Animal Gate Code Documentation

Overview

The Animal Gate Code project comprises two main parts: training the model and using the pretrained model on a Raspberry Pi.

Training the Model

1.Download Data:
Obtain the dataset from this Kaggle link.

2.Model Training Script:

Utilize the TrainModel.py script for preparing and training the model.
Functions in TrainModel.py**:**

a. **Remove Dog Data:**
Use removeDogData to eliminate dog data from the dataset.

b. **Clean Train/Test Name Files:**
The test.txt and train.txt files may contain redundant text. Utilize cleanTrainTestNameFile to remove unnecessary content.

c. **Split Data:**
The dataset needs to be split into training and testing sets. Employ trainTestSplitBasedOnFile for this purpose.

d. **Model Training:**
The ModelHead class is designed for training the model. Utilize the train function to train the ResNet architecture.

4. Pretrained Model for Raspberry Pi:

For deploying the model on Raspberry Pi, the DetectCat.py script is provided. It captures an image, classifies the breed, and prints the result to the console.

a. **Capture Image:**
Capture_image() captures an image from the camera and saves it.

b. **Classify Image:**
Classify_image uses the pretrained model to detect the breed.

c. **Prepare Image:**
Prepare_image readies the image for the model.

d. **Run:**
When run is called, it first captures an image, then opens the saved image, and finally sends it to classify_image.

Code Samples

prepareData = PrepareData()
prepareData.removeDogData()
prepareData.trainTestSplitBasedOnFile()
prepareData.splitData('train')
prepareData.splitData('test')
prepareData.splitData('valid')

Model Training

model = torchvision.models.resnet50(pretrained=True).to(device)

train(model, N_EPOCHS, criterion, optimizer, train_data_loader, valid_data_loader, device, MODEL_SAVE_PATH)