Vehicle Classification Project Report

---

1. Abstract

This project focuses on building an image classification model to identify various types of vehicles using deep learning. A pretrained ResNet18 model was fine-tuned on a custom dataset containing 12 vehicle classes. The model achieved 81% validation accuracy and was exported to ONNX format for deployment and cross-platform compatibility. Evaluation metrics and visual verification confirm the model's robustness and potential.

---

2. Data Analysis and Cleaning

The dataset consisted of labelled images distributed across 12 classes: auto-rickshaw, bicycle, bus, car, e-rickshaw, mini-bus, mini-truck, motorcycle, rickshaw, tractor, truck, and van.

Tools like CleanVision and Fastdup were attempted for automatic data quality checks, but due to environment limitations, a custom script was used to detect and remove:

- Corrupted images
- Images with zero dimensions
- Unreadable formats

---

3. Data Preprocessing

All images were resized to **224×224 pixels**.

- **Training Transforms**: Random horizontal flip, random rotation, color jittering, normalization with mean = [0.5, 0.5, 0.5] and std = [0.5, 0.5, 0.5].
- **Validation Transforms**: Only resizing and normalization.

These transforms ensured robustness to position, lighting, and contrast variance.

---

4. Model Architecture

The model used is **ResNet18** pretrained on ImageNet. Its final layer was replaced to output predictions over 12 classes.

Reasons for choosing ResNet18:

- Lightweight and fast
- Proven performance for image classification
- Easy to fine-tune on limited compute (GTX 1650 GPU)

A Softmax layer was added during ONNX export for interpretable probability outputs.

---

5. Training and Experimentation

- **Loss Function**: CrossEntropyLoss
- **Optimizer**: Adam
- **Learning Rate**: 0.001
- **Epochs**: 10
- **Batch Size**: 32
- **Hardware**: Trained on GPU (GTX 1650 Mobile)

The model achieved a validation accuracy of **81%**. The loss and validation curves indicated proper convergence. The final PyTorch model was saved as vehicle_classifier.pth.

---

6. Results and Key Findings

- **Validation Accuracy**: 81%
- Evaluation metrics (Precision, Recall, F1-score) were calculated using sklearn.
- A confusion matrix showed good separability between most classes.
- Some confusion was seen between **van** and **motorcycle**, likely due to similar shape/lighting or class imbalance.

ONNX model inference on test images predicted:

- car.png → car (0.95)
- autorickshaw.png → auto-rickshaw (0.89)
- van.png → incorrectly predicted as motorcycle (0.58)

Despite one misclassification, the results were consistent with the overall performance.

---

7. Future Work

Given more time and compute, the following enhancements are suggested:

- Experiment with deeper models like **ResNet50**, **EfficientNet**
- Balance the dataset using oversampling or class weighting
- Fine-tune learning rates with schedulers
- Use label smoothing or dropout to improve generalization
