ResNet-50 Architecture & Training Results

1. Methodology

We implemented the **ResNet-50** architecture using the PyTorch framework. The dataset was preprocessed into a 224 × 224resolution with ImageNet normalization stats. We conducted two distinct training experiments to determine the optimal strategy for fine-grained classification.

2. Experiment A: Fixed Feature Extractor (Baseline)

In this initial experiment, we utilized ResNet-50 purely as a feature extractor.

- **Configuration:** All convolutional layers (Layers 1–4) were **frozen** (weights locked). Only the final fully connected classification head was trainable.
- **Hypothesis:** The pre-trained ImageNet weights would be sufficient to distinguish between car models.

Results (Logs Summary)

- **Starting Loss:** ≈5.29 (Random guessing for 196 classes).
- **Ending Loss (Epoch 10):** ≈ 4.65.
- **Observation:** The model failed to converge significantly. The loss decreased very slowly, indicating that the generic features learned from ImageNet (which includes dogs, cats, and generic vehicles) were not specific enough to differentiate between similar car models (e.g., distinguishing a 2011 vs. 2012 version of the same sedan).

3. Experiment B: Fine-Tuning (Optimized Strategy)

In the second experiment, we applied **Fine-Tuning** to adapt the network to the specific domain of vehicles.

- **Configuration:** We **unfroze Layer 4** (the final convolutional block) while keeping early layers frozen. We reduced the learning rate to e^{-4} to prevent destroying pre-trained weights.
- **Hypothesis:** Allowing the deep layers to update would enable the model to learn specific vehicular features (headlights, grille shapes, wheel rims).

Results (Logs Summary)

The improvement was immediate and drastic.

Epoch	Loss	Training Accuracy	Status
1	4.92	3.99%	Initial learning
3	2.35	42.59%	Rapid convergence
5	0.71	83.93%	High performance
8	0.12	98.50%	Near convergence
10	**0.05**	**99.39%**	**Fully Trained**

4. Comparative Analysis & Conclusion

The difference between the two approaches highlights the difficulty of the Stanford Cars dataset. Because the inter-class similarity is high, generic features are insufficient.

- **Fixed Extractor:** Resulted in **underfitting**. The model could not learn the subtle differences between classes.
- **Fine-Tuning:** Resulted in **99.39% Training Accuracy**. This proves that the ResNet-50 architecture, when properly tuned, has sufficient capacity to perfectly memorize and classify the training data.