Animal Classifier

Custom image classification dataset featuring five classes: dog, cow, cat, lamb, and zebra, each with 100 images sourced from the internet or captured using a phone. The goal is to develop a classification model to identify these classes with high accuracy.

1st Fahim Muntashir
Computer Science and Engineering
North South University
ID: 2021183642

2nd Md Abul Bashar Nirob
Computer Science and Engineering
North South University
ID: 2022198042

3rd Tausib Abrar
Computer Science and Engineering
North South University
ID: 2121446642

4th Muhammad Omar Mahin Jinnat
Computer Science and Engineering
North South University
ID: 2012826642

***Abstract*—Image classification is an essential task in computer vision with applications in various domains, including wildlife monitoring, security, and automated sorting. This study focuses on building a classifier capable of distinguishing five different animal species: dog, cow, cat, lamb, and zebra. The dataset consists of 500 images collected from diverse sources to ensure robust classification performance. This paper details the dataset preparation, model development, and initial findings, outlining future steps for model improvement. Additionally, we explore challenges faced in dataset curation, such as variations in lighting conditions, occlusions, and background clutter, which impact model generalization.**

***Index Terms*—Image classification, Convolutional Neural Networks, Deep Learning, Data Augmentation, Transfer Learning**

I. Introduction

Image classification plays a crucial role in modern AI applications, especially in automated animal recognition systems. This research aims to create a machine learning model that accurately classifies five animal species using a custom dataset. The dataset is curated to include varying lighting conditions, angles, and backgrounds to enhance model robustness. The objective is to achieve a classification accuracy of at least 90%. This paper elaborates on the dataset collection, preprocessing methods, and initial experiments leading to the classifier's development. Furthermore, we discuss potential applications, such as wildlife conservation, automated surveillance, and veterinary diagnostics.

II. Methodology

A. Dataset Collection

A custom dataset was created by sourcing 100 images for each of the five classes from publicly available sources such as Google Images and Unsplash. The dataset was organized into separate directories per class to facilitate model training. Care was taken to include images from different angles, lighting conditions, and backgrounds to improve generalization. Additionally, the dataset was manually inspected to remove duplicate and irrelevant images, ensuring data quality.

B. Data Preprocessing

Preprocessing steps were applied to standardize the images and enhance model performance. These steps include:

- **Resizing:** All images were resized to 224x224 pixels to maintain uniformity.
- **Normalization:** Pixel values were scaled to a [0,1] range for optimal convergence.
- **Data Augmentation:** Techniques like rotation, flipping, and cropping were applied to increase dataset diversity and improve model robustness.
- **Noise Reduction:** Gaussian blurring and histogram equalization were tested to reduce unwanted artifacts.

III. Tools and Libraries

The following tools and libraries were employed in this study:

- **TensorFlow/Keras:** Used for building and training the convolutional neural network.
- **OpenCV:** Applied for image preprocessing and transformations.
- **Pandas/NumPy:** Used for data handling and efficient numerical computations.
- **Matplotlib/Seaborn:** Employed for visualizing dataset distributions and model performance.

IV. Model Evaluation

After training the initial CNN model, evaluation metrics were employed to assess its performance:

- **Accuracy:** The initial model achieved an accuracy of 85%.

Identify applicable funding agency here. If none, delete this.

- **Confusion Matrix:** Used to analyze misclassification rates among different classes.
- **Precision, Recall, and F1-score:** Provided detailed performance insights.
- **Loss Function Analysis:** Cross-entropy loss was used to evaluate training stability.

V. Progress Update

The project has reached the following milestones:

- Dataset collection: Completed (500 images total, 100 per class).
- Data preprocessing: Completed (resizing, normalization, augmentation, noise reduction).
- Initial model implementation: A basic Convolutional Neural Network (CNN) was developed and tested on a subset of the data, achieving an initial accuracy of 85%.

VI. Next Steps

To further improve the model, the following tasks will be undertaken:

- Enhance the CNN architecture or employ transfer learning with models like VGG16 or ResNet.
- Conduct hyperparameter tuning, optimizing learning rates, batch size, and epoch count.
- Evaluate performance using additional metrics such as ROC-AUC curves.
- Expand the dataset by incorporating more images and applying advanced augmentation techniques.

References

1) A. Krizhevsky, I. Sutskever, and G. E. Hinton, ÏmageNet Classification with Deep Convolutional Neural Networks,ïn Advances in Neural Information Processing Systems, 2012.
2) TensorFlow Documentation, Ïmage Classification Tutorial,ävailable at: *https://www.tensorflow.org/tutorials/images/classification*.
3) J. Deng et al., ÏmageNet: A Large-Scale Hierarchical Image Database,ïn IEEE Conference on Computer Vision and Pattern Recognition, 2009.