Deep Learning Project

Mammal Species Image Classification

Overview

This project involves building a deep learning model to classify images of different mammal species. Students will use a dataset which contains images of 45 different mammals. The goal is to apply and compare at least two deep learning algorithms to classify these images accurately. Additionally, students will create a user interface using Streamlit or Gradio to demonstrate their models.

Objectives

1. **Understand and preprocess the dataset**: Explore the dataset, understand its structure, and preprocess the images for deep learning models.
2. **Model Development**: Implement at least two different deep learning models for image classification.
3. **Model Comparison and Analysis**: Compare the performance of the models based on accuracy, precision, recall, and F1-score.
4. **User Interface Creation**: Develop a user-friendly interface using Streamlit or Gradio where users can upload images and get predictions.

Dataset

- **Source**: *45 Animals*
- **Content**: Images of 45 different mammal species.

Suggested Deep Learning Algorithms

1. **Convolutional Neural Networks (CNNs)**: A standard algorithm for image classification tasks.
2. **Transfer Learning with Pre-trained Models**: Utilize models like ResNet, VGGNet, or Inception, which have been pre-trained on large datasets like ImageNet.

Tools and Libraries

- **Python**: Programming language.
- **TensorFlow/Keras or PyTorch**: For building deep learning models.
- **OpenCV or PIL**: For image processing.
- **NumPy and Pandas**: For data manipulation.
- **Streamlit or Gradio**: For building the user interface.
- **Matplotlib or Seaborn**: For data visualization.

Steps

1. **Data Exploration and Preprocessing**:
    - Load the dataset.
    - Visualize different mammal species images.
    - Normalize and resize the images.
    - Split the dataset into training, validation, and test sets.
2. **Model Building**:
    - Implement two or more deep learning models.
    - Compile and train the models on the training set.
    - Use validation data to fine-tune and optimize.
3. **Model Evaluation**:
    - Evaluate the models on the test set.
    - Compare the models based on various metrics.
4. **User Interface Development**:
    - Create a simple UI using Streamlit or Gradio.
    - Allow users to upload an image and get a prediction.

Resources

- Deep Learning Tutorial for Beginners
- PyTorch Tutorial for Deep Learning