Fine-Grained Image Classification

Israfel SALAZAR
MSc. MVA Mathématiques / Vision / Apprentissage
ENS-Paris-Saclay
Paris, France
israfel.sr@gmail.com

***Abstract*—Fine-grained image classification aims to identify minor categories between classes, for instance dog breeds or bird species. In this work we present a methodology for fine-grained classification using deep learning. We will discuss the different implementations as well as the problems that arise in this particular task.**

I. Dataset

For this work we are going to be using the bird_dataset. This contains 1800 training samples divided in 20 classes. In the figure below we can see some images of the training set.

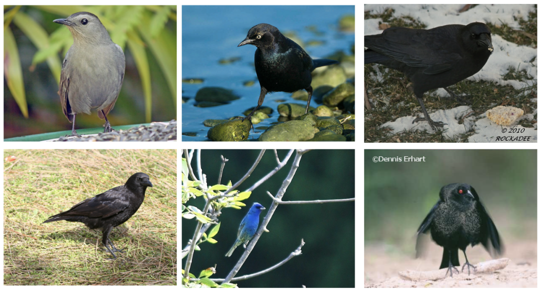

Fig. 1. Sample of the training dataset.

As we can see the images are quite similar between each other. Fine-grained image classification is a hard task in which the models must find small details in the images to distinguish between one class or another.

II. Architectures

To solve this problem we tried several architectures. We tested training a model from scratch but the results were not interesting. Then we moved to a more robust implementation using a pretrained model as a base for the classifier. We took care of taking a model only trained on imagenet and not on more specific datasets. We selected two different pretrained models a convolutional neural network, the resnet152 and a transformer, teh vit. To create the architectures we used pytorch lightning which is great for model prototyping and we created an easy to test pipeline that allows us to change between the pretrained base to be able to compare both.

Over the pretrained base we build a fully connected layer as a classifier layer that takes as inputs the embedded vector of the inputs after passing by the base architecture and outputs the number of classes.

TABLE I
Training Results

Acc	ResNet152	Transformer	B-CNN
Train	1	1	0.94
Val	0.53	0.89	0.75

We also tested a more sophisticated Bi-linear Convolutional Neural Network This network uses a combination of two CNNs to capture pairwise interaction between the image features. It was specially designed for the Fine-Grained image classification task. We used the resnet152 as a base model for this architecture.

III. Training

We train all of the models using GPU. Pytorch lightning allows us to easily move from one device to another. Since we were using pretrained models as base architecures, we trained for maximum 10 epochs to avoid overfitting. We watched the train and validation loss and accuracy. To follow the training we used the weighs and biases (wandb) framework that logs training values. We computed the validation loss and accuracy after each epoch and we also implemented a checkpoint logger that saves the model each time the validation accuracy is maximized, saving the model at that training stage.

IV. Results

Table I shows a summary of the results for the training of the different models. We can see that the best performance is achieved with the transformer and that the Bi-linear CNN performs more than 20 points better than the plain resnet152. We explain these results mainly because of overfitting. The plain resnet quickly achieves train accuracy 1 and stops learning because it does not have many layers that avoid overfitting. On the B-CNN we added a dropout and the transformer architecture has several layers whose only purpose is to avoid overfitting.

V. Conclusion

In this report we presented the implementation for fine-grained image classification. We discuss the implementation of three different architectures and its results on the training and validation bird_dataset. We show that the transformer architecture is the model achieving the highest performance. We believe that B-CNN can be extended using transformers and that their results will be even better.