

# **Computer Vision**

## **Spring 2020**

# **Problem Set #2**

Andrew Samuel Parmar  
[aparmar32@gatech.edu](mailto:aparmar32@gatech.edu)

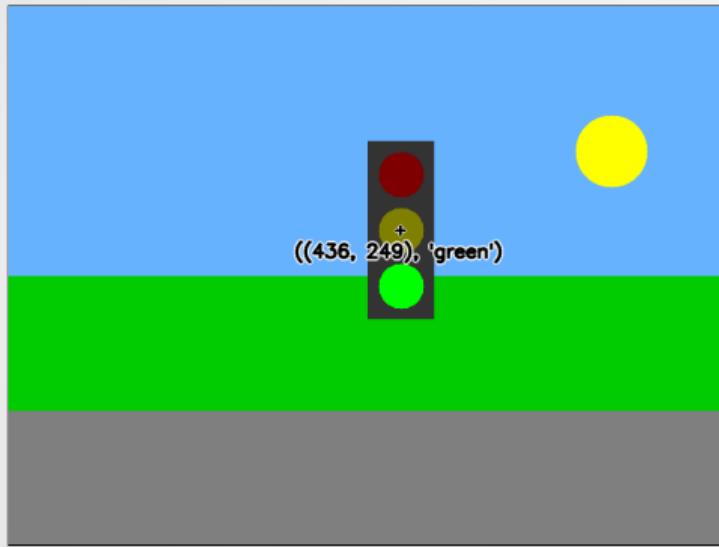
# Traffic Light Detection



Coordinates and State:  
(135, 121), color: green

ps2-1-a-1

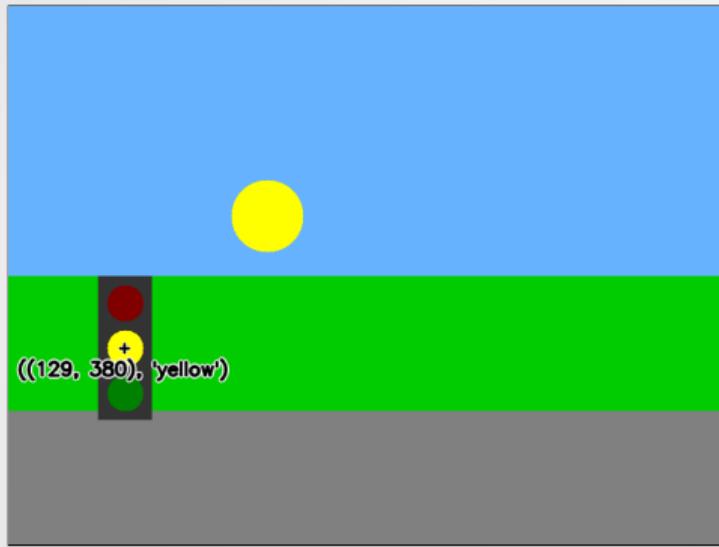
# Traffic Light Detection



Coordinates and State:  
(436, 249), color: green

ps2-1-a-2

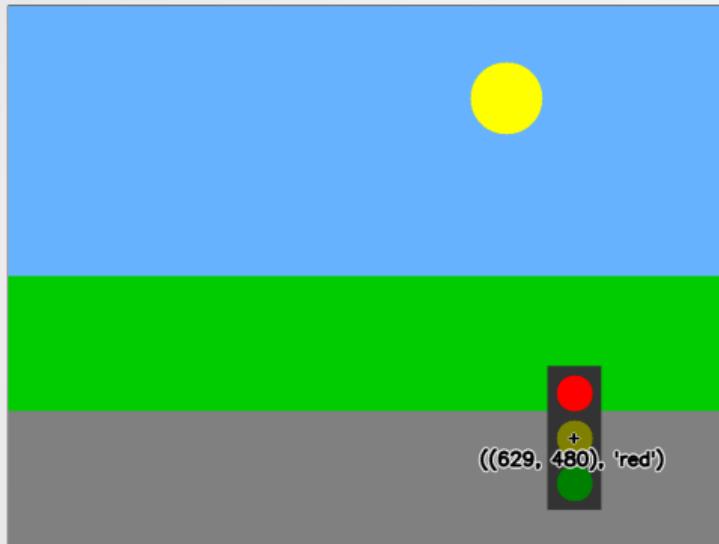
# Traffic Light Detection



Coordinates and State:  
(129, 380), color: yellow

ps2-1-a-3

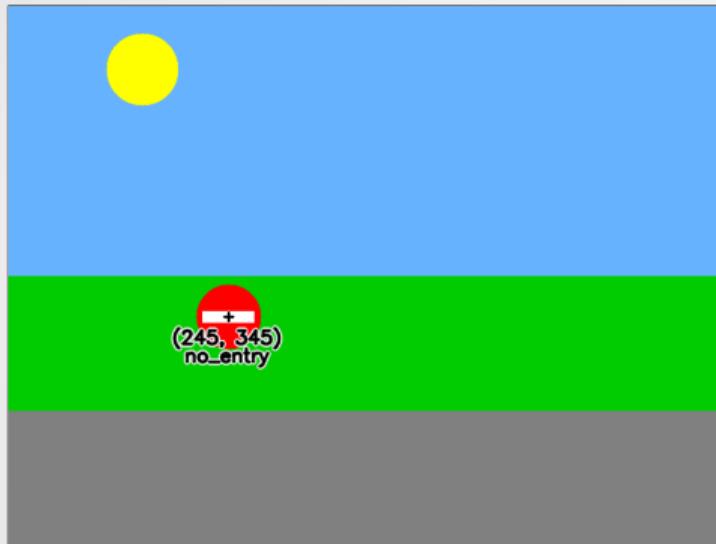
# Traffic Light Detection



Coordinates and State:  
(629, 480), color: red

ps2-1-a-4

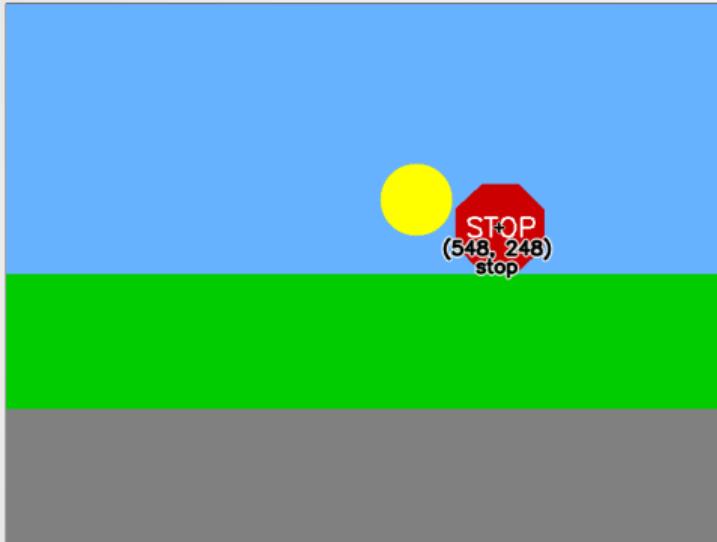
# Traffic Sign Detection - Do Not Enter



Coordinates:  
(245, 345)

ps2-2-a-1

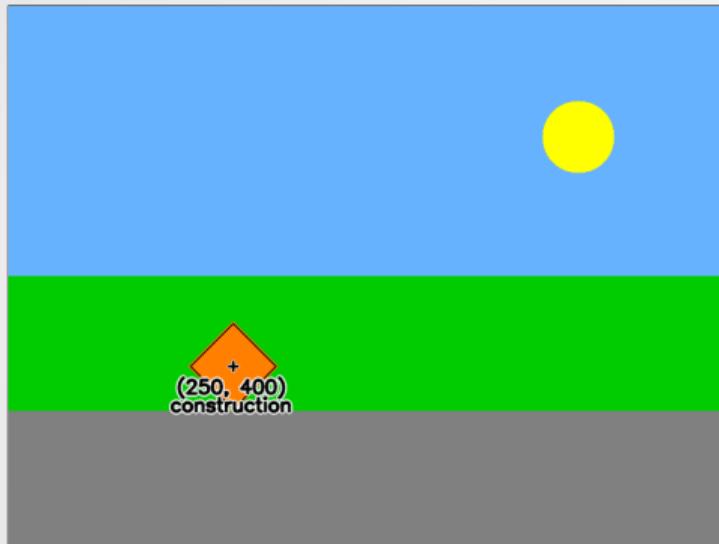
# Traffic Sign Detection - Stop



Coordinates:  
(548, 248)

ps2-2-a-2

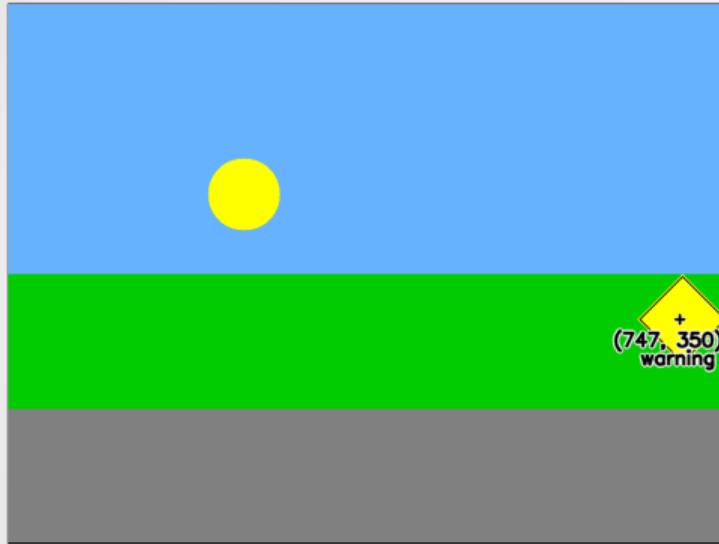
# Traffic Sign Detection - Construction



Coordinates:  
(250, 400)

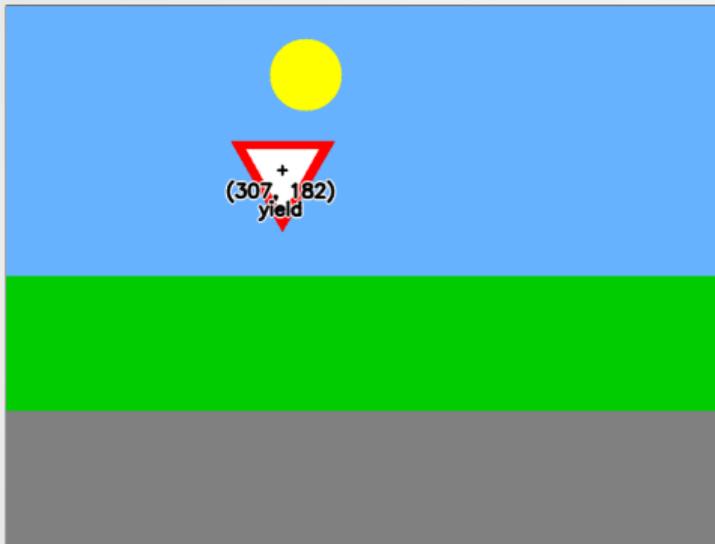
ps2-2-a-3

# Traffic Sign Detection - Warning



ps2-2-a-4

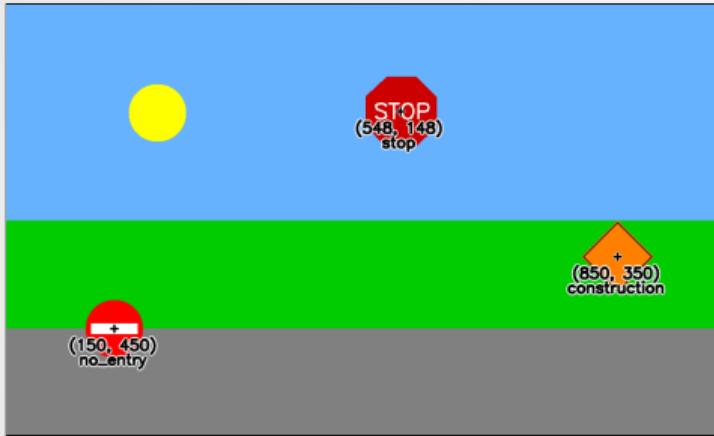
# Traffic Sign Detection - Yield



Coordinates:  
(307, 182)

ps2-2-a-5

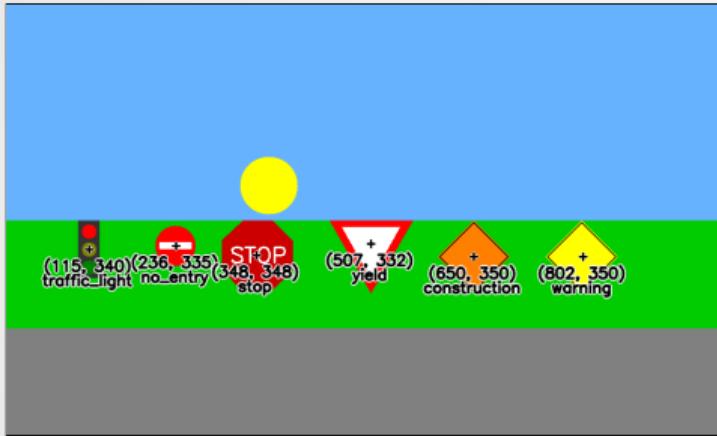
# Multiple Sign Detection



Coordinates and Name:  
No Entry: (150, 450)  
Stop: (548, 148)  
Construction: (850, 350)

ps2-3-a-1

# Multiple Sign Detection



Coordinates and Name:

Traffic Light: (115, 340)

No Entry: (236, 335)

Stop: (348, 348)

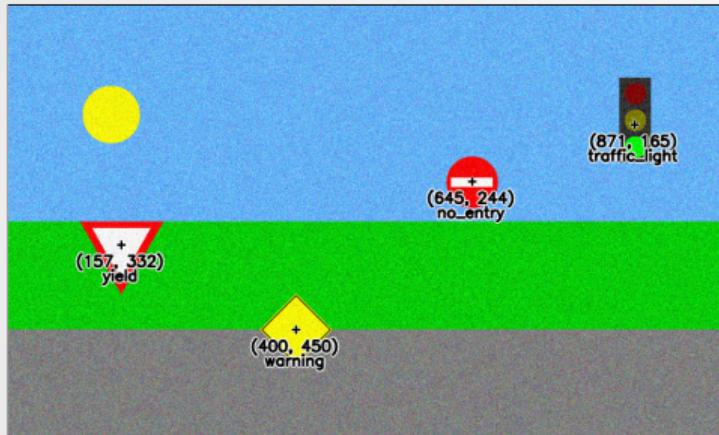
Yield: (507, 332)

Construction: (650, 350)

Warning: (802, 350)

ps2-3-a-2

# Multiple Sign Detection With Noise



Coordinates and Name:

Yield: (157, 332)

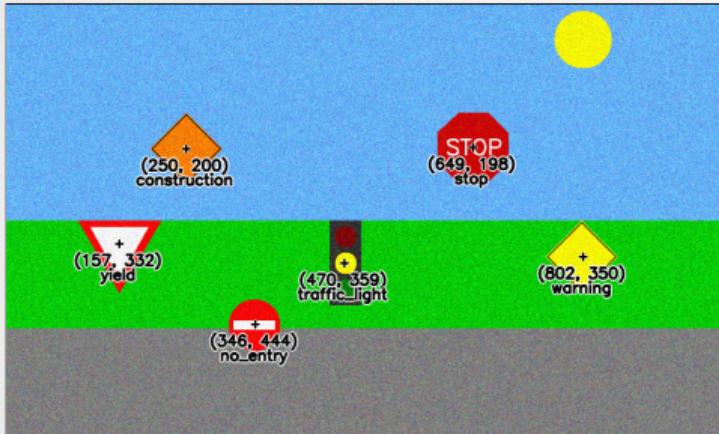
Warning: (400, 450)

No Entry: (645, 244)

Traffic Light: (871, 165)

ps2-4-a-1

# Multiple Sign Detection With Noise



Coordinates and Name:

Yield: (157, 332)

Construction: (250, 200)

No Entry: (346, 444)

Traffic Light: (470, 359)

Stop: (649, 198)

Warning: (802, 350)

ps2-4-a-2

# Challenge problem - A



Coordinates and Name:  
Warning: (353, 351)

ps2-5-a-1

# Challenge problem - A



Coordinates and Name:  
Stop: (290, 498)

ps2-5-a-2

# Challenge problem - A



Coordinates and Name:  
Construction: (472, 340)

ps2-5-a-3

# Challenge problem - B



Coordinates and Name:

Warning: (328, 322)

Stop: (327, 377)

Traffic Light (false +ve): (589, 458)

ps2-5-b-1

# Challenge problem - B



ps2-5-b-2

Coordinates and Name:  
Warning: (345, 151)  
Traffic Light (false +ve): (508, 278)

# Challenge problem - B



Coordinates and Name:  
Stop: (396, 339)  
Warning: (690, 319)

ps2-5-b-3

# Challenge problem - Text

Describe what you had to do to adapt your code for this task. How does the difference between simulated and real-world images affect your method? If you used other functions/methods, explain why that was better (or why your previous implementation did not work)

Color masking is one of the primary stages of my detection pipeline. With the simulated images, I could dial in a color code and filter only for a desired color space. With real images, colors are no longer fixed within a tight tolerance spec. A stop signs color can be different shades of red, depending on lighting conditions, and the weathering effects on the sign. Due to this, I had to increase my tolerance ranges by a significant margin to allow for filtering the required signs. This has the negative effect of also retaining aspects of the pictures that are not the traffic sign we are interested in detecting.

After the color filter, my primary approach was to use cv2.HoughLinesP to get as many lines as possible in the image. I used the min and max x, y values to get the corners of the bounding box which led to determining the centroid. This method worked for simulations where the colors were controlled and localized. Within the real images this technique caused the centroid detection to be off if colors similar to the sign were present in other areas of the picture. An area of improvement would be to use the sign angle information to restrict the line space, while also taking into consideration rotation of the sign.

Image sources:

- img-5-a-1.png - Bob Kerner <https://bit.ly/2uX9JAb>
- img-5-a-2.png - New Jersey Hills Media Groyp <https://bit.ly/31djPco>
- img-5-a-3.png - Niagara Now <https://bit.ly/31e9EnX>
- img-5-b-1.png - Shutterstock <https://shutr.bz/2vAGzHr>
- img-5-b-2.png - Bob Kerner <https://bit.ly/2uX9JAb>
- img-5-b-3.png - Alamy <https://bit.ly/2SbBywP>