Object Recognition and Computer Vision Assignment 3

Imane Elbacha

Abstract

The purpose of this assignment is to identify the classes of a portion of the bird dataset Caltech-UCSD Birds-200-2011. This dataset includes images of 20 different bird species. Training, validation, and test data are separated from it. I will first go through data pre-processing in this report. I'll go into more detail about the categorization model and its variables after that. Finally the model that gives the best results

1. Dataset

The dataset contains 1082 images in the training set, 103 in the validation set and 517 in the test set. Using the transforms of PyTorch, the images were normalized and resized to 224 x 224, after redistributing the validation and training set.

1.1. Calibration of validation and training data sets

We must expand the validation dataset in order to achieve better parameter tuning. To accomplish this, I added photos from the training dataset to this dataset, dividing the total number of images into 25% for the validation data and 75% for the training data, respectively.

1.2. Preprocessing

Looking at the pictures, one can see that the bird sometimes only takes up a little portion of the picture. Therefore, it seems like a good idea to crop the photographs so that only the bird is maintained in addition to normalizing and resizing the images . Since the given boxes cannot be used for this, one can use a pre-trained Mask RCNN to find them (which yields better performance than single-pass networks such as YOLO). This method was tested for the case of this dataset, even though it yeilded good results in other papers it affected the training negatively in our case. It's an area that could be explored further.

1.3. Data augmentation

By creating fresh and diverse instances to train datasets, data augmentation helps to improve model performance and results. Working with PyTorch's basic transformations is one of the simplest ways to go about it. I utilized random rotation with an angle of 20°, random horizontal flip with a frequency of 0.5, and random affine with a degree of 20° in this assignment.

2. Architecture of the model

After testing several models(which are included in the code), we present the model that has the best results.

To get good results, we use pre-trained models (often learned on ImageNet). In fact, the majority of the photos have features in common. For instance, the first layers of a network can recognize edges, which are present in the majority of the images. This makes utilizing a pre-trained model a "good initialization."

I used the pre-trained model called ResNet34. In addition, I substituted two completely connected layers, a Tanh activation function, and a dropout with probability of 0.6 for the final fully connected layer (fc). I modified the model in the forward function by including a block that allows for more precise classification. In fact, this method is utilized to distinguish between insignificant categories like bird species.

There are two steps to learning. I employed the ADAM optimizer in both steps. We only leave the changed (fc) layers unfrozen in the first stage and freeze the ResNet weights and biases after the (fc) layers. The model is then trained across 20 epochs. The weights and biases of the further layers are trained in this step. The second stage involves unfreezing every layer and retraining the network over a period of 20 additional epochs. I added a terminating criterion based on the loss values in this stage.

3. Improvements:

There are several areas of improvement for the model and preprocessing method. First the croping step could yield interesting resultsif explored further the issue is the computation time(this step takes 3 hours to examine the whole dataset). Second, we could explore the Resnet50 pre-trained model (presented in the code) since we could use models pre-trained on iNaturalist.