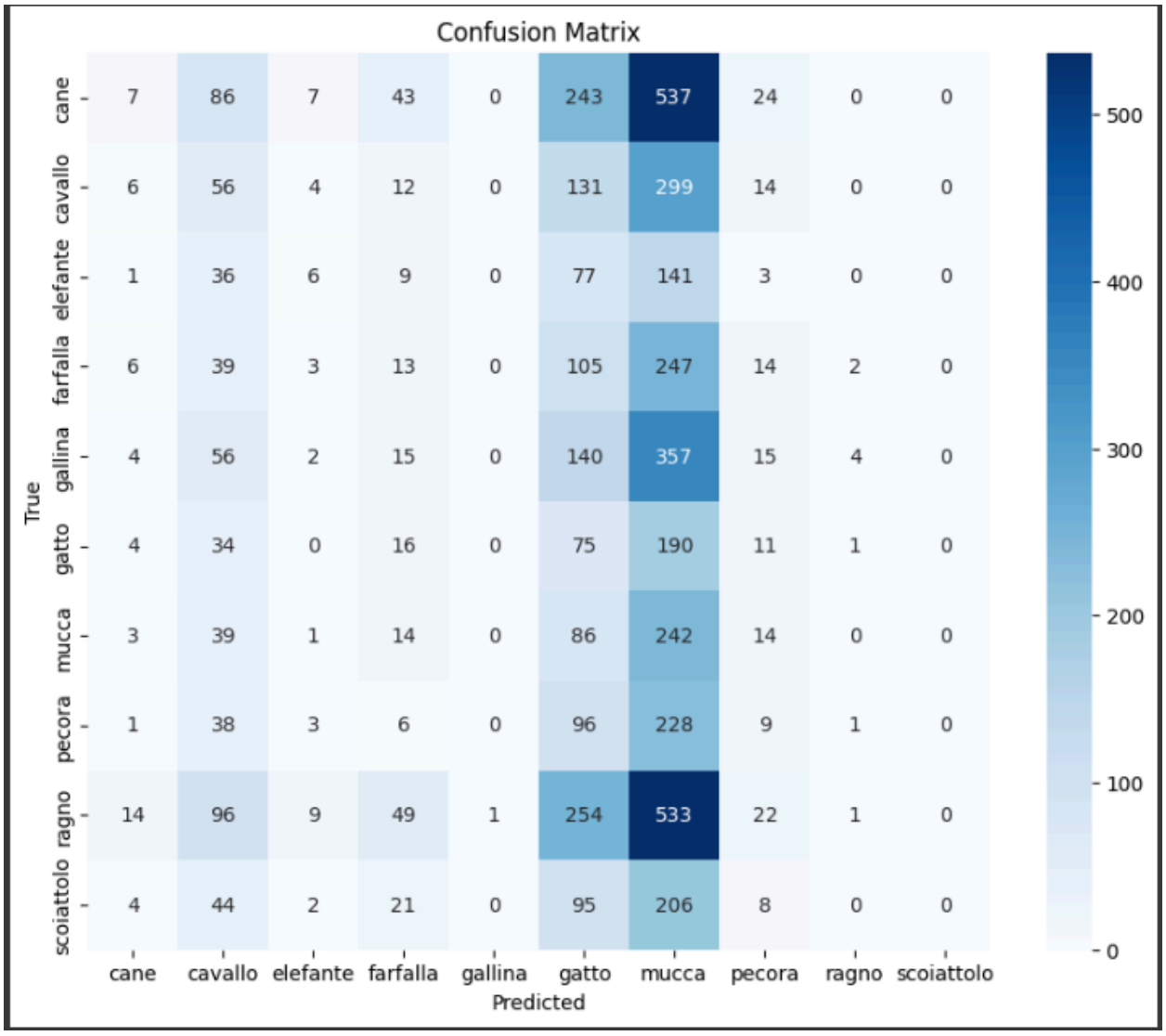
Deep Learning Assessment: Image Classification with CNN

Group 3: Abhi, Lovely, Priya

1. Approach

- Loaded Animals dataset using TensorFlow/Keras after extracting dataset from the zip file into the Colab environment.
- Used 20% validation split and fixed seed, applied image resizing and batching for memory efficiency, verified class labels, and optimized the input pipeline using shuffling and prefetching for improved performance.
- The first model was a custom CNN composed of three convolutional layers followed by max-poolling and dropout layers to prevent overfitting.
- Leveraged two powerful architectures for transfer learning, MobileNetV2 and ResNet50, each initialized with pre-trained ImageNet weights.
- Fine-tuned these models by freezing the base layers and retraining the top layers, with gradual unfreezing to further improve performance.
- Both models were compiled using the Adam optimizer and categorical cross-entropy loss function. We trained them for 10-15 epochs with early stopping based on validation loss.

Figure 1. Confusion Matrix of ResNet50 after fine tuning



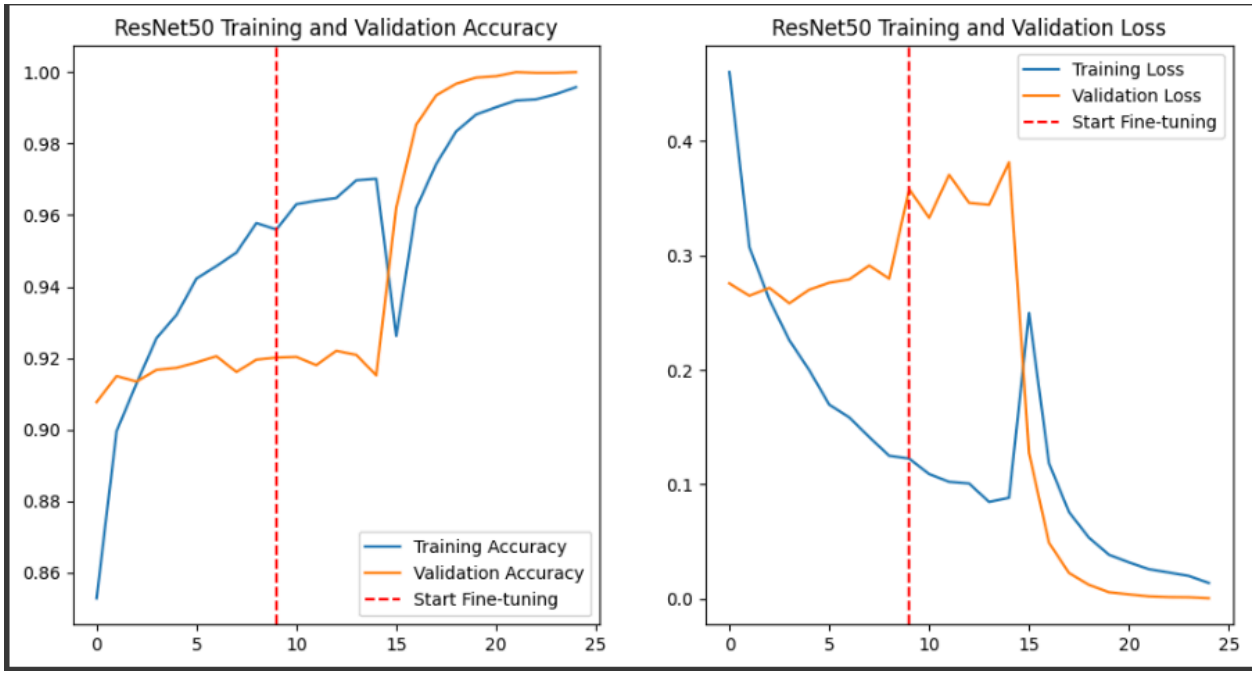
Results

- The fine tuned ResNet50 model demonstrated high performance on the animals dataset, achieving an accuracy of 98%, while the MobileNetV2 model performance at 93%. MobileNetV2 validation accuracy before fine tuning was 94%.
- Confusion matrix revealed misclassifications between dog and cat or butterfly or cow, might be due to visual similarities, insufficient

imbalance training samples or background noise in images. Squirrels are also poorly predicted (Figure 1).

- F1-score is low at macro avg ~0.05, meaning many classes are not well predicted.
- Accuracy might be misleading if a few classes dominate predictions like mucca and cane.

Figure 2. Training and Validation accuracy for the fined-tuned ResNet50 model.



2. Analysis

- Fine-tuning significantly improved both accuracy and loss on training and validation sets.
- Overall, the training was highly successful with a near 100% accuracy and very low loss (near zero).
- Additional experimentation with model architectures and hyperparameter tuning may enhance the model's performance on the more challenging classes.