Autonomous Navigation of Vehicle Using Visual Feedback

Rajeev Arya, B.V. Sai Kiran, Y.S.P. Kumar, J.C.P. Reddy, Ravi T. Nandula,
Mrinal Sen and R. Janarthanan

Abstract— *Planning the path of anAutomated Vehicle (AV) has been a continuous challenge for the researchers. A stream of solutions with prominent limitations has been suggested in the past decades. This paper presents a hardware implementation of amobile robot maneuvering autonomously guided by some popular well established algorithms like Dijkstras, Modified Binary Search, and Dynamic Programming. These algorithmsare implemented to find shortest path avoiding the obstacles bycontinuously monitoringand processing the visual data provided by the camera housed overhead.The functions or commands are calculated in the base computational station and send to the AV. The AV is programmed to attend the functions parallel to the computations. Finally the algorithms compared in terms of their complexity, processing time and limitations.*

Keywords--- *Mobile Robot,Image Processing, Path Planning, Optimization, Microcontroller*

I. INTRODUCTION

IN recent years, design of autonomous wheel robots [13] have received increased attention due to their potential capability and compelling applications in various sectors of automation. From academic level robotic researches to the sophisticated applications in mining[1], space organizations, and military purposes[2] using Global Positioning System (GPS) [3]employs considerable manpowerin the development of automated vehicles.

Navigation of an AV robot in an unknown environment has been studied by a number of researchers in the past decades. Various problemshave been risen in the navigations, path planning and motions control. The robots use efficient path planning algorithms to reach the target avoiding collision.This is the key issue in the mobile robot navigation.Recently various algorithms for navigation of autonomous mobile robots have been introduced.

Many optimization algorithms like Dijkstra [10], Binary search [11], Floyd Algorithm[6], Genetic Algorithm [4][5], Differential Evolution [7], Particle Swarm Optimization(PSO)[8], Ant Colony Optimization (ACO)[9]have been widespread.Out of them some employs simple search technique whereas others are based on the stochastic methods. Stochastic optimization algorithms have robustness but they are out weighted by their complexity in implementation and processing time. The successful implementation of evolutionary algorithms depends on efficient implementation of fitness function to valuate offs spring and robust crossover and mutation functions.

The primary requirement in AV path planning is robustness and speed. This can be served by simple Dijkstra Algorithm proposed by Dutch scientist Dijkstra for shortest path routing problems. Dijkstra is wide spread and can be easily implemented in any of programming languages.

The Binary Search can also be modified to meet the requirement of our problem space. Modified Binary Search can minimize the calculation time with increased obstacles.

Dynamic Programming can be used in planning the path. It can also be used to correct the errors introduced in real time movement of the robot in the challenging arena.

In this paper we utilized all the three algorithms-Dijkstra,Modified Binary Search and Dynamic Programming individually to plan a shortest path to reachthe target, avoiding obstacle detected by a static camera mounted at an altitude above the arena. The neighborhood is being analyzed by continuous images fed to the computer by the camera and the robot navigates through serial communication.

II. PROCESS DESCRIPTION

A. Robot Design

For the vehicle to respond to commands received from the controlling station there is an inevitable requirement blend of electronic and mechanical devices for precise maneuvers and accurate linear and rotational movement of the robot.

This narrows down the selection to robust combination of programmable microcontroller and high precision stepper motors. With economy and availability as selection parameters the microcontroller[15]has been chosen tobe 8051[P89V51RD2] and uni-polar stepper motors for ease. The microcontroller is programmed using native compilers like KEIL UV. The supporting circuitry required for the microcontroller and motors such as crystal oscillators and driving Darlington pairs are embedded on to a Printed Circuit

Rajeev Arya, Department of Electronics Engineering, Indian school of mines, Dhanbad. E-mail: rajeev.arya.ism@gmail.com

B.V. Sai Kiran, Department of Electronics Engineering, Indian school of mines, Dhanbad. E-mail: saikiran.ece13@gmail.com

Y.S.P. Kumar, Department of Electronics Engineering, Indian school of mines, Dhanbad.

J.C.P. Reddy, Department of Electronics Engineering, Indian school of mines, Dhanbad.

Ravi T. Nandula, Department of Electronics Engineering, Indian school of mines, Dhanbad.

Mrinal Sen, Department of Electronics Engineering, Indian school of mines, Dhanbad. E-mail: mrinalsen.ahm@gmail.com

R. Janarthanan, Department Information Technology, Jaya Engineering College, Chennai, India. E-mail: srmjana_73@yahoo.com

ISBN 978-93-82338-02-4 | © 2012 Bonfring

Board(PCB) mount on chassis of the vehicle with a separatepower source for electronic and mechanical systems to provide isolation and protect low power circuits form surge currents. A RS 232 cable through a MAX 232 is attached with the host computer to receive the commands and respond accordingly. The schematic circuit diagram and the image of the wheeled robot are provided in the appendix.

B. Image Acquistion

To be able to guide the robot amid its physical objects hindering its advancement to destination a better view of the space is required. Bird's eye view (Fig. 1) of the arena would suffice to precisely identify the robot and its counter obstacles and destination. This can be achieved by placing image acquisition sensor such as web camera above the space being controlled by computer so as to obtain visual information whenever required. MATLAB has been used to acquire image from the web camera.

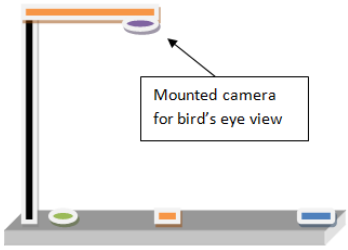

Figure 1: Mounting of Visual Sensor

C. Image Pre-Processing

The crude image data obtained by visual sensor is usually in RGB format (as in Fig. 2) and may increase computational complexity. This is backed by fact that each pixel has three arrays each consisting 8 bit data making it cumbersome to handle and process it for our application in most widely used image processing software package like MATLAB^{R}.

This is why it has to be downgraded suitably in binary image in form of matrices to be handled computationally robust and agile for application of efficient path planning algorithms later on in the process.

D. Path Planning

The preprocessed image or rather matrix thus obtained is suitable to be enhanced for our purpose to find optimum path. The feasible space is obtained by dilating image with structuring element which has same area as that of robot to find space where robot can be placed in any angle.

The space which is populated by obstacles provides a narrow way for the robot such space is identified by dilating the image with rectangle/disk with area same as robot to form feasible path. Then from such space a few random points are selected that are analogous to routers inDijkstra Algorithm. This has been shown in Fig. 4.

The boundary of the available path is obtained after dilation to check the possibility of collision. That is done using canny edge detection [12] (as shown in Fig. 5). The Gaussian transform can also be used for that but that could thicker the boundary thus increasing the boundary points which leads to larger processing time.

Now in order to find form feasible segments i.e.., active routers an adjacency matrix is generated with feasible nodes by checking collision of lines with boundary of feasible space obtained of after dilation.

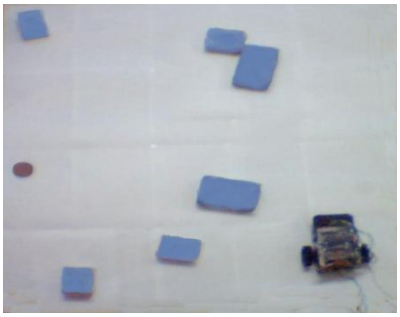

Figure 2: RGB Image obtained i.e prior to Preprocessing of Image

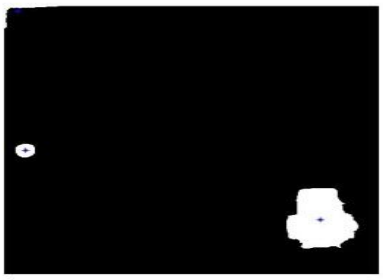

Figure 3: Identifying Source and Destination

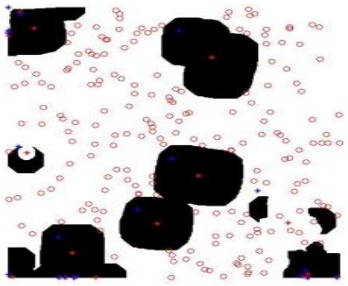

Figure 4: Dilated Image to obtain Feasible Space with Random Selection of Points

Figure 5: Boundary Detected Using Canny Edge Detection.

The algorithm can be easily implemented in any of programming languages, here for package uniformity it has been implemented in MATLAB.Three different algorithms has been implemented and they are discussed in the corresponding

ISBN 978-93-82338-02-4 | © 2012 Bonfring

sections.

1) Pseudo Code For Dijkstra's

```
Function Dijkstra(Graph, source):
For Each vertex v in Graph:// Initializations
        dist[v] := infinity ; // Unknown distance function from
                                 sourceto v
        previous[v] := undefined ;// Previous node in optimal
                                 //path from source
End For
dist[source] := 0 ;  // Distance from source to source
Q := the set of all nodes in Graph ;   // All nodes in the graph
                                       //are unoptimized
WhileQis not empty:  // The main loop
        u := vertex in Q with smallest distance in dist[] ;
        if dist[u] = infinity:
                break;// all remaining vertices are
                 // inaccessible from source
        end if;
        u=target;
End While
```

2) Modified Binary Search With Minimum Expansion

In order to suit for the requirements of input matrix of the binary search, dilated image is complemented. Input Image matrix input to the binary search with first element of matrix as the source and last element as destination. Required number of steps to reach the destination without considering the obstacles are stored in Heuristic matrix. Matrix expansion is done in such a way that first matrix is considered such that number of steps it had already covered summation number of more steps it has to take is minimum such that every matrix value is not expanded making it to take less time to compute. The pseudo code for the Modified Binary Search is as follows.

```
Input: size (a, b)  //Consider the dilated image as input
Heuristic (i,j): a +b- (i + j)
Goal: final destination =x,y
G=0
F=G + heuristic
Open= [G 1 1]
Start
{
        For i= 1 to a
                For j 1tob
                        G=G+1
                        F=G + heuristic (i , j)
                        Open= [F i j];
                        Remove the expanded matrix point
                        Add the expanded matrix in to open
                        array. Sort open array with respect
                        to F
                End For
        End For
}
```

3) Dynamic Programming

Starting from the source point it expands in uni-direction by updating the value matrix, care is taken such that it doesn't cross boundaries. If sum of succeeding step value and cost of step is less than current step then this step is considered. Now entire matrix is been expanded so as to give shortest path from every point on image to destination.Pseudo code is as follows.

```
input; //Consider the dilated image as input
start point(init);
goal;
cost of step;
delta= [-1, 0 ;
       0, -1; %# go left
       1, 0 ; %# go down
       0, 1 ]; %# go right
value;
change=1;
While (change)
     For X=1:size of rows
          For y=1:size of columns
               If goal is reached
                    If value(x,y)>0
                         value(x,y)=0
                         change=1;
                    End If
               Else If input(x,y)==0
                  x2 = x + delta(i,1);
        y2 = y + delta(i,2);
                    If (x2 > 0 && x2 <= number of rows
                         &&y2>0 && y2 <= number of
                         columns)
                  v2=value(x2,y2)+cost of step;
        If v2<value(x,y)
                  change=1;
                  value(x,y)=v2;
                         End If
                    End If
               End If
          End For
     End For
End While
```

The feasible shortest collision free path obtained is as show in Figure. 6.The shortest path obtained is in form of vertices and nodes in software it cannot be communicated directly in physical world.For the linear and rotational movement of the robot in physical world it has to transform into combination of distances and angles to be turned. So we need to go for some image post processing. The distances between individual hops and angles to be turned are transformed into stepping sequences and are serially communicated to the on board microcontroller.

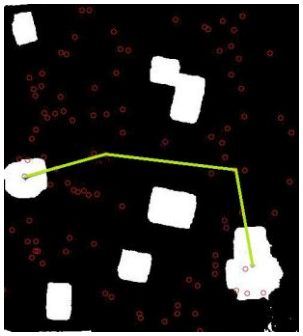

Figure 6: Feasible Path (Avoiding Collision) found by Dijkstra Algorithm

ISBN 978-93-82338-02-4 | © 2012 Bonfring

E. Image Post Processing

Therefore form the nodes, obtained from path planning algorithms, series of instructions in the form of stepping sequence for stepper motor and duration to turn on are to be fed into microcontroller through standard RS-232 serial communication incorporated within MATLAB.

The pseudo code of the implemented program in the microcontroller is given below.

```
While True
    For I=1 to 4
        While RI=0
        End while
        RI:=0
        Input(i):=Serial Input Buffer
    End for
    Sum:=input(2)
    For i=2 to 3
        Sum:=10*Sum+Input(i+1);
    End for
    Switch Input(1):
        Case 'f': both the motors move forward.
        Case 'b': both the motors move backward
        Case 'l': both the motors move left.
        Case 'r': both the motors move right.
    While Sum Not 0
    End while
End while
```

III. RESULTS AND DISCUSSIONS

The Dijkstra's Algorithm implemented does depend upon the randomly selected nodes form the available space for movement. Thus it had been run repeatedly. The results for three consecutive run has been presented in Table I.

Table 1: Processing Details of Dijkstra's Algorithm

Sl.no.	Time	Distance	Path
1	8.27308 sec	951.2542	[1 38 102]
2	6.313062 sec	951.5534	[1 60 78 102]
3	5.925188 sec	953.7612	[1 41 102]

Table 2: Comparison of Processing Details of Binary search and Dynamic Programming

Sl.No	Input Matrix Size	Evaluation Time for Binary Search[Sec]	Evaluation Time For Dynamic Programming[Sec]
1	[64,64]	1.645734	48.368890
2	[64,64]	1.724872	40.643238
3	[120,120]	7.831553	355.742824

Out of all the algorithms implemented Dijkstra's algorithm has shown the best possible shortest path with minimum computational time.

Modified Binary Search produces only one path from source to destination, whereas Modified dynamic search produces possible shortest path from every point on image to destination.

When implemented simultaneously, Modified Binary search takes less time were as Modified dynamic takes a lot of time (as shown in Table II). Modified binary search uses to give path initially and during motion if robot deviates from path it can be retraced using Modified dynamic search. By this error can be compensated.

F. Complexity

Complexity of the Dijkstra's Algorithm isa function of $(|E|+|V|^2)$. Where E is possible number of paths between generated Random points and V represents total number of random points generated. In the shownresults the variation of time has occurred due to the change in E in different cases with V is equal to 100.

In Modified Binary Search and Modified Dynamic Programming, Size of the input matrix decides the number oftimes the recursion has to be done. More area covered by obstacles results in less number of matrix expansions to be made.

IV. CONCLUSION

The performance of single source path planning algorithm Dijkstra algorithm for its robustness is tested along with other techniques like binary search and dynamic programming.

The self sustained path planning autonomous robot can be proved useful and its mettle can be tested in situations where an optimum collision free path has to be planned in areas without human intervention like tunnels .It can also be proved helpful for blind people.

REFERENCES

[1] Dr.Peter Ridley,Dr Peter Corke,"Autonomous control of underground mining vehicle "Queensland uiversity Brisbane,Australia,CSRIO manufactruing science and technology Kenmore,Australia.

[2] Maxim Likhachev school of Computer Scienceu niversity o Pennsylvani Philadelphia, PA,Dave Ferguson Intel Research Pittsburg 4720 Forbes Ave Pittsburgh, PA,"Planning Long Dynamically-Feasible Maneuvers for Autonomous Vehicles".

[3] Kourosh Rahnamai; Kevin Gorman; Andrew Gray; Payman Arabshahi,"Formations of Autonomous Vehicles Using Global Positioning Systems

[4] (GPS)",Jet Propulsion Laboratory California Institute of Technology Pasadena, CA 91109.

[5] Chang Wook Ahn, R. S. Ramakrishna,"A Genetic Algorithm for Shortest Path Routing Problem and the Sizing of Populations".

[6] Sachith Abeysundara*, Baladasan Giritharan+, Saluka Kodithuwakku "A Genetic Algorithm Approach to Solve the Shortest Path Problem for Road Maps".*Department of Statistics and Computer Science, Faculty of Science, University of Peradeniya, Sri Lanka .

[7] S.saiful azhar,A.Ghaney liew,D.Suhardy,K.Farizul Hafiz,M.Narzy Salleh."International Conference on Mathematics and natural sciences.",Institute Teknologi Bandung november-29-30,2006.

[8] Hélder Santos, José Mendes, P. B. de Moura Oliveira and J. Boaventura Cunha," path planning optimization using the differential evolution algorithm" Universidade de Trás-os-Montes e Alto DouroDepartamento de Engenharias Quinta dos Prados, 5000 Vila Real – Portugal.

[9] Yanfang Deng, Hengqing Tong,"Dynamic Shortest Path Algorithm in Stochastic Traffic Networks Using PSO Based on Fluid Neural Network". School of Science, Wuhan University of Technology, Wuhan, China

ISBN 978-93-82338-02-4 | © 2012 Bonfring

[10] Karl O. Jones, André Bouffet,"comparison of ant colony optimisation and differential evolution". International Conference on Computer Systems and Technologies - CompSysTech'07
[11] "Introduction to Algorithms " by Thomas.H.Cormen, Charles E Leiserson, Ronald L Rivest, Clifford Stein.MIT Press1990.
[12] "Classic Data Structures" by Debasis Samanta.Princeton Hall of india2006.
[13] DigitalImageProcessing usingmatlab"By.Rafael.C.gongzalezland.Richard.e.Woods.Stevenl.Eddi ns.
[14] Jens-Steffen gutmann,Masaki Fukuchi,Masahiro fujita,"Real Time Path Planning for humanoid Robot Navigation" Intelligent system research laboratory sony corporation,tokyo,Japan.
[15] Sung-Min Han and Kang-Woong Lee,Mobile robot navigation using circular path planning algorithm.,School of electronics ,telecommunications and computer engineering korea.,aerospace university,seoul.,korea.International conference on contro ,autonomous and system 2008 oct 14-17-2008 in COEX,seoul,korea..
[16] Ajay.v.Desmukh."Microcontrollers[theory and applications]"Tata Mcgrawhill Publishing Co.Ltd published 2005.

II. APPENDIX

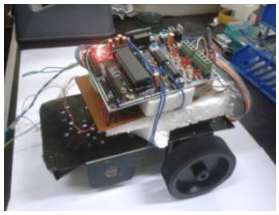

Figure 7: Image of the AV

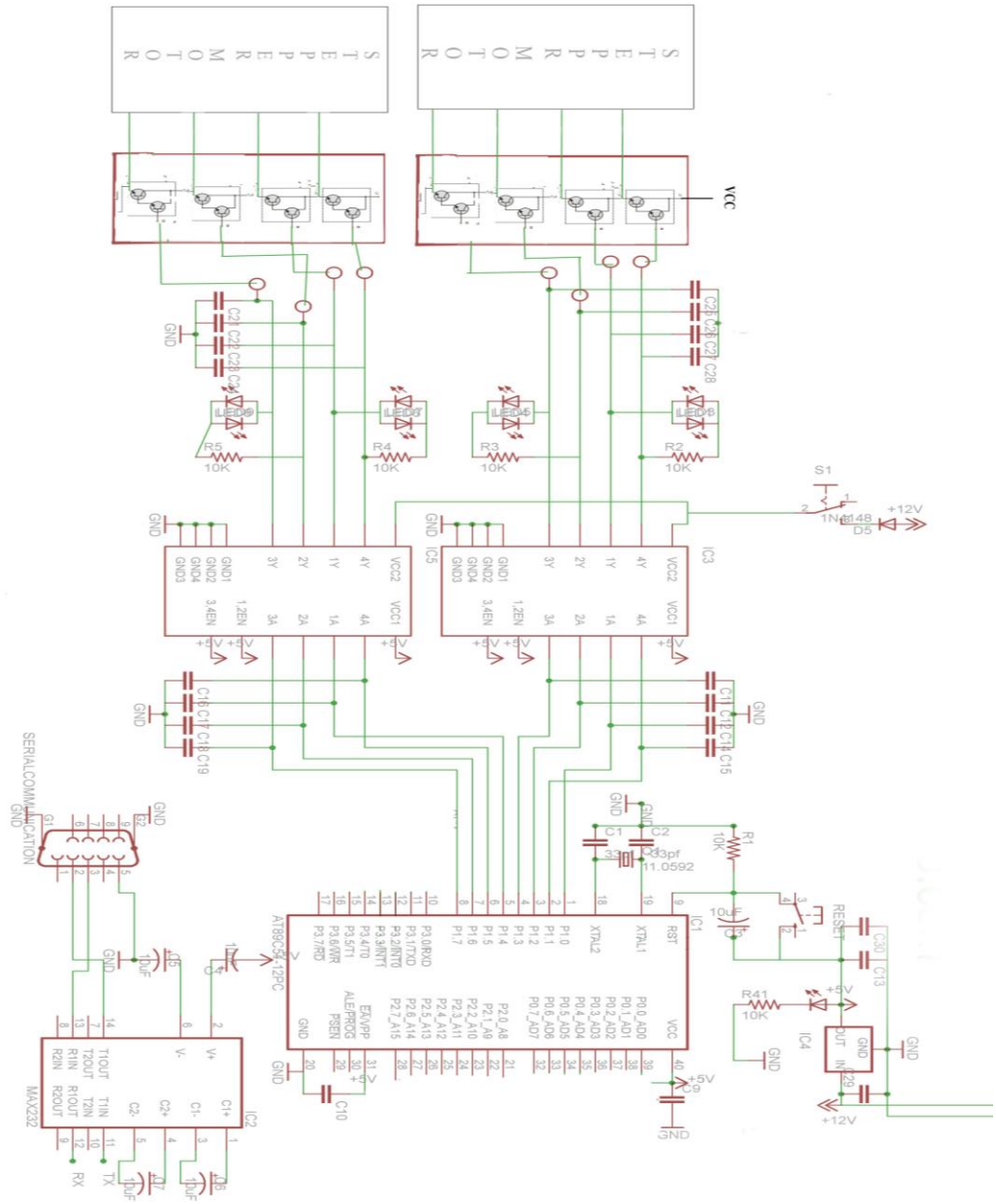

Figure 8: Schematics of the AV Circuit