VISION BASED AUTONOMOUS SECURITY ROBOT

FUNCTIONAL DESCRIPTION
AND
COMPLETE SYSTEM BLOCK DIAGRAM

By:

Kevin Farney

Project Advisor:

Dr. Joel Schipper

Submitted to:

Dr. Winfred Anakwa

EE 451 Senior Capstone Project

November 10, 2009

INTRODUCTION

Computer vision is defined as making useful decisions about real physical objects and scenes based on sensed images [4]. Humans generally take the details of eyesight and the amount of information naturally derived from sight for granted. Machine vision incorporates artificial (not necessarily intelligent) systems with image processing systems. The design of this project is to develop an autonomous vehicle with computer vision its primary sensor for gaining information about its environment.

For most intelligent machines (vehicles) the capability of understanding the visual world is a prerequisite. Typically autonomous vehicles have many different types of sensors all operating simultaneously. This project will use a camera (such as a webcam) for navigation and obstacle avoidance as well as other tasks the robot must perform.

Applications for this type of technology are numerous. This particular design will be used as a security robot. At the most complex level this autonomous vehicle will receive a signal indicating that a disturbance in a particular room has been discovered. The Vision Based Autonomous Security Robot (VBASR) will then travel to that room and investigate, taking pictures of any culprits that it locates there. After the task has been completed it will return to its charging station.

PROJECT GOALS

- Locate and navigate down the center of a corridor.
- Avoiding obstacles in real time in addition to locating and navigating down the center of a corridor.
- Map the hallway using images. Also, recognize locations (rooms/offices) using pre-defined images of those locations. Point-to-point navigation into rooms.
- Security Application. Locate and take a picture of an intruder. Theoretically, lower-cost sensors (motion detectors) would give the VBASR a location to investigate and then the VBASR would locate and take a picture of the intruder.

SYSTEM BLOCK DIAGRAM

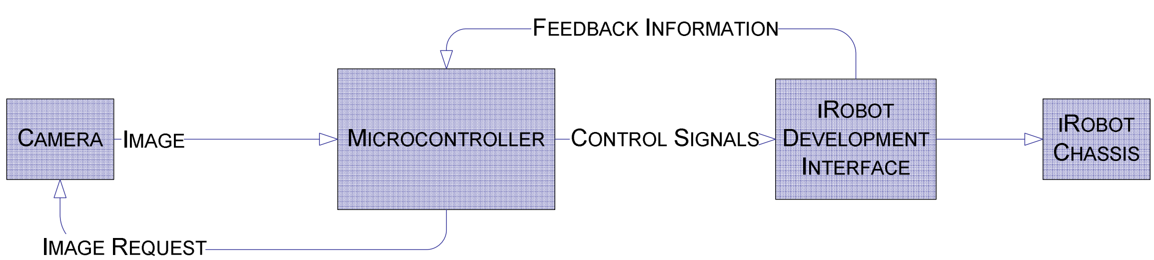

FIGURE 1 – COMPLETE SYSTEM BLOCK DIAGRAM

FUNCTIONAL DESCRIPTION

Due to the unique nature of this project, many of the specifications have yet to be determined. Upon a more complete understanding of the intricacies of computer vision, a much more educated decision can be made regarding exactly which components are necessary and how they will interface together. Generally robots follow the sense-plan-act process. For this application the camera does the sensing, microprocessor does the planning, and the chassis reacts. The following are the descriptions of each component found in Figure One.

CAMERA – This is the primary sensor for the robot. The camera will be continually taking images for the microcontroller to analyze. These images will be still shots, not streaming video. Upon completion of analysis of an image, the next image will be requested while the rest of the system operates as the first image dictates. The following are the possible cameras at this point: DSK-EYE gigabit, CMUcam3 SDK, and SRV-1 Blackfin Camera. Each of these cameras has processing power built in. It has not been determined if this type of elaborate camera is necessary or if a simple webcam is all that is required.

MICROCONTROLLER – The microcontroller is the central nervous system of this project. It will have the computer vision programs uploaded that enable it to process the images from the webcam. Based upon the information from the images, it will dictate to the iRobot exactly what the iRobot needs to do (see the Software Flowchart section). This microcontroller will also continually request images from the webcam and store them as necessary. The best microcontroller for this project is still to be determined. A laptop computer (connected to the iRobot via Bluetooth) may be used for the processing power.

IROBOT OPEN (DEVELOPMENT) INTERFACE – An accessory necessary for this project is the iRobot Create command module. This module consists of an electronic interface (the physical component) and a software interface. The hardware consists of a 7 pin Mini-DIN connector and a DB-25 connector for connecting external hardware (a camera for example). Through the software, the robot's behavior and various sensors can be read via various commands. The command module is powered by an Atmel AVR ATMega168 reprogrammable microcontroller. This microcontroller may be used as the microcontroller described above depending upon its capabilities. Using C or C++ personalized programs can be downloaded and implemented with this development interface.

IROBOT CHASSIS – In Figure 2 the iRobot chassis is pictured along with several of the accessories. The actual chassis of the VBASR (white circular bot) contains the following: Omni directional IR receiver, serial port, control buttons, wheels, batteries, a cargo bay, DB-25 connector, 6 mounting cavities and 4 cliff sensors. Each component in total makes up the iRobot Create unit. The camera and (possibly) external microcontroller will be mounted atop the

iRobot Create. This chassis is controlled by the iRobot Open Interface via the command module (green unit in Figure 2) or simply a serial cable. Also included with the iRobot Create Premium is the following: two virtual walls (gray top-right), a self-charging home base with AC/DC converter (gray top-center), standard remote (black left-center), and a rechargeable battery (yellow) all shown in Figure 2 [2].

IROBOT PREMIUM DEVELOPMENT PACKAGE

SOFTWARE FLOWCHART

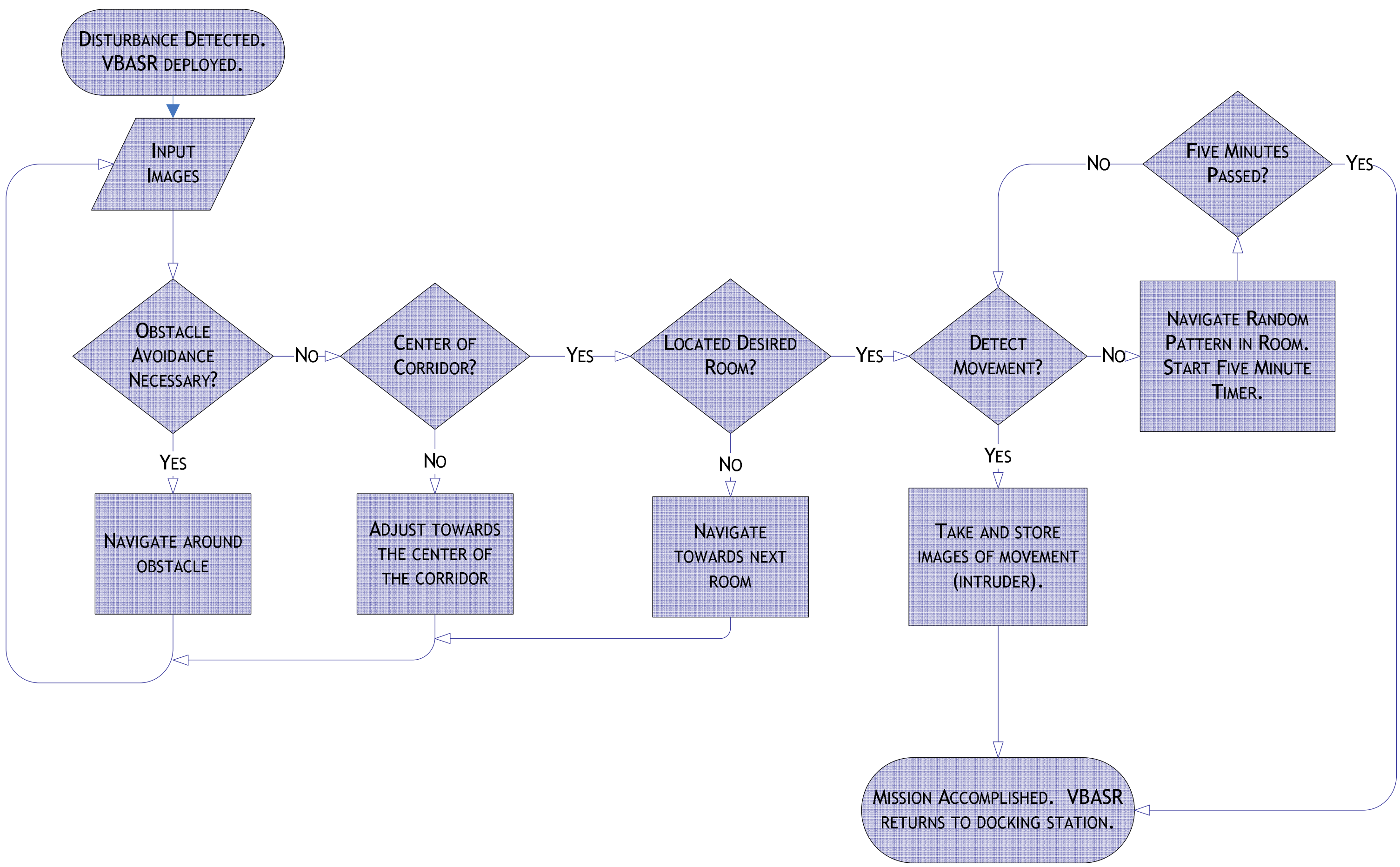

FIGURE 3 – FIRST GENERATION SOFTWARE FLOWCHART

The flowchart shown as Figure 3 above shows the first generation design logic flow. Assuming the 4th project goal is achieved the VBASR would receive a signal indicating that a particular room needs to be investigated. With internal mapping, the VBASR will start taking navigation images and start towards that room. With each new image it will determine if it is necessary to attempt to avoid an obstacle. If not, then it will verify that it is in the center of the hallway. If so it will navigate to the necessary room.

Once it reaches and enters the room it will run a random pattern looking for movement (intruders). If an intruder is discovered in less than five minutes the incriminating photos will be taken and stored. After five minutes and no intruders were discovered the VBASR will return to its docking station and await a new disturbance.

Note that the VBASR will be sampling images as fast as the microprocessor can analyze them. Thus the robot will have the fastest possible reaction time, ideally far greater than any human. In such a manner, the VBASR will be able to quickly identify and avoid any sudden obstacles and also quickly identify any movement to photograph it.

LITERARY REVIEW

Computer vision has been explored in many different projects and papers yet there are few standards for this area of research. Kingsly and Young argue that image processing is traditionally concerned with pre-processing operations (Fourier filtering, edge detection, etc). Computer vision is an extension of these ideas including understanding scene content, tracking and object classification. A good example of a security application for a fixed camera is identifying license plates. One example used in their paper for fixed security cameras is to subtract the normal image with a second image including an intruder. Since the camera is fixed, the background never changes. Thus the difference in the images will leave only an image of the intruder [3].

Andrea Cavallaro attempted to put together a succinct computer vision project for undergraduate students. Because the learning curve is steep for beginners into this field Cavallaro attempted to create several simple yet insightful projects for students to design. OpenCV was used for each of these projects. This is a useful reference because OpenCV is a huge database of open source image processing programs. OpenCV will be used to program the VBASR [1].

REFERENCES

[1] Cavallaro, Andrea. Image Analysis and Computer Vision for Undergraduates. Tech. ICASSP, 2005. IEEEXPLORE. Web. 20 Oct. 2009. <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1416369&isnumber=30654>

[2] "iRobot Create Premium Development Package." IRobot. IRobot Corporation, 2009. Web. 10 Nov. 2009. <http://store.irobot.com/product/index.jsp?productId=2591901&cp=2591511&parentPage=family>.

[3] Sage, Kingsly, and Stewart Young. Security Applications of Computer Vision. Tech. IEEE AES Systems Magazine, 1999. IEEEXPLORE. Web. 20 Oct. 2009. <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=756080&isnumber=16393>

[4] Shapiro, Linda G., Linda G. Shapiro, and George Stockman. Computer Vision. Upper Saddle River: Prentice Hall, 2001. Print