

ANIMAL CLASSIFICATION REPORT

I. INTRODUCTION

1. Task review

This report describes our methods for enhancing the baseline convolutional neural network used for classifying images of **151 different animal species**. The dataset comprises **6,270 RGB images** spread across 151 folders, each representing a unique animal category. **Our primary aim was to boost the model's classification accuracy while also focusing on improving its computational efficiency**. At the end, the result shows ***the optimized model efficiency boost 21.75% compared to the baseline model with the accuracy is as 2.57 times as high as the old model.***

This report covers the *overview of the baseline model and model improvement methods* executed during data training along with experiments of different models combined with result summary. The last two parts mention the *rationale behind the chosen model with its trades-offs* as well as the *limitations and conclusions* implied from the project.

2. Baseline model review

The project focuses on developing a **CNN** to classify images into 151 animal categories, structured with **four convolutional layers** and **a fully connected output layer** optimized for effective feature extraction and classification. The initial convolutional layer utilizes **64 filters sized 5x5** with **a stride of 1** to extract primary visual features, followed by three layers with **128 filters of 3x3 size**, enhancing the model's capability to capture more complex features.

To prevent overfitting and boost generalization, each layer integrates a **ReLU** activation function and a **2x2 max pooling operation**, reducing feature map dimensions and improving computational efficiency. The network concludes with a **Softmax** function in the output layer for classifying the features into animal categories. Image preprocessing includes **resizing, horizontal flipping, and center cropping to standardize inputs to 112 pixels**, enhancing performance and efficiency.

Initial tests showed a **36.09% accuracy** with a computational cost of **0.3555 GFLOPs**, indicating good baseline performance but also room for improvement in scalability and generalizability. Enhancements like data augmentation, transfer learning, and tuning training parameters are being considered to elevate the model's accuracy and practical application efficacy.

II. MODEL IMPROVEMENT IMPLEMENTATION

Here are the following improvement stages executed during training model:

1. Enhancing Training Data with Augmentation Techniques

To combat overfitting and improve the generalization capabilities of our model, we employed a series of data augmentation strategies. These included **resizing** images to 224x224 pixels, applying **horizontal flips, random rotations, color jittering, and random resized cropping**. These techniques are designed to emulate different photographic conditions.

2. Utilizing Transfer Learning and Fine-Tuning

We applied *transfer learning* from models pre-trained on the expansive ImageNet dataset to fast-track our development cycle and enhance feature extraction capabilities. Specifically, we chose **Reset50, ResNet18 Fined-tuned and ShuffleNet Fine-Tuned using ShuffleNet_v2_x0_5**.

Fine-tuning these models involved freezing the *initial convolutional layers* to retain generic features and retraining the final layers to adapt to our specific classification task, efficiently customizing the models to our dataset.

3. Optimizing Learning Rates for Rapid Convergence

Adaptive learning rates were set based on the architectural nuances of each model:

- **ResNet50** and **ResNet18** were trained with a learning rate of 0.001, optimizing for stability.
- **ShuffleNet** used higher rates (0.001 and 0.02, respectively), to exploit their architectural efficiencies for faster convergence.

4. Efficiency Analysis

We measured efficiency by comparing the accuracy percentage to **GFLOPs**, pinpointing the most resource-efficient models suitable for deployment in resource-sensitive settings.

5. Result summary – Ablation Study

Our ablation study demonstrated significant improvements through our methodologies (provided in the following table). Initial **baseline accuracy was 36.09%, with a computational efficiency of 101.53 (acc%/GFLOPs)**. By integrating data augmentation, transfer learning, and fine-tuning strategies, we saw dramatic improvements:

- **ResNet50:** Achieved *61.27% accuracy* with efficiency of *14.78 (acc%/GFLOPs)*.
- **ResNet18 Fine-tuned:** Reached *an accuracy of 96% with efficiency of 52.622* in terms of GFLOPs.
- **ShuffleNet:** Noted for its balance between *efficiency (2075.10)* and *accuracy (92.81%)*, making it highly suitable for deployment where resource efficiency is crucial.

Model	Results			Model defining			
	Accuracy (%)	GFLOPS	Efficiency (acc%/GFLOPS)	Transfer Learning	Fine-tuning	Learning rate	Data Augmentation
Baseline	36.09	0.355	101.526	No	No	0.001	No
ResNet50	61.27	4.143	14.789	Yes	No	0.001	Yes
ResNet18 Fine-tune	96.09	1.826	52.623	Yes	Yes	0.001	Yes
ShuffleNet	92.81	0.044	2075.107	Yes	Yes	0.02	Yes

III. MODEL EFFICIENCY AND TRADE-OFFS

To achieve a balance between computational efficiency and classification accuracy, our approach involved careful model selection and optimization. Below are the details and rationales behind choosing each model:

- **ResNet50:**

Initially chosen for its depth and robust feature extraction capabilities, ResNet50 offered improved accuracy but was computationally demanding, registering **4.143 GFLOPs**. Although it enhanced model *accuracy to 61.27%*, its efficiency in terms of *acc%/GFLOPs* was lower at **14.789**, indicating a substantial computational cost. This prompted the need to explore more efficient architecture.

- **ResNet18 Fine-Tuned:**

As a scaled-down alternative to ResNet50, the fine-tuned ResNet18 model significantly reduced the computational load to **1.826 GFLOPs** and achieved a remarkable accuracy of **96.09%**. While the model's efficiency improved to **52.623 acc%/GFLOPs**, it still fell short of the desired efficiency gains, leading us to consider even more lightweight models.

- **ShuffleNet Fine-Tuned:**

Ultimately, we selected the **ShuffleNet_v2_x0_5** model for its exemplary balance between high accuracy (**92.81%**) and minimal computational demand (0.044 GFLOPs). This model not only achieved an efficiency score of **2075.107 acc%/GFLOPs**, making it **21.75** times more efficient than the baseline, but also provided a viable solution for deployment in resource-constrained environments.

By integrating these focused strategies, we have significantly elevated the classification accuracy while managing computational costs effectively. The chosen models and their configurations ensured a superior trade-off between accuracy and efficiency, as evidenced by the high efficiency scores, particularly of the *ShuffleNet* model. This approach not only meets our accuracy standards but also aligns with our goals for deploying highly efficient and effective models in practical settings.

IV. LIMITATIONS AND CONCLUSIONS

1. Limitations

Despite the successes, the project has notable limitations:

1. **Training Duration:** The models were trained for only 10 epochs, potentially limiting their optimization and learning stability, especially in deeper architectures.
2. **Model Specificity:** The optimizations focused on efficiency may not translate well to different or more complex tasks requiring adjustments for broader applications.
3. **Generalization Concerns:** Although data augmentation was used to enhance robustness, the models' performance on completely unseen and varied real-world data needs more testing.

2. Conclusions

The project demonstrated that strategic model selection and optimization can significantly enhance accuracy and computational efficiency in image classification. *The fine-tuned ShuffleNet* model excelled by offering the best balance between high accuracy and efficiency, suitable for resource-constrained environments. This approach has established a scalable method for similar tasks and lays the groundwork for future enhancements in image classification technology, particularly in optimizing models for practical deployment in diverse settings.