Exploration of an Unknown 2D Environment using a View Improvement Strategy

Rachel Gartshore
CVSSP
University of Surrey
Guildford, GU2 7XH, UK
R.Gartshore@surrey.ac.uk

Phil Palmer
Surrey Space Centre
University of Surrey
Guildford, GU2 7XH, UK

Abstract

In this paper we present a new exploration strategy for a mobile robot moving in a 2D plane. The task of the exploration algorithm is to move around autonomously in an unknown room environment of a limited size whilst building and using a 2D map to avoid obstacles and plan its next move. The strategy is not goal driven, rather seeking new unseen areas to view and explore.

The exploration strategy is designed for a vision system with a limited field-of-view. Improvement from previous views of local obstacles is used with a next best viewpoint calculation to select the next move where recently detected obstacles are given more weight. Example explorations are presented and are shown to find new areas to view without exhaustive searching.

1. Introduction

Exploring an unknown environment is a fundamental mobile robotics problem. Many researchers have concentrated on exhaustive exploration techniques such as raster scan, wall-hugging (Albers et al., 1999) or circumnavigation algorithms (Kutulakos et al., 1993), (Lorigo et al., 1997); way-markers (Newman et al., 2003) and goal positions (Burgard et al., 2000) can be selected on- or off-line to aid navigation, or human path following (Althaus and Christensen, 2003) for post navigation.

This paper addresses the issue of no goal position being provided and no prior knowledge of the scene. The robot is to explore a completely unknown environment. The shape and size of the scene is not known, and we do not assume that all internal areas are visible or accessible. There are no way-markers to guide the robot, and once the exploration starts no human intervention is possible. Using a vision system with a limited field-of-view, the robot must decide which direction to move and to view from its current knowledge of the scene. The only restriction placed upon the exploration algorithm is that only a limited area can be explored (to stop the robot from wandering off). If the vision system detects features outside of this area they are ignored.

We want the robot to explore in an intelligent way, like a human might. For example, you arrive at a hotel in a city to which you've never visited before. You have no map or guide to help you, therefore no knowledge of which direction the interesting areas are. You decide to explore the immediate area of the hotel by setting a target to explore an area of one square mile. On leaving the hotel you take a quick scan of the hotel's surrounding buildings, and then select an area to head towards. This direction may lead to a new area to explore further, or it may lead to a dead-end.

From an initial scan of the environment from the starting position, a representation of the scene is incrementally built. To gain more knowledge of the scene, a calculation of the improvement of the view of detected obstacles allows the next move to be selected. The vision system cannot differentiate between obstacles in the scene and barriers which can be traversed. A confidence measure of whether world obstacles represent scene obstacles is maintained using the visibility constraint described by (Manessis, 2001). When the view of a barrier reveals new areas (within the restricted area), these are also explored. We assume perfect lighting and vision system, map building and position estimation. We acknowledge that errors in these aspects of the system will be important for the effectiveness of the exploration and will need to be addressed for the exploration to be useful in a real system. In this paper however, we wish to draw attention upon the exploration algorithm itself.

In section 2. we discuss the storage requirements for the internal representation of the world being mapped. In section 3., the calculation of hypothesised view improvement of the world map is introduced. Section 4. reviews a next viewpoint calculation to maximise the viewing angle of an opening. Two example explorations of room environments are shown in section 5. The results are discussed and some conclusions drawn.

2. Internal Representation of the World

Vertical edge features detected in images of the scene are used to define the boundaries of obstacles. These vertical features are projected onto a 2D plan and are therefore robust to errors in detection algorithms as it is not important to detect the entire edge. Vertical edges are critical for deciding where the robot can move; detecting other features is not necessary.

Some vertical edge features such as shadows or transient objects may be added into the world map alongside real obstacle features. To allow for such artefacts to be distinguishable from real obstacles, points in the world map are labelled with a confidence measure of being real scene obstacle features; C_f is calculated from the number of sightings of the scene feature S and the number of times that the scene area is viewed V and is defined as $C_f = 1 - V/S^2$.

Many map building algorithms store position information of scene features with no connections between the neighbouring features (Davison and Murray, 2002). Features in any world map representation alone do not provide enough information for a navigation algorithm to successfully avoid obstacles. A triangulation of the detected scene features in two dimensions generates a simple non-overlapping mesh suitable for encoding the information required for the exploration algorithm. The connecting lines between these world map features either represent real obstacles in the scene or are constructed for the triangulation (construction lines).

Construction lines are labelled with a measure of their confidence of being an obstacle using intersections of a visibility line described by (Manessis, 2001). Using the constraint for a scene such that an obstacle in the scene can not intersect any line from the camera to a real obstacle. Where such a line intersects a construction line this constraint is violated, therefore the construction line can not represent a real scene obstacle.

The path of the robot is deemed valid if it does not intersect any construction line labelled an obstacle. The confidence measure of a construction line being a real obstacle C_o is calculated using visibility line violations and the confidence of the features C_f to which the visibility lines are related.

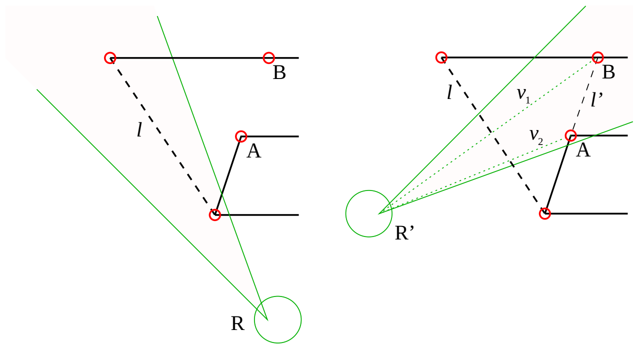
$$C_o = \prod_{i=1}^{N} 1 - C_{f_i} \tag{1}$$

The more times a construction line intersects visibility lines and the higher the confidence of the world feature's existence forming the visibility line, the less likely the construction line is to be an obstacle.

3. View Improvement

Obstacles in the world potentially occlude unseen features in the scene. Construction lines may correctly represent real obstacles, or they should not be labelled as obstacles but have thus far only been viewed from positions where the occluded features are not visible. Figure 1(a) shows the robot at R with its camera's view direction and field-of-view. When first added into the triangulation the construction line l will be labelled as an obstacle (so that a path cannot be planned through it), since the features A and B have not yet been viewed. When the robot moves to position R' (figure 1(b)), A and B are visible. The visibility lines v_1 and v_2 intersect the construction line l, the C_o measure is then re-calculated.

When selecting a position to move to, we use the knowledge that when viewing a construction line from different positions new scene features may become visible. By storing information about the positions from which a construction line has been viewed from, we can quantify the change and therefore calculate an improvement for a potential viewpoint.



(a) Occluded Features (b) Features are Detected

Figure 1: Obstacles Viewed from Different Orientations.

3.1 Obstacle View Improvement

A 1D histogram of equal sized bins is stored with each construction line to store the angles from which the line has been viewed. Figure 2 shows such a histogram for the line l with I bins and normal vector $\mathbf{n}$. For each histogram bin along the length of l within the camera's field-of-view, θ is calculated from the corresponding $\mathbf{w}[\mathbf{i}]$ and stored in the histogram at $l[i]$. When the construction line is viewed from another position and orientation r', the accumulative difference in all previously stored views $p \leq P$ can be calculated, defining the view improvement for the construction line I_e.

$$I_e = \sum_{i=1}^{I} \prod_{p=1}^{P} \mathbf{n}.\mathbf{w}'[\mathbf{i}] - \mathbf{n}.\mathbf{w}[\mathbf{i}]_\mathbf{p} \tag{2}$$

The more histogram bins of a construction line that are seen and the larger the difference in the stored angles, the larger the improvement for the new view. A line viewed from the same angle but different positions does

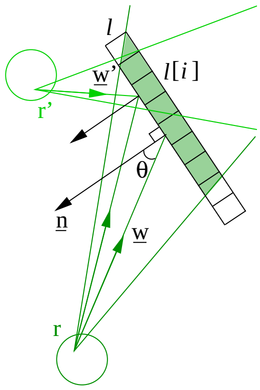


Figure 2: Obstacle Histogram Storing Angles.

not improve the view for the line as much as viewing the line from the same position with a different view orientation.

3.2 Area View Improvement

Detected obstacles stored in the world map provide positive feedback data; if no obstacles are detected, no data would be added into the world map. The information that no features were detected is useful and should also be stored.

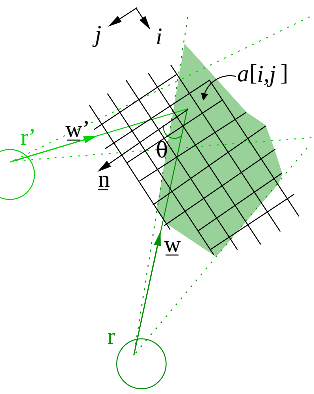


Figure 3: Area Histogram Storing Angles.

A simple mechanism to store the seen parts of the world is a discrete grid of equal sized cells forming a 2D histogram, figure 3. For each histogram bin in area a within the camera's field-of-view, θ is calculated from $\mathbf{n}.\mathbf{w}$ and stored in the histogram at $a[i,j]$. When the area is viewed from another position and orientation r', the accumulative difference in all previously stored views $p \leq P$ can be calculated, defining the view improvement for the area I_a.

$$I_a = \sum_{i,j} \prod_{p=1}^{P} \mathbf{n}.\mathbf{w}'[i,j] - \mathbf{n}.\mathbf{w}[i,j]_p \tag{3}$$

The larger the viewed area and the greater the difference in the stored angles in the area's histogram, the larger the improvement for a new view.

3.3 Similarity in View Position

Similar viewing positions provide little new information about the world map. Viewing the same features from a very similar position can improve the accuracy of the feature positions or the robot position, however to aid the pioneering characteristic of an exploratory robot, a measure of position similarity is used to deter a viewing position from being re-visited.

4. Next Viewpoint Calculation

A utility function that maximises the viewing angle of an opening is presented in (Gartshore, 2005). Figure 4 shows the robot at position r with a limited distance to move d in the presence of an opening formed by the corners C_1 and C_2. The point that gives the maximum viewing angle ν of the opening provides the next best viewpoint r'.

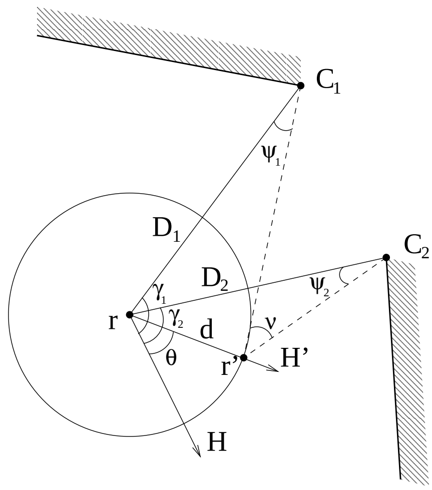


Figure 4: The Two Corner Case

The viewing angle ν is defined by

$$\begin{aligned} \nu &= \angle rr'C_2 - \angle rr'C_1 \\ &= (\psi_1 - \psi_2) + (\gamma_1 - \gamma_2) \end{aligned} \tag{4}$$

$(\gamma_1 - \gamma_2)$ was the viewing angle at the robot position r, therefore we maximise $(\psi_1 - \psi_2)$ to find the position r'.

The direction for the robot to move to reach r' is defined as θ. In the general case where $D_1 \neq D_2$, the solution to the maximum view calculation is re-arranged for θ and shown to be

$$\theta = \left(\frac{\gamma_1 + \gamma_2}{2}\right) - \arctan\left(\frac{q}{p}\right) - \arccos\left(\frac{1}{\sqrt{p^2 + q^2}}\right) \tag{5}$$

$$p = l\cos\mu \quad q = m\sin\mu \quad \mu = (\gamma_1 - \gamma_2)/2$$
$$l = \frac{D_1 D_2 - d^2}{d(D_1 - D_2)} \quad m = \frac{D_1 D_2 + d^2}{d(D_1 + D_2)} \tag{6}$$

This calculation is used to find the next best viewpoint for a given obstacle in the world map, but is only valid when the circle defined by d remains on the same side of the opening. Since the robot path cannot intersect an obstacle, the equation will be valid.

To select which obstacle in the world map should drive the exploration, each obstacle that is visible from its next best viewpoint is considered. The joint improvement for the obstacle from the next viewpoint is calculated using the position similarity along with the addition of obstacle improvement and area improvement.

4.1 Obstacle Importance

When new features are viewed and added to the map, the obstacle map expands. Newly discovered obstacles are more interesting in the exploration algorithm than obstacles that have already been viewed numerous times. When an area is found to contain new obstacles, these obstacles are prioritised above the others. To rank new construction lines, an importance value is assigned to each obstacle, the joint improvement is then multiplied by this rank. The importance value is decreased each time the obstacle is used to define the next viewpoint or when the majority of the obstacle is seen, and is increased when it is first discovered during the exploration. It is increased by a value relative to the size of the new area that was discovered when the obstacle was first seen. If when the new feature is added to the map only a small new area is added, those new obstacles are not highly prioritised.

5. Results

No information of the position of obstacles in the scene is known prior to exploration. The robot builds up the map relative to its starting position of $(0, 0)$. An initial scan of the scene is done to build up a representation of the local area. Neighbouring scene features are assumed to be obstacles until the construction line that defines the hypothesised obstacle boundary is violated by visibility constraints of other obstacles.

5.1 Exploration of a Basic Map

Figure 5 shows the map being built from the test scene. As new scene features are detected they are added into the world map representation which is then re-triangulated, and the confidence of each construction line being an obstacle C_o is re-calculated using the visibility lines from the new view. Previous robot view position and orientations are shown. The current view position has the field-of-view region and visible features

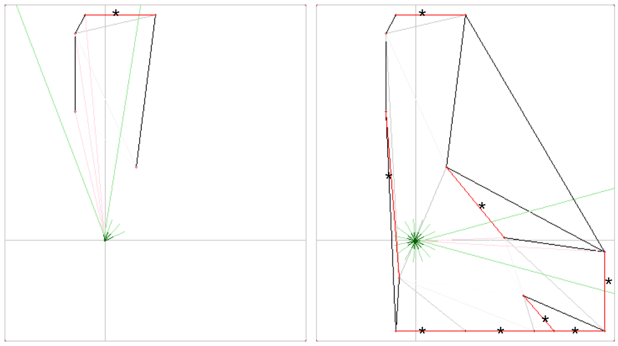


Figure 5: Initial Moves in the Basic Map Scene, robot in central position. Correctly detected obstacles highlighted by asterisks. All construction lines displayed to grey-level of C_o, with $C_o = 1$ drawn in black.

highlighted. The construction lines are drawn to a grey-level value to represent its C_o value, where a black line represents $C_o = 1$. If the construction line has a very high C_o value and has been at least partially seen, it is highlighted by an asterisk (the algorithm does not receive this information) to signify it is a correctly identified obstacle.

After the initial scan, three of the five internal and five of the sixteen external obstacles have been found. The fourteen obstacles with $C_o > 0.7$ (eight which are correctly labelled and the six potentially visible others) are considered for further viewing. Positions providing the next best viewpoint are calculated, and if still visible from the hypothesised position the joint improvement I_o is calculated. The obstacle with the highest I_o provides the next robot move.

It should be noted that three options of distance to move (d) are used to calculate three next viewpoints for a given obstacle: $0mm$ (rotation only), $400mm$ and $1200mm$. This allows the robot to not move if it has found a good position to view several obstacles from, or to quickly move to a new area if the current position does not allow for good viewpoint possibilities. For each viewpoint calculated, if the obstacle is viewable from the new position by a number of orientations, options for view orientation are also calculated.

Figure 7 shows the first few moves selected by the exploration algorithm. Each column shows one snapshot of the exploration. It can be seen in move 4 that the obstacle selected by the exploration algorithm (shown on the line histogram map by circles at each of the hypothesised obstacle's feature points) was occluding an unseen scene feature, C. A new hypothesised obstacle is created (L1) with this new feature (C). This new obstacle, L1, is selected for view in move 5 where three new scene features are discovered. Again a newly constructed hypothesised obstacle (L2) is selected for view in move 6. These moves are an exciting example of how the obstacle importance measure is used alongside the improvement measures to rapidly and repeatedly discover new areas of the scene.

Figure 8 shows the obstacle map for the choices 45 to 50. The robot path is displayed and the new viewpoint is highlighted with the viewing angle and field-of-view region of the camera. This set of results shows the final stages of the exploration. The algorithm continues to select obstacles that yield the highest view improvement, but as this improvement value decreases, the choice of obstacle is made from the position similarity element which pushes the robot to the position where the final new area can be viewed.

Figure 6 shows a plot of the percentage of the area and

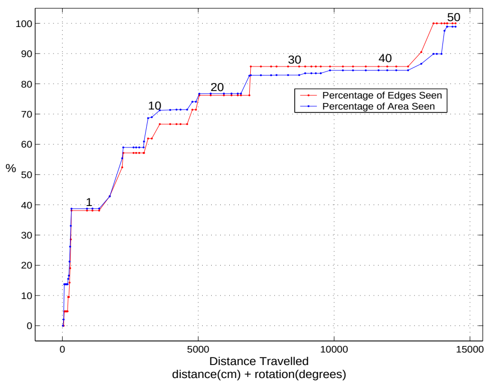


Figure 6: Basic Map Exploration: Percentage of Obstacles and Areas seen versus total distance travelled.

obstacles seen against the distance and rotation travelled. After the initial move 40% of the total area has already been seen. After 28 moves, over 18 of the 21 scene obstacles have been detected and over 80% of the area has been viewed. After 50 moves all of the obstacles have been detected and 99% of the area has been viewed.

5.2 *Exploration of a Hidden Room*

The previous scene map showed the general workings of the exploration algorithm and how a map can be explored. We will now consider a scene where the majority of the area and obstacles can be seen from the starting point, but contains a room whose entrance is hidden from view, figure 9(a). For the robot to find the entrance to the room, it must travel up to the top of the scene and look towards the entrance at an angle such that the interior of the room becomes visible.

Figure 9(b) shows the obstacle map after the initial move and after 32 moves where no new information of the scene has been detected. On move 33, the robot detects the new area behind the room and seeks out the new area by moving to the best viewpoint for the newly detected obstacles. Once the room entrance is labelled as a confident opening, the robot moves inside and looks around. Once the obstacle and area improvement decrease, the robot is pushed away from previously visited positions and exits the room. The last set of interesting obstacles is then viewed further thus completing the exploration of unknown areas of the scene.

6. Conclusions

In this paper we have presented an exploration strategy for a vision system with limited field-of-view. From no prior knowledge of the scene, a map representation is built up incrementally from viewing positions calculated using a next best viewpoint calculation. The next move is chosen from a set of options to improve upon previous views of obstacles.

The exploration has been tested on a large dataset of randomly generated maps. Further information of how the exploration algorithm handles a variety of situations can be found in (Gartshore, 2005).

References

Albers, S., Kursawe, K., and Schuierer, S. (1999). Exploring unknown environments with obstacles. In *10th Symposium on Discrete Algorithms*.

Althaus, P. and Christensen, H. I. (2003). Automatic map acquisition for navigation in domestic environments. In *Proc of ICRA*, volume 2, pages 1551–1556.

Burgard, W., Moors, M., Fox, D., Simmons, R., and Thrun, S. (2000). Collaborative multi-robot exploration. In *Proc of ICRA*, volume 1, pages 476–481.

Davison, A. J. and Murray, D. W. (2002). Simultaneous localization and map-building using active vision. *Transactions on Pattern Analysis and Machine Intelligence*, 24(7):865–880.

Gartshore, R. (2005). *Autonomous Robotic Exploration of Unknown Environments*. PhD thesis, University of Surrey, UK.

Kutulakos, K., Lumelsky, V., and Dyer, C. (1993). Vision-guided exploration: A step toward general motion planning in three dimensions. *Proc of ICRA*, pages 289–296.

Lorigo, L., Brooks, R., and Grimson, W. (1997). Visually-guided obstacle avoidance in unstructured environments. *Proceedings of IROS*, pages 373–379.

Manessis, A. (2001). *3D Reconstruction from Video Using a Mobile Robot*. PhD thesis, University of Surrey, UK.

Newman, P., Bosse, M., and Leonard, J. (2003). Autonomous feature-based exploration. In *Proc of ICRA*, volume 1, pages 1234–1240.

Acknowledgments

We would like to thank John Illingworth for his useful comments and discussions.

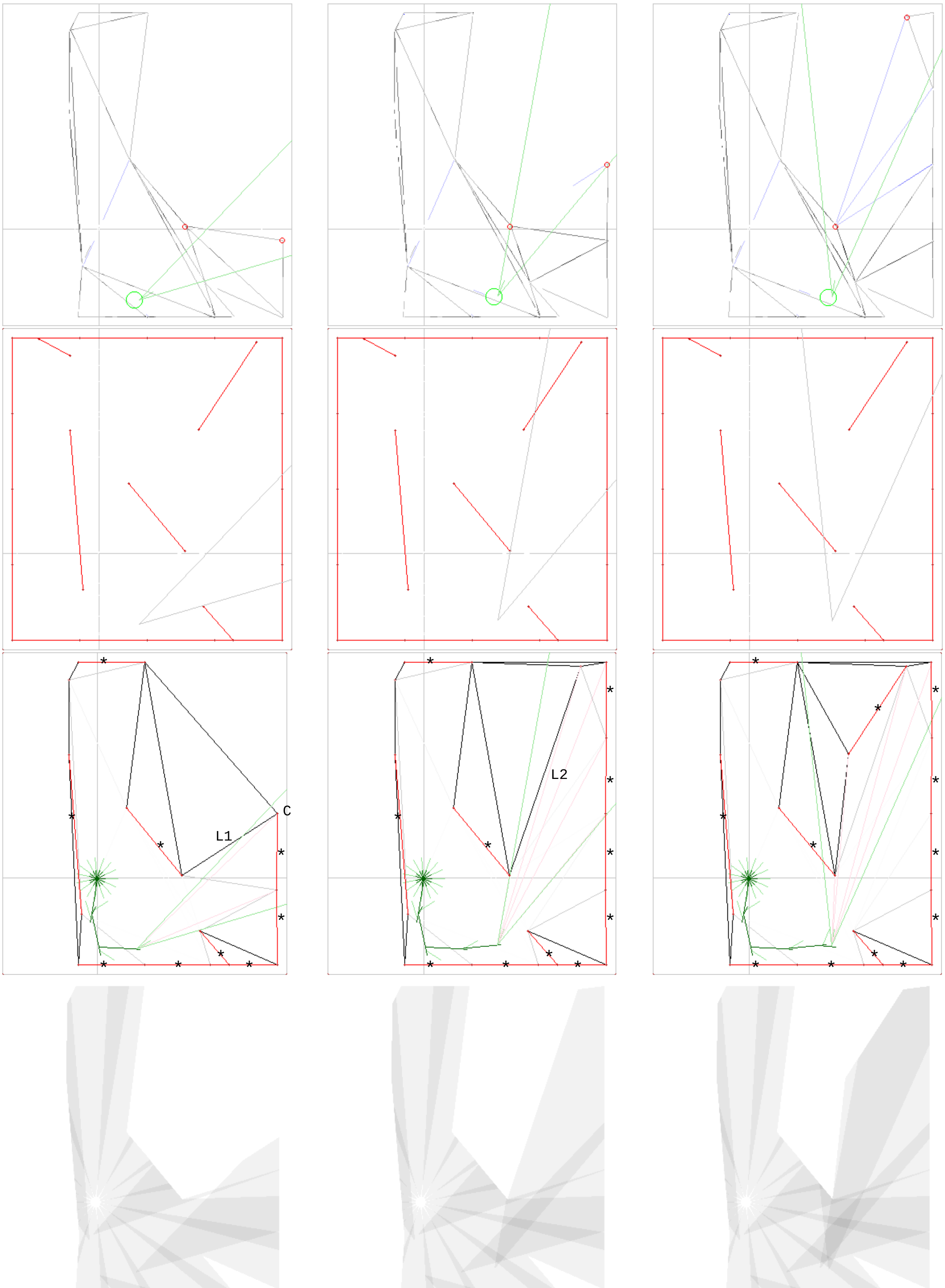


Figure 7: Exploration moves 4–6. (i) next move on line histogram map, selected obstacle highlighted with corner circles (ii) position in scene, (iii) obstacle map with correctly detected obstacles highlighted by asterisks, (iv) seen areas. As new scene features are discovered new hypothesised obstacles are constructed and prioritised in the consideration for improvement.

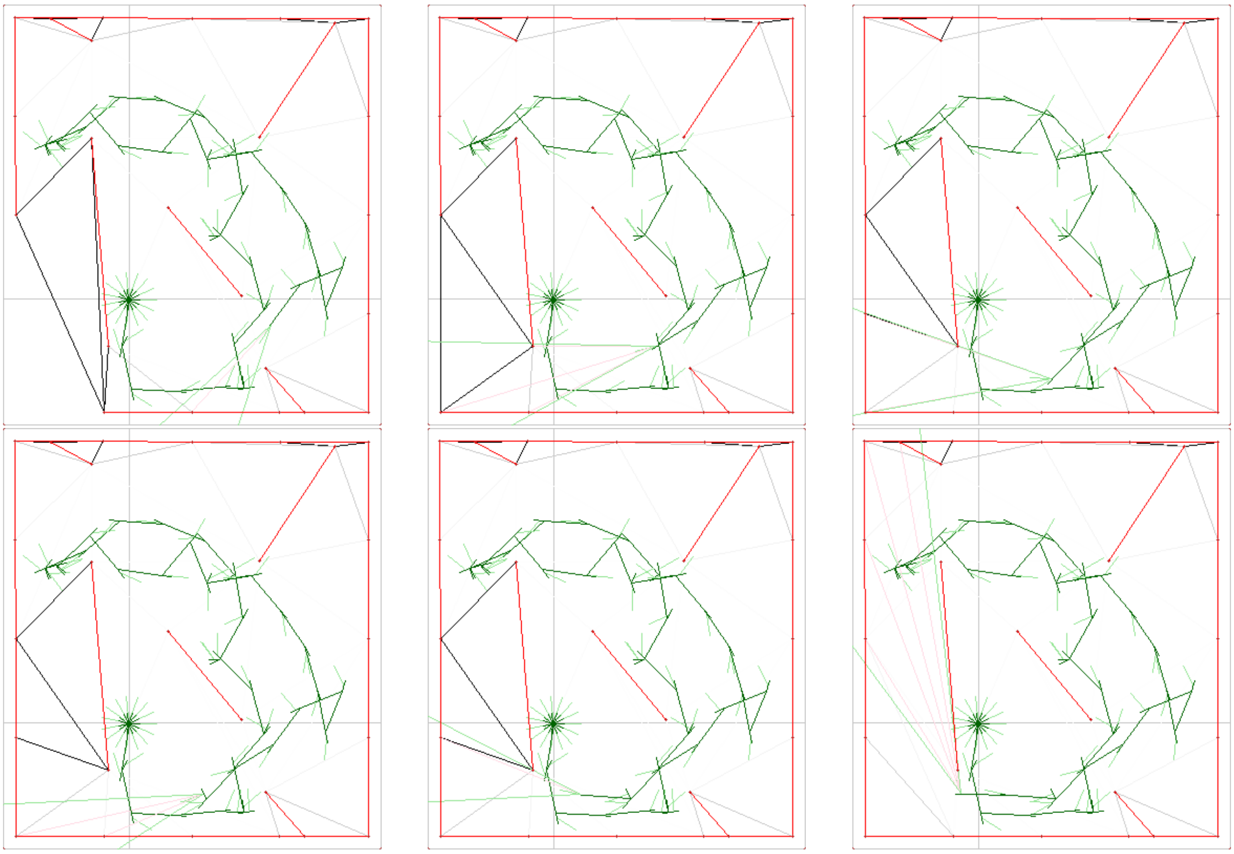
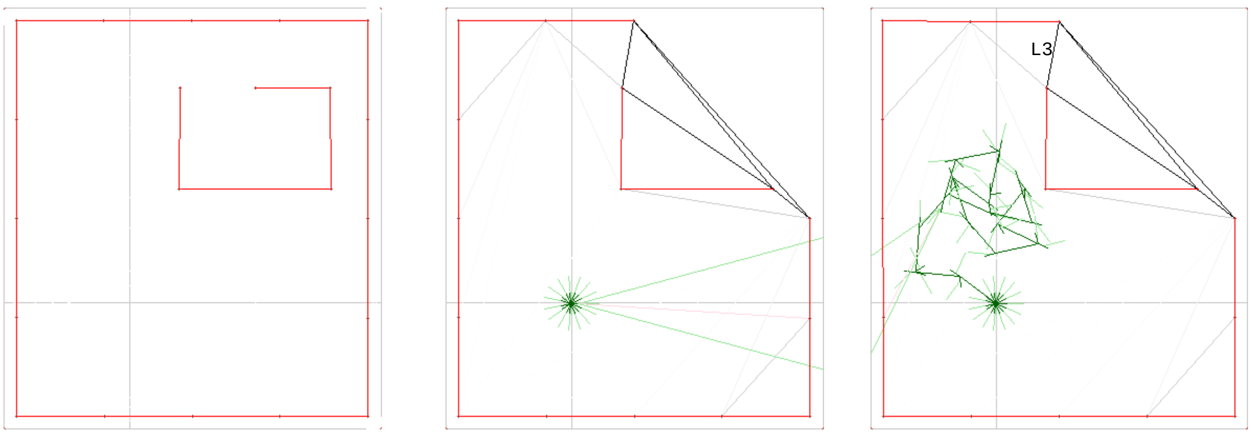


Figure 8: Exploration moves 45–50. Robot path with Obstacle Map. The robot safely navigates around obstacles whilst improving the knowledge of the scene. As the obstacle-improvement and area-improvement values decrease, the position similarity value becomes more prominent and the robot is pushed to positions in the scene which have not previously been visited. This set of results shows the discovery of the final unknown area in the scene. Again, as new features are detected the new hypothesised obstacles are prioritised for selection.



(a) Scene Map

(b) Exploration moves 0–32

Figure 9: Hidden Room Exploration. Most of the obstacles and the area is seen from the initial scan. To view the remaining area the robot must travel to the top of scene and look towards the entrance at an angle such that the interior of the room becomes visible. In the first 32 moves, the hypothesised obstacle L3 is not viewed at such an angle to yield this information.

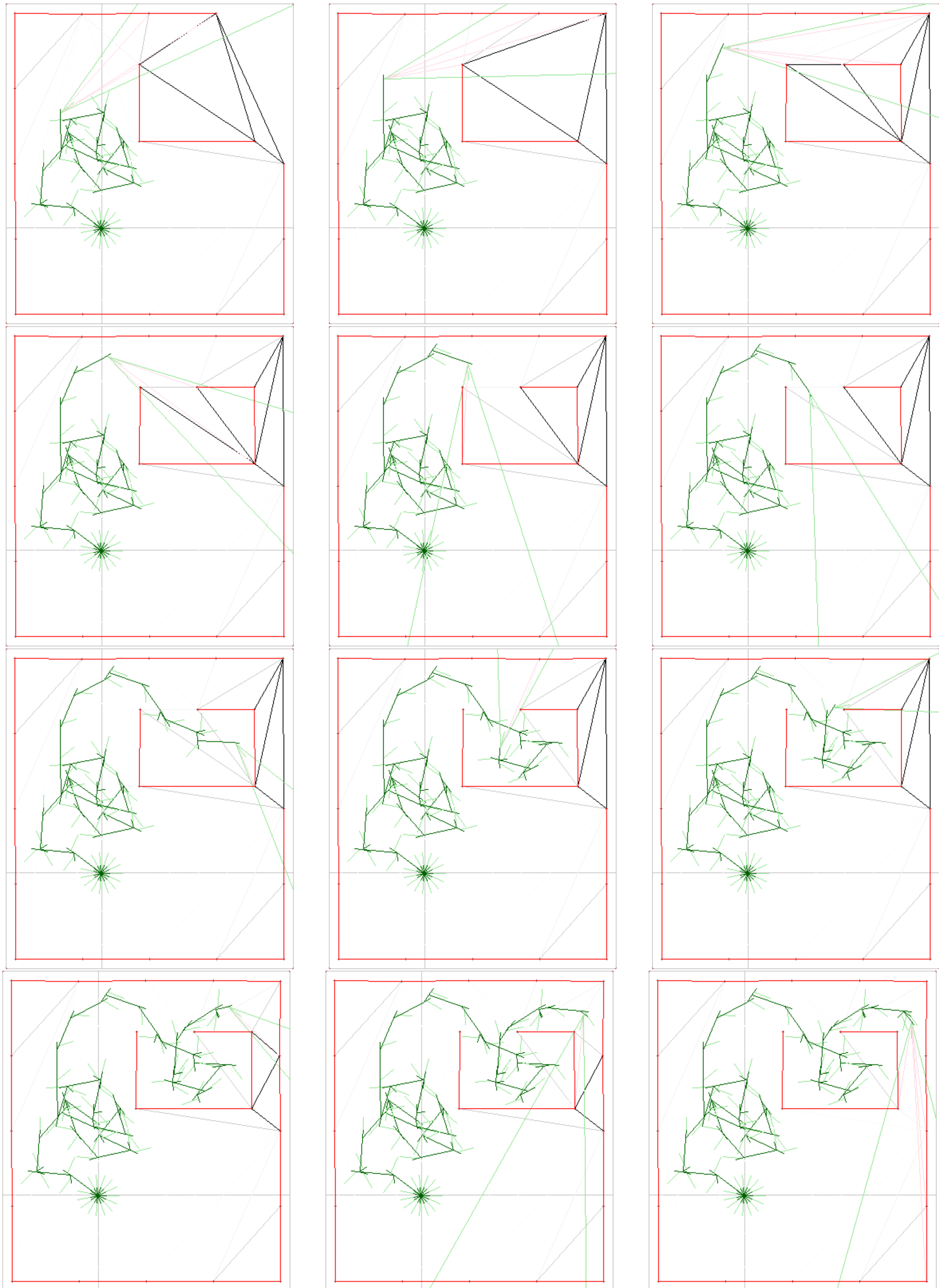


Figure 10: Hidden Room Exploration. Obstacle map with robot path for selected moves 33–66. Move 33 selects the obstacle L3 (figure 9(b)) to view, allowing new features to be detected. Subsequent hypothesised obstacles selected yield further information about the scene. After just 66 moves the entire area has been viewed and all obstacles discovered.