In the appendix, I documented the step-by-step codes I used in carrying out the research:

1. Imports Section

I imported all the necessary open-source libraries such as TensorFlow, PyTorch, Scikit-learn, NumPy, and OpenCV to support the model development and evaluation.

2. Dataset Loading

I loaded the CIFAR-10 dataset directly through TensorFlow's built-in function, which provided 60,000 images across 10 classes, including the animals used in my research.

3. Preprocessing

I resized and normalized the images, and applied augmentation techniques like rotation, flipping, and zooming to improve the robustness of the model.

4. Model Setup (MobileNetV2)

I used the MobileNetV2 pre-trained model from ImageNet, froze its base layers, and added my own custom classifier layers for the classification task.

5. Training & Fine-Tuning

I first trained the model in a feature extraction phase, and then fine-tuned the top layers with a smaller learning rate to improve accuracy. I optimized with the Adam optimizer and applied early stopping to avoid overfitting.

6. SVM Baseline

I built a baseline model using Support Vector Machine (SVM) with HOG features, implemented through Scikit-learn, to compare against the transfer learning approach.

7. Evaluation Metrics

I evaluated the models using accuracy, precision, recall, F1-score, and confusion matrices, implemented with Scikit-learn functions.

8. Robustness Testing

I tested the model on rotated, flipped, noisy, and adversarial images to assess its resilience under challenging conditions.

9. Computational Efficiency

I measured training time, inference speed, and memory usage to confirm that the model is efficient enough for deployment in resource-limited settings.