Assignment 4

Due on 01.06.2025 (23:00)
Programming Language: Python 3.9

Introduction

Classification is one of the core problems of human being. Human being's first aim was to classify if an object is a threat or not at the time they were living in their caves. The question at there was is s/he the hunter of the object or prey of the object. Human being gained this ability to run away from threats and run into goals. This ability is involved in that long time period too. Nowadays, we are trying to classify structures into more detail, instead of calling them as prey, we are classifying them into more detail, such as animals, plants etc. As the task gets more detailed the harder it becomes. Human being is the one that wants to maximize its profit and minimize the effort to consume for it. So, we invented the machines and then computing machines to make them compute for us the things that we are in need of computing. Classification is one of these tasks, and image classification is the subfield of that task. Nowadays, we can even predict species of a bird with relatively simple programs. In this project, your aim is to classify given bird species into correct classes with help of neural networks.

Dataset

The dataset[1] contains twenty-five species of birds in India with 1,500 samples for each. **The samples are separated as 1,200 for training and 300 for validation, but you are supposed to split each validation set into half randomly to separate as 150 for test and 150 for validation in order to achieve 80-10-10 ratio for training, testing, and validation purposes with respect to order of ratios.** The dataset contains 37,500 images in total and each image is approximately 1 MP.

You are free to use the dataset as you desire in this project, which means you can downscale the images and/or you can work with a subset of the dataset (be careful about conserving ratio of the sets; train, test, and validation as 80-10-10; also be considerable while downscaling or taking subset, do not take only a little portion of the dataset or do not apply too much downscaling) if computational power of your machine (graphical power etc.) is not enough for this project, so, you are not required to buy Colab Premium or something like that, but keep in mind that using whole dataset and applying as less downscaling as you can will generate better results which gives you opportunity to do better interpretations, because the important thing in this project is your interpretations and implementations rather than the accuracy, however, keep in mind that if your accuracy is way lower than the others, it will also be penalized.

Steps to Follow

Part 1: Classification According to Feature Extraction

Extract features from the images as you did in your first project assignment, but this time it will be more complex than the first assignment as it is the last one. The features you can extract are: Color, Color Histogram, SIFT, HoG, Gabor Filter, and the features that you think they give good insights. Later on, apply some basic ML (Machine Learning) algorithms such as SVM (Support Vector Machines), Random Forest, Naive Bayes, MLP (Multilayer Perceptron), and algorithms that you think they are worthy to try. **You have to try at least three feature extraction methodology and try at least three ML algorithms. Do not forget to try each feature that you extracted with model separately. Remember to report all your results and comment about them.**

Part 2: Principal Component Analysis and Feature Selection

Apply PCA (Principal Component Analysis) and a feature selection method (you are supposed to select a method that eliminates some features instead of creating new features from them as in PCA) to the all features that you have extracted at the first part **separately**. Then try the same algorithms as in the first part. **Compare results of these two approaches with each other and the results of the first part. Do not forget to experiment with all algorithms and the features as you experimented in the first part.**

Part 3: Fine-Tuning Pretrained CNN Models

Select at least three well-known pretrained CNN (Convolutional Neural Network) models and fine-tune them to adapt them to the given dataset. **Do not forget to report your results and comment about them.** Also note that **you are supposed to provide loss graphs and decide where to stop the training according to the losses**; you may implement other approaches to stop the training of the models in addition to this one, **remember to provide a graph or a material about your other methodologies if you applied one.**

Part 4: Training Randomly Weighted CNN Models

Assign random weights to the CNN models that you selected for third part and train the models with the given dataset. **Compare your results with the third part. Do not forget to provide the same outcomes (in meaning of comments, graphs etc.) as in the third part.**

Part 5: Implementing and Training Your Own CNN Model from Scratch

You can use any library (PyTorch, TensorFlow etc.) to create your own CNN model. You can use any of the methods of these libraries. The aim here is creating a CNN model from scratch such as determining the best layers (in meaning of count, shape etc.) that fits best to the given dataset. **Do not forget to share results of all your considerable trials in the way of the achieving the best. Note that you are supposed to provide the same outcomes (in meaning of comments, graphs etc.) as in the third and fourth parts.**

Extras

- Import and visualize the data in any aspects that you think it is beneficial for the reader's better understanding of the data.
- The most important part of this project is doing as much experiment as you can to show strengths and weaknesses of the algorithms. In short, you are supposed to experiment with different scenarios and comment about them, note that commenting is as much important as the experimenting, so, please explain your reasoning and inference for every experiment that you did. **Remember to compare the result of the each step with the all others, and also compare the sub-results of the steps in the step itself.**
- **Remember to compute accuracy of your model** at least with following metrics: Accurracy, Precision, Recall, F_1-Score. You can use libraries to obtain these values. You can also use any other metrics that you think it is beneficial in addition to them.
- Show the examples that your models are unable to classify correctly, discuss why they are mislabeled. Also discuss the examples that one model successfully predicts but the other not, and vice versa, discuss the reason behind it.
- You are free to implement each step on your own, there is no limitation, you can use any libraries, any methodologies. You can use any libraries that you think you are in need of, but **use Python 3.9 and your report also has to contain necessary libraries to be installed with the versions that are used (!pip install commands are preferred).** The important things in this project are your interpretations and implementations, so, please be neat at your interpretations and do not hesitate to comment about anything or draw a plot about it that you think it is beneficial, also please be neat at your code too, your code must also be thoughtfully commented.

What to Hand In

You are required to submit all your code in a Jupyter notebook, along with a report in ipynb format, which should also be prepared using Jupyter notebook. **The code you submit should be thoroughly commented and your notebook must be ran and have outputs for each cell in the order of the cells before submission.** Your report should be self-contained and include a concise overview of the problem and the details of your implemented solution. Note that your report also has to contain necessary libraries to be installed with the versions that are used (!pip install commands are preferred). Feel free to include pseudo-code or figures to highlight or clarify specific aspects of your solution.

Submission hierarchy must be as follows:

```
- <GroupID>.zip
    - assignment4.ipynb
    - *.(jpg|jpeg|png|gif|tif|tiff|bmp|svg|webp) (optional)
```

Do not send the dataset.

Preparing a good report is important as well as the correctness of your solutions! You should explain your choices and their effects on the results. You can create a table (or any content you believe that it is beneficial to show your all work) to report your results.

Note that submission format is crucial and submit system is set to give you score as one if you follow the submission hierarchy, which is really easy (there might be some issues for the MacOS users but it can be overcomed via the mini guide that is shared at the Piazza). If you do not score one from the submit system **you will penalized by 20% even if your submission hierarchy is correct**.

Academic Integrity

All work on assignments must be done on your own group unless stated otherwise. You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudo-code) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the material found on the web as everything on the web has been written by someone else.

References

[1] Kaggle - indian birds species image classification dataset. `https://www.kaggle.com/datasets/ichhadhari/indian-birds/data` (Last access: 05.05.2025).