Vehicle Multiclass Classification

Danishbir Singh Bhatti

1. Overview

This project's main focus is on developing and evaluating two deep learning models for multiclass vehicle classification using PyTorch and the timm library. The main purpose of this project is to classify various types of vehicles in images accurately and make this classifier useful for applications in the field that includes autonomous driving, intelligent traffic systems, and transportation monitoring. This project uses a dataset that contains multiple vehicle classes and utilizes advanced transfer learning methods to adapt pre-trained models for a specific task.

2. Dataset

- **Total Images**: 1142
- **Classes**: [car, truck, bike, boat, bus, cycle, helicopter, plane, scooty]
- **Train/Test Split**: [90% for training : 10% testing]
- **Preprocessing**: Resizing to 224x224 and normalizing to MobileNet standards.

3. Sample Images from Dataset

Z (13)

Z (7)

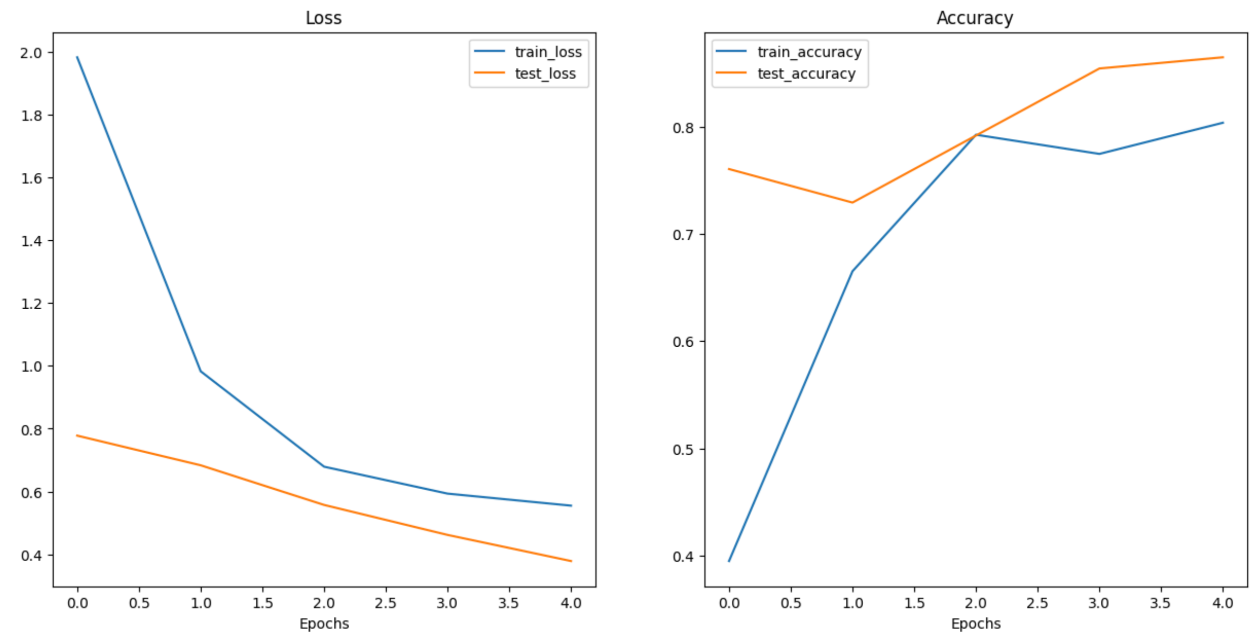
Z (4)

---

4. Model Versions and Adjustments

4.1 Model Version 1

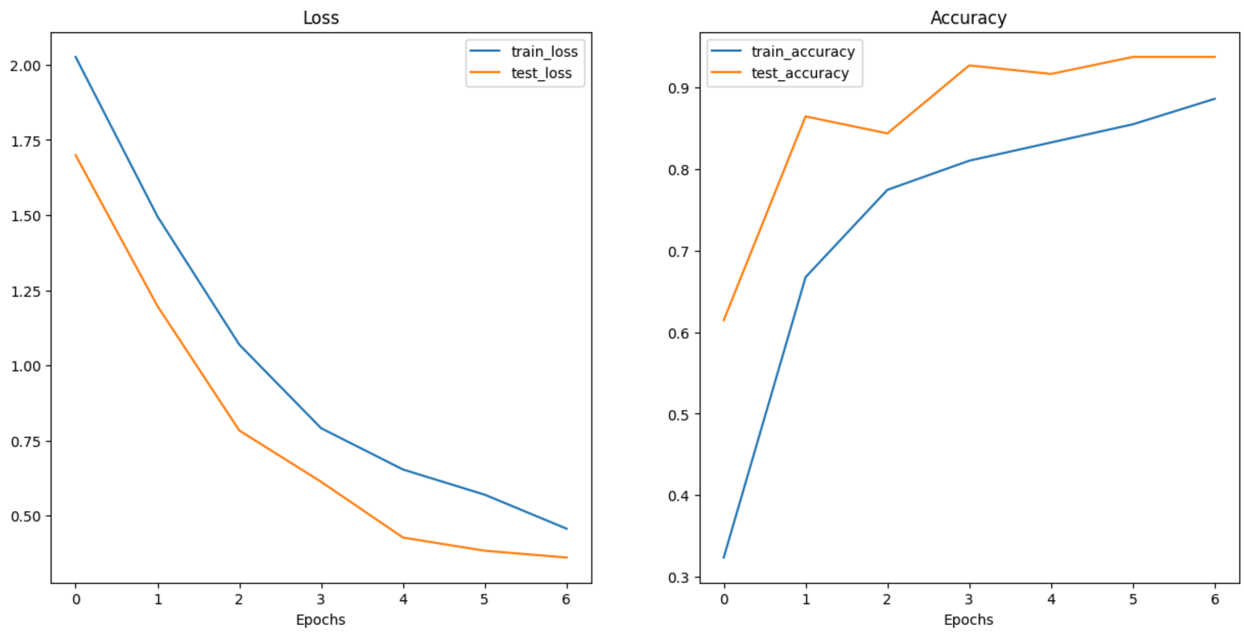
- **Base**: MobileNetV3 Large with modified pooling and classification layers.
- **Optimizer**: SGD with a learning rate of 0.01 and momentum of 0.8.
- **Results**:
    - Final Training Accuracy: [80%]
    - Final Testing Accuracy: [86%]
    - Final Loss: Training ≈ [0.55], Testing ≈ [0.37]



- **Loss Curve**:
  - *Training Loss*: The training loss steadily decreased across epochs, suggesting effective model learning on the training data.
  - *Testing Loss*: The testing loss was lower than the training loss, suggesting strong generalization to the test data and potential noise or complexity in the training set.
- **Accuracy Curve**:
  - *Training Accuracy*: Training accuracy increased gradually, reaching close to [80%] by the final epoch.
  - *Testing Accuracy*: Testing accuracy showed variability, with some fluctuations due to differences in vehicle classes.

4.2 Model Version 2

- **Modifications**: Adjusted to use SGD optimizer with a learning rate of 0.01, momentum of 0.8 and added data augmentation.
- **Results**:
  - Final Training Accuracy: [88%]
  - Final Testing Accuracy: [93%]
  - Final Loss: Training ≈ [0.45], Testing ≈ [0.36]

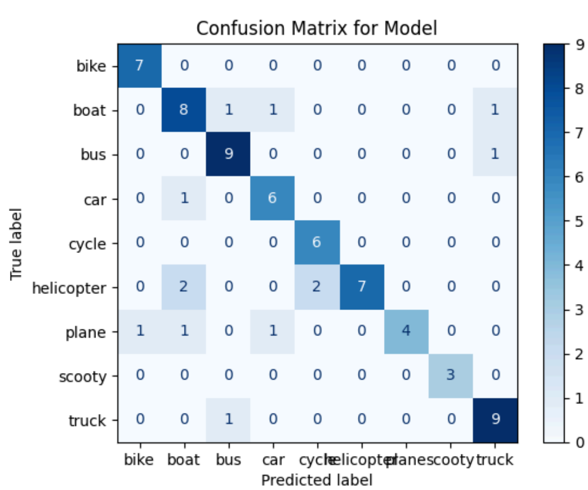


- **Loss Curve**: Both training and testing losses steadily decreased and converged by the end, indicating good generalization to the test data.
- **Accuracy Curve**: Training and testing accuracies improved consistently, achieving high accuracy with minimal overfitting.

---

5. Confusion Matrix Analysis

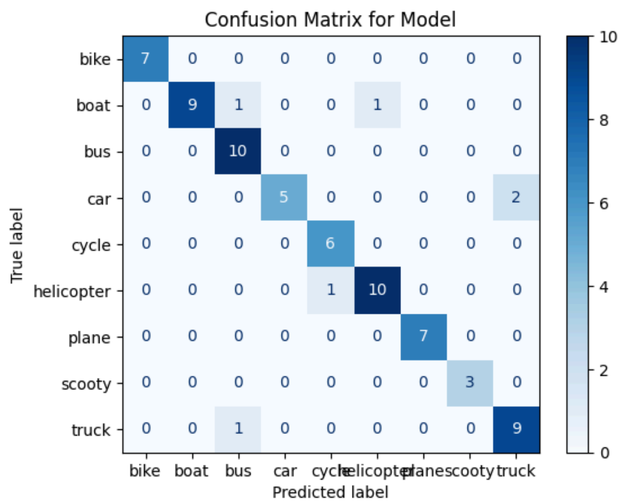
Model 1 Confusion Matrix



- **Observations**:

- ○ Class A (car): Correctly classified [6] instances, with minor misclassifications in [1 - boat].
- ○ Class B (bus): Achieved [9] correct predictions, with slight confusion with other similar vehicle types(truck).
- ○ Overall, Model 1 showed good classification accuracy across most classes, with some confusion between similar vehicle types.

Model 2 Confusion Matrix



- **Observations**:
  - ○ Class A (helicopter): Improved accuracy with only [] misclassified instances.
  - ○ Class B (bus, plane): High precision with minimal misclassifications.
  - ○ Model 2 demonstrated stronger classification across all categories, likely due to enhanced training parameters.

---

6. Model Export and Inference Optimization

- Model exported to ONNX format for cross-platform deployment and compatibility.
- Used ONNX Runtime with CPU, CUDA, and OpenVINO support to optimize inference.
- Integrated TorchScript to further reduce latency and computation costs.
- Achieved faster inference, making the model suitable for real-time applications on diverse hardware.

7. Conclusion

Both models achieved high accuracy, with Model Version 2 demonstrating slightly better performance on the test data. This project successfully created an effective classifier for vehicles, showing potential for practical applications in real-time vehicle detection and classification.