University of
Hawaii at Hilo
Fall 2016

Developing a Computer Vision System for Autonomous Rover Navigation

ASTR 432 | FINAL REPORT | FALL 2016

DARYL ALBANO

Table of Contents

Abstract 2
Introduction 3
Background 3
Project Findings 3
Next Steps 4
Conclusion 5

Abstract

The NASA Robotic Mining Competition is hosted every year, in which several universities design, build, and compete in planetary mining competition. The competition consists of two 10-minute rounds where the goal is to mine and retrieve the most regolith as quickly and efficiently as possible. Thus, the project for this semester is targeted towards improving the Vulcan Robotics Team's competition rover, as well as design an autonomous navigation system.

The project involves development on a miniature prototype of the competition rover. This includes similar movement functions and on-board computers. However, the main program for the navigation system has switched from the previous programming language, C, to Python. Additionally, the Arduino microcontroller has been removed in favor of the Raspberry Pi.

Project findings have shown that the computer vision system can detect edges of objects however, there is difficulty distinguishing the edges of objects with the edges of shadows. With limitations under various lighting environments, a different design was introduced in which the camera would follow a reference object for navigation. Using a reference object removes the issue of misread information. From there the video data can be manipulated to include the size of the object, as well as the color of the object.

Overall, the project has opened opportunities for other forms of navigation systems. Improvements were made in the navigation and communications portion of the rover, which should optimize data transfer size and power usage. Some downfalls did occur with the Raspberry Pi, involving networking issues and software issues. Solutions were also composed during the construction of the prototype. With this project, it has provided better insights on software development of a rover and has established future iterations of the autonomous navigation system.

Introduction

The goal of this project is to learn and understand effective methods of rover communication, as well as develop a computer vision system for autonomous rover navigation. The research would facilitate the development of a much more efficient planetary rover for the 2017 NASA Robotic Mining Competition. The competition, hosted by NASA, favors autonomy in the competition. Thus, the project's goal is to begin research and development of an autonomous system. Having a rover that can continue operations without a user would greatly improve exploration and retrieval of resources on planets like Mars.

Some of the issues that plagued previous versions of the rover include communication issues and a lack of autonomy. After several months of research and implementations, improvements were made in reducing network overhead and user control, as well as a better understanding of computer vision.

Background

OpenCV is an open source computer vision software that is used to translate the video data into usable information. The software analyzes objects in the real world and translates the data. For example, the software is capable of outlining objects and recognizing human faces.

For rover navigation, the computer vision system will search for an object and follow it. Depending on the object's position, it will drive the motors to follow the item. For example, if the object being searched for is a green ball, its distinct color and shape will make it easy to identify. In practical applications, such as a foreign planetary environment, a barcode or a quick response code will take the place of the green ball.

From there, an initial prototype was developed that utilizes two 6V DC motors, a Raspberry Pi 3 Model B computer, and an L298N H-Bridge motor controller. Its primary purpose was to replicate the full planetary rover's basic navigation system. In other words, the turning methods share similarities between both platforms.

Project Findings

What the project did succeed in was improving communications by replacing the original secure shell (SSH) connection with virtual network computing (VNC). Connection to the Raspberry Pi Computer has also been improved by configuring it as a network access point. Thus, the command center can directly connect to the Raspberry Pi without needing access to a separate access point, improving connection speed and reliability.

The computer vision system uses an open source software called OpenCV. This software provides several libraries that allows for video data from an input device to be manipulated for various tasks. An example would be face detection and color manipulation. The reason behind selecting OpenCV was for its focus on software. OpenCV utilizes multiple programming languages (C++, Python, Java) and is available on multiple operating systems (Windows, MacOS, Android, iOS). Since the desktop webcam on the initial prototype has certain limitations (providing only vision), development was focused on software. This does pose an issue since OpenCV utilizes more resources, resulting in slower performance on the Raspberry Pi computer. Methods of reducing computer lag can be achieved with improved vision algorithms. Thus, the focus for upcoming iterations is to research and develop efficient software.

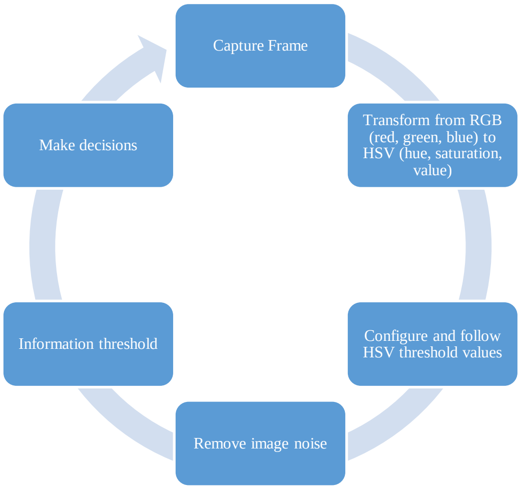
Capture Frame
Transform from RGB (red, green, blue) to HSV (hue, saturation, value)
Configure and follow HSV threshold values
Remove image noise
Information threshold
Make decisions

Figure 1 Diagram demonstrating the process for detecting a reference object

During the development of the initial prototype, much time was spent debugging and fixing issues, leaving limited time for navigation testing. The results show that although the Raspberry Pi can be efficient in cooling and energy consumption, it is limited in processing capabilities. Although other options, like the Intel NUC computer or Android mobile device, offer better performance, a Raspberry Pi is still a viable option since it provides an inexpensive Linux-based environment for computer vision development. With much needed improvements in software and OpenCV, there is potential in using computer vision with the navigation system.

Next Steps

With limitations in hardware, methods of improving software will be the primary focus for the upcoming semester. The idea is that the vision system locates a reference device that the computer could easily identify, such as a green ball or a quick response code. It could potentially identify its current location by determining the distance of the reference object based on its relative size, resulting in both distance and location awareness.

The project findings have shown that computer vision can be difficult to implement with multiple factors to consider. Given the various lighting environments, the vision system would obtain inaccurate information. A possible solution would be to develop a program that contains exception handling methods for misread information. As stated previously, the implementation of a reference object could potentially improve the overall vision system by having an object that the rover would distinctly identify.

Conclusion

In conclusion, development on the computer vision system has led to new discoveries and improvements for the current planetary rover. In terms of navigation and communication, it has become much more streamlined compared to previous iterations. Although the computer vision system did not achieve its original goal, much of the research and development leaves much potential for other navigations methods. Thus, having a better understanding would lead to better iterations in the future. Overall, the project has succeeded in improving the current rover and to continue development of an autonomous navigation system.

References

[1] D. A. Forsyth, Computer Vision: A Modern Approach (2nd Edition), Pearson, 2011.

[2] L. Matthies, M. Maimone, A. Johnson, Y. Cheng, R. Willson, C. Villalpando, S. Goldberg and A. Huertas, "Computer Vision on Mars," *International Journal of Computer Vision,* pp. 67-92, 2007.

[3] "The R.R.O.P. - RaspRobot OpenCV Project - All," Autodesk Inc., 2016. [Online]. Available: http://www.instructables.com/id/The-RROP-RaspRobot-OpenCV-Project/?ALLSTEPS. [Accessed 24 08 2016].