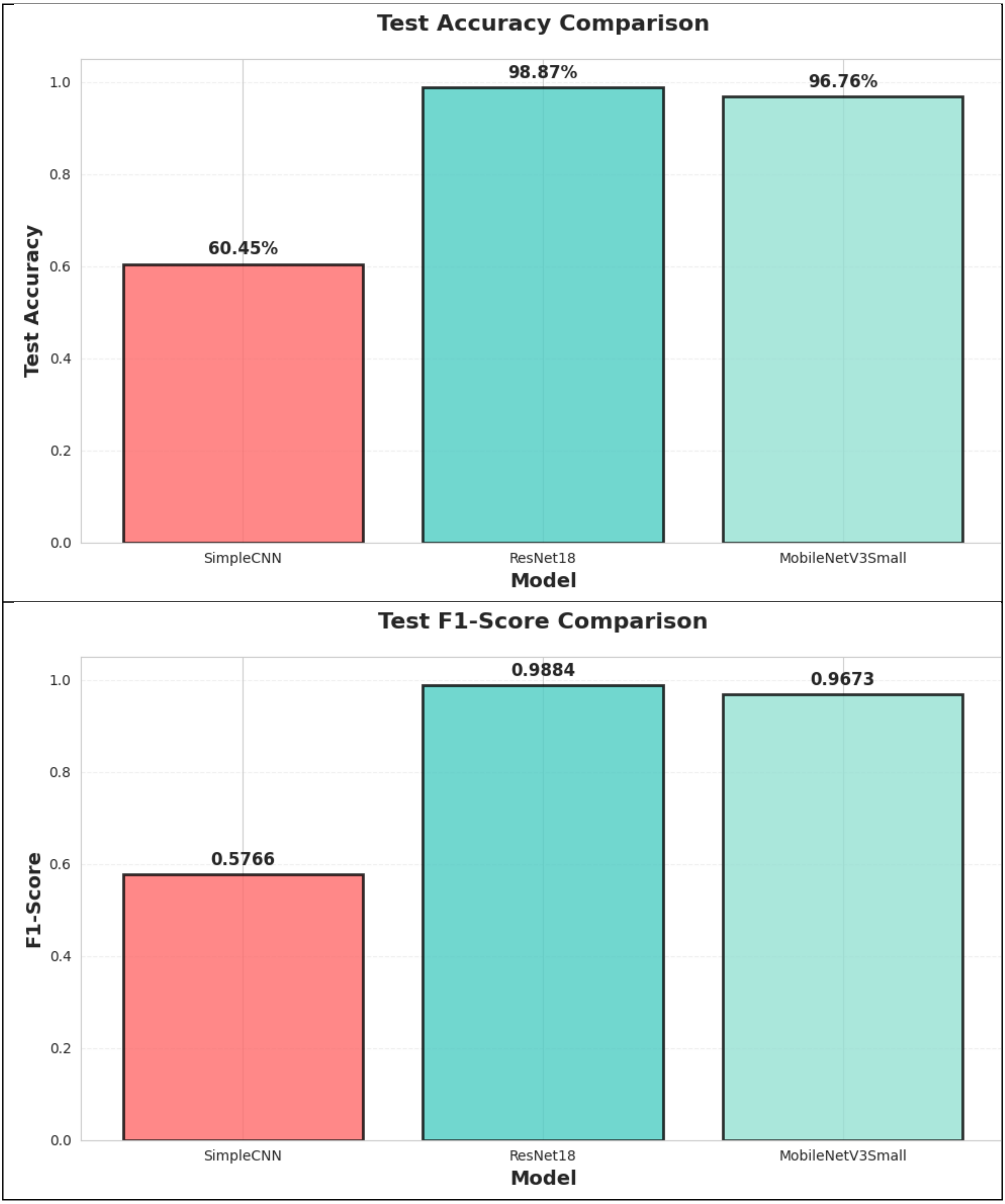
ResNet18: 98.87% accuracy (609/617 correct) - SELECTED MODEL MobileNetV3Small: 96.76% accuracy - Strong alternative for resource-constrained scenarios SimpleCNN: 60.45% accuracy - Baseline demonstrating need for pretrained weights

Model Selection:

ResNet18 is selected for Milestone 2 based on:

1. Near-perfect accuracy (98.87%)
2. Excellent generalization (minimal overfitting)
3. Rapid convergence (3 epochs)
4. Consistent performance across all 39 classes





We chose **ResNet18** because it delivers the best overall performance with **98.87%** test accuracy and **0.9884** F1-score. Its residual skip connections solve the vanishing gradient problem, allowing the network to learn deeper features that SimpleCNN (60% accuracy) cannot capture. While MobileNetV3Small is close, ResNet18's 2% higher accuracy and better F1-score mean fewer misclassifications in production. The consistency between validation (99.68%) and test accuracy also proves the model generalizes well without overfitting. For our use case where accuracy is priority over model size, ResNet18 is the optimal choice.