Assignment 3 : Bird classification challenge

Yassine NAJI : yassine.naji@ip-paris.fr

Abstract

The objective of the assignment is to classify bird images taken from the Caltech-UCSD Birds 200-2011 into 20 classes representing bird species. Provided only with 1185 images , our goal is to build a robust model that could reach high accuracy on test images. We will discuss in this report our choice of the model and the main techniques that we used to improve its performance.

1. Data preprocessing :

1.1. Data distribution :

The distribution of the 1023 train images is balanced according to the classes, however it's not the case for validation. I had to do a balanced split into 3 folds that I used for the ensemble method.

1.2. Data augmentation:

To deal with the small amount of training data, I applied data augmentation techniques such as : **rotation, translation, horizontal flip, adding noise and gaussian blur.** I noticed that in some images, birds occupy a small part. I managed to crop them using **Fast RCNN** pretrained on COCO dataset**,** so that the model focuses more on bird features. In the same approach, I used **FCN-RESNET101** (also pretrained on COCO) to perform segmentation of train images in order to delete background. The last technique, unlike the others, didn't improve the performance of the model because the bird segmentation wasn't always done perfectly and the resulting images weren't realistic (adding new background could have improved performance). Finally, I applied the **pseudo-labeling** technique which consists of predicting the class of unlabeled bird images and adding them to the training set if the model is confident with 99% about the label.

1.3. Preprocessing :

Since the model requires square input, I padded images before resizing to avoid distorting them. I also chose a relatively high resolution 300x300 (instead of 224x224) so the models can catch more details of birds .

2. Model:

I tested models built from scratch and others based on pretrained models on Image Net (ResNet, VGG ..) in which I modified the last layers. I reached the best performance on test images by training 3 models using bagging with 3 folds, and made them vote. The first model is **ResNet 152** pretrained on ImageNet in which I replaced the last layer by 2 fully connected layers with ReLU activation (the last hidden layer has 1000 units). I also added dropouts and froze the 6 first layers of the pretrained network to prevent overfitting. I did the same manipulation using the two other pretrained models on ImageNet : **ResNext101 32x8d** and **Wide ResNet 101.** I trained each model on two folds and test it on the third using **Adam** optimizer with a **scheduler** for the learning rate, I set the initial value of the learning to 10^{-4} and I multiplied it by 0.75 at each epoch. I trained the 3 networks independently for 20 epochs each with a small batch size (16) to prevent memory issues and I saved for each model, parameters which provided the highest validation accuracy. I obtained the following validation accuracy scores:

- **ResNet 152 : 94%**
- **ResNext101 32x8d : 95%**
- **Wide ResNet 101 : 93%**

By combining the 3 models and using the output: $argmax_{c \in \{1,..,20\}} p_1(c) + p_2(c) + p_3(c)$

I got the score of 80% on test images.