Abstract

This paper presents an approach for outdoor robot navigation using Distance Transform Methodology (DT). DT has been used in path planning for indoor robot research and can also be used to perform obstacle avoidance simultaneously. However, when it comes to outdoor robot navigation, the operating environment becomes too large and DT becomes inefficient when performing both the tasks of path planning and obstacle avoidance. Usually both tasks have to be decoupled and DT is normally only used for path planning. The newly proposed DT methodology solves this problem by optimising the DT algorithm and reducing the processing area. Simulation and actual tests had also been carried out on an autonomous mobile robot to verify that the DT methodology can integrate both the tasks of path planning and obstacle avoidance and yield encouraging results in outdoor navigation.

1. Introduction

Two important tasks in autonomous robot navigation are path planning and obstacle avoidance. Path planning is defined as the work of evaluation of the information received about the robot’s localisation, the structure of the operating environment and the goal location, through multi-sensor data acquisition and finally devised a strategic action path for the robot to manoeuvre to the goal location. Obstacle avoidance is the task of devising a safe path for the robot to manoeuvre through the immediate surrounding without collision with any obstacle that is present.

Some researchers [1][2][3][4] preferred to decouple both tasks in their literature and others [5] find it more efficient to integrate them together. DT methodology is versatile and can be used for path planning alone or integration of path planning with obstacle avoidance. In this paper, we will propose an implementation of DT to achieve the latter case in outdoor navigation where the operation vicinity can easily stretch to several kilometre squares in size.

The paper is organised as follows: Section 2 gives a description of the conventional DT methodology and the environmental map used by DT. The problems encountered when using DT for performing path planning and obstacle avoidance simultaneously in real time outdoor robot navigation. Section 3 presents the newly proposed DT methodology and explicates how it solves the problems mentioned in Section 2. Simulation results, comparing the performance of the newly proposed DT methodology and the conventional DT methodology are presented in Section 4. Section 5 shows an actual implementation of the proposed DT methodology in an outdoor autonomous robot. The proposed DT methodology is simultaneously used for both path planning and obstacle avoidance. Finally the conclusions and directions for future work are drawn in Section 6.

2. Conventional DT Methodology

The conventional DT methodology in space/time proposed by Jarvis [5][6] works in a three dimensional domain formed by a two dimensional tessellated map or grid map [7][8] and time, providing a basis for optimal collision-free path finding. The methodology is simple but sufficient for navigation in indoor environment where the operation area is fixed and relatively small as compared to the case in outdoor environment. The methodology is shown below:

(i) The destination cell in the tessellated map is given a distance propagation cost of zero for all time instant during distance transform.

(ii) All obstacle occupied cells and boundary cells are assigned with an infinity distance propagation cost (eg. 9999) for all time instant during distance transform.

(iii) All unoccupied cells in the tessellated map except the destination cell are assigned with a large distance propagation cost (eg. 5000) initially.

(iv) Evaluation of the distance propagation cost for every cells in the map is done in a systematic manner from top to bottom and left to right, for a period of time T,
which the obstacle information in the map must remain true. Distance propagation cost for each cell is derived based on its previous cost and also the propagation cost of the surrounding cells. The algorithm is as follows:

Let the tessellated map size be M x N cells. The spatial cell positions be indexed by i, j where 0 ≤ i ≤ M and 0 ≤ j ≤ N. The time dimension be indexed by t where 0 ≤ t ≤ T, with t = 0 is the present and t = T represents the time instant most distant into the future which the map still remain true.

for t = T-1 down to 0 do
    for i = 1 to M do
        for j = 1 to N do
            if cell (i,j) not zero and infinite then
                cell(i,j,t) = min { cell(i,j,t+1),
                cell(i-1,j+1,t+1) + d,
                cell(i,j+1,t+1) + v,
                cell(i+1,j+1,t+1) + d,
                cell(i+1,j,t+1) + h,
                cell(i+1,j-1,t+1) + d,
                cell(i,j-1,t+1) + v,
                cell(i-1,j-1,t+1) + d,
                cell(i-1,j,t+1) + h
            }
where v is the cost of vertical movement,
h is the cost of horizontal movement,
d is the cost of diagonal movement.

From the algorithm, it is obvious that each cell in the map has to be processed a number of times. If the map is large, processing time will be longer, as there are more cells to process. In outdoor navigation, the operation area involved can be as large as several kilometre squares, making the conventional space/time DT methodology inefficient and too slow for real time application. Therefore a new DT methodology specifically for outdoor navigation is needed.

3. Proposed DT Methodology for Outdoor Navigation

In the proposed DT methodology for outdoor navigation, each cell in the map needs to be processed once and only a portion of the map is used at each time. Thus reducing the number of cells to process and number of time to process each cell. Hence drastically reduced the processing time to allow real time application. The short processing time also allows obstacle avoidance to be incorporated simultaneously.

3.1 Algorithm

In the proposed DT methodology, the algorithmic steps in Section 2 remain the same from Step (i) to Step (iii), except Step (iv). The evaluation of DT value at each cell remains the same, except that the raster approach of determining the next cell to process is no longer used. Processing now starts from the destination cell. While processing each cell, a search for new cells to process in the immediate eight neighbouring cells is also carried out. If the neighbouring cell is not occupied, it is added into a first-in-first-out linked list buffer (FIFO). The next cell to process is then retrieved from the FIFO linked list buffer on a first-in-first-out basis. Evaluation of DT value is carried out and followed by a search for new cells to process. The whole process repeats until the FIFO linked list buffer is empty implying that all valid cells in the map are being processed.

The new algorithm will be faster than the raster approach used. Each cell in the map needs only to be processed once, regardless of the difficulties of the map structures. Whereas in the conventional DT methodology, the number of times that each cell is processed depends on the complexity of the map. Each cell usually has to be processed more than once. Furthermore, the new algorithm will execute faster in a cluttered environment as more occupied cells in the map means less cells to process. The number of cells need to process in each map in the new DT methodology is given by

\[ \text{No. of cell} = \text{total no. of processed cells in map} - \text{total no. of occupied cells} \]  

where 

\[ \text{data in} \]

\[ \text{FIFO buffer} \]

\[ \text{data out} \]

**Fig1:** Cell searching technique in new DT

3.2 Determination of Local Map

The size of the map used in DT calculation directly affects the execution time. In conventional DT methodology, the entire map is used for the calculation to give optimal solution to path planning. This is impractical when it comes to outdoor navigation, as the operational map can cover several kilometre squares. In this case, the processing time becomes too long for the conventional DT methodology to be useful in real time application.

One solution to this problem is to reduce the resolution of the map (bigger cell size), as the operation area becomes larger. However, this will increase the possibility of the
algorithm making a wrong deduction of cul-de-sac, especially when there are a lot of sparsely scattered obstacles in the operation area. Another better solution is to reduce the size of the DT map. Only a small portion of the map in the robot’s immediately surrounding is used for DT path planning. We call this map the “local map”. An intermediate rendezvous point, which is within the local map will be defined. DT is performed on the local map and with the aim of finding an optimal path to the rendezvous point, instead of the predefined destination point. However, this rendezvous point will ultimately bring the robot closer to the predefined destination point. Each time the robot moves into a new grid cell, new information about it surrounding is acquired, so a new intermediate rendezvous point will be redefined. This process repeats until the predefined destination point is within the local map.

The latter solution might explicitly seem to be a compromise in optimal path planning, as only a portion of the entire map is used for path planning. However, this is not entirely true. Firstly, if the robot is operating in unknown or partially known environment, the assumption that all untravelled cells are empty is not really useful for these cells to be included in the DT calculation. They only contributes to longer processing time. Secondly, if the robot is operating in a highly dynamically changing environment (e.g. on a road), it is also unnecessary to plan a long path. The operation environment might change drastically within seconds and a new DT calculation is needed. Therefore, short processing time is critically needed in DT calculation for it to be useful in real time application. Obstacle avoidance can also be integrated into DT calculation provided that the processing time is fast enough to register and treat any slow moving obstacle as a static obstacle in DT calculation.

For the latter solution, a suitable local map size and an intermediate rendezvous point must be determined. The lower bound limit for the local map size can be clearly defined that it must cover all sensor’s view from the robot. The upper bound limit depends on the processing time allowed for DT calculation, which in turn depends on the robot and obstacle traversing speed and the map resolution used. Usually a large DT map is preferred. After deciding on the suitable local map size, we need to define an intermediate point at each local map. The intermediate rendezvous point must fulfilled two conditions: it should be empty and should lead the robot closer to the predefined destination point. It is observed that if there exists a straight unblocked path from the robot to the destination point, this path should be the optimal route for the robot and DT calculation will also support it. Based on the observation, a shortest path equation (2) is formulated using the robot’s coordinates in the 2D map and the destination point as references. The point where the shortest path equation intersects the border of the local map is chosen as the intermediate rendezvous point. Figure 2 illustrates the determination of a suitable intermediate rendezvous point.

Shortest path equation,
\[
y_i = \frac{(y_d - y_i)}{(x_d - x_i)}(x_r - x_i) + y_i
\]  

If the initial rendezvous point is occupied, a semicircle equation (3) is used to scan the neighbouring area to look for another suitable replacement point. The semicircle will have its centre at the initial rendezvous point. Figure 3 illustrate the search for a replacement point.

Semicircle search equation to find new intermediate rendezvous point \((x_n, y_n)\),
\[
y_n = y_o - \sqrt{d_o^2 - (x_n - x_o)^2}
\]  

where \(d_o\) is the radius of the semicircle, \(d_o = 1, 2, 3, \ldots\)
\(x_n\) is integer in the range of \(x_o - d_o \leq x_n \leq x_o + d_o\)
Simulation Results

A computer simulation was carried out to compare the processing speed between the implementation of conventional DT methodology in Section 2 and the newly proposed DT methodology in Section 3.1. A 2D grid map of 200 by 200 cells were used in the simulation to emulate closely to the actual local map size that would be used in the actual run. The simulation was done on a Pentium 400 machine. Three different scenarios were used in the comparison. The first scenario featured an empty grid map in which no obstacle was present (Figure 4) and the second scenario featured a grid map with a few blocks of obstacles (Figure 5). The final scenario featured a map filled with obstacles such that only a zig-zag path existed from the robot’s location to the destination point (Figure 6). Both the processing time and number of passes per cell in the map were recorded in Table 1.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Conventional DT</th>
<th>Newly Proposed DT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No. of passes per cell</td>
<td>Total time taken (sec)</td>
</tr>
<tr>
<td>1</td>
<td>200</td>
<td>3.090694</td>
</tr>
<tr>
<td>2</td>
<td>394</td>
<td>6.160578</td>
</tr>
<tr>
<td>3</td>
<td>5000</td>
<td>41.524940</td>
</tr>
</tbody>
</table>

Table 1: DT Performance Comparison Table

From Table 1, it is obvious that the proposed DT methodology is much faster than the conventional DT methodology. The short processing allows moving obstacles to be tracked and included in the path planning. Thus eliminates the need for a standalone obstacle avoidance algorithm. Therefore, we can claim that the newly proposed DT methodology integrates both the tasks of path planning and obstacle avoidance. The table also implicitly shows that the proposed DT methodology is much faster in cluttered environment when many of its map cells are occupied. This is favourable, especially when the robot is moving at high speed in a cluttered environment where collision is more likely to happen. The worst scenario in term of processing time for the proposed DT methodology is shown in the first scenario. This information is an added advantage when planning the size of the map to use. For the case of conventional DT methodology, it is difficult to determine because the processing time depends on both the number of obstacle cells and also the placement of these cell in the map.

4. Actual Implementation

The proposed DT methodology is implemented on an commercially available outdoor autonomous robot to perform both the tasks of path planning and obstacle avoidance simultaneously (Figure 7).
Various sensor systems such as laser rangefinders, differential global positioning system (DGPS), Triclops camera, sonar sensors, gyroscopes, roll and pitch sensors are mounted on the robot. Through multi-sensor data acquisition and proper fusion methods, the environmental information is recorded on a 2D grid map [9] and the robot pose is known. The proposed DT methodology is then used to navigate the robot from point to point.

An experiment was conducted in an indoor environment (Fig 9a). The robot was made to transverse at a speed of 0.2 m/s. A 2D grid map was built and Distance Transform planned an initial optimal path to the target (Fig 9b). Upon detection of an obstacle in its optimal path (Fig 10a), the DT replanned a new path (Fig 10b). Thus avoiding collision with the obstacle.

Another experiment was conducted in an outdoor open space (Fig 11a, Fig 11b). The robot was made to transverse at 0.5 m/s. The robot is capable of reaching its destination amid moving obstacles transverse at a speed about 0.5 m/s.
5. Conclusions

In this paper, we have introduced a new implementation method that allows real time simultaneous path planning and obstacle avoidance using Distance Transform. Both indoor and outdoor testing amid moving obstacles had shown encouraging results. However, there are still minor clinches in the implementation that need to be solve to make robot motion smoother. One obvious problem is when a moving obstacle crosses the DT path in the near vicinity in front of the robot. The new DT path planned may be very different from the previous path planned. This will cause the robot which is moving at high speed to make abrupt turning. This results in robot skidding and introducing huge error to the odometry reading. An estimator may be required to predict the projectiles of any moving obstacle near to the robot, such that DT can account and the robot can have ample time to react. With better and faster sensors and computers available, the performance of DT can be greatly improved.

6. References


