Stereoscopic Vision for Autonomous Navigation

Functional Description and
Complete System Block Diagram

By:
John Hessling

Advisor:
Dr. Huggins

November 03, 2009

Introduction

There are many forms of navigational aids that can be used in autonomous navigation. There are sonar sensors that can measure distance to the nearest object in a cone of direction. Then there are laser sensors that are more precise, have a smaller cone of direction. However, these are much more expensive and still cannot distinguish between objects. Then there is mono imaging that consists of one camera. This will provide a clear view of objects in the path, but does not provide information regarding depth. Therefore to combine the best traits of all of these methods, two cameras can be used and the pictures merged to provide detailed information. This is called Stereoscopic Vision. This system has been used on mars rovers and experimental cars.

Goals

- Understand the complexities of Stereoscopic vision
- Create a working prototype Vision system
- Produce a self-contained Stereo Vision unit for use on autonomous vehicles

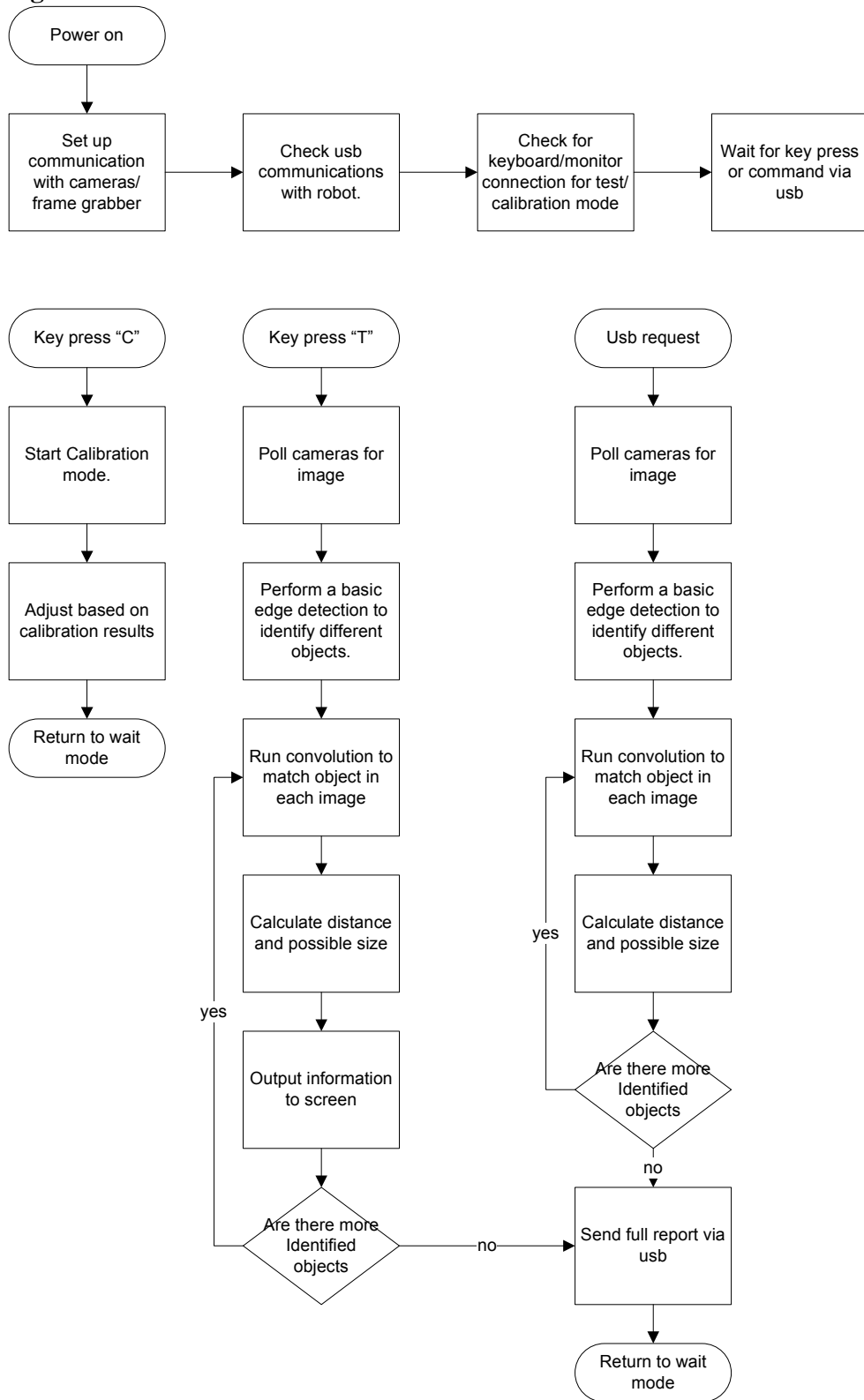**Functional Description and Block Diagram**

The system will consist of only two digital cameras mounted on an adjustable rig. This rig must be capable of making directional adjustments during calibration and then maintain position for the duration of use. There will also be a desktop/laptop computer for the processing power needed. Since the implementation will not be mobile a desktop computer will be used for this project. However, the final project will be a self-contained unit that will require a built-in processing unit. The basic hardware will be of a setup of the two mounted cameras feeding information through a frame grabbing device into a computer as shown in figure 1. The output will be sent through the usb connection for normal run mode. While in calibration mode or in testing mode, communication will be from keyboard and to the screen. Testing mode will still send a copy of the information through the usb connection but only for use in testing.

Figure 1
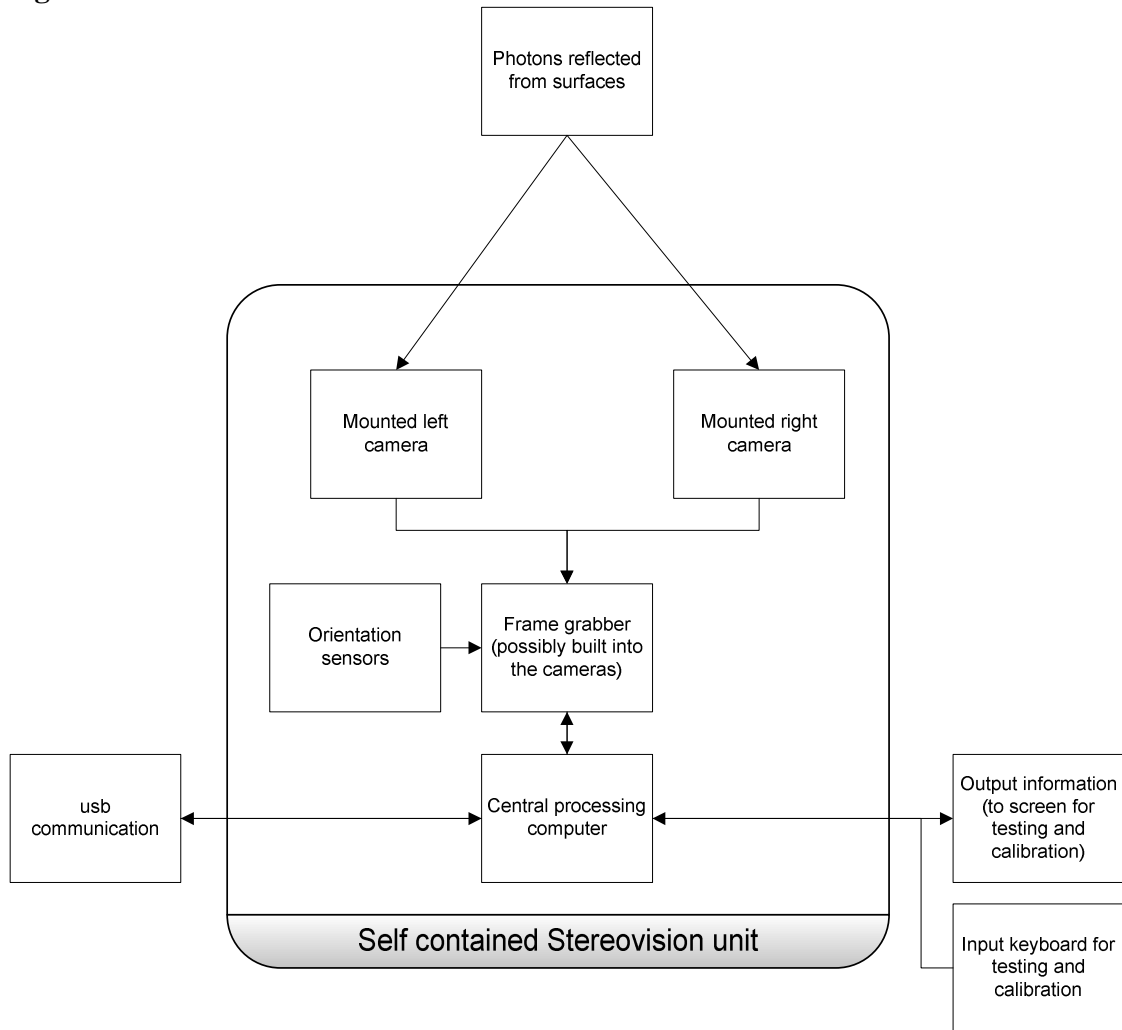
The system will consist of three different modes: Calibration, Testing, Run mode. This is illustrated in figure 2 shown below.

Figure 2

Power on

↓

| Set up communication with cameras/ frame grabber | → | Check usb communications with robot. | → | Check for keyboard/monitor connection for test/ calibration mode | → | Wait for key press or command via usb |

Key press "C"

↓

Start Calibration mode.

↓

Adjust based on calibration results

↓

Return to wait mode

Key press "T"

↓

Poll cameras for image

↓

Perform a basic edge detection to identify different objects.

↓

Run convolution to match object in each image

↓

Calculate distance and possible size

↓

Output information to screen

↓

Are there more Identified objects — yes

Usb request

↓

Poll cameras for image

↓

Perform a basic edge detection to identify different objects.

↓

Run convolution to match object in each image

↓

Calculate distance and possible size

↓

Are there more Identified objects — yes

no

Are there more Identified objects — no → Send full report via usb

no

Send full report via usb

↓

Return to wait mode

Calibration mode

The calibration mode is designed to ensure that the measurements taken in later modes will be accurate. There will be two parts to the calibration mode. The first being a basic camera alignment and this will be to ensure the cameras are pointing directly forward. Properly aligned cameras will ensure the largest field of stereo vision. The adjustments will be made onto the physical camera mounts by the user. This portion of the calibration can be skipped by user override. The second will require a special calibration object to be centered three meters away from the cameras. The system will then capture a picture from both cameras. Then use this information to adjust color differences and distance calculations. The user will be asked to accept changes to adjustments. If the user declines the previous calibration will be used.

Testing mode

Testing mode will be used mostly for development and proof of operation. This mode will perform all the same tasks as Run mode, but will also display all the information to the user screen for verification. The raw images will be saved in addition to the intermediate processing steps. This will allow for full testing, development, and debugging. These images will not need to be outputted during Run mode. Therefore, this will increase the overall speed of the stereo vision unit.

Run mode

Run mode will be the main operation mode for the stereo vision unit and will only be accessible through usb connection. The run request and the desired format request will be sent to the stereo vision unit. If a desired format is not sent a default format will be selected. The stereo vision unit will grab the images from the two cameras and retrieve orientation information from the sensor. The images will then be processed to extract object information from the field of vision. This will be adjusted based on the sensor orientation information. The object information will be formatted as requested and sent back via usb connection. Although speed is not the primary concern for this project, every effort will be made to minimize the time delay between request and completion. To this end one request mode will include a less accurate but faster format.

Conclusion

There are several uses for Stereoscopic vision. This project will create a useful self-contained unit that will perform stereo vision and relay the information back. Future autonomous projects will be able to utilize this unit as a drop-in system knowing its ability and reliability.

References:

[1] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1/2/3):7-42, April-June 2002. Microsoft Research Technical Report MSR-TR-2001-81, November 2001. http://vision.middlebury.edu/stereo/taxonomy-IJCV.pdf

[2]