PR2: Image Classification

Published Date:
Mar. 16, 2018, 5:00 p.m.

Deadline Date:
Mar. 30, 2018, 11:59 p.m.

Description:

****
This is an individual assignment.
****

Overview and Assignment Goals:

The objectives of this assignment are the following:

- Experiment with different image feature extraction techniques.
- Try dimensionality reduction techniques (must try at least one, even if you do not use it in the final solution – include it as a function in the code)
- Experiment with various classification models.
- Think about dealing with imbalanced data.

Detailed Description:
Develop predictive models that can determine, given an image, which one of 14 classes it is.

Traffic congestion seems to be at an all-time high. Machine Learning methods must be developed to help solve traffic problems. In this assignment, you will analyze features extracted from tiny traffic images depicting different traffic-related objects to determine their type as one of 14 classes, noted by integers 1-14: car, suv, small_truck, medium_truck, large_truck, pedestrian, bus, van, people, bicycle, and motorcycle, signal_green, signal_yellow, signal_red. The object classes are heavily imbalanced. For example, the training data contains 31,775 cars but only 280 motorcycles and 197 buses. Classes in the test data are similarly distributed.

The input to your classifiers will not be the images themselves, but rather features extracted from the images. You can experiment with any kind of image feature extraction techniques you would like, as long as the output is a vector representation of the image. You may not use models that automatically learn features and classify (e.g., YOLO). Instead, you must build a vector representation of the images and then experiment with using classification algorithms to achieve the best classification performance. A few classic examples of image features are Histogram of Oriented Gradients (HOG) features, Normalized Color Histogram (Hist) features, Local Binary Pattern (LBP) features, Color gradient (RGB) features, Depth of Field (DF) features, etc.

Since the dataset is imbalanced, the scoring function will be the F1-score.

Caveats:

+ Remember that not all features will be good for predicting the object class. Think of feature selection, engineering, reduction (anything that works).

+ Use the data mining knowledge you have gained until now, wisely, to optimize your results.

Data Description:

The training dataset consists of 100,000 records and the test dataset also consists of 100,000 records. We provide you with the training class labels and the test labels are held out. We have included a small subset of the data in the *traffic-small* dataset which you may use to practice your codes locally. However, we recommend you use the HPC for the final computation on the *traffic* dataset. Note that the dataset file is very large (**474 MB**, expands to **1050 MB**). Ensure you have enough space on your drive before expanding the file. Moreover, you will need to think carefully about how you will organize computations so you do not run out of RAM during training.

The dataset file contains two dataset directories: *traffic*, and *traffic-small*. In each directory, you will find *train* and *test* sub-directories with images numbered 1 to 100,000 (e.g., 000001.jpg) for the traffic dataset and 1 to 4209 for the traffic-small dataset. The train and test sets contain the same number of images. The image ID for the ith image corresponds to the label on the ith line of the train.labels file found in the main dataset directory. The traffic-small dataset contains a test.labels file, but the traffic dataset does not. Your task is to predict those labels for the images in the test set and create a test.txt file containing those labels, which you will submit to CLP. Note that CLP only accepts files with extensions .txt or .dat.

Rules:

- This is an individual assignment. Discussion of broad level strategies are allowed but any copying of prediction files and source codes will result in an honor code violation.
- Feel free to use the programming language of your choice for this assignment.
- You are allowed 5 submissions per day.
- After the submission deadline, only your chosen or last submission is considered for the leaderboard.

Deliverables:

- Valid submissions to the Leader Board website: https://coe-cmp.sjsu.edu/clp/ (username is your MySJSU email and your password is your MySJSU password).

Canvas Submission for the report:

- Include a 2-page, single-spaced report describing details regarding the steps you followed for feature extraction, feature selection, and classifier model development. The

report should be in PDF format and the file should be called **<SJSU_ID>.pdf**. Be sure to include the following in the report:

1. Name and SJSU ID.
2. Rank & F1-score for your submission (at the time of writing the report). If you chose not to see the leaderboard, state so.
3. Your approach.
4. Your methodology of choosing the approach and associated parameters.

- Ensure you submitted the correct code on CLP that matches your output. Code does not need to be submitted on Canvas.

Grading:

Grading for the Assignment will be split on your implementation (70%), report (20%) and ranking submissions (10%). Extra credit (1% of final grade) will be awarded to the top-3 performing algorithms. Note that extra credit throughout the semester will be tallied outside of Canvas and will be added to the final grade at the end of the semester.

Files: Due to the large size of the dataset, it has been posted on the CLP web server. Please download the dataset from: https://coe-cmp.sjsu.edu/clp/static/datasets/traffic.tar.gz