Final Project: Enhanced Road Sign Detection

## Description

For this topic you will revisit road sign detection and classification using any of the techniques taught in this course. Your final algorithm must correctly identify road signs and traffic lights in real world images including images with adverse lighting, partial occlusions, and difficult weather conditions. Your code will be expected to take in an image and return a dictionary with road sign names and locations.  
  
Sample Dataset: <http://cvrr.ucsd.edu/LISA/lisa-traffic-sign-dataset.html> Related Lectures (not exhaustive): 2B, 4A-4C, 8A-8C.

## Problem Overview

**Methods to be used:** Implement a method that is capable of recognizing road signs and traffic lights using real world images. Unlike PS2 this project is not limited to the use of Hough-Tools.

**RULES**:

* **Don’t use external libraries for core functionality**You are encouraged to use libraries while writing code for your final report. However you will receive a low score if the main functionality of your code is provided via an external library.
* **Don’t copy code from the internet**The course honor code is still in effect during the final project. All of the code you submit must be your own. You may consult tutorials for libraries you are unfamiliar with, but your final project submission must be your own work. Any instance that does not follow the Honor Code and the class rules will be directly reported to the Office of Student Integrity.
* **Don’t use pre-trained machine learning models**If you choose a topic that requires the use of machine learning techniques, you are expected to do your own training. Downloading and submitting a pre-trained model is not acceptable for any project topic.
* **Don’t rely on a single source**We want to see that you performed research on your chosen topic and incorporated ideas from multiple sources in your final results. Your project must not be based on a single research paper and definitely must not be based on a single online tutorial.

**Please do not use absolute paths in your submission code. All paths must be relative to the submission directory. Any submissions with absolute paths are in danger of receiving a penalty!**

## Programming Instructions

You may use the python 2 environment that you have been using for all the assignments or the [python 3 environment provided for the project](https://gatech.instructure.com/courses/26111/files/folder/Project Files?cv_proj.yml). This new environment is simply a list of the versions of libraries that will be used during grading. You may install them however you wish. We recommend conda. Include a README.md file with usage instructions that are clear for the grader to run your code. Remember to specify what version of python you are using. Notice that despite having Tensorflow and Pytorch in the environment, you are not allowed to use them.

Windows Users Warning:

Be warned that TA’s grade exclusively on linux machines. Thus, it is your responsibility to make sure that your code is platform independent. This is particularly important when using paths to files. If your code doesn’t run during grading due to some incompatibility you will incur a heavy penalty.

## Write-up instructions

The report must be a PDF of 3-6 pages including images and references. Not following this requirement will incur in a significant penalty and the content will be graded up to page 6. **Note that the report will be graded subject to a working code.** There will be no report templates provided with the project materials.

The report must contain:

1. A clear and concise description of the algorithms you implemented. This description must include references to recently published computer vision research and show a deep understanding of your chosen topic.
2. Results from applying your algorithm to images or video. Both positive and negative results must be shown in the report and you must explain why your algorithm works on some images, but not others.
3. Performance statistics obtained by applying your algorithm to a public imagery or video database. You are expected to determine appropriate quantitative performance metrics based on your own research.
4. A technical discussion of how your results compare to the state of the art and how your results could be improved.

You report must be written to show off your work and demonstrate a deep understanding or your chosen topic. The discussion in your report must be technical and quantitative wherever possible.

## Video presentation:

Present your work in a video (screen, camera / cellphone recording) showcasing your work and results. This video must not be more than 3.5 minutes long (3:31 will not be accepted). **Videos longer than this will not be watched and therefore not graded.** Submit a link to your video hosted somewhere the grader can access it. We recommend providing multiple links in case one of them does not work or is removed. We will only use the link you provide at the time of submission. **Please do not send private posts with updated links as these will not be used.**

## How to submit

Unlike the class assignments, **you must submit your project files in one compressed (.zip) folder via Canvas.** Find the assignment labeled “Final Project”, attach all the required files, click on submit. Late submissions will not be graded so plan accordingly.

**Important: Submissions sent to Bonnie, Piazza or anything that is not Canvas will not be graded.**

## Grading

The report will be graded following the scheme below:

* Code (20%): We will verify that the methods and rules indicated above have been followed.
* Report (70%): Subject to a working code.
  + Description of existing methods published in recent computer vision research.
  + Description of the method you implemented.
  + Results obtained from applying your algorithms to images or videos.
  + Analysis on why your method works on some images and not on others.
  + Performance statistics analysis.
  + Discussion on how your results compare to the state of the art methods.
  + Proposals on how your methods can be improved.
  + References and citations.
* Video Presentation (10%)

## Assignment Overview

This project is an enhanced version of Problem Set 2. Unlike PS2 you are not restricted to only use Hough tools. As part of your research, you must study state of the art methods that perform similar tasks. This should give you an idea of how to build your road detection algorithm. It is up to you to define how to approach this problem following the project requirements.

### 1. Method Requirements

Your method must be able to perform traffic signs and traffic lights detection and classification using real images. Some sample images can be found in <http://cvrr.ucsd.edu/LISA/lisa-traffic-sign-dataset.html>. Your code must locate all traffic signs and traffic lights in an image, identify what type of sign they are, and annotate their location and label in an output image. This approach differs from PS2 in that it is limited to real images and must address different scenarios as explained in the next section.

### 2. Input Requirements

Implement a method that is able to process real world images and correctly identify road signs and traffic lights. These images must not be limited to ideal conditions and will include scenes with adverse lighting, partial occlusions, and difficult weather conditions. All input images cannot be a single cropped road / traffic sign, they must be a scene including other objects such as roads, sidewalks, cars, etc.

### 3. Final Results

#### 3.1 Road signs description in the scene

For each input image, your code will be expected to process an image and return a dictionary with the road sign names and locations.

#### 3.2 Output images

Render each road sign position and label on the output image. This visualization must be readable by the grader. It is recommended that a dark background is used where annotations are placed. Make sure you present all cases mentioned in the project instructions.