Project: Rover Search and Sample Return

By Lingzhi Kong

Udacity Robotics Nano Degree

Sep. 19, 2018

Abstract

This project aims to program a rover to navigate autonomously in simulation and create a map of its environment while searching for rock samples of interest. In particular, first we implement perspective transformation, which will convert image seen by the rover camera into a top-view image that will be used to map the navigate area, obstacle area, and golden rocks. Second we used color threshold to identify the three areas. Third we convert the top view pixels coordinates that was generated from the previous perspective to rover centric coordinates. Finally, we map the navigable terrain from rover centric coordinates to world coordinates to create our map.

.

2 Details

2.1 Perception

2.1.1 Perspective transformation

The goal of perspective transformation is convert the camera image coming from the rover into the top-view image. To do this, we used a reference grid. To perform the perspective transform, we will identify the four corner points of a grid cell in the field of view of the camera, which will be our source points. Then, we will define four destination points which represent the same grid cell in a top down view of the world. With our source and destination points defined, we are able to use OpenCV to perform a perspective transform. This operation will transform our rover camera images into a top-down perspective or map perspective. The below figure shows the result of the perspective transform.

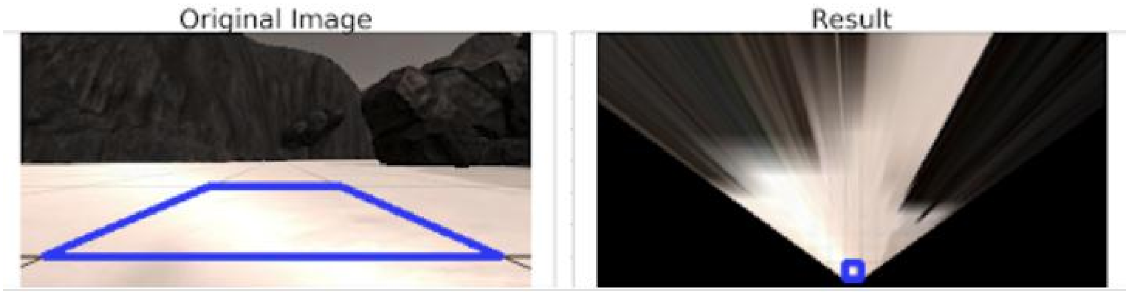

Fig. 2.1 perspective transform

2.1.2 Color Threshold

By applying color threshold function, we will be able to identify the navigable, obstacle, and golden rock areas. For navigable thresholding, we used rgb_thresh=(160, 160, 160) as the threshold value. For golden rocks, we used rgb_levels=(110, 110, 50).

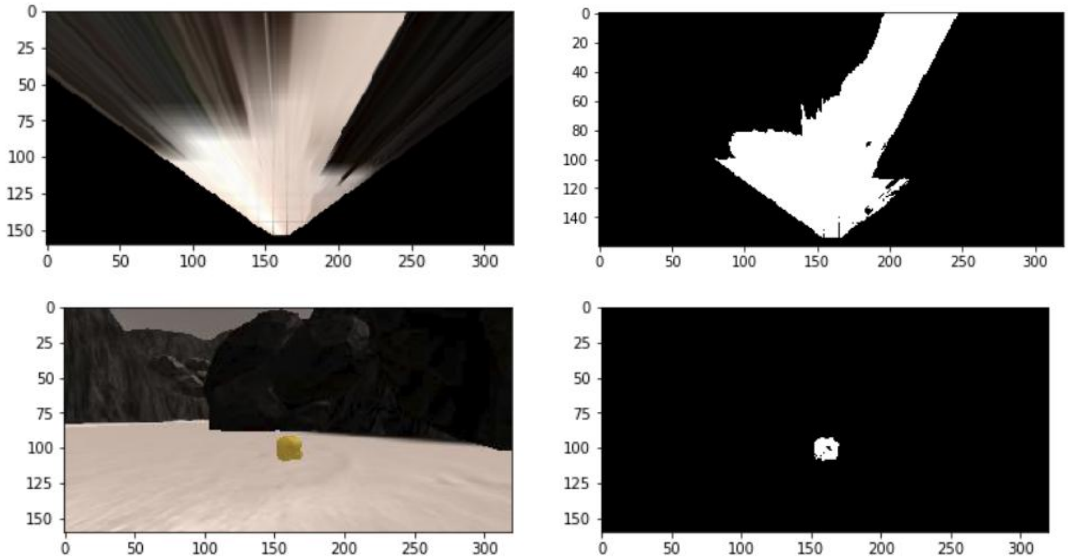

Fig. 2.2 Color Thresholding (navigable terrain & golden rocks)

2.1.3 Rover Centric Coordinates Functions

In order to describe the positions of objects in the environment with respect to the rover’s camera, we need to convert the top view pixels coordinates that was generated from the previous perspective to a rover centric coordinates. This can be done by removing the x, y position of the rover and rotating the image based on the robot yaw.

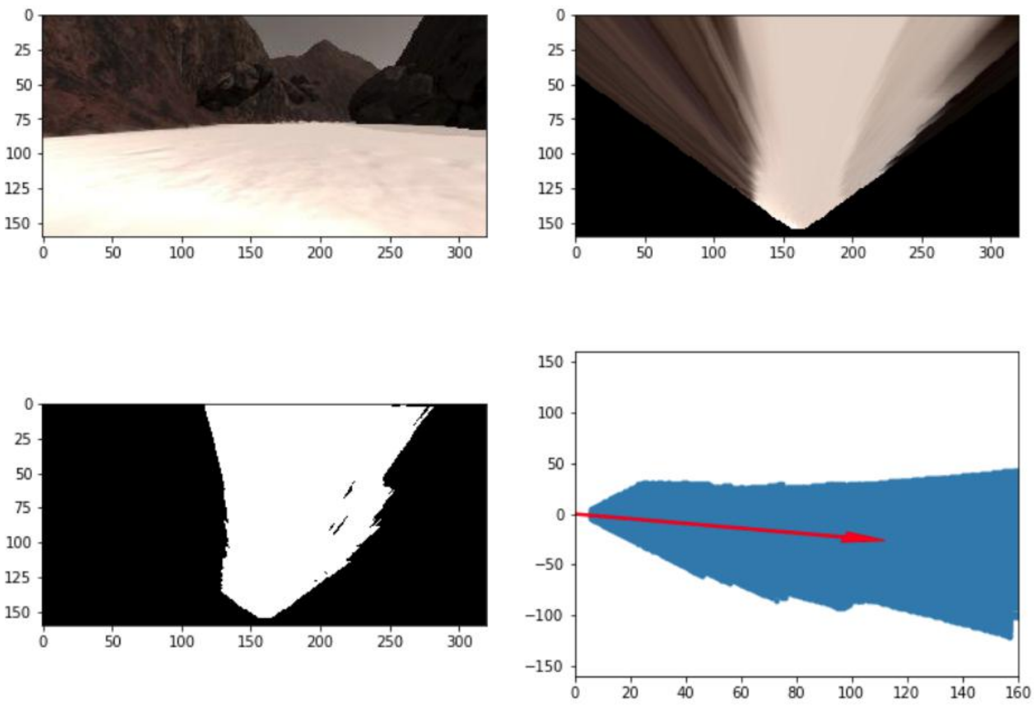

Fig. 2.3 Rover Centric Coordinates Transform

2.1.4 Map to World Coordinates

Now that we have the navigable areas from the rover centric coordinates, we are able to apply the processed image to the world map to mark navigable/obstacle/rock areas.

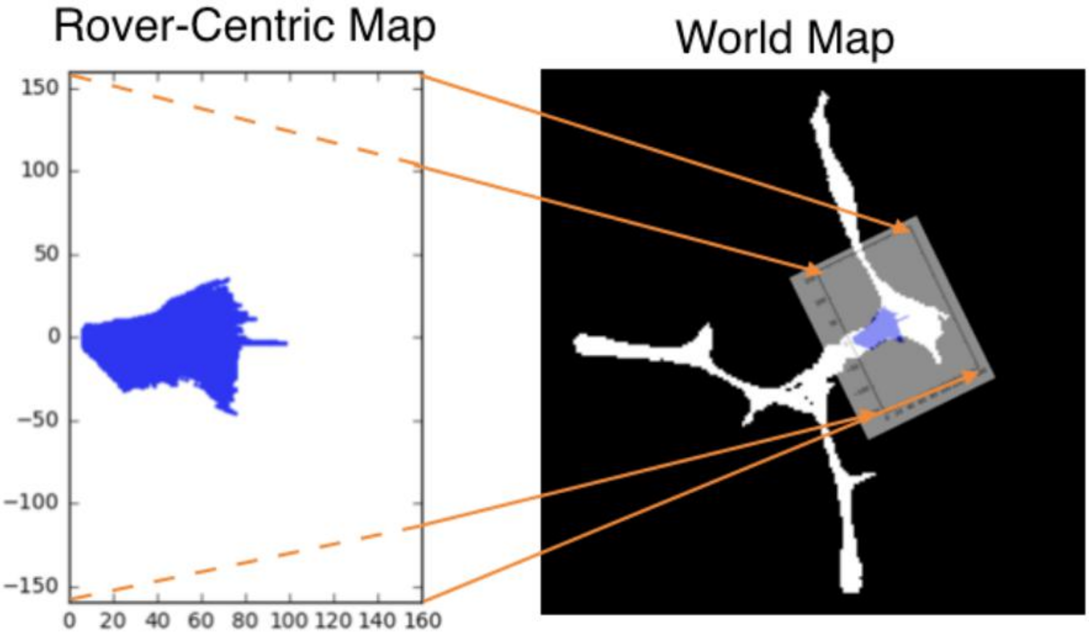

Fig. 2.4 Map to world coordinates

2.1.5 Image Processing Pipeline

Now, we have completed all components needed for perception. Next, we will populate the process_image() function. The pipeline is shown in below figure.

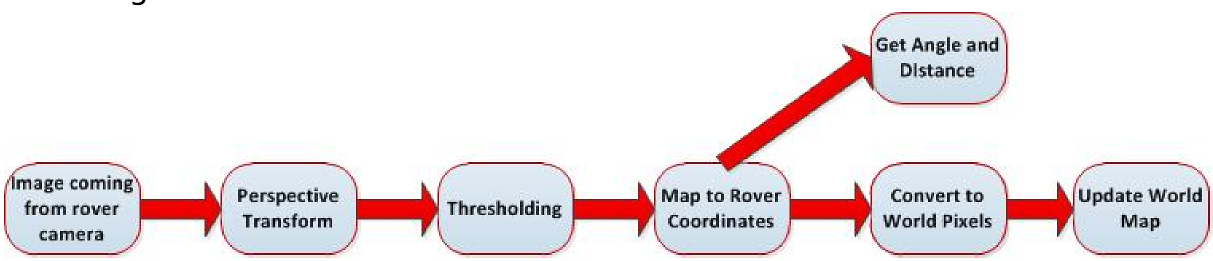

Fig. 2.5 Image processing pipeline

2.1.6 Mapping video generation

Previously, we have updated the world map.Next, we will be able to generate a mapping video. First we create output image consisting of original image, warped image and worldmap. Then we create video using moviepy function. The upper left of the video is the original image, the upper right of the video is the warped image, and the bottom left is the worldmap. The below figure shows a screenshot of the video.

Fig. 2.6 Mapping video generation

2.2 Decision making and actuation

2.2.1 Features extraction

Rover must be able to decide what to do after receiving the inputs. After deciding what to do, rover will actuate. In particular, the decision-actuation process involves a loop of steps:

1. Obtain image from the rover camera;
2. Process the image to get the coordinates and direction where it would move towards;
3. Based on the possible "place and direction to go" and on its current state(moving, stopped of stuck), rover will decide about:
   a) Keep moving
   b) Stop moving
   c) Spin
   d) Pick up a rock sample
4. Rover will obtain a new state based on:
   a) Current position, speed
   b) Its status
   c) It sensorial data

5. Update statistics about:
    a) Map
    b) Rocks

3 Improvement & Discussion

5.1 Improvement

1) Implement path planning algorithm in order to return to the original point.

2) Instead of processing images by RGB model, we can process in HSV model.

3) We can also consider using deep reinforcement learning method to do the navigation.