ISSN 2320 - 2602
Volume 2, No.3, March 2013
International Journal of Advances in Computer Science and Technology
Available Online at http://warse.org/pdfs/2013/ijacst02232013.pdf

Autonomous Mine Sweeping System Using Image Processing

Anuja Vaidya, Prajakta Pardeshi, Shalom Cohen
University Of Pune, India, vaidya.anuja732@gmail.com
University Of Pune, India, prajakta.191@gmail.com
University Of Pune, India, shalomcsaoe@gmail.com

ABSTRACT

The Mine Sweeping System is being developed for the sole purpose of efficiently detecting mines within a user defined area .It comprises of a Java application from the user end and a robot grid system which work together to form a fully automated, optimized and efficient system. The Server (i.e. the application) and the Robots communicate with each other through wireless transmission and at no point of time any human contact with the robot system is required.

The Server Application is provided to administer the entire mine detection process. The optical flow vector result, Robot Alignment result and virtual map of the location of the robots are received on the Server Computer. Database results of previously searched locations are made available to user. These database results are then used to optimize future mine sweeping processes through a modified LRU Algorithm.

The Robot System is mounted with a camera and uses a Self Navigating algorithm with Obstacle Avoidance implemented using Optical Flow. The optical flow generated within the surroundings of the robot is registered using the on board camera and using Image Processing and Vector Calculations the Time to Contact (in case of obstacle avoidance) or Tracking (in case of aligning algorithm) is obtained. The robots function within a self- generated virtual grid and a real time map of the mine sweeping operation is provided on the Server system. The robots calculate the shortest path within the grid while searching and sweeping the mines. The robots also communicate with each other using Swarm technology in order to complete the task.

The Autonomous Mine Detection System is developed for the following reasons:

- Human interaction involved in the process is minimized.
- The task is performed at a higher efficiency rate.
- A practical solution for Self Alignment of an autonomous system is developed.
- The Robot system features obstacle avoidance using the most efficient method.
- User is provided with added functionality and the entire system is made to benefit user functionality.
- This System will reduce the overall time complexity of the mine detection process.

Key words: Serial Communication, Obstacle Avoidance, Image Processing-Time to contact, Shortest Path Algorithm

1. INTRODUCTION

Traditionally mine detection used to be carried out by humans themselves, by wearing the necessary protective gear and instruments which were used to detect mines. The risk involved in the task was life threatening in spite of the protective gear used since the proximity and severity of the blast could not be predicted. Hence the introduction of manual robots in which they were controlled by the user to survey a particular area for mines. The problem in such an alternative is that the manual maneuvering of the robot over the given area becomes tedious especially over a large terrain.

Thus the need for autonomous robots comes in, where the user is not at all involved in maneuvering the robot. The autonomous robots search in a specified area for the mines and the co-ordinates of their positions are constantly sent back to the server system. Thus the robots are being mapped on a virtual grid which is generated on the server system and their position can be tracked down. When a mine is detected, that particular co ordinate is sent back to the server system and a symbol is marked on the virtual grid indicating the detection of a mine. Immediately the necessary information related to the capturing of the mine such as co ordinates, place, time, area all are stored in the database. The user can then examine the database to get the necessary information and thus narrow down the search area. Using methods such as the shortest path algorithms, self alignment algorithms and image processing for obstacle avoidance, the efficiency of the entire operation is maintained and thus the involvement of humans in the entire mining detection operation is negligibly zero.

2. IMPORTANT CONCEPTS AND FEATURES

2.1 Optical Flow Calculation Using Lucas Kanade Method

This is used for obstacle avoidance and self-alignment of robot system. Optical flow[9] or optic flow is the pattern of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer (an eye or a camera) and the scene. The concept of optical flow was first studied in the 1940s and ultimately published by American psychologist James J. Gibson as part of his theory of affordance. Optical flow techniques such as motion detection, object segmentation, time-to-collision and focus of expansion calculations, motion compensated encoding, and stereo disparity measurement utilize the motion of the objects' surfaces and edges.

2.1.1 Estimation of the optical flow*:*

Sequences of ordered images allow the estimation of motion as either instantaneous image velocities or discrete image displacements. Fleet and Weiss provide a tutorial introduction to gradient based optical flow. John L. Barron, David J. Fleet, and Steven Beauchemin provide a performance analysis of a number of optical flow techniques. It emphasizes the accuracy and density of measurements.

The optical flow methods try to calculate the motion between two image frames which are taken at times t and $t + \Delta t$ at every voxel position. These methods are called differential since they are based on local Taylor series approximations of the image signal; that is, they use partial derivatives with respect to the spatial and temporal coordinates.

For a 2D+t dimensional case (3D or n-D cases are similar) a voxel at location $(x, y, t)$with intensity $I(x, y, t)$will have moved by Δx, and Δt.

$$I(x,y,t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \quad (1)$$

Assuming the movement to be small, the image constraint at $I(x, y, t)$with Taylor series can be developed to get

$$I(x+\Delta x, y+\Delta y, t+\Delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t \quad (2)$$

From these equations it follows that:

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0 \quad (3)$$

Or

$$\frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t}\frac{\Delta t}{\Delta t} = 0 \quad (4)$$

From this above equation we can now write

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial t} - 0 \quad (5)$$

Where V_x, V_y are the x and y components of the velocity or Optical flow of $I(x, y, t)$ and $\frac{\partial I}{\partial x}$, $\frac{\partial I}{\partial y}$ and $\frac{\partial I}{\partial t}$ are the derivatives of the image at (x,y,t) in the corresponding directions. Ix, Iy and Iz can be written for the derivatives in the above equation. Thus the final equation for optical flow is deduced to be

$$I_x V_x + I_y V_y = -I_t \quad (6)$$

Or

$$\nabla I^T \cdot \vec{V} = -I_t \quad (7)$$

This is an equation in two unknowns and cannot be solved as such. This is known as the aperture problem of the optical flow algorithms. To find the optical flow another set of equations is needed, given by some additional constraint. All optical flow methods introduce additional conditions for estimating the actual flow.

2.1.2 Uses of optical flow

Motion estimation and video compression have developed as a major aspect of optical flow research. While the optical flow field is superficially similar to a dense motion field derived from the techniques of motion estimation, optical flow is the study of not only the determination of the optical flow field itself, but also of its use in estimating the three-dimensional nature and structure of the scene, as well as the 3D motion of objects and the observer relative to the scene, most of them using the Image Jacobian.

Optical flow was used by robotics researchers in many areas such as: object detection and tracking, image dominant plane extraction, movement detection, robot navigation and visual odometry. Optical flow information has been recognized as being useful for controlling micro air vehicles.

The application of optical flow includes the problem of inferring not only the motion of the observer and objects in the scene, but also the structure of objects and the environment. Since awareness of motion and the generation of mental maps of the structure of our environment are critical components of animal (and human) vision, the conversion of this innate ability to a computer capability is similarly crucial in the field of machine vision

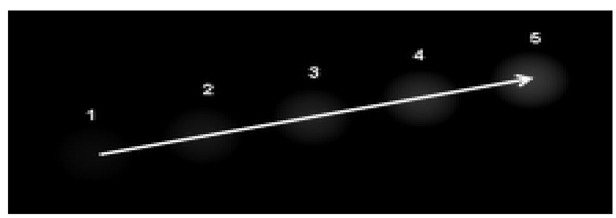

Figure 1: *The optical flow vector of a moving object in a video sequence*

Consider a five-frame clip of a ball, Figure 1, moving from the bottom left of a field of vision, to the top right. Motion estimation techniques can determine that on a two dimensional plane the ball is moving up and to the right and vectors describing this motion can be extracted from the sequence of frames. For the purposes of video compression (e.g., MPEG), the sequence is now described as well as it needs to be. However, in the field of machine vision, the question of whether the ball is moving to the right or if the observer is moving to the left is unknowable yet critical information. Not even if a static, patterned background were present in the five frames, could we confidently state that the ball was moving to the right, because the pattern might have an infinite distance to the observer.

Using this Optical Flow, the Focus of expansion/contraction of an object is calculated and from this available data the Time to Contact [3] from each object is established.

2.1.3 Image segmentation

A fundamental step for video/image retrieval by content is the calculation of the visual features. In some applicative contexts the most relevant content consists in represented subjects rather than the whole scene. For example, let prefigure an application that takes in input a photo or a video sequence, extracts the focused objects, searches similar images in its database and outputs the corresponding descriptions. In this perspective, the distinction between focused objects and background permits to calculate the visual features of each object alone, making more effective the retrieval task. To this end, a segmentation method for the extraction of images of single complete objects from a digital photo or a short video sequences has been developed. Image segmentation, widely employed in medical imaging, computer vision, production quality control, etc., is the process to extract meaningful regions from an image.

The necessity to detect the object as a whole and not only part of it has been approached using a hybrid segmentation technique based on edge detection, region growing, and optical flow procedures.

Image Segmentation [4] proves to be an important step in calculating the navigational results of the robot system.

2.2 Atmega 32 Microcontroller

The microcontroller used for autonomous coding is the Atmega 32 which has the following features and Pin Out description. Figure 2 shows the above details.[6]

- High-performance, Low-power Atmel® AVR® 8-bit Microcontroller.
- Advanced RISC Architecture
- 133 Powerful Instructions – Most Single Clock Cycle Execution
- 32 x 8 General Purpose Working Registers + Peripheral Control Registers
- Fully Static Operation
- Up to 16 MPS Throughput at 16 MHz

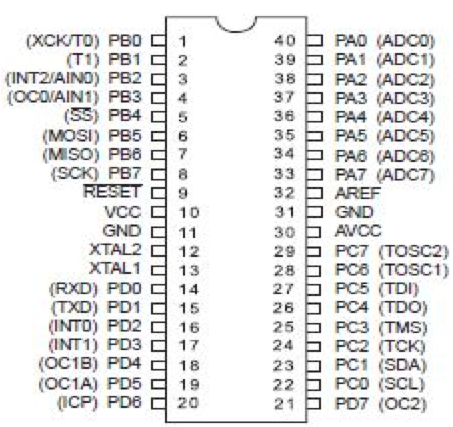

Figure 2: *Pin Configuration Atmega32*

This microcontroller along with the L298 (motor driving IC's)[8] will be used to run the Robot autonomously**.**

2.3 Xbee Pro Communication Module

The communication between the Robot System and Server System is established through a Xbee module which acts as a Transreceiver by securing Serial Communication using the UART (Universal Asynchronous Receiver Transmitter)**.** [7]

2.3.1 Serial communication

The XBee/XBee-PRO OEM RF Modules interface to a host device through a logic-level asynchronous serial port. Through its serial port, the module can communicate with any logic and voltage compatible UART; or through a level translator to any serial device (For example: RS-232/485/ 422 or USB interface board).

2.3.2 UART data flow

Devices that have a UART interface can connect directly to the pins of the RF module as shown in the figure below.

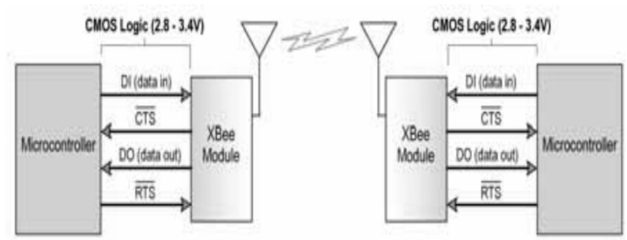

Figure 3: *Xbee Pro System Data Flow Diagram in UART Environment*

2.3.3 Flow control DI (Data In) buffer

When serial data enters the RF module through the DI pin (pin 3), the data is stored in the DI Buffer until it can be processed.

Hardware Flow Control (CTS): When the DI buffer is 17 bytes away from being full; by default, the module de-asserts CTS (high) to signal to the host device to stop sending data [refer to D7 (DIO7 Configuration) parameter]. CTS is re-asserted after the DI Buffer has 34 bytes of memory available.

2.3.4 Flow Control DO (Data Out) buffer

When RF data is received, the data enters the DO buffer and is sent out the serial port to a host device. Once the DO Buffer reaches capacity, any additional incoming RF data is lost.

Hardware Flow Control (RTS): If RTS is enabled for flow control (D6 (DIO6 Configuration) Parameter = 1), data will not be sent out the DO Buffer as long as RTS (pin 16) is de-asserted.

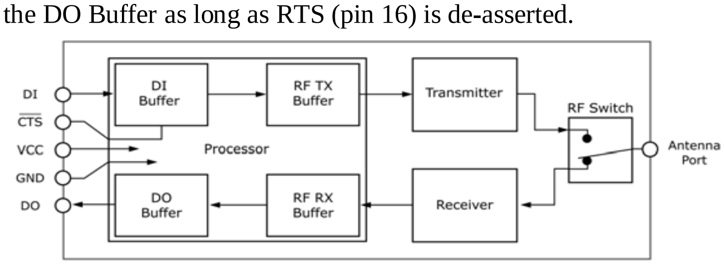

Figure 4: *Xbee Pro Internal Data Flow Diagram*

Thus in this manner the Xbee Pro module proves to be an efficient Transreceiver which establishes a secured UART channel.The developer will have to create a template first to fulfil the document processing requirements. The other services will only require single time activation.

3. SYSTEM DESIGN

3.1 System Architecture

The entire autonomous mine sweeping system is divided into 3 main modules, shown in Figure 5, which include:

- Server System with Java Application
- Robot System
- Communication System

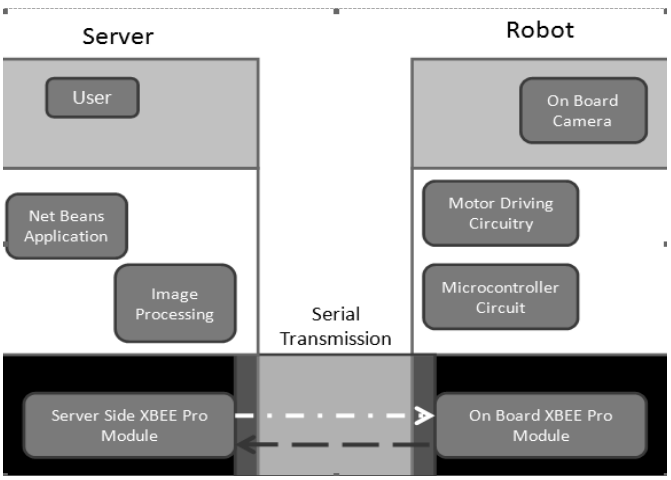

Figure 5: *System Architecture Block Diagram*

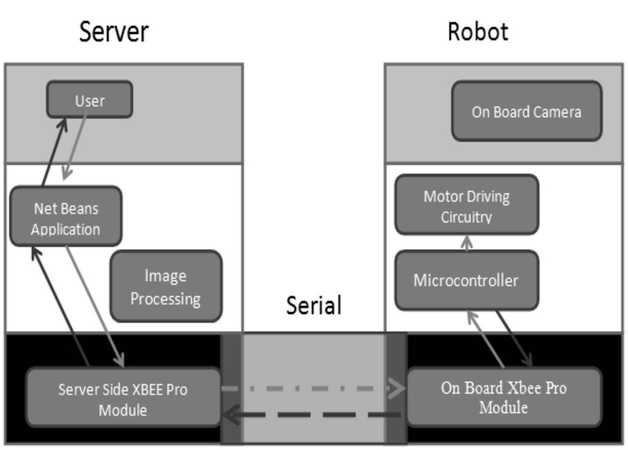

Figure 6: *Communication between Microcontroller and rest Of Architecture*

- The User enters the Net Beans Application the application sends a 'start' command to the UART channel via Serial Communication using javax.comm package, on User Click.
- The 'start' command is interpreted by the Xbee module on the Robot System and triggers the code within the Atmega 32 m/c.
- Here after the Microcontroller sends signals to the motor driving circuits and continuously transmits characters onto the UART channel which describe the navigational direction and status of mine locations within the traversal field.
- These commands are interpreted by the Java Application and the real time Map of the Mining Operation is generated on the GUI simultaneously
- This process continues in a loop until the Robot reaches the end of the self generated virtual grid.
- At the end of the run a 'stop' command is sent by the Robot system to the Java Application.
- This entire process is depicted in Figure 6

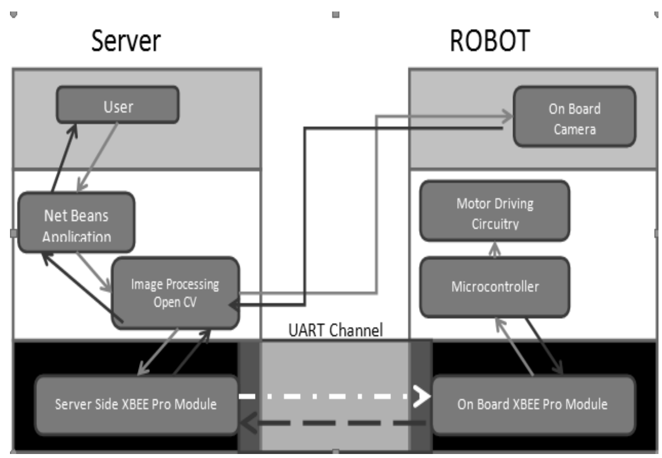

Figure 75: *Communication between On Board Camera and System Architecture*

- Once The User enters the Net Beans Application the application sends a 'start' command to the UART channel via Serial Communication using javax.comm package, on User Click.
- The 'start' command is interpreted by the Xbee module on the Robot System and triggers the code within the Atmega 32 m/c.
- As the m/c is started, the Java Application also sends data to the on board IP camera through a wireless network router to start the Image Processing module.
- The data received from the IP camera will be processed for Optical Flow [2], Time To Contact [3], Object Segmentation[4] results and the corresponding command to be sent to the Robot is generated on the Server Computer.
- The generated command is interpreted by the m/c and a corresponding action is performed by the Robot System.
- At the end of the run a 'stop' command is sent by the Robot system to the Java Application which also puts a stop to the Image Processing Module.
- This entire process is depicted in Figure 7.

4. SOFTWARE REQUIREMENTS SPECIFICATIONS

4.1 Project Scope

The Scope of this project covers:

- An autonomous solution for the existing Mine Detection System for military organizations in detecting land mines.
- Highly efficient obstacle avoidance and Self Alignment modules which can be used by other self maneuvering autonomous systems used for Security, cleaning and other navigation related activities.
- Can also be used for commercial mining purposes – Searching and Retrieval. Searching and Retrieval algorithm can be easily mapped for another object retrieval purposes.

4.2 Product Perspective

The proposed Autonomous Mine Sweeping System in terms of a product consists of:

- Server Application (Software)
- Image Processing Module (Software)
- Robot System. (Hardware)

The following Steps are executed by the Mine Sweeping System after the User enters the application:

- User enters Place /City at which Mine Searching operation should commence along with area to be searched for mines in terms of Length and Breadth within the Java Application.
- The Robot i.e. the Search and Collector Robot starts the Landmine/ Metal search operation on User start input from Server Computer base.
- The robot traverses within a self-generated virtual grid using Obstacle Avoidance and Self Alignment algorithms which function using the optical flow generated by the environment of the robot registered using the On board Camera.
- The Robot co-ordinates with the server system using Wireless Serial Communication channel and IP networking technology.
- The robot calculates the shortest path within the grid to detect / sweep the mines.
- A Real time map of the mine sweeping process of the robot is obtained simultaneously as the robot is functioning within the area.
- Metals /Landmines within the virtual grid area are detected using a magnetic flux sensor.
- The co-ordinates of the Mines within the field, as well as other parameters of the mine sweeping process are fed back to the server and it is stored on the server database accordingly.
- The option of a modified LRU (Least Recently Used) algorithm is provided within the system to improve metal mining search efficiency.

4.3 Product Features

The Product Features include:

- User friendly application GUI.
- Real time Map Generation of Robot Co-ordinates in Virtual Grid.
- No Human Interaction **-** Human Contact with Mine field or Robot System at any point of time.
- Wireless Communication between Robot and Application and Network connection between Onboard IP Camera (Robot) and Server System.
- Shortest Path Algorithm For mine sweeping.
- Obstacle Avoidance and Self Alignment modules allow Robot System to function in any given environment using Optical flow calculations.
- Database provided to store values retrieved from mine sweeping operation.
- Artificial Intelligence included in Robot System where in robots co-ordinate with each other using Swarm technology.
- Manual Override Functionality given to User.

4.4 User Classes and Characteristics

There is only one type of User for the application and User is defined as follows:
User: Person who wishes to run mine sweeping operation within a specified area using the robot system.

4.5 Operating Environment

1. Software: The product application will be operating in Java (Windows) environment and uses a .dll file made of the Image Processing which is done in Visual C++ OpenCV environment.
2. Hardware: The Robot system will be able to function on any given outdoor terrain.

4.6 User Dependencies

The product will include user manual. The user manual will include:

1. Product overview
2. Complete configuration of Software System and Hardware Robot System
3. System setup details
4. Technical details
5. Maintenance procedure
6. Contact information

5. ASSUMPTIONS AND DEPENDENCIES

5.1. Assumptions

1. The distance between the Client System and Robot System cannot exceed more than 500 to 700 meters for accurate results.
2. The movements of objects within the environment are considered to be gradually changing for object detection by the Robot System.

5.2 Dependencies

1. Power Supply of the Robot System must be constant at all times for accurate results.

2. Connection strength of UART Communication System is constant.

6. SYSTEM FEATURES

6.1 Database Storage

An SQL database has been created which stores results of a particular mine retrieval run by the Robot System. It is linked to the Java Application through a JDBC connection.

6.2 Description and Priorities

Proposed Database is intended to store, retrieve, update, and manipulate information related to mine sweeping operation performed by the Robot System.

1. Mine No.
2. X – Co-ordinate
3. Y- Co-ordinate
4. Location
5. Count

7. FUNCTIONAL REQUIREMENTS

This section gives the list of Functional requirements which are applicable to the Autonomous Mine Detection System. Functional requirements are nothing but the services provided by the system to its End users. There are three sub modules in this phase.

1. Java Server Application module.
2. Image Processing module.
3. Robot System module.

The functionality of each module is as follows.

7.1 Java Server Application Module

The User starts the entire mine detection process by firstly entering the Length and Breadth of the field out of which the Robot is supposed to create a virtual grid. The Application then shows a real time map of the mine sweeping operation performed by the robot continuously through the UART communication channel. At the end of the operation, the application displays the database results for the User. The application is built using Java NetBeans 7.0.

7.2 Image Processing Module:

Once the Mine Sweeping operation is initiated by the User from the Server Application, the Robot System uses the image processing module present on Server system through a wireless network connection. It collects and process data results for Obstacle Avoidance and Self Alignment of the Robot System by using Optical Flow calculations.

7.3 Robot System Module:

The Robot System module consists of an on board microcontroller which generates the virtual grid and actually performs the mine sweeping operation upon User command.

8. EXTERNAL INTERFACE REQUIREMENTS

8.1 User Interfaces

Application will be accessed via a Java run time instance. The interface would be viewed using 1024 x 768 resolution setting.

8.2 Hardware Interfaces

8.2.1 Server system side:

1. Operating System: Windows
2. Processor: Pentium 3.0 GHz or higher
3. RAM: 256 Mb or more
4. Hard Drive
5. COM Port
6. RS 232 XBEE Pro Module - Serial Transreceiver[7]
7. Wireless connection to robot system onboard IP camera

8.2.2 Client robot system side:

1. Robot Base Design
2. Battery Supply
3. Microcontroller: Atmega 32[6]
4. Motor Driving Circuitry[8]
5. IP Camera
6. Wireless Network Router
7. XBEE Pro Module-Serial Transreceiver[7]

8.3 Software Interfaces

8.3.1 Client Side:

Robot LCD Display

8.3.2 Server side:

Java Application with Image Processing .dll file.

8.4 Communications Interfaces

The communication interface is provided by the communication system which uses a Xbee Pro module.

1. Server Side Communication is established by connecting the Xbee Transreceiver to the RS232 COM Port in order to carry out serial communication using UART.
2. Similarly on the Robot System, the microcontroller sends commands serially on the UART channel by communicating through a Xbee Transreceiver.

9. NON FUNCTIONAL REQUIREMENTS

9.1 Performance Requirements

The results collected by the Robot in a specified search location can be used again and a better optimized result will be available if the robot is made to perform the operation within the same search field.

9.2 Safety Requirements

This System may crash due to connection failures in Communication channel and also due to Microcontroller lapses. At such times a separate routine with an interrupt is run on the Robot System which resets the Robot to its initial state and displays an Abort Message on Server System.

9.3 Security Requirements

The Security Requirements are as follows:

1. The Robot at no time is supposed to be manually handled by the User if such an instance does occur the Application is locked and User is directed out of the application. The Robot is reset and is made to traverse up to its initial location (X – Co-ordinate, Y Co-ordinate)
2. The Communication channel is made Point to Point with Specific predefined Baud rates and Safety bits used in communication data.
3. The Wireless Network established consists of a secure connection established only through a SSID user and password value.

10. SOFTWARE QUALITY ATTRIBUTES

The Quality of the System is maintained such that it is easy to use. The Software Quality attributes are assumed as under:

1. Accuracy.
2. Time Complexity.
3. Computational Requirement.
4. Degree of Autonomy.

11. OTHER REQUIREMENTS

11.1 Database Requirement

The mine sweeping system will hold one logical database. The first form enters the place and dimensions which are entered in the database. From next GUI the x and y coordinate of where the mine is located is stored in the database. The various tables used in here are mine table(which will store coordinates, date and time when the mine was located),place table(has place and dimensions stored in it).

11.2 Internalization Requirement

No Internalization Requirement is needed for this project.

11.3 Legal Requirement

The Mine Sweeping System application is developed using all open technologies. There are no Legal requirements for this application.

12. REUSE OBJECTIVES FOR THE PROJECT

The System can be reused for other objectives which involve object searching, path retrieval, and object detection autonomous solutions. Modules may include:

1. Autonomous Self Navigating Robot
2. Environments used for Transportation of material in Construction, hospital and other public service environments.
3. Home Automation Systems.
4. Optical Flow, Image processing module can be reused for various purposes such as Gesture recognition, object detection and tracking etc.

13. ANALYSIS MODELS

13.1 Data Flow Diagram (DFD)

Data flow diagrams illustrate how data is processed by a system in terms of inputs and outputs. Figure 8, figure 9, figure 10, show Level 0, level 1 and 2 data flow diagrams respectively.

- **Level 0:**

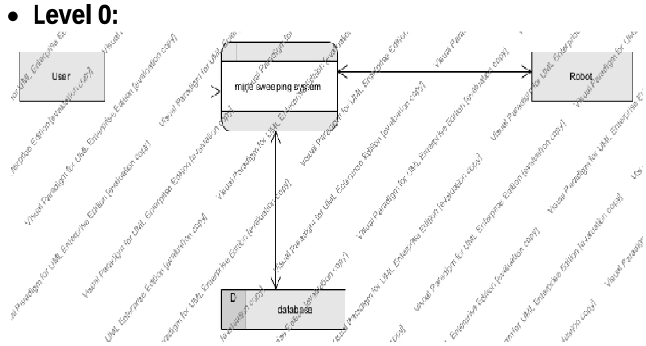

Figure 8: *Level 0 Data Flow Diagram*

- **Level 1:**

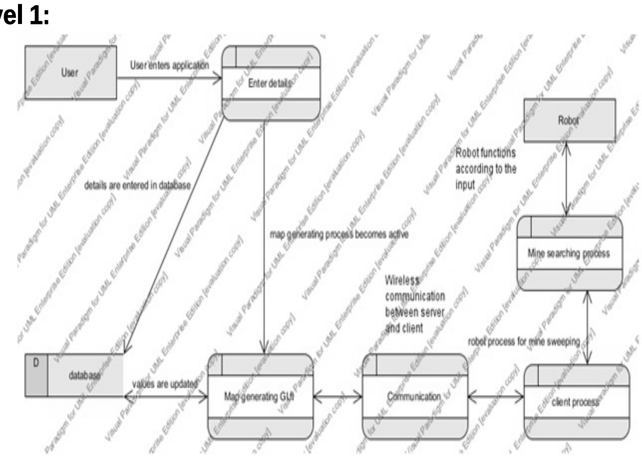

Figure 9: *Level 1 Data Flow Diagram*

- **Level 2:**

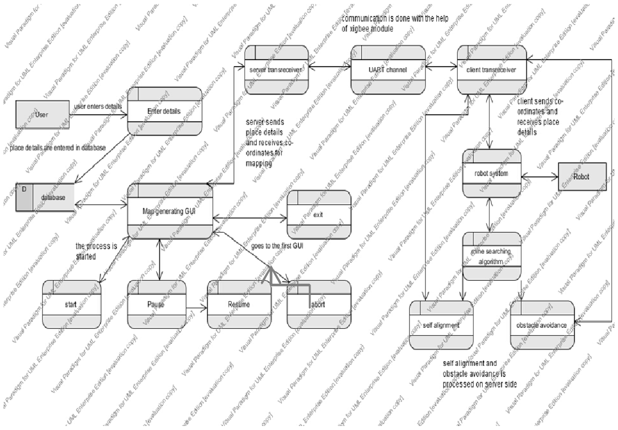

Figure 10: *Level 2 Data Flow Diagram*

14. EXPERIMENTAL RESULTS

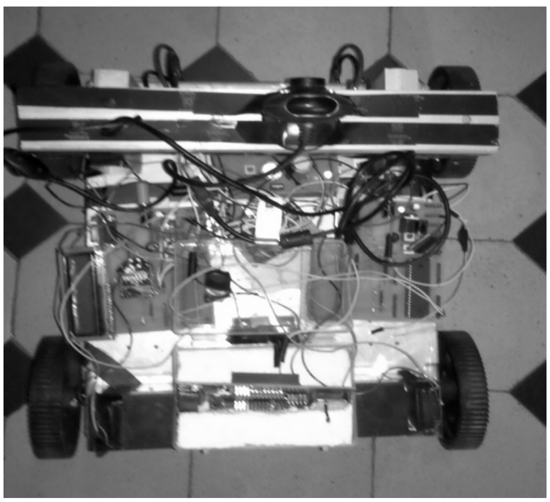

Figure 11: *The Mine Sweeping Robot Design*

Figure 11 shows the design of the robot made for sweeping of the mines in a given particular area.

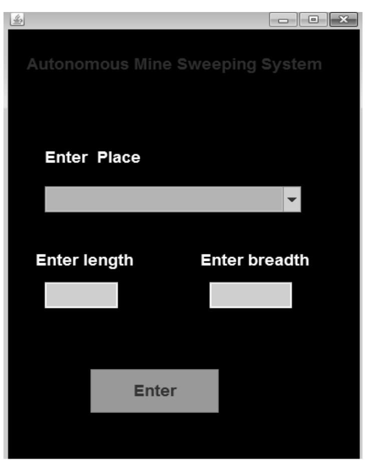

Figure 12: *First Page of the Java application*

Figure 12 shows the application made in java where the user can enter the dimensions for mine sweeping.

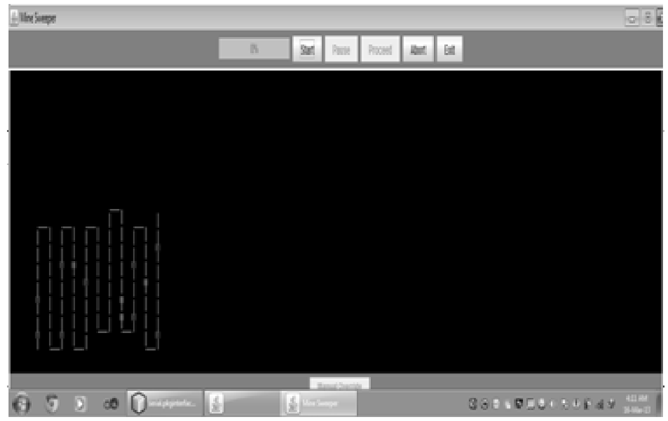

Figure 13: *Virtual Grid Display Page*

Figure 13 shows the screenshot of the second page of the Java application where the user can start the mine sweeping process and can also view the map of the movement of the robot.

14.1 Methods used for Obstacle Avoidance

14.1 Optical Flow Density Comparison Method.

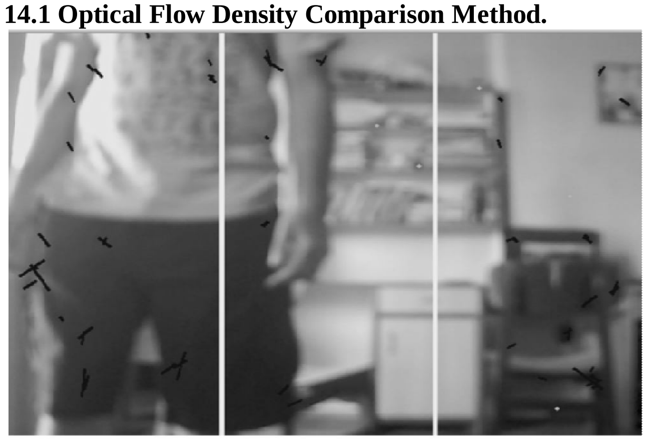

Figure 14: *Video Input for Density Comparison*

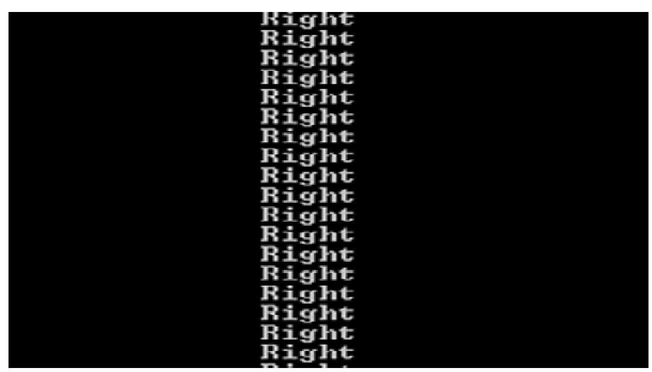

Figure 15: *The instructions given to robot with respect to the input*

The Video Input Screen of the Robot System is divided into 3 windows and based on the density of the optical flow in each window; a decision is taken as to where to move as shown in Figure 14. This output can be further improved by separating the optical flow vectors into still and moving which gives us more number of conditions and hence greater accuracy. The instructions to the robot are as given in Figure 15.[4]

14.1.2 Length of the Segmented Object Comparison Method.

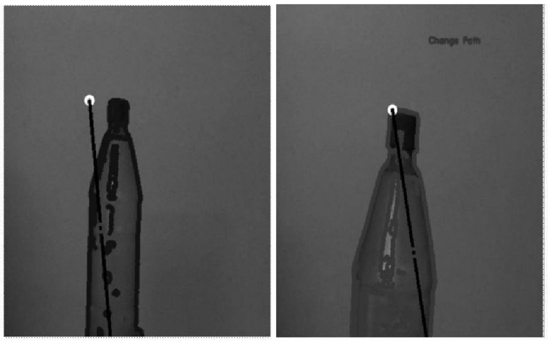

Figure 16: *(MinX, MinY) -Midpoint Method*

The midpoint of an object is found and a line is drawn from the midpoint to the (MinX, MinY) point of the object. The length of this line of the object increases as the object comes closer to the camera. In every new frame the new length is compared to the length of the object in the initial frame as shown in figure 16. Tracing multiple objects is easy.[1]

15. EDITORIAL POLICY

Optical flow method is used which was developed by Lucas Kanade in 1981[2].

16. CONCLUSION

This research's aim is to provide blueprints for developing a system, an autonomous robot, capable of identifying and classifying an object (a landmine specifically an anti-tank, a cluster bomb or unexploded ordnance). This paper investigated different sensing techniques, image processing techniques, and recognition and classification techniques. Based on the constraints and specifications required for the system, the best techniques were adopted for the design and implementation of the robot system: the designed system employs a camera, as a sensing technique, to capture real-time images of the scanned area. The captured images are then fed into a computing unit to be digitally processed.

17. ACKNOWLEDGEMENT

We express our sincere thanks to all those who have provided us valuable guidance towards the completion of this report as a part of the syllabus of the degree course. We express deepest sense of gratitude towards our guide Prof. G.Gandhi (H.O.D of Computer Department) who consistently helped us and gave valuable suggestions, guidance and encouragement. His innovative and practical ideas were very important in giving useful tips at every step. We express our special thanks and immense gratitude to project coordinator Prof. B. B. Gite and our respected staff members for inspiring us throughout the completion of our project report. The acknowledgement will be incomplete without a word of thanks to our Principal Dr. A.G. Kharat for providing us with the necessary facilities to work in the project.

REFERENCES

1. Gary Bradski, Adrian Kaehler. **Learning Vision Computer Vision with the OpenCV Library**, Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.
2. Bruce D. Lucas, Takeo Kanade. **An Iterative Image Registration Technique with an Application to Stereo Vision** in Proceedings of Imaging Understanding Workshop, pp. 121-130 (1981).
3. Ted Camus, **Real-Time Quantized Optical Flow in** Real-Time Imaging **3**, 71–86, 1997.
4. Etienne M´emin, Patrick P´erez. **Dense Estimation and Object-Based Segmentation of the Optical Flow with Robust Techniques, in** IEEE Transactions On Image Processing, VOL. 7, No. 5, MAY 1998.
5. Tomas Malmer. **Image Segmentation using GrabCut**. May 12, 2010.
6. **Atmega 32** Datasheet
7. **Xbee Pro** Datasheet
8. **L298** Datasheet
9. http://en.wikipedia.org/wiki/Optical_flow