ISSN 2395-1621

Visualization of Disaster Prone Area by a ROBOCAM for Rescue Operation

#1Mallamma R Rudraswamimath, #2K. Nirmala Kumari

1pooja.rdmath@gmail.com,
2knkbit@gmail.com

#1PG Student, BIT Bengaluru,
#2Associate Professor Dept. of Electronics and Communication,
BIT, Bengaluru.

ABSTRACT

This paper focuses on the development of a Robot which is an Unmanned Ground Vehicle (UGV) for human search and rescue operations in disaster locations due to natural calamities. Disasters such as terrorist attacks, earthquakes, wildfires and floods etc, causes loss of life and property. These disasters create emergency situations to provide basic services to the victims that must be coordinated quickly. The emergence of Urban Search and Rescue Robotics (USAR) as a field of research is largely attributed to the failure of robots in this field. Existing platforms are unable to fulfill the task of finding survivors because they are expensive, specialized and require a very long lead time. Further, they cannot be localized once a survivor is found. This project seeks to address the issues by developing an unmanned ground vehicle to map its location for facilitating two-way communication with the onboard Raspberry Pi micro-computer. In the current prototype, wireless communication is achieved over a Wi-Fi connection. The developed unmanned ground vehicle allows easy and intuitive visualization of the location and allows users to mark the model for rescue purpose. The developed prototype has successfully demonstrated its ease of construction, intuitive user interface and capability of location mapping.

Keywords--UGV, USAR, ROBOCAM, Disasters, Raspberry Pi micro-computer, Location mapping.

ARTICLE INFO

Article History

Received: 24th July 2017

Received in revised form :

24th July 2017

Accepted: 27th July 2017

Published online :

4th August 2017

I. INTRODUCTION

The tragic loss of life in the Great Hanshin Earthquake in Japan and the Oklahoma City Bombings, both in 1995, heralded a new era for Urban Search and Rescue (USAR). Over the past twenty years, countless research labs devoted time and resources into specialist robotics research for USAR applications. The input by the Center for Robot-Assisted Search and Rescue (CRASAR) during the 2001 World Trade Center disaster response marked the first time usage of robots. The last few years has seen more and more disaster response teams using unmanned robots in efforts to find survivors, but to date, only remains have been found, and frequently the robots have failed for various reasons. Thus, there is still a long way to go before USAR robots prove their worth and are fully accepted in the community.

Robotics is a tool used by responders, rather than a replacement. Specialized robots may be tasked with searching, mapping, rubble removal, structural inspection or simply to provide logistics. Out of the 34 recorded USAR robot deployments between 2001 and 2013, 24 have included UGVs (Unmanned Ground Vehicles), as opposed to 11 for UAVs (Unmanned Aerial Vehicles) and 7 for UUVs (Unmanned Underwater Vehicles)- a possible cause being UUVs and UAVs are often disregarded due to environmental constraints (their working envelope). Because of this, significant research effort is based on advancing UGV capabilities. However, in general, robots reduce the risk to rescuers through valuable information, as well as environmental manipulation.

Small man-packable UGVs are used for first response, their primary role being to locate and communicate with trapped survivors. As a secondary role, often first response robots are also tasked with reconnaissance and mapping, giving responders an overview of the situation, as well as a model of the destroyed environment.

The "UGV 10 minute HRI (Human Robot Interaction) rule" defined in states that if a human cannot figure out how to use the main function of the product within 10 minutes, the software is flawed. Many robots and robot interfaces currently in use do not comply with this, somewhat unofficial, rule, and require additional training for operators, which is not always a possibility. As outlined in the Handbook of Robotics, a good interface should facilitate both control of the mobile robot, and robot feedback to the operator. Basic mobility requirements for any USAR robot entail forward and backward movement with left and right turning, in all environmental conditions, including on a slope. The feedback should specifically include at least the robot's perspective, the robot's status and pose, and also a birds-eye map for localization.

Therefore, the first focus of this research is to propose an easily accessible and construct UGV that is capable of mapping the location using off-the-shelf (OTS) components coupled with custom software. The ability to profile and map the terrain in real time offers users (rescue team) valuable information about the physical environment, danger zone, and survivor locations. The second focus is ought to offer control and feedback to users in real time, equipped with wireless communication capability. The gathered data about the GPS and location mapping, allows intuitive and meaningful visualization and interpretation by users (rescue team).

II. SYSTEM DESCRIPTION

A robot is an electromechanical machine that is controlled by computer program to perform various operations. Robots can perform dangerous and accurate work to increase the productivity as they can work 24 hours without rest. Industrial robots have designed to reduce human effort and time to improve productivity and to reduce manufacturing cost. Today human-machine interaction is moving away from mouse and pen and becoming much more pervasive and much more compatible with the physical world. Now-a-days the world is optimizing and is becoming more precise by switching from the world of personal computers to laptops to android phones. Human is moving and is accepting compact technologies so that, the gap between personages and the machines is being reduced to ease the standard of living. The purpose of this project is to design and implement a prototype robot. Android app is used to control the robot motion from a long distance using Bluetooth communication to interface controller and android.

Smart phones are becoming a basic need in day to day life with massive storage capacities, fortified processors, ample divertissement functions and vast communicating methodologies. Bluetooth is mainly used for exchanging data between different devices be it two smart phones or be it a robot and a smart phone. It has changed the medium of how people use digital devices at home or offices and has brought wireless devices in existence. The basic element of a Bluetooth is piconet, which is a collection several slave devices operating together with one master. Maximum of seven slaves can share a common master through a same link. Even several piconets can link together and forms scatter net. It is useful in home environments, looking at its range or normal working area is 8 meters. Bluetooth has gradually increased users to prosecute smart phones, which have gingerly turned into a multipurpose portable device and are accessible to people for their quotidian use.

Present day, android is widely accepted as an open source platform. Android consist of a complete package involving an operating system, middleware layer and core applications. A Smartphone is a cell phone built on a mobile computing platform, which has big number of boosted connectivity and computing ability than what a feature phone has. In this paper, we are overcoming the problem of traditional robots, which are usually handled with any remote controller. Reducing the remote work we are making the robot move by just a click on the cell phone with android operating system.

2.1 System Assembly

The project proposes a mobile rescue robot that moves in the disaster, earthquake prone area and helps in identifying the live people, injured people, location and rescue system operations. Hence due to the on timely detection in natural calamities this can save precious life & great loss even without the help of large number of rescue operators. The proposed system consists of mobile robot, Arduino, Bluetooth module, Ultrasonic sensor, Raspberry pi with camera and L293D motor driver. The robot mobility is mainly based on Arduino interfaced with a Bluetooth module controlled by an android app. The robot system is a harmony of hardware and software design. The software implemented in the Arduino and Raspberry Pi provides specific instructions to the hardware components in the vehicle, such as the motors, sensors, and camera, to perform specialized tasks.

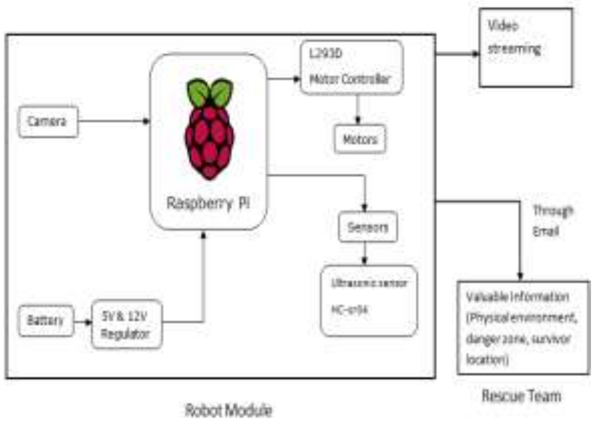

Fig.1. Block diagram of the overall system

2.2 Robot Control Approach

The robot is controlled using Bluetooth module HC-05 and Arduino ATmega 2560 microcontroller with android Smartphone device. The controlling devices of the whole system are a microcontroller, Bluetooth module and DC motor interfaced to the microcontroller.

The data received by the Bluetooth module from android smart phone is fed as input to the controller. The controller acts accordingly on the DC motor of the robot. The robot can move in all the four directions i.e., forward, backward, left and right using the android phone. In achieving the task the controller is loaded with program written using Embedded 'C' Language.

2.2.1 Arduino ATmega 2560

Arduino is an open source computer hardware and Software Company, that designs and manufactures microcontroller based devices that can sense and control objects in the physical world. Arduino uses Atmel ATmega AVR series of chips, specifically the ATmega8, 168, 328, 1280 and 2560.

Arduino ATmega2560 is a microcontroller board based on ATmega2560. It contains everything needed to support the microcontroller. It can be connected to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

The ATmega2560 board can be programmed with the Arduino IDE software. It has an editor that uses the processing/wiring language; the commands resemble the C language and the supporting utilities for the projects (Library).The programming work can easily be performed by making the necessary settings and definitions in the IDE program.

2.2.2 HC-05 Bluetooth Module

This Bluetooth module is capable of communicating with PC, mobile phone or any other Bluetooth enabled devices. It is interfaced with the microcontroller over the serial UART (Universal Asynchronous Receive Transmit) port of the microcontroller. Bluetooth is a wireless communication protocol running at 2.4GHz, with client-server architecture. Bluetooth allows the connectivity between the two devices using their MAC address. HC-05 is an easy to use Bluetooth SPP (Serial Port Protocol) module and is fully qualified of V2.0+EDR (Enhanced Data Rate) of 3Mbps modulation with complete 2.4GHz transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping). The module works on 3.0V lower power operation and 3.0 to 4.2V I/O controls. It has integrated antenna, edge connector and UART interface with programmable baud rate. Default baud rate: 38400, Data bits: 8, Stop bits: 1, Parity: No parity and supported baud rates are 9600, 19200, 38400, 57600, 115200, 230400, 460800.

2.2.3 L293D Motor Driver

Microcontroller cannot supply the current required to run DC motor. So to satisfy this requirement, IC's are used to drive the motor. L293D is a motor driver IC which allows the motor to move in any direction. With the help of motor driver, two DC motor can be attached on a single IC and both of them can be moved in either directions. L293D is a 16 pin IC which can control a set of direct current motors, Dual H-bridge Motor Driver integrated circuit(IC).This driver drives small as well as quite big motors, and it works on the concept of H-bridge circuit which allows the voltage to be flown in any direction. H-bridge is ideal for driving a DC motor as the voltage needs to change the direction of the motor to make it move it in either clockwise or anti-clockwise direction.

2.2.4 Ultrasonic Sensor HC-sr04

This sensor is attached to detect the distance of the obstacle from the robot. It uses sonar to govern distance of an object. It inaugurates non-contact range detection, and provides stable reading in an easy to use package. Its range varies from 2 cm to 400 cm or 1" to 13 feet. Sensor is not affected by sunlight or black material but it is difficult to detect the distance from any soft material like cloth. It is a combination of both ultrasonic transmitter and receiver module. Its output is greatly perturbed by Echo signal, so the output never goes Low if Echo is not received. Even timeout parameters are needed to alter the output according to the user aspirations. Its resolution is 0.3 cm and trigger input pulse width is 10 µS.

2.3 Working Module

The system consists of two sections. The first section is the mobility of the robot module. The second section of the system is the feedback control to the rescue team. The feedback control includes the information regarding the disaster prone area and location mapping to other rescue teams. The feedback control module consists of Raspberry pi micro-computer with a camera unit. Raspberry pi is chosen as the micro-computer for this project because of its superior prototyping and video processing capabilities. Raspberry pi has the ability to execute python coded programs and allows General Purpose Input Output (GPIO) pins that can control anything.

2.3.1Robot mobility control by Bluetooth module interfaced with the Arduino board

Procedure: The Arduino ATmega2560 board works according to the code burnt on it, code is burnt on the board with the help of a software. The code includes the maximum and minimum ranges of detecting distance from the obstacle through the Ultrasonic sensor, where the input is taken from the Echo pin while output is sent to Trigger pin. Initially, connect the HC-05 Bluetooth module and Ultrasonic sensor HC-sr04 to the Arduino ATmega2560 and also connect the Arduino with the input pins of L293D for sending signals to drive motors. With an android app named by Bluetooth terminal HC-05 is used to operate the Bluetooth module, connect the Bluetooth device in the android phone. Power up the connection, scan the Bluetooth device and pair it with the Bluetooth module through the application using the default passkey 1234. After the connection is established and the devices are paired up, start operating the robot with giving commands in the application. Arduino will receive the commands via the serial port of baud rate 9600. The program functions for the following commands: 1-forward, 2-backward, 4-right, 5-left and 0-stop. After receiving the data, Arduino switches the function to the L293D motor driver, which uses 12V DC gear motors to move the robot in the desired direction.

Fig.2. Arduino based Bluetooth controlled Robot module

2.3.2 Raspberry Pi micro-computer with camera unit

Raspberry pi is an all in one micro computer that operates on Raspbian, a free Linux operating system. Its compatible video camera module makes setup easy and is capable of producing high definition videos at high frame rates. Multithreaded python script is utilized to transmit the data from the robot. Once executed, the script will run continuously until the user terminates it. In order to provide real time data access to the PC, the overall processing time of these tasks are kept to a minimal. Thus processor intensive tasks are avoided by the Raspberry Pi and are only performed by the PC.

The Raspberry Pi and the PC were connected on the same network through a wireless router via Wi-Fi. This enabled easy access to the Raspberry Pi and also provided a pathway for transmitting video and data streams to the PC. Video captured by the on-board camera uses a dedicated streaming protocol for video, while Transmission Control Protocol (TCP) was used for the data stream. User control signals were also transmitted to Raspberry Pi via a separate port; this allows an independent transmission that simplifies the data transmitted. Many different techniques exist, for streaming live video from the Raspberry Pi Camera over a Wi-Fi network. Most attempt to stream the raw video, allowing the host computer to render the stream using tools such as VLC Media Player.

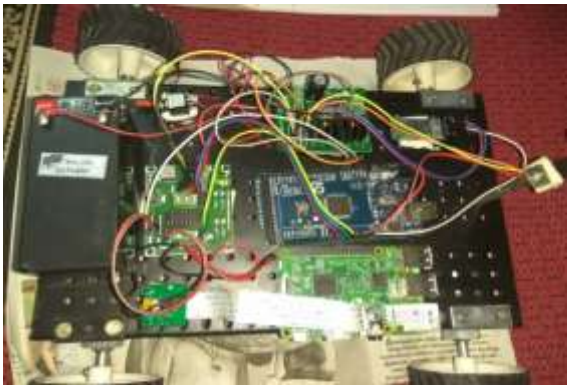

Fig.3. Robot module with Raspberry Pi and camera unit

However, this induces a very large overhead on the network, and observations of latency during initial tests proved to be in an order of tens of seconds - simply not good enough for a time-critical USAR application. An open source service[10], coupled with the RPi-Cam[11] provides a unique approach to viewing a live video stream. The latter hosts a web page locally on the Raspberry Pi (typically the target port of an HTTP request), accessing and updating the

HTML element for the video stream, as new frames are captured and stored locally on the Raspberry Pi. Tests with this technique show a latency of less than 1 second for extended periods. This is achieved by the fact that the network is only supporting the PC to access a web page on the Raspberry Pi and not transmitting large amounts of live video data.

III. FEEDBACK TO RESCUE TEAM

The Arduino based Bluetooth controlled robot with the raspberry pi and camera unit is used to visualize and monitor the disaster area. A 5MP camera module is mounted on the robot and live video stream is captured. The video covers the live visualization of the location with the robot mobility.

Ustream is used to broadcast the live video stream to the rescue team. The video streaming can be broadcast only to the authorized person by sending the website link of Ustream. Ustream members can also record and save videos for future broadcast distribution. Ustream's video platform is known for its ability to provide viewers with different ways to interact with the presenter during a live broadcast, providing broadcasters with chat and instant polling features, as well as allowing integration with social network sites. Python programming is used to obtain the GPS data from the disaster location. GPS data provides the latitude and longitude values with the direction of the robot. This data is refreshed for every 10 seconds. These values are then converted into degrees and mapped in Google maps for display. The map keeps on refreshing in the Google maps and can also be sent to the rescue team.

IV. PERFORMANCE ASSESSMENT AND RESULTS

The evaluation of the result depends on the robot quality, its mobility and visualization of the robot location. The python programming enables the ease of updating the GPS data with latitude and longitude values of the robot location.

Fig.4. GPS- latitude & longitude

Fig.5. Location Mapping

V. CONCLUSION

The developed robot system offers the benefit of exploring disaster location with the live video streaming that enables the rescue team to visualize the disaster location in real-time. The GPS data with latitude and longitude is obtained and the map is also displayed from the Google maps.

VI. ACKNOWLEDGEMENT

It's a great privilege to place on record my deep sensor gratitude to My guide Mrs K.Nirmala Kumari, Associate Professor,Dept of ECE, HOD Dr. K. V.Prasad, Dept of ECE and Principal Dr. A.G.Nagaraj and the Management of Bangalore Institute of Technology, who patronized throughout our carrier and for the facilities provided to carry out this work successfully.

REFERENCES

[1] Davids, A. (2002). Urban search and rescue robots: from tragedy to technology. Intelligent Systems, IEEE, 17(2), pp. 81-83.

[2] Siciliano, B., and Khatib, O. (2008) 'Search and Rescue Robotics.' In: Springer Handbook of Robotics, Springer, Berlin, pp. 1151 - 1173.

[3] Murphy, R. R. (2004). Trial by the [rescue robots]. Robotics and Automation Magazine, IEEE, 11(3), pp. 50-61.

[4] Murphy, R. R. (2012). A decade of rescue robots. Intelligent Robots and Systems (IROS), pp. 5448-5449.

[5] Murphy, R. (2014). Disaster robotics. The MIT Press, Cambridge, Massachusetts.

[6] Murphy, R. R., Kravitz, J., Stover, S., and Shoureshi, R. (2009). Mobile robots in mine rescue and recovery. Robotics & Automation Magazine, IEEE, 16(2), pp. 91-103.

[7] silvanmelchior RaspiMJPEG. Retrieved July 6, 2014 http://www.raspberrypi.org/forums/viewtopic.php?t=61771.

[8] Pellenz, J., Lang, D., Neuhaus, F., and Paulus, D. (2010). Real-time 3D mapping of rough terrain: A field report from Disaster City. Safety Security and Rescue Robotics (SSRR), pp. 1-6.

[9] POLOLU CORPORATION (2013). Pololu TReX User's Guide. Center for Robot-Assisted Search and Rescue.