Background Subtraction and Accessibility Analysis in Evidence Grids

Peter Anderson-Sprecher, Reid Simmons, and Daniel Huber

Abstract— **Evidence grids are a popular representation for fused data from multiple sensors. Previous attempts at background subtraction within evidence grids either do so prior to sensor fusion or do so naively, simply ignoring any cells with a high background occupancy probability. A key weakness of these approaches is that they cannot reason about interiors of objects or other unobserved regions. Recognizing and removing solid object interiors is important for any application that must be able to differentiate between occupied and unknown space after background subtraction.**

In this paper, we propose accessibility analysis as a method for the removal of interior regions. We then present and compare two approaches for performing background subtraction with accessibility analysis in evidence grids. Performance is measured using a 3D evidence grid in a test bed for a sensing system designed for use in safety monitoring of an automated assembly workcell. Within the parameters of the present study, both techniques allow for precise detection of foreground objects while fully removing background objects. Subtraction runs in near real-time, even for large grids.

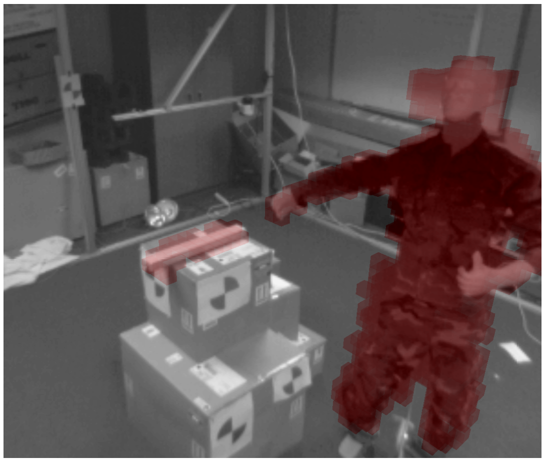


Fig. 1. Background subtraction visualization. Red indicates sensed foreground regions.

I. INTRODUCTION

Evidence grids, also called occupancy grids, were first introduced by Elfes and Moravec [3], [9], and are a popular technique to store probabilistic occupancy data about environments based on an arbitrarily large collection of sensor data. As with any sensor-based system, background subtraction is commonly used to remove static, unmoving objects in the environment and focus on objects of interest. Figure 1 shows foreground regions in a 3D evidence grid following background subtraction.

Most previous approaches to multi-sensor background subtraction have done so directly on sensor images using 2D vision techniques [1], [4], [6]. Background is subtracted from each individual sensor, then remaining foreground data are combined to determine occupancy of foreground objects.

However, performing background subtraction directly at the evidence grid level confers several advantages. For instance, by simply discarding an individual reading when it fits a background model, we lose the ability to differentiate the region in-between the sensor and the sensed object, which is known to be empty, from the region beyond the reading, which is unobserved. Distinguishing between unknown and unoccupied regions is important for many applications, such as safety systems, where the potential cost of an object occupying an unobserved region is high. Since evidence grids keep track of whether regions are occupied, unoccupied, or simply unknown, subtraction at the grid level can remove occupied areas while leaving empty regions alone.

In mobile robotics, Simultaneous Localization and Mapping (SLAM) is another application for which evidence grid-based background subtraction is advantageous. SLAM approaches implicitly generate a background map either through a static environment assumption (e.g., [11]) or through moving object detection (e.g., [13]). Background subtraction in these applications can give important information about unexplored areas, as well as highlighting newly observed objects.

One simple approach to background subtraction within evidence grids is to simply ignore any cells that are occupied, according to some background map. However, the regions subtracted by this method are limited to the exterior surface of objects, as object interiors cannot be directly observed. For large objects, interior regions will always have prior (unobserved) occupancy probabilities, which will remain unchanged during background subtraction. Again, this poses a problem for applications that must conservatively interpret unknown regions as potentially occupied.

To address the problem of interior region removal, we introduce the notion of *accessibility analysis,* a method of measuring and removing regions that may be assumed to be empty due to surrounding background objects. We define a region to be *accessible* if and only if there exists a path to it from some known open region that does not traverse through any occupied region. However, note that, because of sensor error and gaps in coverage, there will often exist small cracks or occlusions in what is an otherwise solid object, leading to

This work was supported by the Office of Naval Research, USA.

Peter Anderson-Sprecher, Reid Simmons, and Daniel Huber are with the Robotics Institute at Carnegie Mellon University, Pittsburgh, PA 15213 USA {peanders@andrew.cmu.edu, reids@cs.cmu.edu, dhuber@cs.cmu.edu}

an overly generous estimate of accessibility. For this reason, we allow for an adjustable degree of tolerance to gaps by measuring accessibility with respect to an object of some given radius.

With this concept in mind, determining accessibility is similar to a standard robotic motion planning operation. This gives a simple algorithm for measuring accessibility:

1) Derive a configuration space by erosion of empty regions.
2) Find all locations reachable from some known accessible region within configuration space.
3) Revert to physical space by dilation of accessible areas.

Figure 2 illustrates these steps on a sample background map.

In this paper, we present two methods for performing background subtraction and accessibility analysis within a fused evidence grid. The first is a natural extension of the simple threshold-based approach to include accessibility analysis. Second, we present a novel probabilistic algorithm that computes accessibility probabilities for all cells based on a min-cost path analysis. Finally, we test and compare both methods using simple background subtraction scenarios in a 4-sensor test workcell. Our results demonstrate that accessibility analysis is an effective method for interior region removal, and that probabilistic subtraction is particularly well-suited for detecting small foreground objects.

A. Prior Work

To date, there has been little research in background subtraction within the evidence grid framework. Work on background subtraction in multi-sensor systems has historically performed subtraction on single sensors prior to data fusion. For cases using monocular cameras with silhouette-based 3D projection, background subtraction has been performed using standard computer vision techniques [1], [4]. Other work has used statistical analysis at the point cloud level, such as in [6].

Background subtraction on fused data is implicit in approaches to Simultaneous Localization and Mapping (SLAM) as well as Structure from Motion (SfM) in non-static environments [8], [13], [14]. There has also been research in object segmentation for fused data. Standard data clustering techniques may be used on either point clouds or evidence grids [7]. There are also studies particular to segmentation in point clouds [5] and in evidence grids [12]. However, while useful for finding certain foreground features, these methods do not directly address the problem of background subtraction and accessibility.

B. Applications and Approach

The research described here is part of the ongoing Intelligent Monitoring of Assembly Operations (IMAO) project at Carnegie Mellon University. The goal of the project is to allow humans to work closely with assembly robots while maintaining a safe environment. The approach uses 3D sensing to closely monitor positions of all persons and other potential hazards in the environment.

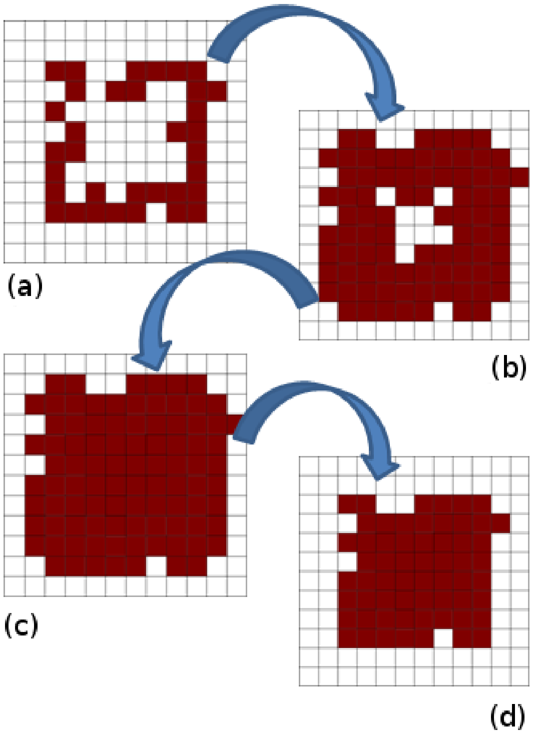


Fig. 2. Accessibility analysis example (thresholded, $r = 1$). (a) The initial background map; (b) Configuration space generated by erosion of empty regions; (c) Configuration space with inaccessible regions filled; (d) Final background map after dilation of accessible regions.

Effective background subtraction is crucial to the success of the project. Interior region removal is particularly important because the safety policy must be conservative, and therefore any unknown regions must be treated as potentially occupied. A background subtraction method for any safety system must also operate in real-time.

The next two sections describe our thresholded and probabilistic approaches to background subtraction.

II. THRESHOLDED BACKGROUND SUBTRACTION

In the thresholded approach, we perform accessibility analysis based on a binary thresholded background map. Thresholding simplifies the problem, but has the disadvantage that once a cell is classified as background, it will always be removed even if the occupancy probability for the cell later increases.

The algorithm starts with a background evidence grid G_B, which contains for each cell the probability of occupancy by a background object. This grid can be gathered by building a standard fused evidence grid at a time when it is known that all foreground objects are out of the environment.

A cell c in the grid can have one of two states, occupied or empty. The probability of occupancy for a cell c according to an evidence grid G_B is denoted $P(c = occ; G_B)$.

Next, we threshold G_B according to a background probability threshold t_B, which produces the set of empty cells E.

$$E = \{c | P(c = occ; G_B) < t_B\} \tag{1}$$

In practice, this threshold can be set to any value above the prior probability, typically just barely above.

To perform accessibility analysis, we use standard morphological operations with a structuring element R, which is defined as an object of uniform radius r. Choosing r is application dependent; it should be approximately the size of the smallest foreground object. Higher values will make the system more robust to sensor noise and gaps in coverage, but also may introduce false negatives.

To derive the configuration space E_R of R in E, we erode E:

$$E_R = E \ominus R \tag{2}$$

Now, we need to determine which regions in E_R are accessible. An accessible cell in configuration space is defined as any cell that can be reached starting from a cell that is known a priori to be accessible.

To compute accessibility, first a connected components algorithm is used to group the cells in E_R. Then, given a set of known accessible cells A_0, a connected group E_i is accessible if and only if it intersects with A_0. This gives a set A_R of cells that are accessible in configuration space:

$$A_R = \{E_i | E_i \cap A_0 \neq \varnothing\}. \tag{3}$$

The method of selecting A_0 is application dependent. In our implementation, we have a grid with a bottom boundary on the floor, a top boundary on the ceiling, and open sides, so A_0 is defined to be all cells on the four sides of the grid. This choice is valid since any foreground entities will need to travel through one of those boundaries to enter the grid.

Finally, we obtain the set of all accessible cells A by dilation of A_R, reversing (2):

$$A = A_R \oplus R. \tag{4}$$

Now, we can use this accessibility map to filter probabilities at run-time by introducing a prior foreground occupancy probability $P(fore)$. This can be derived from the accessibility probability $P(acc)$:

$$\begin{aligned} P(fore) \quad = \quad & P(acc)P(fore|acc)+ \\ & P(\neg acc)P(fore|\neg acc). \end{aligned} \tag{5}$$

Naturally $P(fore|\neg acc) = 0$, and for our purposes we use $P(fore|acc) = 0.5$. This gives our final foreground prior

$$P(fore) = \frac{P(acc)}{2}. \tag{6}$$

Since we are dealing with thresholded cells, $c \in A \iff c = acc$, and we have

$$P(c = fore) = \begin{cases} c \in A & 0.5 \\ c \notin A & 0 \end{cases}. \tag{7}$$

Application of this prior will have the effect of reducing foreground occupancy probabilities to zero for inaccessible cells, and leaving accessible cells unchanged.

III. PROBABILISTIC BACKGROUND SUBTRACTION

The probabilistic version is conceptually similar to the thresholded version, but instead of thresholding and producing a set of accessible cells A, we estimate for each cell its *probability* of being accessible, $P(c = acc; G)$. Thus, our goal is to create a new evidence grid G_A that contains the accessibility probability for each cell

$$G_A(c) = P(c = acc; G). \tag{8}$$

The meaning of a cell being accessible in this case is the same as before: a cell is accessible if and only if in the background grid there exists a path to it from some known accessible cell by an object of radius r. Our approach follows the same motion planning-based steps as in the thresholded method, but is generalized to the continuous domain.

A. Configuration space probabilities

First, we perform an analog of the erosion operator to generate configuration space probabilities. A cell c is in the configuration space E_R of our structuring element R if and only if the set of cells R_c it occupies when centered at c are empty:

$$R_c = emp \iff c \in E_R, \tag{9}$$

where $P(c = emp)$ is the probability that a cell is empty, and $P(c = emp) = 1 - P(c = occ)$.

We define R_c for an object of radius r as

$$R_c = \{c' | ||c' - c||_1 \leq r\}. \tag{10}$$

Now, we can write the configuration space probabilities:

$$P(R_c = emp) = P(\bigcap_{c' \in R_c} c' = emp). \tag{11}$$

Note that (11) is the probabilistic analog to (2). As is typical in occupancy grids, we assume that cell probabilities are independent, so this simplifies to

$$P(R_c = emp) = \prod_{c' \in R_c} (1 - P(c' = occ)). \tag{12}$$

For practical purposes we do not actually want to use the straight background probabilities in computing the configuration space. For large sizes of r, and for long accessibility paths, even relatively low values of $P(c = occ; G)$ will cause accessibility to approach zero rapidly. The problem is even more pronounced with unobserved cells, as they have a prior probability of 0.5. This is mainly a result of the cell independence assumption, and to mitigate it we set unobserved cells and cells with an occupancy probability lower than a fixed threshold t_E to 0, using a thresholding function $f(c, G)$.

$$f(c,G) = \begin{cases} P(c = occ; G) < t_E & 0 \\ c \text{ is unobserved} & 0 \\ \text{else} & P(c = occ; G) \end{cases} \tag{13}$$

Now we can define our configuration space grid G_{E_R} based on the background probability grid G_B, using (12). Also, from here on we choose to work in the negative log space for computational stability.

$$G_{E_R}(c) \quad = \quad -\sum_{c' \in R_c} \log(1 - f(c', G_B)) \tag{14}$$

B. Configuration space accessibility

Next, we must compute the probability that cells are accessible in configuration space, $P(R_c = acc)$. This is analogous to (3):

$$R_c = acc \iff c \in A_R.$$

A cell is accessible in configuration space if and only if there exists some path P of adjacent cells traveling from a known accessible configuration $R_0 \subset A_0$ to a configuration R_d such that $\forall c \in P, R_c \in E_R$. However, because of the infeasibility of combining all possible paths, we approximate this with the most probable path:

$$P(R_c = acc) \approx \max_{\{P|c \in P\}} P(\bigcap_{c' \in P} R_{c'} = emp). \tag{15}$$

In practice, this is not an unreasonable approximation. Most areas will consist of unoccupied space, where the path taken is irrelevant. Areas that do require entry into occupied space will typically have a single point of entry that gives maximum accessibility probability, with all less probable paths being insignificant in comparison.

The log probability of accessibility via a path P is then the conjunction of all probabilities along that path:

$$\begin{array}{rcl} \log P(R_d = acc; P) & = & \log P(R_{P_0}, ..., R_d) \\ & = & \sum_i \log P(R_{P_i} | R_{P_0}, ..., R_{P_{i-1}}) \end{array} . \tag{16}$$

We can now derive the cost of moving from a cell c_i to an adjacent cell c_j from the conditional probabilities if we assume $P(R_{P_i})$ is independent of $P(R_{P_0}), ..., P(R_{P_{i-2}})$ given $P(R_{P_{i-1}})$.

$$\begin{array}{rcl} \text{cost}(c_i, c_j) & = & -\log P(R_{c_j} = emp | R_{c_i} = emp) \\ & = & \sum_{c \in R_{c_j}, c \notin R_{c_i}} G_{E_R}(c) \end{array} \tag{17}$$

The conditional independence assumption here does introduce inaccuracy, in that any time there is a turn in the path there will be cells that are in $R_{P_{i-2}}$ but not in $R_{P_{i-1}}$. This has the effect of penalizing turns in the path; for an object radius r, there are r cells that are double counted in any turn. However, as R incorporates $\approx 2r^2$ cells (depending on grid dimensionality), this is usually a minor problem.

With these costs, we can use Dijkstra's algorithm to compute the most probable path from some R_0 to each R_c. This yields a configuration space accessibility grid G_{A_R} with values

$$G_{A_R}(c) = \min_P \sum_{i=1}^{n} \text{cost}(P_{i-1}, P_i). \tag{18}$$

C. Individual cell accessibility

Finally, we need to extract the individual accessibility probabilities $P(c = acc)$, analogous to (4) in that

$$c = acc \iff c \in A.$$

Since $R_c = acc \rightarrow \forall c' \in R_c, c' = acc$, we can write

$$P(c = acc) = P(\bigcup_{c' \in R_c} R_{c'} = acc). \tag{19}$$

Now for any two cells $c_i, c_j \in R_c$where $P(R_{c_i} = acc) > P(R_{c_j} = acc)$, we assume that the path to R_{c_i} travels through R_{c_j}. This is a reasonable approximation, at least for relatively small values of r. Then $P(R_{c_i} | R_{c_j}) = 1$ and $P(R_{c_i} \cup R_{c_j})$ can be simplified to

$$\begin{array}{rcl} P(R_{c_i} \cup R_{c_j}) & = & P(R_{c_i}) + P(R_{c_j}) - P(R_{c_j}) P(R_{c_i} | R_{c_j}) \\ & = & P(R_{c_i}) \end{array} \tag{20}$$

Therefore, we can take the maximum probability, giving

$$\begin{array}{rcl} P(c = acc) & = & \max_{c' \in R_c}(P(R_{c'} = acc)) \\ & = & \exp(-\min_{c \in R_c} G_{A_R}(c)). \end{array} \tag{21}$$

D. Foreground filtering

During run-time, foreground extraction uses accessibility probabilities in computing the prior $P(fore)$, as shown in (6). This prior may be used during fusion in accordance with standard practice. In our experiments we use the independent opinion pool method with probabilities stored in log-odds format as defined in [3], so foreground filtering consists of adding the prior to the current fused evidence grid.

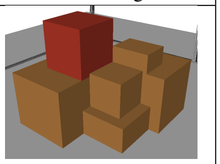
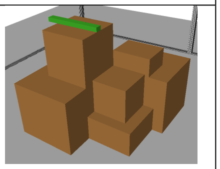
IV. EXPERIMENTAL METHODS

Our test environment consists of a 4m x 4m x 2m frame, with one 3D sensor mounted in each upper corner near the ceiling (Figure 3). Two sensors are Swissranger SR4000 flash LIDARs and the other two are TYZX G3 EVS stereo cameras. All senors are oriented inward at the center of the environment.

Sensor evidence models are generated using a parametrized sensor model similar to the one used by Moravec et al in [10]. Evidence models for each point are one-dimensional, as the sensor resolutions are dense enough to give sufficient coverage without conical evidence profiles, and single-line profiles significantly speed up processing. The model used for all tests has a peak occupancy probability of 0.56 at the reading and a trough at the sensor origin of 0.44. Rasterization into the evidence

TABLE I

TEST SETUPS

Scenario	Foreground	CAD image
Large	One additional box	
Small	One 2x4 placed on top	

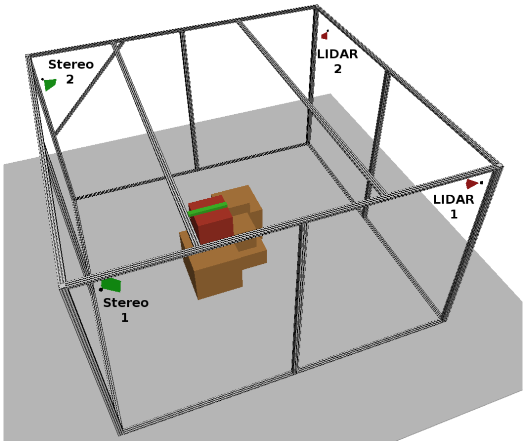


Fig. 3. CAD model of the test environment

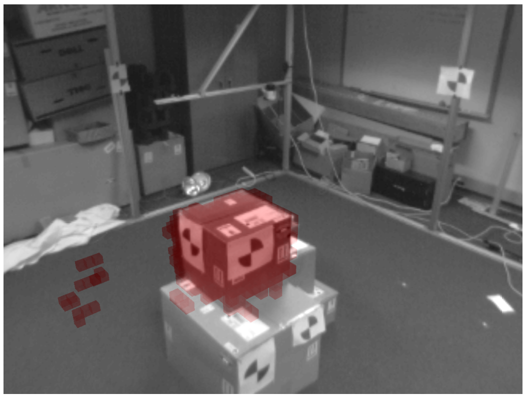


Fig. 4. Foreground occupancy for large object test, using thresholded subtraction and $t_f = 0.73$.

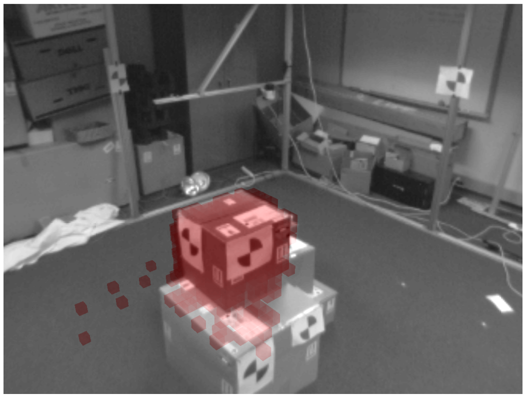


Fig. 5. Foreground occupancy for large object test, using probabilistic subtraction and $t_f = 0.73$.

grid is performed using a 3D version of Bresenham's line algorithm.

All tests were performed using arrangements of boxes with known ground truth. Ground truth evidence grids were generated within the OpenRAVE simulator [2] using simulated sensors. All simulated sensors had the same calibration, field of view, and resolution as the real sensors. We used an ideal 0-1 sensor model for generating evidence grids in simulation, where the voxel containing the sensor reading is marked as occupied with $P(occ) = 1$ and all voxels leading up to it are marked as unoccupied with $P(occ) = 0$.

Tests used a 3D evidence grid containing an area with dimensions 1.7m x 1.5m x 1.5m in the center of the environment. The voxel size for all experiments was 5 cm. For accessibility analysis, all tests used an accessibility object radius $r = 10\,\text{cm}$. The thresholded method used a background threshold $t_b = 0.51$, and the probabilistic method an empty threshold $t_e = 0.1$.

To measure the performance of background subtraction, we compared the observed evidence grid with the simulated ideal grid using a standard precision-recall curve. We used a foreground threshold t_f to classify voxels in the observed grid as occupied or empty, then checked the classification against the ideal grid. Any voxels that were unknown in the ideal grid, i.e., voxels in the interiors of foreground objects, were ignored. Precision-recall curves were generated by measuring error rates across different values of t_f.

In addition to the probabilistic and thresholded background subtraction techniques presented in this work, we also tested baseline versions of these methods with no accessibility analysis. For the thresholded algorithm, we used the set of all empty cells E instead of the accessible cells A when computing the prior probability $P(fore)$ in (7). For the probabilistic algorithm, we substituted the probability that a cell is empty in the background grid $1 - P(occ)$ in place of the accessibility probability $P(acc)$ in (6).

All measurements were gathered in two test scenarios, with background and foreground objects as indicated in Table I. One scenario tested detection of large objects, and the other detection of small objects.

V. EXPERIMENTAL RESULTS

Error rates for the large object test are shown in Figure 8 and for the small object test in Figure 9. Images showing foreground voxels, taken from the perspective of sensor Stereo 1 (see Figure 3), are shown in Figures 4 and 5 (large object), and in Figures 6 and 7 (small object).

In the large object test, results are good for both the thresholded and probabilistic methods. In the versions without accessibility analysis, results are significantly worse. This is

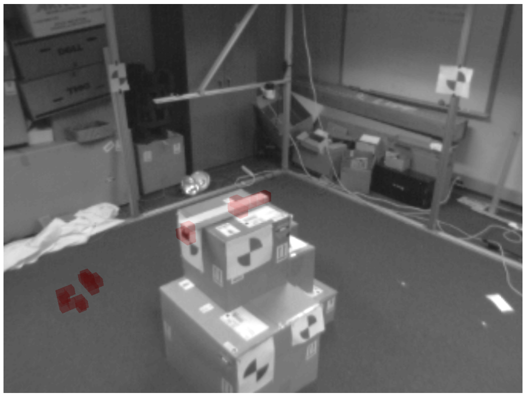


Fig. 6. Foreground occupancy for small object test, using thresholded subtraction and $t_f = 0.73$.

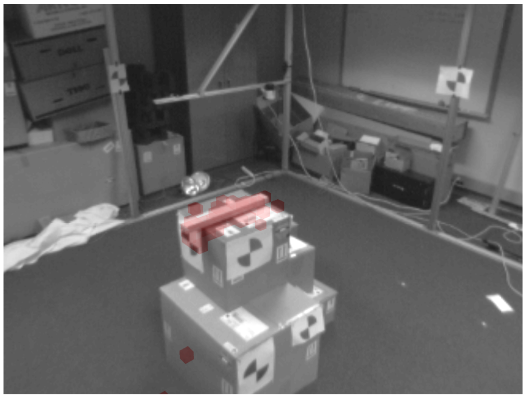


Fig. 7. Foreground occupancy for small object test, using probabilistic subtraction and $t_f = 0.73$.

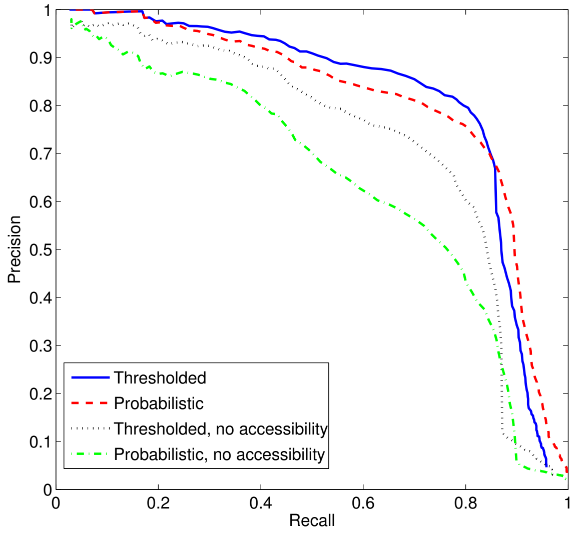


Fig. 8. Precision-recall for large object test

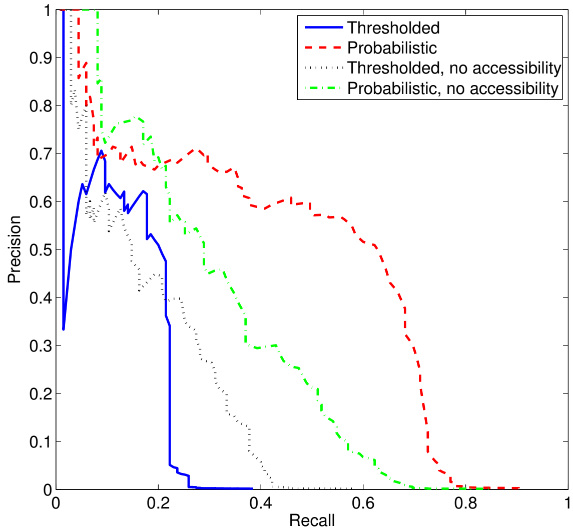


Fig. 9. Precision-recall for small object test

to be expected, as the baseline methods will naturally classify interior regions of the background boxes as occupied for any threshold $t_f < 0.5$. In the small object test, performance is dramatically better with the probabilistic method, particularly under high recall.

Accessibility analysis does negatively affect performance under some circumstances, such as those seen in the low-recall portion of Figure 9. When detecting a flat surface such as the top of the background box, sensor noise will naturally produce scattered false positive voxels, making it uneven. During the accessibility computation, the morphological operations in effect perform a closing operation, which will exacerbate the problem by marking many of the gaps in-between the noise-generated voxels as inaccessible. However, our results indicate that this is not a major problem in practice.

Differences between the thresholded and probabilistic methods are most visible in boundary areas at the edges of background objects, where there is a moderate accessibility probability. In the thresholded method, these regions will likely be part of the background map and always removed, but in the probabilistic version the regions are still sensitive to changes in evidence. Since they are at a boundary between empty and occupied space, the probabilistic accessibility path does not need to go through occupied regions to reach them, and base subtracted foreground probability will be only slightly below 0.5.

In many situations this sensitivity is helpful. For small object detection, the increase in evidence may be a result of the addition of a thin foreground object touching a background object. This means that small objects may be detected even though they occupy voxels with high background occupancy probability (Figure 7), something that is not possible using the thresholded version (Figure 6). This accounts for the drastic difference in recall between thresholded and probabilistic subtraction in Figure 9.

However, this feature can also introduce unwanted noise, as there are circumstances where occupancy probabilities may increase even though no foreground object is added. In Figure 5 we see one example of this type of false positive, which results from occlusion of an area from one or more sensors by a foreground object. While intuitively objects should have higher occupancy probability if there are more sensors reading it, in reality this is often not the case, especially near object boundaries. Since the evidence ascribed to a voxel by a given sensor reading is a function of the distance between them, for voxels near a surface a sensor reading normal to the surface will give higher evidence than one at an oblique angle. The phantom voxels seen on top of the lower box in Figure 5 are a result of the occlusion of low-angle sensor readings. Note that the thresholded results in Figure 4 do not exhibit this behavior.

Adding the background prior during run-time to perform subtraction is fast; our 30,600 voxel grid took under 1 ms on a standard desktop computer. However, computing the background prior is more lengthy: the thresholded method requires approximately 20 ms, and the probabilistic version nearly 400 ms. However, because we use a static background, this computation only needs to be performed once at start-up and is not a significant consideration.

The video clip accompanying this work demonstrates the IMAO system in action with several people moving about the environment. The entire sensing, fusion, and visualization system runs at 10 hz, distributed across two computers.

VI. CONCLUSIONS AND FUTURE WORK

Our results indicate that background subtraction at the occupancy grid level is highly effective. Foreground filtering is reliable using either method presented, accurately detects objects down to single-voxel resolution, can be done in real-time, and is easily parallelizable.

For most applications that require coarse-grained detection of large objects, the thresholded version is sufficient. When greater sensitivity and detection of small objects is required, the probabilistic method performs significantly better, but is also more vulnerable to false positives.

In all cases, accessibility analysis is shown to be effective under the conditions of this study. Even with a small gap tolerance, interior regions are removed completely, allowing for much greater precision than would otherwise be possible. However, we expect the success of the interior region removal to be highly dependent on sensor coverage. Our test environments were designed such that all portions of the objects were visible to at least one sensor, but with less dense sensor coverage determining accessibility would be impossible, leaving interiors of background objects unknown.

A. Future Work

While current results are successful within the parameters of this study, there are several areas where the approach could be extended to improve results and broaden the potential applications of this work.

First, this work was limited to the study of static backgrounds. Testing this approach with a dynamically changing background map would increase its versatility and allow it to be used in less controlled environments. The methods developed in this work are compatible with any standard background modeling technique, as accessibility analysis is performed after the background is gathered. However, using a dynamic background would require recomputing accessibility every time it changes, which will slow performance.

Also, the use of an evidence grid with a cell independence assumption requires many approximations, particularly in the probabilistic algorithm, which leads to decreased accuracy. The evidence grid independence assumption could be relaxed to a first-order Markov assumption, which may allow more accurate accessibility estimates, but likely at significant computational cost.

VII. ACKNOWLEDGMENTS

The authors gratefully acknowledge the contributions of all persons involved in the IMAO project. Special thanks to Paul Rybski, Kartik Babu, Thomas Koletscha, George Brindero, and Chris Niessl for their work in developing the sensor research platform. The IMAO project and this research is funded by the Office of Naval Research under contract #N00014-09-D-0584.

REFERENCES

[1] G.K.M. Cheung, T. Kanade, J. Bouguet, and M. Holler, "A Real Time System for Robust 3D Voxel Reconstruction of Human Motions," *2000 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2000, vol 2, pp 2714.

[2] R. Diankov and J. Kuffner, *OpenRAVE: A Planning Architecture for Autonomous Robotics,* Robotics Institute, Carnegie Mellon University, July 2008, CMU-RI-TR-08-34. *http://openrave.programmingvision.com.*

[3] A. Elfes, "Occupancy Grids: A Probabilistic Framework for Robot Perception and Navigation," PhD Thesis. Carnegie Mellon University, Pittsburgh, 1989.

[4] J. Franco and E. Boyer, "Fusion of multiview silhouette cues using a space occupancy grid", *2005 IEEE International Conference on Computer Vision*, vol 2, pp 1747-1753, 2005.

[5] A. Golovinskiy, T. Funkhouser, "Min-cut based segmentation of point clouds," *IEEE Workshop on Search in 3D and Video (S3DV) at 2009 International Conference on Computer Vision,* 2009.

[6] D. Hahnel, R. Triebel, W. Burgard, and S. Thrun, "Map building with mobile robots in dynamic environments," *2003 IEEE International Conference on Robotics and Automation*, vol 2, pp 1557- 1563 September 2003.

[7] A. Jain, M. Murty, and P. Flynn, "Data clustering: A review," *ACM Computing Surveys,* vol. 31, pp 265-322, September 1999.

[8] A. Kundu, M. Krishna, and C.V. Jawahar, "Realtime motion segmentation based multibody visual SLAM," *Proceedings of the Seventh Indian Conference on Computer Vision, Graphics and Image Processing,* pp 251-258, 2010.

[9] M. C. Martin and H. P. Moravec, "Robotic Evidence Grids," Robotics Institute, Carnegie Mellon University, Pittsburgh, March 1996, CMU-RI-TR-96-06.

[10] H. Moravec and M. Blackwell, "Learning Sensor Models for Evidence Grids", *CMU Robotics Institute 1991 Annual Research Review,* pp 8-15, 1991.

[11] H. Moravec and A. Elfes, "High resolution maps from wide angle sonar," *1985 IEEE International Conference on Robotics and Automation,* Vol 2, pp 116-121, 1985.

[12] D. Vasquez, F. Romanelli, T. Fraichard, and C. Laugier, "Fast Object Extraction from Bayesian Occupancy Grids using Self Organizing Networks," *2006 9th International Conference on Control, Automation, Robotics and Vision*, pp 1-6, 2006.

[13] C. Wang and C. Thorpe, "Simultaneous localization and mapping with detection and tracking of moving objects," *2002 IEEE International Conference on Robotics and Automation,* vol 3, pp 2918-2924, 2002.

[14] S. Wangsiripitak and D.W. Murray, "Avoiding moving outliers in visual SLAM by tracking moving objects," *2009 IEEE International Conference on Robotics and Automation,* pp 375-380, 2009.