# Research Paper

By Shakti Biplab

## Title: Development of an Animal Classification System Using Deep Learning and Transfer Learning


## Abstract

The rapid advancement of deep learning techniques has enabled the development of robust image classification systems. This project focuses on building an animal classification system capable of identifying 15 different animal species from images. Leveraging transfer learning with the MobileNetV2 architecture, the system achieves high accuracy while being computationally efficient. The software includes a user-friendly graphical interface that allows users to drag and drop images for real-time predictions. This paper discusses the methodology, implementation, and evaluation of the system, highlighting its potential applications in wildlife monitoring, education, and conservation.


## 1. Introduction

Image classification is a fundamental task in computer vision, with applications ranging from medical diagnosis to autonomous driving. In this project, we address the challenge of classifying animals in images, which has significant implications for wildlife monitoring, biodiversity studies, and educational tools. Traditional methods for image classification often require extensive computational resources and large datasets. However, by using transfer learning, we can leverage pre-trained models to achieve high accuracy even with limited data and resources.

This project aims to:

1. Develop a deep learning model capable of classifying 15 animal species.
2. Create a user-friendly interface for real-time predictions.
3. Evaluate the model's performance and address limitations.

## 2. Methodology

### 2.1 Dataset

The dataset consists of 15 classes of animals: Bear, Bird, Cat, Cow, Deer, Dog, Dolphin, Elephant, Giraffe, Horse, Kangaroo, Lion, Panda, Tiger, and Zebra. Each class contains images of size 224x224x3, resized to match the input requirements of the MobileNetV2 model.

### 2.2 Model Architecture

We used **MobileNetV2**, a lightweight and efficient convolutional neural network (CNN) pre-trained on the ImageNet dataset. The model was fine-tuned by adding a global average pooling layer, a fully connected layer with 512 units, and a softmax output layer for multi-class classification.

### 2.3 Training

The model was trained using the Adam optimizer with a learning rate of 0.0001. Data augmentation techniques, such as rotation, zooming, and flipping, were applied to improve generalization. The model achieved a validation accuracy of over 95% after 20 epochs.

### 2.4 Confidence Threshold

To handle cases where the model is uncertain, a confidence threshold of 70% was introduced. If the model's confidence is below this threshold, the system responds with "Sorry, I am unable to detect that."

---

## 3. Implementation

### 3.1 Software Design

The software consists of two main components:

1. **Deep Learning Model**: Trained using TensorFlow and Keras.
2. **Graphical User Interface (GUI)**: Built using Tkinter and TkinterDnD for drag-and-drop functionality.

### 3.2 Features

- Drag-and-drop image input.
- Real-time animal classification.
- Confidence-based feedback for uncertain predictions.
- A "Clear" button to reset the interface for multiple predictions.

---

## 4. Results and Discussion

The system was evaluated on a test set of images, achieving an accuracy of 95.2%. The confidence threshold effectively filtered out uncertain predictions, ensuring reliable results. However, the system occasionally misclassified non-animal images, highlighting the need for further improvements in dataset diversity and model robustness.

---

## 5. Applications

1. **Wildlife Monitoring**: Automatically classify animals in camera trap images.
2. **Education**: Teach students about animal species using interactive tools.

3. **Conservation**: Assist researchers in tracking endangered species.

---

## 6. Conclusion

This project demonstrates the effectiveness of transfer learning in building accurate and efficient image classification systems. The developed software provides a user-friendly interface for animal classification, making it accessible to non-technical users. Future work will focus on expanding the dataset, improving model robustness, and adding support for more animal species.

---

## References

1. Howard, A. G., et al. "MobileNetV2: Inverted Residuals and Linear Bottlenecks." *arXiv preprint arXiv:1801.04381* (2018).
2. Chollet, F. "Deep Learning with Python." *Manning Publications* (2017).
3. TensorFlow Documentation. https://www.tensorflow.org/