U UDACITY

PROJECT SPECIFICATION

Search and Sample Return

Writeup

CRITERIA	MEETS SPECIFICATIONS
Provide a Writeup / README that includes all the rubric points and how you addressed each one. You can submit your writeup as markdown or pdf. Here is a template writeup for this project you can use as a guide and a starting point.	The writeup / README should include a statement and supporting figures / images that explain how each rubric item was addressed, and specifically where in the code each step was handled.

Notebook Analysis

CRITERIA	MEETS SPECIFICATIONS

CRITERIA	MEETS SPECIFICATIONS
Run the functions provided in the notebook on test images (first with the test data provided, next on data you have recorded). Add/modify functions to allow for color selection of obstacles and rock samples.	Describe in your writeup (and identify where in your code) how you modified or added functions to add obstacle and rock sample identification.
Populate the `process_image()` function with the appropriate analysis steps to map pixels identifying navigable terrain, obstacles and rock samples into a worldmap. Run `process_image()` on your test data using the `moviepy` functions provided to create video output of your result.	Describe in your writeup how you modified the `process_image()` to demonstrate your analysis and how you created a worldmap. Include your video output with your submission.

Autonomous Navigation and Mapping

CRITERIA	MEETS SPECIFICATIONS

CRITERIA	MEETS SPECIFICATIONS
Fill in the `perception_step()` (at the bottom of the `perception.py` script) and `decision_step()` (in `decision.py`) functions in the autonomous mapping scripts and an explanation is provided in the writeup of how and why these functions were modified as they were.	`perception_step()` and `decision_step()` functions have been filled in and their functionality explained in the writeup.
Launching in autonomous mode your rover can navigate and map autonomously. Explain your results and how you might improve them in your writeup.	By running `drive_rover.py` and launching the simulator in autonomous mode, your rover does a reasonably good job at mapping the environment. The rover must map at least 40% of the environment with 60% fidelity (accuracy) against the ground truth. You must also find (map) the location of at least one rock sample. They don't need to pick any rocks up, just have them appear in the map (should happen automatically if their map pixels in `Rover.worldmap[:,:,1]` overlap with sample locations.) **Note: running the simulator with different choices of resolution and graphics quality may produce different results, particularly on different machines! Make a note of your simulator settings (resolution and graphics quality set on launch) and frames per second (FPS output to terminal by** `drive_rover.py` **) in your writeup when you submit the project so your reviewer can reproduce your results.**

Suggestions to Make Your Project Stand Out!

To have a standout submission on this project you should not only successfully navigate and map the environment in "Autonomous Mode", but also figure out how to collect the samples and return them to the starting point (middle of the map) when you have found them all!

Student FAQ