2nd International Conference on Autonomous Robots and Agents
December 13-15, 2004 Palmerston North, New Zealand

Distance Transform Based Visibility Measures for Covert Path Planning in Known but Dynamic Environments

Ray Jarvis
Australian Research Council Centre for Perceptive and Intelligent Machines in Complex Environments: Intelligent Robotics.
Monash University
Ray.Jarvis@eng.monash.edu.au

Abstract

The notion of visibility evaluation in known, initially unknown and dynamic obstacle strewn environments is relevant for many robot navigation scenarios and is particularly relevant in covert robot operations where the robot should try and remain hidden from nominated 'sentry' points whilst simultaneously finding an efficient path to a goal. This paper presents an elegant way of measuring visibility in cluttered spaces using the Distance Transform and shows its application in covert path planning in known but dynamic environments. Emphasis is placed on simplicity and generality.

Keywords: Robot navigation, visibility, distance transform, covert robotics and path planning.

1. Introduction

Sometimes simplicity is very appealing, especially when intuitively illuminating. For example, Rosenfield [1] suggested that 'elongation' was the property of a shape that had a lot of its inside near its outside. Also, ideas like 'circularity' being the property of shapes where the ratio of perimeter to area was as small as possible fit into this category.

Visibility is usually defined as a mutual property between two points whose joining straight line does not penetrate an obstacle. Variants may include range and/ or angular limits and could also include the opposite of being 'hidden' where being hidden may be more than just being out of sight. The pure form of visibility evaluation is used in a number of path planning algorithms such as A* [2]. In real Euclidean space, with polygonal obstacles, there are published formulations for calculating visibility [3]. In rectangularly tessellated spaces which are the domain of Distance Transforms (DT) [4], ray tracing along jagged lines of cells to limits of line-of-sight visibility can certainly be done but the calculations are time consuming and inelegant.

In what follows a simple method to calculate visibility using DTs is introduced and its application in covert path planning in known but dynamic environments is demonstrated. Some very promising results are presented. This is followed by discussion of the algorithms used and an indication of future work. Conclusions follow.

2. Distance Transforms (DT)

The original DT algorithm was devised by Rosenfield and Pfaltz [1] as a tool to study the shape of objects in 2D images by propagating distance in tesselated space from the boundaries of shapes into their centres. Various properties of shape can be extracted from the resultant transform and it can be shown that the skeletons of local maxima can be used to grow back the original shapes without information loss.

Jarvis [4] discovered that by turning the algorithm 'inside out' to propagate distance from goals in the free space surrounding shapes interpreted as obstacles and by repeating a raster order and inverse raster order scan used in the algorithm until no further change takes place, a space filling transform with direct path planning applications resulted. Multiple starting points, multiple goals and multidimensional space versions are easily devised. Use in partially known stationary environments and known dynamic environments [5] has already been demonstrated. When the DT has been completed, filling free space with distance markers, the goal is achieved from any starting point through a steepest descent trajectory without the possibility of plateau and local entrapment.

Other path planning methods, such as the potential field approach, can suffer from entrapment in cul-de-sacs. In some cases special consideration must be applied to permit the paths to escape such traps. The rapidly exploring random tree (RRT) [7] method, whilst efficient in known environments, is difficult to apply in initially unknown or dynamic environments and it is difficult to see how visibility measures might be integrated with it.

The Pascal-like pseudo code (taken from [5]) for the DT in 2D and the trajectory calculation follows.

2D Distance Transform in a 0..xmax+1 by 0..ymax +1 rectangularly tessellated, wall enclosed space:

```
(*Initialize cell & blocked border around cell at
x=0, xmax+1 & y=0, ymax+1*)
 for y:=0 to yMax+1 do
  for x:=0 to xMax+1 do
   if goal [x,y] then cell [x,y]:=0;
                else cell; [x,y]:=xMax*y Max; (*A
large number *)
(*Calculate distance transform in cell []*)
repeat
 for y:=2 to yMax do
  for x:=2 to xMax do
   if not blocked [x,y] then
    cell [x,y]:= min (cell[x-1,y]+1, cell[x-1,y-
                1]+1, cell [x,y-1]+1, cell[x+1,y-
                1]+1, cell [x,y]);
 for y:=yMax-1downto 1 do
  for x:=xMax-1 downto do
   if not blocked [x,y] then
    cell[x,y]:=min(cell[x+1,y]+1,cell[x+1,y+1]+1,
             cell[x,y+1]+1,cell[x-
             1,y+1]+1,cell[x,y]);
until no change;
```

By modifying this outline by weighting moves in x and y directions by 2 and diagonal directions by 3 (3/2 ≈√2) better approximations to Euclidean distances result. The original code counts steps rather than calculates distances.

Goal seeking trajectory from xs,ys:

```
(*Trace all paths*)
 for ys:=1 to yMax do
  for xs:= 1 to xMax do
   if start [xs,ys] then (*Trace path to nearest goal
   for start point (xs,ys)*)
    begin
     x:=xs;y:=ys;
    while not goal [x,y] do
      begin
       next(x,y,xn,yn);
       x:=xn;y:=yn;
      end;
    end
```

Procedure next (x,y,xmin_neighbour ,ymin_neighbour) examines the eight neighbours of (x,y) and returns the one with minimum distance transform potential field value in (xmin_neighbour, ymin_neighbour).

Subsequently, Jarvis [5] showed that optimal paths through space/ time could be found using a variant of the DT in perfectly known but dynamic environments (goals and obstacles can move, obstacles can even change shape and/ or size). Since goals now exist in time and space, they are called rendezvous. For a three dimensional array with two spatial and one temporal dimension, the space/ time DT is as follows:

Begins at time t=T-1 and move backwards in time towards t=0 in one pass. The sought after distance transform in free space/ time is found as follows:

```
for t=T-1 downto 0 do
for i=0 to N do
for j=0 to N do

If cell [i,j,t] not zero and not infinite then
cell [i,j,t]=min (cell[j,t+1]+n,

        cell [i+1,j,t+1] +h,
        cell[i-1,j,t+1] +h,
        cell[i,j+1,t+1]+v,
        cell [i,j-1,t+1]+v
        cell[i+1,j+1,t+1]+d
        cell[i-1,j-1,t+1]+d
        cell[i+1,j-1,t+1]+d
        cell[i-1,j+1,t+1]+d);
```

where n is the cost of no move
v is the cost of a vertical move (j variation, say)
h is the cost of a horizontal move (i variation, say)
and d is the cost of a diagonal move (both i and j variation)

Setting n=1,v=h=2 and d=3 works.

This allocation is associated with standing still for a time unit costing 1 unit, moving in x or y directions 2 units and moving diagonally 3 units (3/2 ≈√2). The inner two loops can be processed in parallel. Only one pass is involved.

3. DT Visibility

Most robot path planners look for optimal paths (shortest time, distance or energy) or safe paths [8] or feasible paths. A fairly new idea [9,10] is to find covert paths which allow movement from a nominated starting point to a goal whilst being minimally observed from specific (but perhaps) moving locations (sentry points) or from anywhere in general free space. In this age of terrorism such possibilities now seem quite important in terms of tracking and perhaps apprehending an enemy agent or at least proceeding to a destination without the knowledge (or perhaps minimal knowledge) of the enemy.

If one can fill free space with costs associated with risk of observation (through visibility analysis) one can combine (using weighting coefficients) distance and observation risks costs in developing the new kind of DT (perhaps D/RT) where steepest descent trajectories in the transform space achieve the desired result of a short/ covert approach to the goal.

With respect to a single (enemy) observation cell (sentry point), the visibility status with respect to that point throughout free space can be determined very elegantly by combining the results of two pure DT operations in that space. The basic idea is that if the direct distance is the same as the minimal obstacle-free distance from the sentry point to any other free cell is the same, that point is visible from the sentry point. Anomalies in tessellated space distances cause some distortions but these are not serious. One can also develop more subtle definitions of 'visibility' other than the binary result of visible or not visible.

Let d(i,j), i=0..N-1, j=0..N-1 (in the mxn map space) be the DT result for a single nominated sentry point (in free space) when no obstacles are included.

Let c(i,j),i=0..N-1,j=0..N-1 be the corresponding DT result when the obstacles are present.

Then, for every cell (i,j), V1(i,j)=1 (visible) when [d(i,j)/c(i,j)>0.95], else V1(i,j)=0 (not visible). Figure 1 shows some typical results. Obstacles are shown as blobs and the black areas are non-visible regions from the view point of a location near the top right corner. These look like shadows cast by the obstacles when illuminated from that location. Note the results are not perfect but are good enough and are obtained quickly and elegantly. If more than one sentry points are present, this binary evaluation can be done separately for each. Summing the binary results will produce a count of how many sentry points can observe a free cell. In the extreme situation where locations of possible sentry points are not known, a general visibility from anywhere in free space can be evaluated by summing binary visibility values calculated for each free cell in turn. Some variations of visibly measure are reflected in V2, V3 and V4, defined below.

V2(i,j)= [1.0/ (c(i,j)-d(i,j)+1.0)], i=0..N-1,j=0..N-1

This value tends towards 1.0 as (i,j) becomes visible and gets smaller when (i,j) is not merely not visible but also 'hidden' in a 'tucked away' sense.

V3(I,j)=[V1/d(i,j)], i=0..N-1,j=0..N-1

This is inverse distance weighted which could reflect that visibility becomes less of an issue at some distance from the sentry point.

V4(i,j)=1.0 when d(i,j) ≤ k1 and d(i,j)/c (i,j) >0.95
i=0..N-1
j=0..N-1
=0 otherwise

This is visibility within distance k1 from sentry point, further away being considered not visible despite clear line-of-sight.

Any one of VI, V2, V3 or V4 can be used in modifying the DT path planning formulation to take visibility as well as distance into account by using a weighted sum of the distance between adjacent cells (nominally = 1, although $\sqrt{2}$ can be used for diagonal neighbours) and the visibility value calculated for each cell. The more the visibility measure is weighed relative to the distance measure, the more covertness dominates the path at the expense of path length and vice versa.

4. Covert Goal Seeking Trajectories in Time/ Space

The time/space DT algorithm given in section 2 can be used for covert time/space trajectory determination by simply modifying the costing formula. Instead of using distance and time related costs simply add visibility costs to the formula (weighted if desired) and the expected result happens. For example, using the V1 formulation, simply add (factor x V1 (i,j)) values in the one unit forward time slice to the n, h, v and d values shown and process as before, where 'factor' is a weight.

The resulting trajectories tend to hang about in the visibility shadows of the visibility field but get to the goal if possible in the time available, depending on the distance to travel.

5. Covert Time/ Space Trajectory Results

These are best seen as video clips (at presentation time) since time/space trajectories are involved. If 'factor' is set to 0, the original minimal time/ space trajectory results. Factors of 1.0 and 2.0 are shown for comparison with the latter showing more covert behaviour than the first, as expected. Of course, the extra price for a covert trajectory shows itself as extra distance traveled and longer time taken to get to the goal (some standing still often being involved). Only one sentry point was used for these results.

6. Moving Sentry Point

If the sentry point moves in time along a known time/ space trajectory, the covert trajectory will accommodate to these movements without modifying the algorithm. Of course, the visibility DTs at each time slice must use the appropriate sentry point location to propagate from. Multiple moving sentry points can also be handled but the visibility evaluations have to be handled individually for each such point.

7. Discussion

This paper shows that the DT has many capabilities and can be manipulated towards a variety of applications. Both the space only and the space/time variants are demonstrated here. The basic algorithm is so simple that it literally takes only a few hours of programming to try out new ideas in using it. It was mentioned in passing that sometimes safer rather than optimal paths are sought. Here safety is not used in the sense of being free from risk of observation but of collision if the robots' position can not be controlled accurately enough or the estimation of the locations of obstacles is not accurate. The DT can be used in yet another mode to provide a simple safe path algorithm [8]. Regarding all obstacles as made up of goals, the DT propagation out from them thought out free space will have local maxima at places where the distance to the nearest part of any obstacle is greater locally. Small values indicate closeness to obstacle cells. The difference between the largest DT value in the whole space and that of each individual free cell is a measure of risk of collision. Adding these into distance and/ or observation risk evaluations (with weighting factors) allows the simple generation of trajectories with customized collision risk/distance/observation risk. A variety of trajectories can be generated for the same environments, starting points and goals, depending on what emphasis one wants regarding risks of collision, visibility and long paths.

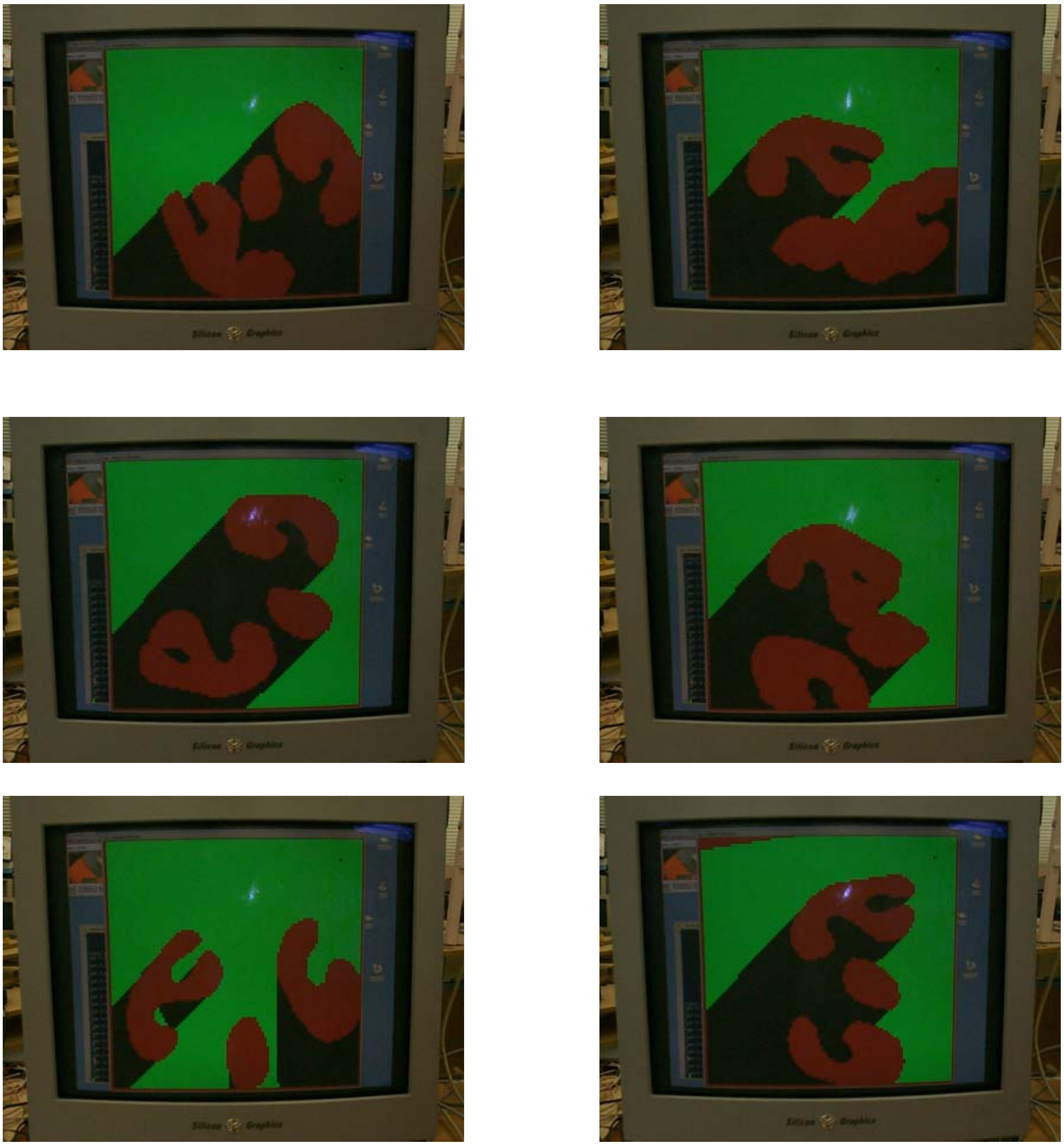

Figure 1: Distance Transform based Visibility Evaluations: some typical results

The only limitation on the use of the DT for generating the field within which steepest descent trajectories are the way to go, is that costs must be positive; this is an easy constraint to meet.

Yet to be investigated are covert path planning in situations where environments are stationary but initially unknown and where environments are both initially unknown and changing, perhaps with some constraints on the unpredictability of change. Also, combinations of specific and generalized visibilities should be investigated with respect to covert path planning for various types of environmental knowledge.

8. Conclusion

This paper has introduced a DT based visibility analysis tool for covert path planning and demonstrated its use in time/ space trajectory planning for known but time varying obstacle fields and observation points (sentry points). Further variations have yet to be researched. In an era of concern over terrorism the notion of covert path planning comes into its own.

Bibliography

[1] Rosenfeld, A., and Pfaltz, J.L, "Sequential Operations in Digital image Processing", *J.A.C.M., Vol.13, No.4, Oct. 1966*, pp.471-494.

[2] Nilsson, N.J., Problem Solving Methods in Artificial Intelligence, McGraw-Hill, 1971.

[3] Munson, J.H. Efficient Calculations on Points and Lines in the Euclidean Plane, SRI AI Group Technical Note No. 61, Oct. 1971

[4] Jarvis, R.A., Collision-Free Trajectory Planning Using Distance Transforms, Proc. National Conference and Exhibition on Robotics – 1984, Melbourne 20-24 August 1984. Also in Mechanical Engineering Transactions, Journal of the Institution of Engineers, Australia, 1985. (five pages).

[5] Jarvis, R.A., On Distance Transform Based Collision-Free Path Planning for Robot Navigation in Known, Unknown and Time-Varying Environments, invited chapter for a book entitled 'Advanced Mobile Robots' edited by professor Yuan F. Zang. World Scientific Publishing Co. Pty. Ltd. 1994, pp: 3-31.

[6] Latombe, J.C., Robot Motion Planning. Kluwer Academic Publishers, 1991, Boston, MA.

[7] Kuffner, J.J. and LaValle, S.M. RRT-Connect: An efficient approach to single-query path planning. In Proc. IEEE Int'l Conf. on Robotics and Automation (ICRA'2000), pages 995-1001, San Francisco, CA, April 2000.

[8] Zelinski, A., Robot Navigation with Learning, The Australian Computer Journal, Vol. 20, No.2, May 1988, pp.85-93.

[9] Marzouqi, M. and Jarvis, R.A. Covert Path Planning for Autonomous Robot Navigation in Known Environments, Australian Conference on Robotics and Automation 2003, Brisbane 1-4 December, 2003, pp:

[10] Birgersson, E., Howard, A., Sukhatme, G.S. Towards Stealthy Behaviours, Proc. 2003 IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems, Las Vegas, Nevada, Oct. 2003, pp:1703-1708.