Foundations of Machine Learning - Fall 2018 - Project #00

Project Due: October 9, 2018, 11:59pm

Your goal in this project is to develop a machine learning approach to find all *red cars* in high resolution satellite imagery. You have been provided with some ground-truthed imagery that indicates the location of all red cars ('data_train.npy' and 'ground_truth.npy') and some imagery that does not have corresponding ground-truth ('data_test.npy').

You can view the imagery with the following commands:
import matplotlib.pyplot as plt
plt.imshow(data_train)

You may use any approach presented in the class so far. These methods include regression using a squared error objective (with any basis function, with or without a regularization term), probabilistic generative classifiers, k-nearest neighbors, and k-means. (You may **NOT** use methods not yet covered in class - e.g., neural networks/deep learning) You may also use any extension of a method from class that you develop yourself or find in the literature or any combination thereof. For example, we discussed the standard k-nearest neighbors (k-nn) algorithm with Euclidean distance. However, you may use k-nn with a different distance metric, or the fuzzy k-nn algorithm[1], or the possibilistic k-nn algorithm[2], or k-nn with spatial constraints, k-nn followed by k-means, etc.

You may use the RGB values for each pixel as features directly or you may extract additional features. (Note: This is not required! It is optional.)

In this project, you should develop and test your approach. Your testing should include experiments that highligh advantages and disadvantages to your approach. Your experiments should answer the following questions at a minimum:

- Why did you select this approach over others?
- How did you select parameter settings?
- How robust is your method? How do you think it will perform on unseen test data? Why?

Project Report: You should write a report that includes the sections listed below. Your report should follow the IEEE transactions format (single spaced, double column). Focus your report on your training and testing strategies for the contest and any unique implementations. Templates for the IEEE transactions format can be found here:
http://www.ieee.org/publications_standards/publications/authors/author_templates.html

[1] J. M. Keller, M. R. Gray and J. A. Givens, "A fuzzy K-nearest neighbor algorithm," in IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-15, no. 4, pp. 580-585, July-Aug. 1985. doi: 10.1109/TSMC.1985.6313426

[2] H. Frigui and P. Gader, "Detection and Discrimination of Land Mines in Ground-Penetrating Radar Based on Edge Histogram Descriptors and a Possibilistic$K$-Nearest Neighbor Classifier," in IEEE Transactions on Fuzzy Systems, vol. 17, no. 1, pp. 185-199, Feb. 2009. doi: 10.1109/TFUZZ.2008.2005249

The maximum number of pages for the report is 4. If there are any pages beyond page 4, they will discarded and not read or graded. Be precise - use pseudo-code or equations to be precise.

Abstract A summary description of the contents of the report and your findings. This should provide a brief motivation of your project.
Introduction Overview of your experiment and a literature review. For the literature review, include any references to any relevant papers for your experiment. So, whatever you decide to do, search the ACM and IEEE (or other) literature for relevant papers to read and refer to.
Implementation Describe and outline any specific implementation details for your project. A reader should be able to recreate your implementation and experiments from your project report.
Experiments Carefully describe your experiments. How did you construct training, validation and test data sets? Include a description of what the goal each individual experiment is and what your findings are. This is the bulk of what you will be graded on - if your experimental design is not sound or your experiments do not make sense, you will lose points.
Conclusions Describe any conclusions or things you learned from the project. Your conclusions must follow from what you did. Do not copy something out of a paper or say something that has no experimental support in the Experiments section.
References Listing of all references in IEEE bibliography format.

Submission Details: Turn in your project report on Canvas and your code and report on GitHub. Be sure your code contains the following files: train.py (includes a *function* that will run your training code on an input data set X and return the location of red cars in Y. The output should be a numpy array containing the row- and column- location of each car. Any parameter settings must be easy to find and modify.), test.py (includes a *function* that will run your testing code on an input data set X. Your test.py code should already be trained. A run.py script that runs your training and testing functions on a small sample image (this script should not take longer than approximately 5 minutes to run) and does not require any inputs or adjustment to parameter settings (i.e., we will run the command python run.py and your script should execute without error). Any parameter settings must be easy to find and modify. It should return a vector with the location of each red car found in the input image X) and a concise README.txt file that clearly illustrates how to run your code.

Grade Details: Your grade will be determined using the following breakdown:

- 25% Code Implementation
  - Turn in code that runs correctly and easily on my machine. This requires a very clear README and easy to modify parameter settings. This also requires clearly listing what packages/libraries are needed to run your code - and checking with me before the due date to ensure I have those libraries. If the library is distributed by default with Anaconda, I have it.
  - Turn in code that follows the submission requirements described above
  - Your code should implement whatever approach you choose correctly and match the description in your project report.
- 75% Project report.
  - This component will be graded based on the requirements described above with the following breakdown:

∗ IEEE Formatting and Grammar: 5%

∗ Abstract and Introduction: 5%

∗ Implementation: 25%

∗ Experimental design, clear description of experiments and their results, and corresponding conclusions: 40%