ISSN 2395-1621

Development of an Obstacle Avoidance Algorithm and Path Planning Algorithm for an Autonomous Mobile Robot

[#1]S. S. Parate, [#2]J. L. Minase

[1]smrutikaprt400@gmail.com
[2]jlminase.scoe@sinhgad.edu

[#1]Appearing in M.E. (Mechatronics), Department of Mechanical Engineering,
Sinhgad College of Engineering, SavitribaiPhule Pune University
[2]Assistant Professor, Department of Mechanical Engineering,
Sinhgad College of Engineering, Pune, India

ABSTRACT

Robot mobility is a key feature for any mobile robot. Path planning and Obstacle avoidance are the two important aspects of autonomous mobile robot navigation. Path planning in mobile robots depends on the static or dynamic environment in which it operates. Based on the availability of information about environment, there are two categories of path planning algorithms, namely global and local. In this paper, the well-known heuristic A-star (A*) algorithm is implemented for global path planning to make the mobile robot navigate through known static obstacles and find the shortest path from an initial position to the target position quickly and safely. If the mobile robot detects any unknown static obstacles in the shortest path using sonar sensors mounted on its front panel, the dynamic steering algorithm is used to avoid the obstacles. Then, the mobile robot will update the information about the surrounding environment and recalculates the new shortest path by avoiding the unexpected obstacles in its way from new initial position to the target position, if any. Finally, several simulations and experiments are performed to indicate the practicability of the proposed system.

Keywords— **Mobile Robot, Global Path Planning, Obstacle Avoidance, A-Star (A*) Algorithm, Dynamic Steering Algorithm.**

ARTICLE INFO

Article History

Received :18th November 2015

Received in revised form :

19th November 2015

Accepted : 21st November , 2015

Published online :

22nd November 2015

I. INTRODUCTION

Recently, automation techniques are applied widely in many fields such as factory automation, office automation, hospital automation, mining automation etc. The purpose of automation is to save both time and manpower and to improve the service quality provided to the customer. Mobile robots have the capability to move around in their environment and are not fixed to one physical location. Mobile robots can be autonomous which means that they are capable of navigating through an uncontrolled environment without the need of physical or electro-mechanical guidance devices.For an autonomous motion, the robot should have the ability to localize its current position, perform path planning for the future movement and move to the next position as expected by avoiding the unknown stationary obstacles in its way. Moreover, to reduce the certain performance criterion such as processing time, distance and energy consumption, the planned path is naturally required to be optimal with the shortest length;distance being the most commonly adopted criterion.

The path planning problem for the mobile robot is typically formulated as follows: given a mobile robot and the description of an environment, plan a collision free path between two specified locations which satisfies certain optimization criteria (i.e., shortest cost path). To solve path planning problems, researchers classify various methods based on two factors: type of environment (i.e., static or dynamic) [1] and path planning algorithms (i.e., global or local) [1].The static path planning refers to environment which contains no moving objects and dynamic path planning refers to environment which contains moving obstacle.In global path planning, the complete information about stationary obstacles and trajectory of moving obstacles are known in advance. Whereas, in local path planning, the complete information about environment is not available in advance, mobile robot gets information through sensors, as it moves through the environment.

II. Related Work

Due to rapid increase in automation technology, the research on the mobile robot has been raised tremendously. Many approaches have been developed over the year for global and local path planning problems. The path planning algorithm can be categorized into classical approach and evolutionary approach. A classical approach for formulating and solving the path planning problem is the configuration space (C-space) approach [2].The central idea of this approach is the representation of the robot as a single point. As robot is reduced to a point, each obstacle is enlarged by the size of the robot to compensate. Using C-space as the fundamental concept, there are many path planning approaches like roadmap approach, cell decomposition approach, etc. The roadmap methods are Visibility Graph [2] and Voronoi diagram [3]. The Visibility Graph methodin the robot motion planning is used when the geometry of the environment is known.But, this method is efficient only in sparse environments as the number of roads is dependent on the number of polygonal obstacles and their edges. Also, the obtained path is often very close to the obstacles and thus, may lead to crashing of the robot. The survey on fundamental geometric data structure to explain the use of Voronoi diagrams in global path planning is presented in[3]. It can generate the origin of the robot path points and reduces the path search time. But the distance detected by the sensors affect the normal robot path planning. Also, robot should not be detected more than one obstacles at the same time. The cell decomposition approach [5] computes the C-space of the mobile robot decomposes the resulting space into cells and then searches for a route in the free space cell graph.The grid method [6] is popular cell decomposition approach in which the grids are used to generate the map of the environment. The main difficulty is how to find the size of the grids, the lesser the size of grids, the more accurate will be the representation of the environment. But, when the space increases, this method will lead to sudden increase in storage space that required, and decision was made slowly. The Quadatree method [7] is another algorithm for searching the collision free path for the robot. It takes up a lot of space. Most of the memory will be taken up by the links. They are shift sensitive in that their space requirements are dependent on the position of the origin.

Classic approaches, though found to be effective, take more time in the determination of feasible collision-free path [1]. It becomes more complicated when the environment is dynamic. These drawbacks make the classic approaches to be incompetent in complex environments. Hence, many evolutionary approaches such as Genetic Algorithm (GA), Stimulated Annealing (SA), Graph-Search Algorithms etc. are employed to solve the path planning problem quickly.An adaptive fuzzy controller based on GAis adopted in [8] to process the path planning of the mobile robot. With the increasing number of obstacles in the environment and changes of the moving speed and the direction of the robot and the obstacles, the size of the problems to be solved increases rapidly.To overcome this drawback, Simulated Annealing algorithm [9] is employed for collision-free path among static polygonal obstacles in C-space setting. It is a type of heuristic random search method. The major drawback of this algorithm is that the shape of the robot is ignored though the dimensions of the obstacles are considered for calculation. The calculations are more complex than the GA.The graph search algorithms are based on node-edge notation but this notation lacks when a system like GPS gets an image frame, converts it to a map matrix and uses this map matrix as the grid based map. In these situations using matrix notation gives the advantage of simplicity and comprehension.Yuksel*et al.* (2006) presents the comparative study of different path planning algorithms for the shortest path problems such as Breadth First Search (BFS) algorithm, Dijkstra algorithm [10,11] and A* algorithm [12-14].

Many researchers have studied the usage of A* algorithm for global path planning which is applicable to the situation that the information about the environment is already known. It is also applicable to the quadratic programming of the path. It is faster than Dijkstra's algorithm since it uses heuristic function. It uses less memory space than Dijkstra algorithm. It gives an optimal path for itself by avoiding the obstructions in its way and minimizing a cost such as time, energy and distance.

For obstacle avoidance of the mobile robot, the primary objective of the mobile robot is to avoid unknown static obstacles from starting position to the goal location safely and smoothly in the constrained environment. There are many obstacle avoidance algorithms have been developed, such as edge-detectionmethod, Potential Field method [15] (PFM), Virtual Force Fieldmethod (VFF) [15], Vector Field Histogram method[16] (VFH),Fuzzy Controller method[17] (FC), the combination of FuzzyController, Artificial Neural Network (ANN) [18], etc. In all those obstacle avoidance algorithms, the algorithms which use the sensor data directly will be disturbed more or less, so they are more likely to make the false actions. The algorithms which adopt certainty grid method for obstacle representation are more robust, such as the VFF method and VFH method. Certainty grid method is suggested by Elfes in 1980's,who used the certainty grids for mobile robot off-line globalpath planning. Vision-based navigation by a mobile robot can avoid the stationary obstacles based on Adaptive Thresholding [19]. The author partitioned the difference edge image into five vertical regions to find a direction for safe passage in the presence of obstacles.If the sum of the number of pixels in each of the five regions exceeds the threshold value, then that region will be regard as a dangerous for obstacle avoidance.But the measurement of the distances between robot and obstacle is not easy by using image processing methods. Hence, many approaches to the distance measurement apply the ultrasonic sensory devices. So, to avoid the unknown static obstacles, the dynamic steering algorithm based on sensorial information is presented in this paper.

In this paper, the path planning problem for the mobile robot is solved using the well-known heuristic A* algorithm in the global environment. Also, the dynamic steering algorithm [20] based on sensorial information is used to avoid the unknown static obstacles in the shortest path obtained from A* algorithm.

The rest of the paper is outlined as follows. Section 3 describes theproblemformulation for path planning and obstacle avoidance of the mobile robot.Section 4 illustrates the methodology of the path planning and obstacle avoidance algorithm for the mobile robot.Section 5 presents the experimental resultsto indicate the practicability of the proposed system. Section 6 concludes the paper.

III. PROBLEM FORMULATION

In the grid map environment, the mobile robot has plan the optimal and shortest path from starting position to the destination by avoiding the presence of obstacle in the environment. The formulation considers the evaluation of the next position of the robot from the current position in the given environment map. The path planning and obstacle avoidance algorithm are based on the following assumptions to compute the shortest length path.

1) The current position of the mobile robot is known with respect to the given coordinate system.
2) The goal is fixed and known which may represent any cell of the grid map.
3) The unknown static obstacles can be present on any cell of the grid map.
4) The mobile robot have fixed set of actions for motions. It can select only one action at a given time.
5) The path planning problems for the mobile robot is executed in steps until it reaches the destination.

IV. METHODOLOGY

This section explains the methodological approach for obstacle avoidance and path planning of an autonomous mobile robot shown in figure 1.The purpose of this robot is to find an optimal and shortest path from starting position to the destination quickly by avoiding the unknown static obstacles of any dimensions in its way. The predefined environment map should be necessary to draw on the floor in the fixed pattern. Then, the A* algorithm should be used for path planning of the mobile robot and the dynamic steering algorithm is used to avoid the unknown static obstacles in the shortest route found from the path planning algorithm.

Fig. 1 An autonomous mobile robot

A. The pre-defined environment map

The mobile robot navigates on 2-dimension grid which is dividedinto number of cells that are identified with Cartesian coordinates. It is necessary that all the cells should be of the same size but not necessary that the robot knows the dimension *(n,m)* of the grid where n and m are the number of rows and columns respectively. Depending on the size of the robot and the sensitivity of the sensors, different cell sizes and contrast between plane and grid are possible. The robot start its motion from the pre-defined starting position to destination in this pre-defined environment.

B. A Star (A) Algorithm for Path Planning*

The path planning aims to let the mobile robot travel on a collision-free path. A* algorithm is an artificial intelligence and informed search algorithm represented by a tree or a graph. It is formally defined as best-first, graph search algorithm that finds the least-cost path from a given initial node to the goal node. Least cost path is the path which determine the most cost-effective route between a source and destination. Cost can be a function of time, distance or other criteria that is defined by the user. The objective of this algorithm is to identify a specific goal in the process of generating new nodes. This algorithm has three attributes in addition to holding the map location. These are fitness, goal and heuristic commonly known as *f, g,* and *h*respectively which are defined as follows:

1) *g* is the cost of getting from the start node to the current node i.e. the sum of all the values in the path between the start and the current node
2) *h* stands for heuristic which is an estimated cost from the current node to the goal node (usually the straight line distance from this node to the goal)
3) *f* is the sum of *g* and *h* and is the best estimate of the cost of the path going through the current node. The lower the value of *f*, the more efficient the path.

The *g* cost function can be computed but the h cost function can just be estimated. The purpose of *f, g,* and *h* is to quantify how promising a path is up to the present node.The evaluation function at any point *n* is given by equation 1.

$$f(n) = g(n) + h(n) \quad (1)$$

There are several methods for this estimation such as Euclidian Distance, Manhattan Distance, Chebyshev Distance, etc. Manhattan method is the most used method which is given by equation 2.

$$h(currentcell)=abs(currentX-goalX)+abs(currentY-goalY) \quad (2)$$

Additionally, A* maintains the two lists arrays:OPEN LIST and CLOSED LIST. The OPEN LIST contains all the nodes in the map that have not been fully explored yet, whereas the CLOSED LIST consists of all the nodes that have been fully explored. The step-by-step procedure of A* algorithm is summarized as follows:

1) Let P = starting point.
2) Assign *f, g* and *h* values to P.
3) Add P to the OPEN LIST. At this point, P is the only node on the OPEN LIST.
4) Let B = the best node or parent node from the OPEN LIST (i.e. the node that has the lowest f-value).
   a) If B is the goal node, then quit i.e. a path has been found.
   b) If the OPEN LIST is empty, then quit i.e. a path cannot be found
5) Let C = a valid node connected to B.
   a) Assign *f, g,* and *h* values to C.
   b) Check whether C is on the OPEN LIST or CLOSED LIST.

i) If so, check whether the new path is more efficient (i.e. has a lower *f*-value). If so, update the path.
ii) Else, add C to the Open list.

c) Repeat step (5) for all valid children of B.

6) Repeat from step (4).

C. The Dynamic Steering Algorithm for Obstacle Avoidance

For the obstacle avoidance of the mobile robot, eight ultrasonic sensors which are set separately on every 30° on the froe half of the circumference of the mobile robot and six sharp infrared (IR) sensors are mounted on the front plane of the mobile robot to detect the distances between the mobile robot and obstacles in every direction. These sensors are connected with the Hydra board. The Hydra board is connected to Atom-PC for motion control of the robot which receives command from the PC for speed control and path. It executes these commands using motor control board Cheetah-CB and Cheetah.

A particular path is designed having the staring position and the goal position. The learning procedure of the dynamic steering algorithm for obstacle avoidance to avoid the unknown obstacles in front of the mobile robot with the help of sensors in the constrained environment can be explained from the flowchart as shown in figure 2.

1) When the system gets started, the robot starts its motion.
2) The robot will determine whether the given distance is covered. If not, it will move along the given path.
3) Then the robot will determine whether the unknown obstacle is detected with the help of sensors mounted on it. If not, go back to step 2 and repeat the procedure.
4) If the unknown obstacle is detected, then check whether the detected obstacle is at left side or right side.
5) If the detected obstacle is at left side, the robot will take the right turn. Again it will check whether the obstacle is present at left side.
   i) If yes, then the robot will move straight till it reaches a goal point.
   ii) If no, the robot will take the left turn and move the distance equal to the width of an unknown obstacle. Then the robot will take the right turn and subtract the length of an obstacle from the total distance to be moved.
6) If the detected obstacle is at right side, the robot will take the left turn. Again it will check whether the obstacle is present at right side.
   i) If yes, then the robot will move straight till it reaches a goal point.
   ii) If no, the robot will take the right turn and move the distance equal to the width of an unknown obstacle. Then the robot will take the left turn and subtract the length of an obstacle from the total distance to be moved.
7) After step (5) or step (6), go back to step (2) and repeat the procedure till the given distance is covered.
8) If the given distance is covered, the robot will stop i.e. it will reach to the goal point.

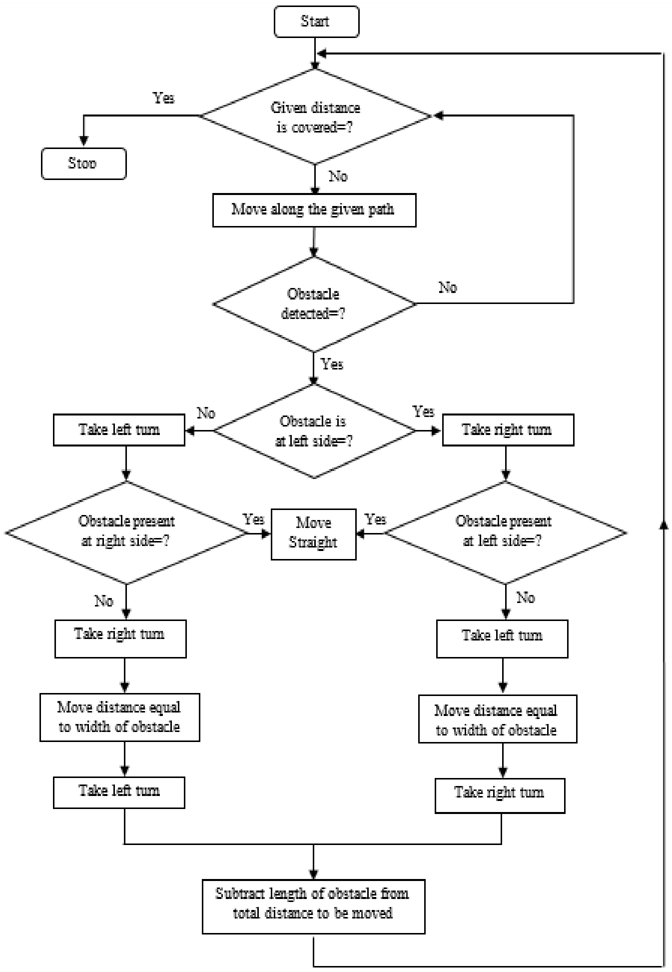

Fig 2. Flowchart of an Obstacle Avoidance Strategy

V. EXPERIMENTAL RESULTS

The experimental validation is done on a robotic platform available in Mechatronics Lab of Department of Mechanical Engineering at SCOE. The algorithms are implemented on the 8x7 grid matrix which is divided into 40cmx40cm cells as shown in figure 3. An autonomous mobile robot encounters 56 junctions. Figure 4 shows the complete pre-defined environment having start and goal position with known static obstacles respectively. This information is fed to the central station (desktop) which will calculate the shortest path. Then, the mobile robot will start moving through the assigned shortest path quickly. If the robot detects any unknown obstacles of any dimensions, it will avoid them without stopping. This information will be communicated to the central station for updating for further calculation of the shortest path. Figure 2 shows the 8x7 grid map of size 40cmx40cm were arranged on the ground. Figure 3 shows the complete pre-defined environment having start and goal position with known static obstacles respectively.

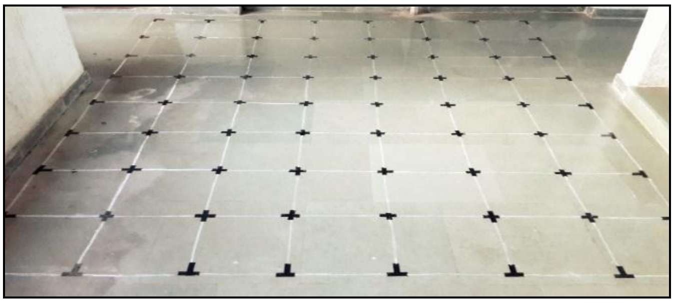

Fig. 3 8x7 Grid Map

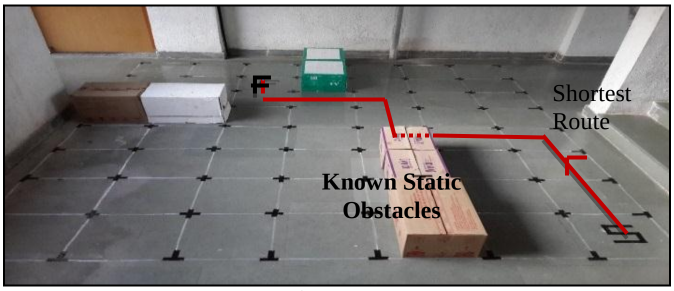

Fig. 4The pre-defined Environment Map

The central station is used to calculate the shortest path using A* algorithm by avoiding the unknown static obstacles. The inputs to the central station includes the start position (represented by S), target position (represented by F) and the obstacle matrix to indicate the location of static obstacles (represented by O) in the 8x7 grid map shown in figure 5.

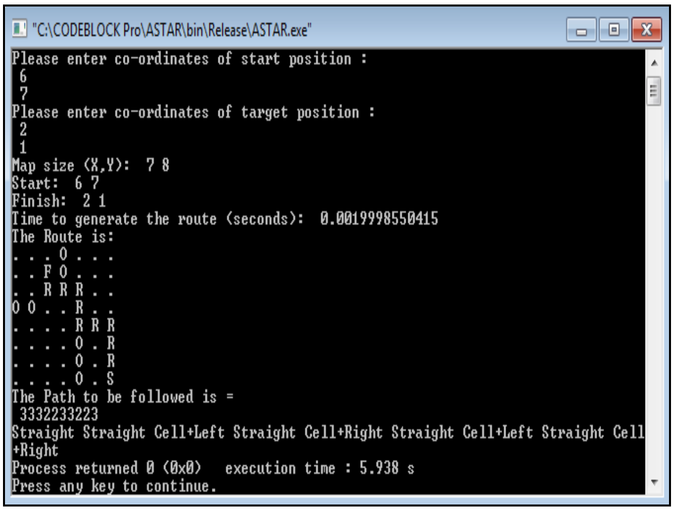

Fig. 5Output Window at Central Station

The outputs to the central station includes the computation time (i.e. time to generate the shortest route from start position to the destination), grid to grid movement (i.e. the path to be followed represented by R) and the commands to the mobile robot. The execution time to complete the process is 5.938 s.

Figure 6 shows the experimental setup for path planning algorithm in the pre-defined environment. The starting point, goal point and the known and unknown static obstacles are initialized shown in figure 6 (a). Robot will avoid the unknown static obstacles in its shortest path with the help of dynamic steering algorithm shown in figure 6 (b). Again it will move to the assigned shortest path shown in figure 6 (c). Finally, it will reach the goal safely and quickly shown in figure 6 (d).

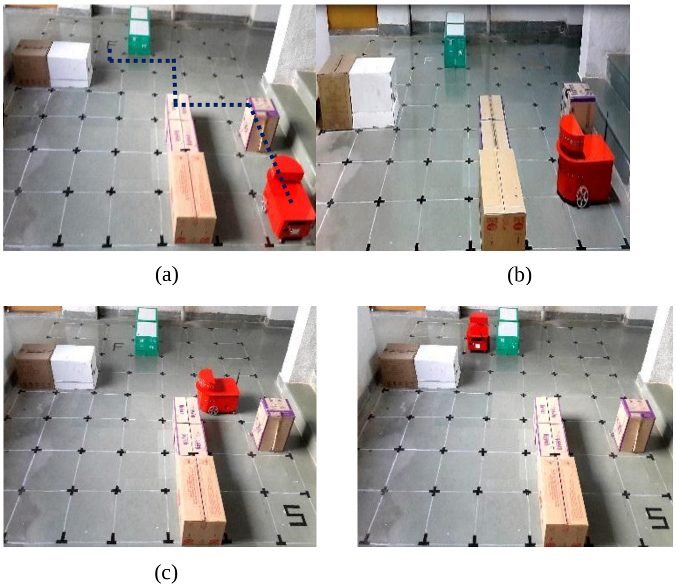

Fig. 6 shortest path planning scenario in pre-defined grid map with known and unknown static obstacle

VI. Conclusions

The mobile robot was designed to overcome the problems associated with the path planning, obstacle avoidance and navigation control. The proposed work gives an optimal path for the mobile robot by avoiding the obstructions in its way and minimizing a cost such as time, energy and distance. A* algorithm is faster than Dijkstra algorithm since it uses heuristic function and less memory space. It is proposed to assist in the constraint environment by combining information decision making system, ultrasonic and infrared sensing system to get the smoother, faster and better performance for avoiding the real time unknown stationary obstacles. The robot contributes the novelty in implementing the dynamic steering algorithm for the mobile robot to avoid the unknown static obstacles in the pre-defined grid-based environment map based on the sensorial information.

Acknowledgment

I would like to express a sincerest gratitude to the guide Dr. J. L. Minase for his dynamic and valuable guidance. I also extend my thanks to the head of P. G. Studies Dr. Y. P. Reddy for their valued support.

References

[1] NacerHacene, BoubekeurMendil, “Autonomous Navigation and Obstacle Avoidance for a Wheeled Mobile Robots: A Hybrid Approach”, *International Journal of Computer Applications*, Vol. 81, No.7, 2013

[2] Lozano-Perez, Tomas, Wesley, Michael A, “An Algorithm for Planning Collision-free Paths among Polyhedral Obstacles”, *Communications of the ACM*, Vol. 22, No. 10, pp. 560-570, 1979.

[3] Franz Aurenhammer, “Voronoi Diagrams – A Survey of a Fundamental Geometric Data Structure”, *ACM Computing Surveys*, Vol. 23, No. 3, pp. 345-405, 1991.

[4] Lozano-Perez T, “Spatial planning: A configuration approach”, *IEEE Trans. on Computers*, Vol. 32, No. 2, pp. 108-120, 1983

[5] V. Boschian, A. Pruski, "Grid Modeling of Robot Cells: A Memory-Effcient Approach", *Journal of Intelligent and Robotic Systems*, pp. 201-223, 1993.

[6] H. Samet and R. E. Webber, "Storing A Collection of Polygons using Quadtrees", *ACM transactions on graphics*, Vol. 4, No. 3, pp. 182-222, 1985.

[7] Wang Y, Mulvaney D, Sillitoe I, "Genetic-based mobile robot path planning using vertex heuristic", *Proc. of the IEEE International Conference on Cybernetics and Intelligent Systems*, pp. 463-468, 2006.

[8] Miao H, Tian YC, "Robot path planning in dynamic environments using a simulated annealing based approach", *Proc. of the tenth International Conference on Control, Automation, Robotics and Vision*, pp. 1253-1258, 2008.

[9] Zhang Fuhao, Liu Jiping, "An Algorithm of Shortest Path based on Dijkstra for Huge Data", *6th International Conference on Fuzzy Systems and Knowledge Discovery*, Vol.4, pp. 244-247, 2009.

[10] Jia-Heng Zhou, and Huei-Yung Lin, "A Self-Localization and Path Planning Technique for Mobile Robot Navigation", *Proceedings of the 8th World Congress on Intelligent Control and Automation*, pp. 694-698, 2011.

[11] T. Goto, T. Kosaka, H. Noborio, "On the Heuristics of A* or A Algorithm in ITS and Robot Path Planning", *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS*), Vol. 2, pp. 1159-1166, 2003.

[12] Pradipta Kumar Das, Amit Konar, RomeshLaishram, "Path Planning of Mobile Robot in Unknown Environment", *International Conference (ACCTA)*, Vol.1, No. 2, pp. 26-31, 2010.

[13] Xiang Liu, Daoxiong Gong, "A Comparative Study of A-Star Algorithms for Search and Rescue in Perfect Maze", *International Conference on Electric Information and Control Engineering (ICEICE)*, pp. 24-27, 2011.

(d)

[14] Koren Y, Borenstein J, "Potential Field Methods and their Inherent Limitations for Mobile Robot Navigation", *IEEE Int. Conf. Robotics and Automation*, pp. 1398 - 1404, 1991.

[15] Borenstein J, Koren Y, "The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robots", *IEEE Int. Conf. Robotics and Automation*, pp. 278-288, 1991.

[16] K. Wu, C. Chen and J. Lee, "Genetic-based Adaptive Fuzzy Controller for Robot Path Planning", *IEEE Fuzzy System*, Vol. 3, pp. 1687-1692, 1996.

[17] Vadakkepat P, Tan KC, Ming-Liang W (2000),"Evolutionary artificial potential fields and their application in real time robot path planning"*Proceedings of the 2000 Congress on Evolutionary Computation*, pp. 256-263.

[18] Ohya, A. Kosaka and A. Kak, "Vision-Based Navigation by a Mobile Robot with Obstacle Avoidance using Single-Camera Vision and Ultrasonic Sensor," *IEEE Transactions on Robotics and Automation*, Vol. 14, No. 6, pp. 969-978, 1998.

[19] VivekHanumante, Sahadev Roy and SantanuMaity, "Low Cost Obstacle Avoidance Robot",*International Journal of Soft Computing and Engineering (IJSCE)*,Vol. 3, pp. 2231-2307, 2013.

