CS308 README



CS308 – 2011 Project

Project: Damar Bot : The Pothole Filling Robot

The objective of this document is to help someone else run the code that is delivered as part of this project.

**Project Title:** Damar Bot : The Pothole Filling Robot

**Students:**

|  |  |  |
| --- | --- | --- |
| Name | Roll No. | Email |
| Garvit Juniwal | 08005008 | [garvitjuniwal@gmail.com](mailto:garvitjuniwal@gmail.com) |
| Ravi Bhoraskar | 08005002 | [ravi@cse.iitb.ac.in](mailto:ravi@cse.iitb.ac.in) |
| Kunal Shah | 08005005 | [kunalshah@cse.iitb.ac.in](mailto:kunalshah@cse.iitb.ac.in) |
| Namit Katariya | 08005007 | namitk@cse.iitb.ac.in |

**Project Objective**

This project aims to develop an automated system which can be used to help filling sunken trenches in roads. As we all know potholes are a major problem in India. Hence this idea has wide applicability and scope for solving a real life problem.

Users of this product will use the interface to initialize the system and monitor it.

The achievements are as follows:

* Image processing based navigation of the robot on an arena using MATLAB.
* Automated detection of pothole in the robot, locally on the robot using sharp sensor.
* Automated filling of the detected potholes by an appropriate amount of filling material using a hopper mechanism with flow control using real time feedback from the sharp sensor.

**Hardware Platform**

1. ZigBee wireless communication interface on both robots and the server.
2. Web cam interface on the top of the arena.
3. Sensor mechanism on the robot to detect potholes. Proximity sensor at the base of the robot that determines the depth of the road below it.
4. Hopper mechanism at the robot.

**Software**

1. MATLAB; for image processing
2. WinAVR/ESTEREL for programming the robot.

**Code Description**

|  |  |  |
| --- | --- | --- |
| Filename | **Purpose** | **Executes on** |
| main.c | Main Program | Robot |
| lcd.c, adc.c, xbee.c, motion.c, servo.c, buzzer.c | Helper files for main program. | Robot |
| motion.m | Image processing, sending commands to robot for navigation. | Centralized server(PC) |

**Deliverables**

|  |  |  |
| --- | --- | --- |
| Folder | **Contains** |  |
| c code | Source code of programs to be burnt on Robot. |  |
| matlab code | Contains Matlab files. |  |
| documents | Contains Project related documents. |  |

**Execution Instructions**

The following steps must be performed in order to run the code provided.

1. **Setting up the hardware on the robot :** The project requires certain specialized hardware. The Hopper Mechanism and the Pothole Detecting Sensor. The Pothole Detector is a Sharp Sensor, that is attached at the front of the robot, at a height of 13 cm from the ground, pointing downwards. This detects the height of the ground immediately in front of the robot, thus detecting potholes. The Hopper Mechanism consists of a bottle filled with beads (the filling material ). The opening of this bottle is fitted with a flap that is opened and closed using a servo motor. Opening the flap by the servo motor leads to beads falling out, thus filling the road. The outlet of the hopper must be suitably adjusted so that the filling material drops close to the Sensor, so that the hole which is detected by the Sensor is filled.
2. **Setting up the image processing hardware:** Create an arena, made of white thermocol with a white cardboard sheet on top of it. Place an overhead camera at a height of 5 feet above the arena. Attach the camera to the Base Station. Put patches of two different colours, one on the front, and one on the back of the robot.
3. **XBee:** Connect a XBee module to both the firebird, and the base station. Configure them properly so that they can communicate with each other.
4. Download the file DamarBot.tar.gz, and extract it into a suitable folder.
5. Open WinAVR, and and create a New Project in it. Import the C-files from the folder into this project. Place main.c as the main file, and all others as the supporting files. Compile the code with all default options for the Firebird (refer the Firebird Manual for more details).
6. Burn the hex file formed onto the robot using the AVR programmer. Now the Firebird is ready to use.
7. Open matlab, and run the file motion.m in it. On the Matlab terminal, enter the RGB threshold values of the two patches when asked. The code will display an image of the arena, from which the RGB values can be extracted using Color Detector, which is a freeware utility available for windows. Also enter the size of the logical grid that you expect Matlab to divide the screen into.
8. Once the matlab code is also set up, the robot is now ready to run. The Matlab code will give it suitable instructions to move in appropriate directions(using XBee), as per the path coded into the Matlab file. This path can be changed by suitably modifying the Matlab file.
9. You are now ready to go fill up the potholes in the road!

**Coding Guidelines**

Please find source code along with this document in the archive. Please refer to this code to write your own code.

Please use the standard “Copyright statement in your code declaring the code to be open source and property of ERTS Lab”.