

Occupancy-elevation grid: an alternative approach for robotic mapping and navigation

Anderson Souza^{†*} and Luiz M. G. Gonçalves[‡]

[†]*Department of Informatics, University of State of the Rio Grande do Norte, Natal, Rio Grande do Norte, Brazil*

[‡]*Department of Computing Engineer and Automation, Federal University of the Rio Grande do Norte, Natal, Rio Grande do Norte, Brazil*

E-mail: lmarcos@natalnet.br

(Accepted February 27, 2015. First published online: April 15, 2015)

SUMMARY

This paper proposes an alternative environment mapping method for accurate robotic navigation based on 3D information. Typical techniques for 3D mapping using occupancy grid require intensive computational workloads in order to both build and store the map. This work introduces an Occupancy-Elevation Grid (OEG) mapping technique, which is a discrete mapping approach where each cell represents the occupancy probability, the height of the terrain and its variance. This representation allows a mobile robot to know with an accurate degree of certainty whether a place in the environment is occupied by an obstacle and the height of such obstacle. Thus, based on its hardware characteristics, it can make calculations to decide if it is possible to traverse that specific place. In general, the map representation introduced can be used in conjunction with any kind of distance sensor. In this work, we use laser range data and stereo system data with a probabilistic treatment. The resulting maps allow the execution of tasks as decision making for autonomous navigation, exploration, localization and path planning, considering the existence and the height of the obstacles. Experiments carried out with real data demonstrate that the proposed approach yields useful maps for autonomous navigation.

KEYWORDS: Grid maps; Navigation; Occupancy-elevation mapping.

1. Introduction

Environment mapping is considered an essential task for autonomous robot navigation.¹ With the determination of obstacles and free pathways, the robot can interact coherently with the environment. Constructed maps can be used for safety navigation, obstacles avoidance and, can be applied for the robot dealing with unexpected situations.

Several approaches (or models) have been proposed to the representation of environment maps. They can be roughly classified in topological and metric maps.² Topological maps represent the robot environment using a graph-like structure, where nodes usually correspond to places and edges correspond to paths between the places. This representation can be used efficiently for path planning and localization in large-scale environments.³ On the other hand, in the approach based on metric maps the geometric properties of the environment are stored. This technique can generate accurate and fine grained descriptions, allowing either grid-based approximations or geometric primitives to represent the environment.⁴ Among the metric map techniques, the occupancy grid model proposed by Elfes⁵ is one of the most used. The mapped environment is represented as a regular grid composed of cells that contains, each one, an occupancy probability value. Therefore, all places of the environment can be classified into occupied by an obstacle, free of obstacle, or not mapped. This probabilistic approach allows adequate manipulation of the sensory uncertainties.⁶

The occupancy grid idea is spread in the robotics community and widely used until today, and the approach can be naturally extended to three-dimensional environment mapping. The use of a

* Corresponding author. E-mail: andersonabner@uern.br

3D occupancy grid map allows robust autonomous navigation for mobile robots, considering the elevation of obstacles and other structures inside the environment. Furthermore, Liu *et al.*⁷ justified the use of such approach arguing that height information can be used to solve the data association problem in robotic mapping, because 3D models have more information than 2D models thus having less ambiguity.

Despite these advantages, the construction of full three-dimensional occupancy grid models typically has excessively high computational demands for building and storing. Therefore, they are not suitable for direct application on mobile robots.⁴ The use of 2.5-dimensional maps has been introduced as an interesting alternative. This kind of structure does not represent a mapped environment in a 3D array. The structure of height is retained in a more compact two-dimensional array, in which each cell stores the height of a mapped point that represents a small region in the environment. A 2.5D map, named here as elevation map, considers the existence of an obstacle in a place with an elevation. That is, it ignores the existence *probability* of an obstacle, thus *assuming as true* the existence of the obstacle. This deterministic behavior is somewhat undesirable, since it ignores the influences of uncertainties present in the sensory data during the obstacle existence estimation.

In resume, it is known that three-dimensional occupancy grids have high computational demands and that elevation maps do not offer an adequate treatment for occupancy information, considering the fact that robotic mapping is characterized by uncertainty and sensory noises.⁶ As a solution, in the current work we propose an alternative approach for environment mapping with three-dimensional information using a compact structure named OEG. Basically, the proposed representation consists of a two-dimensional grid where each cell stores a probabilistic estimation of occupancy, the height of the terrain and its variance. This approach overcomes the high memory demand imposed by 3D occupancy grid to represent height information, and offers a coherent probabilistic technique to deal with sensor data uncertainties, in the estimation of occupancy and height. Occupancy-Elevation maps can be applied for accurate robot navigation with classification of terrain traversability. That is, in the case of a short obstacle, after considering the height and its variance, that allows the robot to traverse over it, the robot can take this decision. Other cases as passing under a tree branch or advertisement boards can also be treated by the proposed methodology indeed.

Unlike the traditional occupation grid methods, the approach presented in this paper gives the possibility for the robot to identify if an area is occupied or empty, according to the probability values. Besides, if it is occupied, the robot can decide whether it is possible to overcome the obstacle by checking the elevation values and variance and by considering its locomotion skills and hardware characteristics as its own height and wheels diameter. For example, our robot, a Pioneer 3AT, can overcome obstacles up to 3.5 cm, tested in practice. Thus, occupied area of the map with elevation up to 3.5 cm can be overcome by our robot during a navigation task. Another possibility not dealt in this paper is that a ramp up to a certain degree of slope can also be passed by robots. So if from a point to another of the grid the robot detects an inclination degree say of 30°, the robot can pass through this ramp.

In this work, OEGs are constructed from stereo vision data and from laser range find data. We have used a robust inverse sensor model in the estimation of occupancy values and we have applied the Kalman filter method to update the elevation value and its variance. We have confirmed the feasibility and robustness of the proposed mapping method with several experiments performed in indoor environments, in outdoor spaces, and also in mixed sites, which present typical scenes of indoor and outdoor places.

This paper is organized as follows: Section 2 discusses some related works. The OEG formalism is introduced in Section 3. After, Section 4 depicts the range sensor modeling adopted in this work. Section 5 shows the steps to build a local occupancy-elevation map from range data. Practical experiments are presented in the Section 6 to validate our method and Section 7 presents a final analysis of our approach and possible improvements.

2. Related Works

Grid mapping has been largely studied at least during the last three decades. The first works on this subject are presented by Elfes⁵ and Moravec,⁸ where the authors developed mapping frameworks based on Bayesian statistical foundation. In a later work Moravec⁹ introduces an approach based

on Dempster–Shafer evidence theory named Evidence Grid, which accumulates diffuse evidence about the occupancy of a grid of small volumes. Other approaches for grid representation emerged to produce better approximations of the environments. For example, Stachniss and Burgard¹⁰ introduce the coverage maps, which represent in each cell of a given discretization a posterior probability about how much this cell is occupied by an object.

Three-dimensional grid mapping approaches arisen as a natural extension of 2D grid mapping. They represent an important step towards accurate robotic navigation, considering obstacles heights. Liu *et al.*⁷ state that height information in a 3D grid mapping can be used to solve the data association problem since 3D models have less ambiguity. In this context, Moravec⁹ presents an early work using 3D grid mapping approach for robot perception. Furthermore, other researchers use similar approaches. For example, Chen and Xu¹¹ developed a real time 3D occupancy grid mapping for an autonomous land vehicle (ALV). Andert¹² presents a 3D occupancy grid mapping and world modeling for unmanned helicopter navigation. And, Azim and Aycard¹³ use a 3D occupancy grid representation to model the dynamic environment surrounding a vehicle and to detect moving objects.

However, as already mentioned, the construction of full three-dimensional models typically has excessively high computational demands for its direct, online application in a mobile robot.⁴ In large-scale outdoor scenarios or when there is the need of fine resolution memory consumption can become prohibitive.¹⁴ To overcome this question, some researches introduce the elevation map (or 2.5D maps), which is a compact two-dimensional grid representation where each cell of the map stores the height value of the mapped place. Bares *et al.*¹⁵ and Hebert *et al.*¹⁶ have done the first investigations with elevation maps in robotics applications employing the proposed approach for planetary rovers exploration. Recent researches have applied elevation maps for outdoor localization and mapping and for loop closing. Pfaff *et al.*⁴ present a system for mapping outdoor environments with elevation maps. Their formulation uses Kalman filter to estimate the elevation of places and, after this, locations in the environment are mapped into four classes, namely: locations sensed from above, vertical structures, vertical gaps, and traversable cells. Marks *et al.*¹⁷ introduce a different approach based on the gamma distribution, where each cell contains the posterior distribution over the variance of elevations. This representation has been applied for navigating in a wide variety of off-road environments, using a Simultaneous Localization and Mapping algorithm named Gamma-SLAM.

Despite greatly reducing the memory requirement, elevation maps and similar approaches only record positive sensor data, and provide no mechanism for decreasing the occupancy value of objects located on the map.¹⁸ Thus, any erroneous readings such as false sensor positives are never removed from the map. This deterministic treatment ignores the influences of the noises present in the sensory readings in the obstacle existence estimation. Such characteristic is an important drawback of the most 2.5D maps.

In the work of Triebel *et al.*,¹⁹ for example, the authors present an approach for environment mapping with three-dimensional information named Multi-Level Surface (MLS) map, which enables a mobile robot to model vertical structures like bridges, underpasses, buildings or mines. A MLS map consists of a 2D grid of variable size where each cell stores a set of surface patches, and each surface patch is represented as the mean and variance of the measured heights of a 3D point detected by the sensor. This method allows to build a compact map and it can be used as features that support the alignment of maps. However, as the traditional elevation maps, this work assumes as a certain existence of an obstacle, considering uncertainties only in the height estimation. Erroneous readings about the existence of an obstacle, such as false sensor positives never will be correctly treated. Additionally, in some situations this technique will require more memory for map storage that the approach presented in this work, since a map cell can store a large amount of surface patches.

In a subsequent work, Rivadeneyra *et al.*²⁰ propose some improvements to the MLS technique relative to treatment of sensory uncertainties, suggesting a more robust probabilistic model. However, just like its predecessor, this work deals with the existence of obstacle in a deterministic way.

Dryanovski *et al.*¹⁸ present an approach for 3D mapping named Multi-Volume Occupancy Grid (MVOG). The MVOG map explicitly stores information about 3D spaces occupancy. This proposal allows the authors to correct potentially previous erroneous readings by incrementally fusing in new positive or negative sensor information and outperforms existing probabilistic 3D mapping methods in terms of memory usage, due the fact that observations are grouped together into continuous vertical volumes to save space. Nevertheless, this approach underperforms when compared to our method, in terms of memory consumption.

Ocree structures can also be used to building 3D maps, where each node in an octree represents the space contained in a cubic volume, usually called voxel.²¹ This volume is recursively subdivided into eight subvolumes until a given minimum voxel size is reached. Hornung *et al.*¹⁴ introduces an evolution of the work presented by Fairfield *et al.*²¹ Their work presents an integrated mapping system based on octrees named Octomap, which achieves probabilistic, compact 3D maps, however, this approach, in some situations, underperforms when compared to our method.

Our approach, named OEG, represents three-dimensional environments on a 2D horizontal grid, where each cell contains the value of the occupancy probability, the height and its variance, corresponding to the mapped point (region). Thus, we guarantee the use of a compact form for representing 3D information. Moreover, noisy sensory information is adequately addressed through probabilistic tools, allowing that positive and negative sensor data can update the map coherently. OEG maps can be used for robot navigation, considering traversability costs to constructing optimal paths. Path planning methods, as potential fields,²² can be directly applied over OEG, aiming safety and accurate navigation.

3. Occupancy-Elevation Grid Framework

We define OEG as a regular two-dimensional grid with three-dimensional information in which each cell stores the occupancy probability, the height or elevation and its variance for a mapped point in the scene (a small region). Mathematically, an OEG map \mathbf{M} can be described by expression (1).

$$\mathbf{M} = \{\langle O_{0:t,n}, \mu_{0:t,n}, \sigma_{0:t,n}^2 \rangle, \quad n = 1, \dots, N\}, \quad (1)$$

where N represents the number of cells in the grid. The grid mapping algorithm decomposes the high-dimensional mapping problem into smaller one-dimensional problems by modeling the map as a Markov Random Field, where the state of each individual cell can be calculated independently. This means that the occupancy probability, the height and its variance for an individual cell do not affect the estimation of the state for neighboring cells. This is a simplification generally used to make easy grid representations.^{3,5,12} Each cell m_n of the map \mathbf{M} can be described as a tuple in Eq. (2),

$$m_n = \langle O_{0:t}, \mu_{0:t}, \sigma_{0:t}^2 \rangle, \quad (2)$$

where $O_{0:t}$ is the occupancy probability value of the cell, $\mu_{0:t}$ is the height (or elevation) estimation, and $\sigma_{0:t}^2$ is the variance value of such height. These variables are estimated considering the temporal evolution until the instant t . Figure 1 illustrates the composition of these information in the proposed map building algorithm. Elevation values are described by a normal distribution characterized by $\mu_{0:t}$ and $\sigma_{0:t}^2$, that is, $\mathcal{N}(\mu_{0:t}, \sigma_{0:t}^2)$ and occupancy probability $O_{0:t}$ is given by the probability density function, $p(n|\mathbf{s}_{0:t}, \mathbf{z}_{0:t})$.

This representation stands out for meeting the following characteristics:

Three-dimensional modeling: The proposed approach allows the representation of three-dimensional information of the mapped area. Structures and obstacles can be depicted in the OEG map. This characteristic provides useful data for a robust navigation, considering traversability costs based on terrain elevation.

Compactness: OEG map stores 3D information in a 2D grid. This means a significant decrease of memory requirement to represent three-dimensional data.

Updatable: It is possible to update the map (occupancy, elevation and variance values) whenever new sensory readings are provided. This is an important issue when there are dynamic changes in the environment, where the obstacles can change at any time.

Probabilistic approach: Sensory data suffers the influence of noise, which yield uncertainty. Probabilistic tools provide a coherent way for modeling and treating this effect.

In summary, OEG is an alternative to full 3D mapping with a compact representation of the three-dimensional data. Furthermore, both occupancy and elevation values are estimated through stochastic

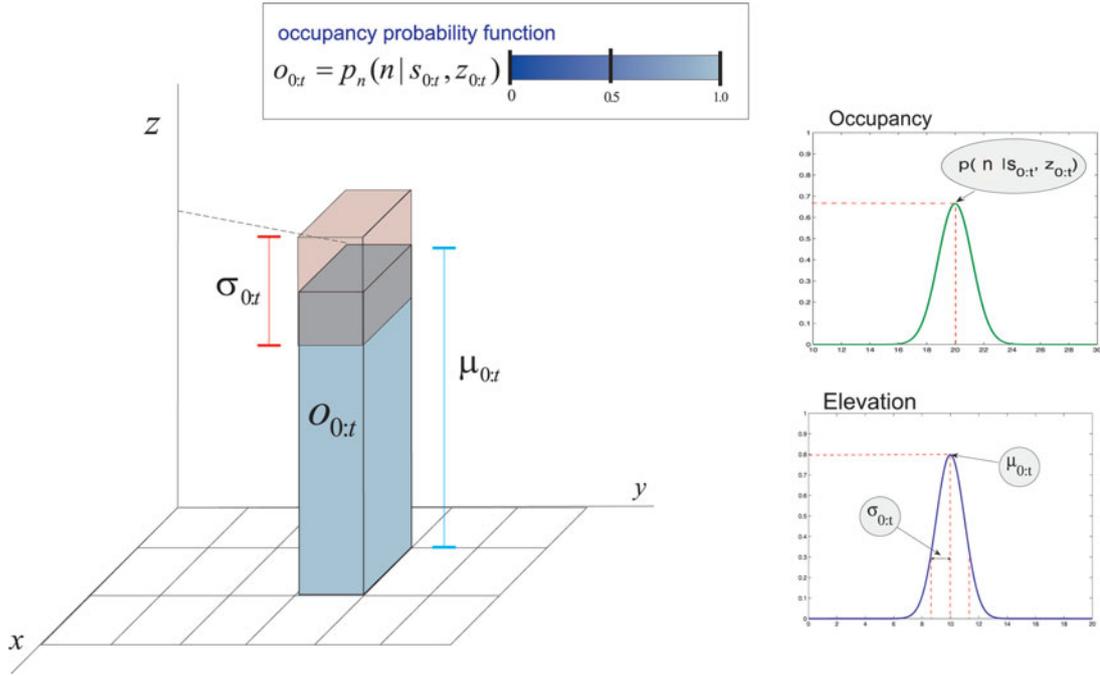


Fig. 1 Occupancy-Elevation Grid composition. Each cell stores occupancy $O_{0:t}$, elevation $\mu_{0:t}$ and variance $\sigma_{0:t}^2$ values.

tools, which allow the correct treatment of the uncertainties inherent to sensory readings. With OEG erroneous readings can be removed from map updating occupancy values from new sensory data. Thus, OEG overcomes the drawbacks inherent to both full 3D grids and elevation maps.

This approach can be directly used to navigation with path planning and obstacle avoidance. Moreover, the robot can classify the obstacles as traversable or non-traversable, based on elevation values and its variance, according to its locomotion skills.

3.1. Occupancy value updating

In order to determine the occupancy probability $O_{0:t}$ of each cell m_n of the map \mathbf{M} , a Gaussian based approach is used like proposed by Elfes.⁵ The occupancy probability value of a cell m_n is calculated by the probability density function, $O_{0:t} = p(n|s_{0:t}, \mathbf{z}_{0:t})$, given by Eq. (3).

$$p(n|s_{0:t}, \mathbf{z}_{0:t}) = 1 - \frac{1}{1 + e^{L_{0:t,n}}}, \tag{3}$$

where

$$L_{0:t,n} = L_{0:t-1,n} + \log \frac{p(n|s_t, \mathbf{z}_t)}{1 - p(n|s_t, \mathbf{z}_t)} - \log \frac{1 - p(n)}{p(n)}. \tag{4}$$

A log notation is used to avoid numerical instabilities for probabilities near zero or one. Note that the occupancy of a cell m_n , depends on the historic of localizations of the robot, $s_{0:t}$, and the historic of sensors readings, $\mathbf{z}_{0:t}$. The probability $p(n|s_t, \mathbf{z}_t)$ specifies the occupancy probability of m_n conditioned to the sensor reading \mathbf{z}_t and to the robot position s_t at time t . This function is named inverse sensor model, and is explained on Section 4.1. $p(n)$ is the prior occupancy value for the cell m_n assigned before any sensory measurement. The common assumption of a prior probability leads to $p(n) = 0.5$, thus, we can rewrite Eq. (4) as is shown in Eq. (5).

$$L_{0:t,n} = L_{0:t-1,n} + L_{t,n}, \tag{5}$$

with

$$L_{t,n} = \log \frac{p(n|\mathbf{s}_t, \mathbf{z}_t)}{1 - p(n|\mathbf{s}_t, \mathbf{z}_t)}, \quad (6)$$

In practice, when occupancy grid map is used for navigation, a threshold on the occupancy probability value calculated by Eq. (3) is applied. An m_n cell is considered occupied when the threshold is reached and is considered free otherwise. This assumption is also used to change the elevation of the cell m_n .

With a more detailed observation over Eq. (4), it is possible to note that in order to change the state of a cell (occupied and free) we need to integrate as many observations as have been integrated to define its current state. That is, if a cell was considered free during the last k times, then it must be observed occupied at least next k times in order for its state to be changed to occupied according to the specified threshold.¹⁴ This property is not desirable in dynamic environments, because the maps do not update quickly following changes in the environments. To overcome this situation, Yguel *et al.*²³ proposes an update approach that defines an upper and lower bound on occupancy estimate given by Eq. (7)

$$L_{0:t,n} = \max(\min(L_{0:t-1,n} + L_{t,n}, L_{max}), L_{min}), \quad (7)$$

where L_{max} and L_{min} represent the upper and lower bound for \log value. With this modification the map can adjust more quickly to changes in the environment.

3.2. Elevation value updating

In order to estimate the elevation and its error for each cell, we apply the Kalman filter formulation, which considers a new measurement, h_t , with its uncertainty represented by σ_t^2 at time t . Those values are calculated by the model presented in the Section 4.2, where the elevation is associated to the occupancy probability value of a cell m_n . In order to integrate a new sensor reading, h_t , we follow the same formulation presented by Pfaff *et al.*⁴ Equations (8) and (9), derived from Kalman formulation, estimate the elevation, $\mu_{0:t}$ and its variance $\sigma_{0:t}^2$ respectively, for a cell m_n , considering all sensor readings up to the time t .

$$\mu_{0:t} = \frac{\sigma_t^2 \mu_{0:t-1} + \sigma_{0:t-1}^2 h_t}{\sigma_{0:t-1}^2 + \sigma_t^2}, \quad (8)$$

$$\sigma_{0:t}^2 = \frac{\sigma_{0:t-1}^2 \sigma_t^2}{\sigma_{0:t-1}^2 + \sigma_t^2}. \quad (9)$$

In our current system, we apply a sensor model in which the variance of the elevation of a measurement σ_t^2 increases with the distance. In this paper, we present experiments with two sensors, a laser range finder and a stereo camera system. Characteristics of each sensor are analyzed to calculate the value of σ_t^2 .

At this point, it can be remarked that the proposed OEG overcomes the high memory requirement imposed by 3D occupancy grid to represent three-dimensional information and offers a coherent way to deal with sensory data uncertainties based on probabilistic tools, allowing estimation of occupancy and elevation.

4. Sensor Model

In general, the map representation introduced in the previous section can be used in conjunction with any kind of range sensor. In our experiments, we have used laser range and stereo camera data to acquire 3D information from the robot environment. To efficiently determine the cells which need to be updated, a ray-casting operation is performed using a 3D variant of the Bresenham algorithm along a beam from the sensor origin to an endpoint (the obstacle). Figure 2 shows this process when there is a new measurement data.

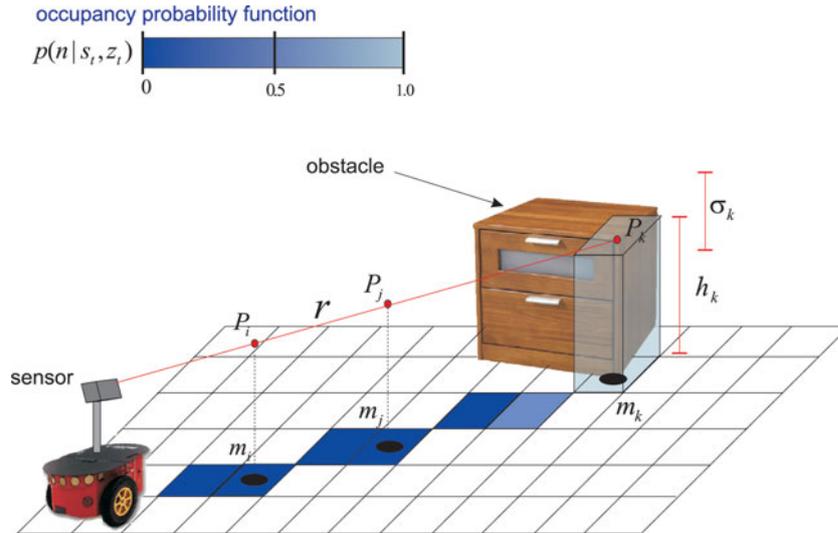


Fig 2. Illustration of points projection on grid map and map updating.

Sensor beam r is discretized by Bresenham algorithm, and each point (sample) along r will have a corresponding grid cell (projection) updated with the sensor model for occupancy, elevation and its variance values. In Fig. 2, there are some examples, three points represented by P_i , P_j and P_k , and its projection on the cells m_i , m_j and m_k . Thus, occupancy probability, elevation and variance values of m_i , m_j and m_k will be updated. Occupancy probability $p(i|s_t, z_t)$, $p(j|s_t, z_t)$ and $p(k|s_t, z_t)$ are calculated by occupancy sensor model, which will be explained in the next section. Elevation h_i , h_j , h_k and variance values σ_i , σ_j , σ_k are estimated by elevation sensor model, explained later.

4.1. Occupancy sensor model

A measurement model makes the interpretation of the sensor data and uses it to populate the map. For the occupancy values, we adopt a modeling that is similar to the one depicted by Andert.¹²

Occupancy sensor model calculates the occupancy probability values for each cell m_n , which has the projection of a point P_n along r , resulting of Bresenham approximation (Fig. 2). Equation (10) computes the occupancy probability value for m_n , relative to the measurement at time t ,

$$p(n|s_t, z_t) = p_{occ}(l) + \left(\frac{k}{\Delta l_p \sqrt{2\pi}} + 0.5 - p_{occ}(l) \right) e^{-\frac{1}{2} \left(\frac{l-l_p}{\Delta l_p} \right)^2}, \tag{10}$$

with

$$p_{occ}(l) = \begin{cases} p_{min}, & \text{if } 0 < l \leq l_p \\ 0.5, & \text{if } l > l_p \end{cases}. \tag{11}$$

Parameter k specifies the significance of a single measurement, l represents the distance from the sensor frame origin to a sample along r , l_p is the distance measured by the sensor, Δl_p is the error in the measured distance, according to the accuracy of the sensor, and p_{min} specifies the minimum probability value considered by the model. This one-dimensional Gaussian model translates noisy sensory data into occupancy information. The function denoted by Eq. (10) iterates over the distance l along the ray r and defines occupancy probability values for cells on the 2D grid at the time t .

The graphic shown in Fig. 3 illustrates a typical occupancy sensor model behavior with occupancy values. Note that measurements at near distance ($l_p = 10$ m) results in more significant occupancy probability value and far distance measurements ($l_p = 15$ m) in low occupancy probability value. This occurs due to effects caused by sensory uncertainties.

From the occupancy values calculated by Eq. (10), it is possible to update the map with Eq. (3).

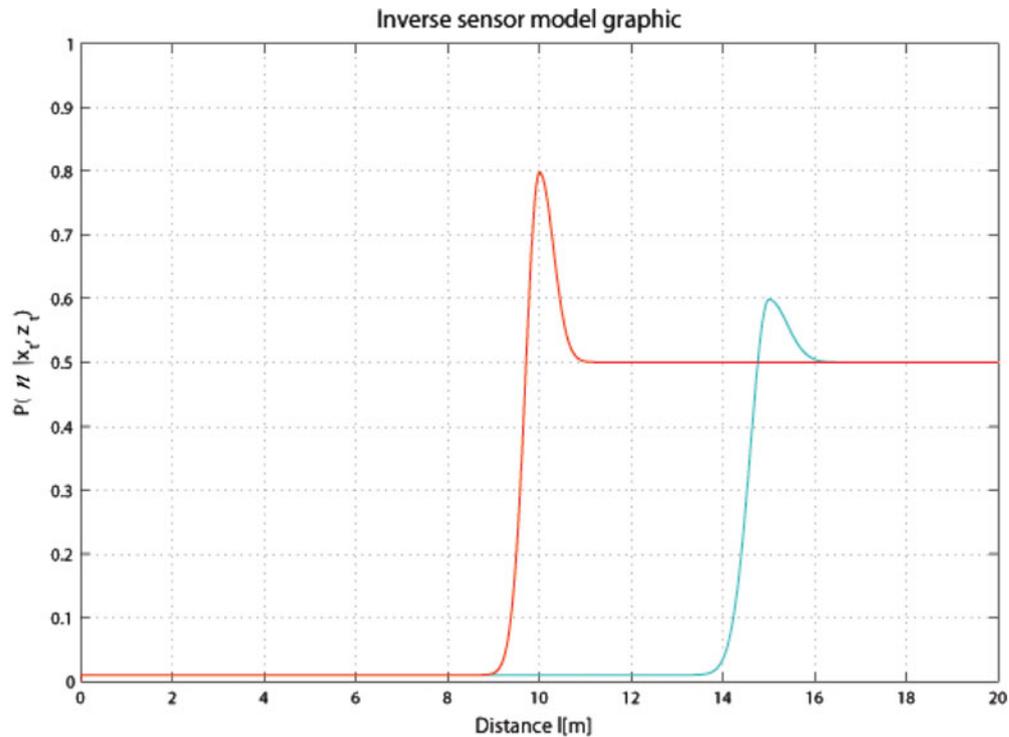


Fig 3. Sensor model simulation, with $l_p = 10$ m and $l_p = 15$ m. In the red curve we have $\Delta z_c = 0.2$ and the cyan curve $\Delta z_c = 0.3$.

4.2. Elevation sensor model

In order to calculate the elevation value h_t for a cell m_n , at the time t , is used the distance l calculated iteratively along r , to estimate 3D coordinates in world frame of a sample, $P_n = (x, y, z)^T$. The height is given by the z coordinate, that is, $h_t = z$. However, the elevation is related to the occupancy, that is, the elevation exists when there is an obstacle. In other words, the occupancy probability must be higher than a threshold l_{occ} , which indicates that probably exists an obstacle. Thus

$$h_t = \begin{cases} z, & \text{if } p(n|s_t, z_t) > l_{occ} \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

Probability and elevation values are estimated at the same iterations over the distance l along the ray r , used to calculate occupancy probability values. It is important to mention that the elevation is computed considering the higher point of the obstacle.

Variance of h_t , represented by σ_t , can be calculated from Δl_p , which increase according to the distance, l_p , measured. The computation of σ_t depends on the sensor properties, taking into account sources of uncertainties which influence its measurements. Figure 4 illustrates the behavior of the sensor model for the elevation mapping. In this figure, the vertical lines (green, red and blue) represents the variation of the height measured, $h_t = 2$ m, which is according to distance estimated from laser range finder data.

With these values, h_t and σ_t , the elevation value and its variance can be updated in the grid map by Kalman filter formulation, Eqs. (8) and (9).

5. Building a Local Occupancy-Elevation Map

Algorithm 1 builds a local map given: the grid map \mathbf{M} , estimated *a priori*; the sensory readings \mathbf{z}_t ; and the robot pose \mathbf{s}_t . This procedure returns the updated OEG map, \mathbf{M} . The occupancy probability and elevation values at time t are estimated iteratively through Bresenham algorithm over a sensor beam r . For each sample \mathbf{p}_n (or point) over r is verified its projection on the bidimensional grid (cell

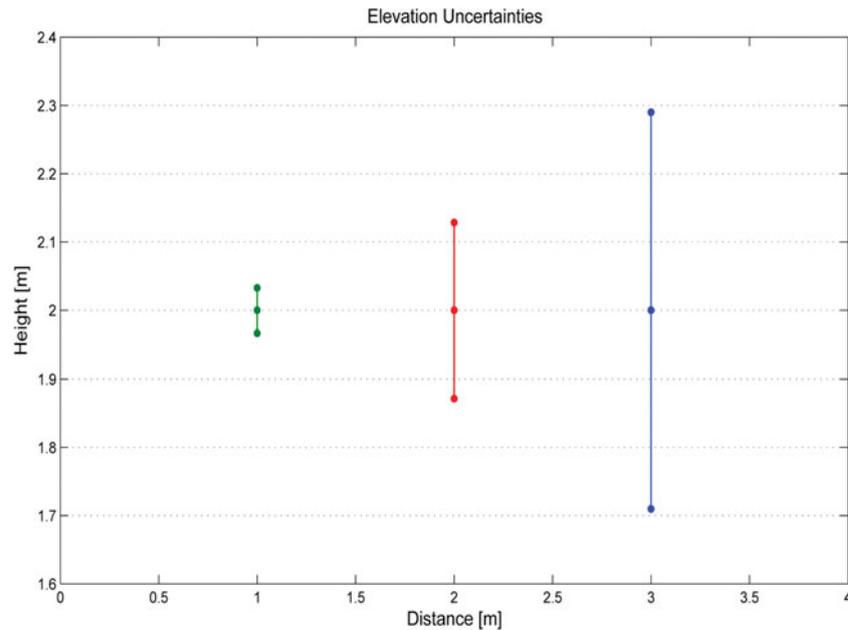


Fig 4. Simulated elevation uncertainties for the sensor model measurement $h_t = 2$ m, with distances $l_p = 1$ m, $l_p = 2$ m and $l_p = 3$ m, $\sigma_t = 0.032$ m, $\sigma_t = 0.13$ m and $\sigma_t = 0.28$ m.

m_n in line 4), and then are calculated the occupancy value, elevation value and its variance for m_n , according to the sensor model presented in Sections 4.1 and 4.2. After, those estimates are integrated to the map \mathbf{M} through Eqs. (3), (8) and (9). At the end of the algorithm, the grid map \mathbf{M} has integrated the sensory input taken on the time t .

Algorithm 1: BuidLocalMap

Data: \mathbf{M} , \mathbf{z}_t , \mathbf{s}_t .

Result: \mathbf{M} .

```

1 begin
2   Consider the ray  $r$  from sensor to the obstacle with length  $l_p$ ;
   // Bresenham algorithm extended to 3D
3   for each sample  $\mathbf{p}_n$  over  $r$  do
4     Identify the cell  $m_n$  where is the  $\mathbf{p}_n$  projection;
5     Calculate  $l$ ; // distance from sensor to  $\mathbf{p}_n$ 
6     Calculate  $p(n|\mathbf{s}_t, \mathbf{z}_t)$ ;
7     Calculate the elevation value  $h_t$  for  $\mathbf{p}_n$ ;
   // Updating M
8     Update the occupancy value on  $\mathbf{M}$ , calculating  $p(n|\mathbf{s}_{0:t}, \mathbf{z}_{0:t})$ ;
9     Update the elevation value on  $\mathbf{M}$ , calculating  $\mu_{0:t}$ ;
10    Update the variance of the elevation on  $\mathbf{M}$ , calculating  $\sigma_{0:t}$ ;
11  end
12 end

```

6. Experiments and Results

One of the most important characteristics of the OEG mapping is that it is *updatable* (for occupancy, elevation and variance of elevation). That means, noisy sensory information are adequately addressed through probabilistic tools, allowing that positive and negative sensor data can update the map coherently, differently from traditional elevation maps. This is an important issue when there are dynamic changes in the environment. Figure 5 illustrates an example of this situation. The robot



Fig 5. Views of a dynamic environment. (a) A person is perceived. (b) No dynamic obstacle is near.

is mapping an indoor environment while people were walking in front of its field of view. Some sensor readings detect the people (Fig. 5(a)), and the OEG formulation translates them as obstacles. Figure 6(a) shows elevation values relative to those readings. Later, there are no people in front of the robot (Fig. 5(b)), and the OEG is updated after new readings, according to the formulation presented in Section 3 like is shown in Figs. 6(b) and (c). New occupancy information produces an updating in the elevation data.

In order to validate the proposed mapping, we have performed several experiments using real sensory data. We use data acquired by our robot based on stereo vision measurements and also other set of range data available on this subject is also used, for comparison purposes.²⁴ In general, our approach can be used with any kind of distance sensor. As said, we use data provided by both stereo vision system and laser range finders.

6.1. Stereo vision data

In the first experiments described, an outdoor environment is mapped using a Pioneer 3AT robot equipped with a very simple stereo vision system. In this experiment, the robot has to deal with bushes, poles and with an irregular terrain. Figure 7 shows some images captured by the robot during its mapping trajectory. Figure 8 shows an overview of the mapped space, highlighting the trajectory followed by the robot (red line).

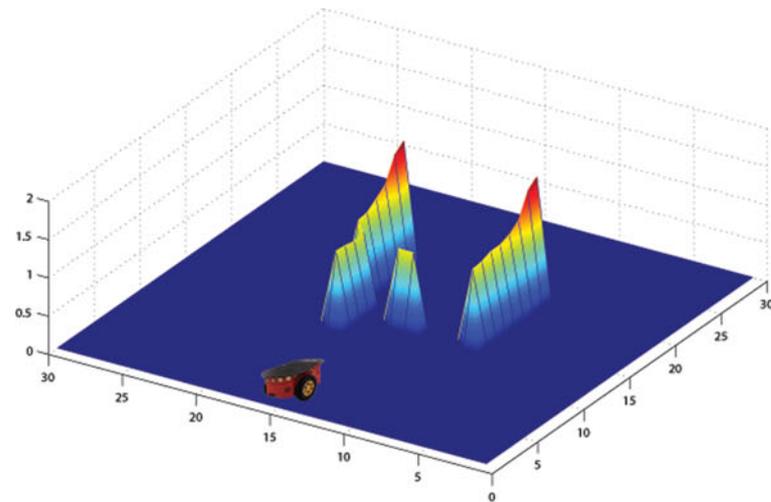
Results of the mapping process can be seen in Fig. 9, where the map was constructed with 0.2 m of cell resolution. The values of each parameter (occupancy, elevation and variance of the elevation) are identified through color bars. Figures 9(a) and (b) highlight specific places, for example, the yellow ellipse corresponds to the mapping of the scenario shown in Fig. 7(a), the white ellipse shows the constructed map of the scene displayed by Fig. 7(b) and red circle emphasizes the grid map of the scene shown in Fig. 7(c).

Obstacles in the environment are correctly identified and their elevations were computed in a coherent way considering the real scenario. Yet, we can see in Fig. 9(b) regions in the map whose elevation values are within the range between 0.1 m and 0.2 m, next to the robot path. These regions cannot be traversed by the robot.

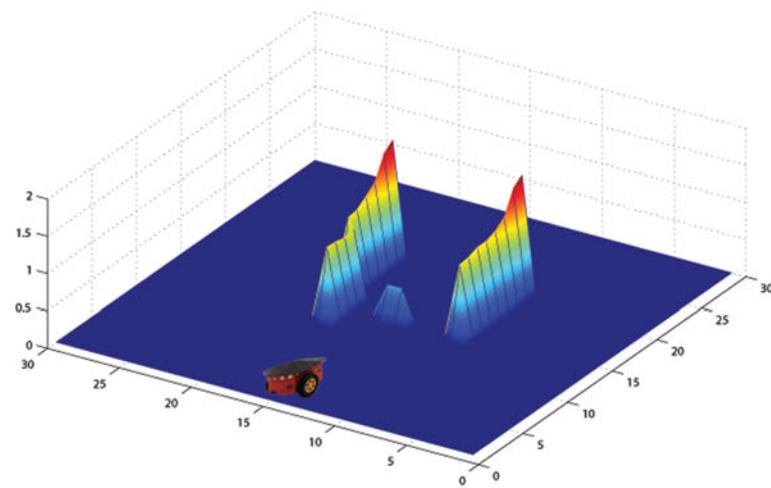
6.2. Laser range data

In this subsequent experiment, we considered a real world data set introduced by Smith.²⁴ The sensor readings have been acquired using two laser range finders LMS 291-S14 scanning over 90°. We apply the sensory models introduced in Section 4, in order to determine the occupancy probability and elevation of a cell following a 3D variant of the Bresenham algorithm. The mapped area can be seen in Fig. 10. The robot's trajectory specified by the red line is estimated through a visual odometry process from the camera images.

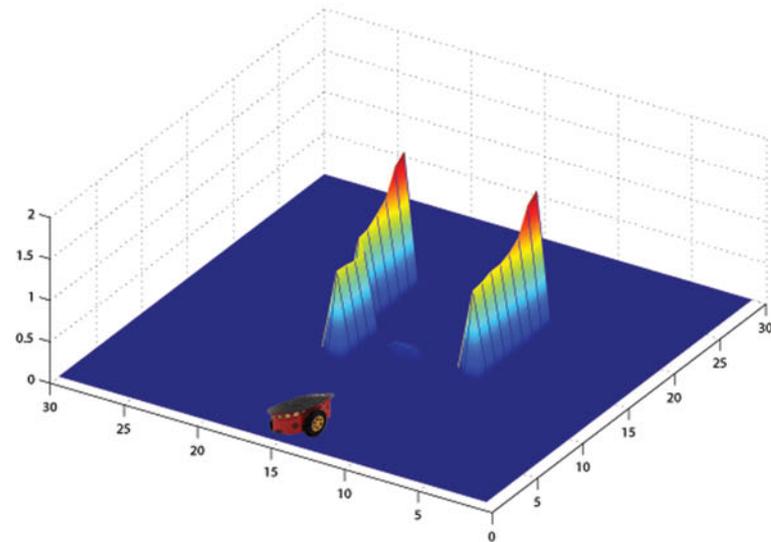
Figure 11 shows the resulting map. Figure 11(a) expresses the values of the occupancy probability while Fig. 11(b) shows the elevation values and Fig. 11(c) shows the values of the variances relative



(a)



(b)



(c)

Fig 6. Estimated elevation values on the OEG map in a dynamic environment. (a) Elevation values with a dynamic obstacle. (b) Elevation values updated by the first readings without dynamic obstacle. (c) Elevation values updated by the second readings without dynamic obstacle.



Fig 7. Scenes captured during the mapping. (a) Free path and bushes. (b) Pole. (c) Bush.



Fig 8. Overview of the mapped space.

to elevations. The values of each parameter are identified by color bars. Low variance values of the elevations map mean that the elevation values present high accuracy and higher variance values indicates low accuracy of the sensor. The robot can use these values (occupancy, elevation and variance of elevation) to identify traversable areas inside its environment considering its motion skills.

Furthermore, it is important to note that mainly in outdoor environments a robot can face some overhang structures such as tree branches and advertisement boards. The proposed mapping can be

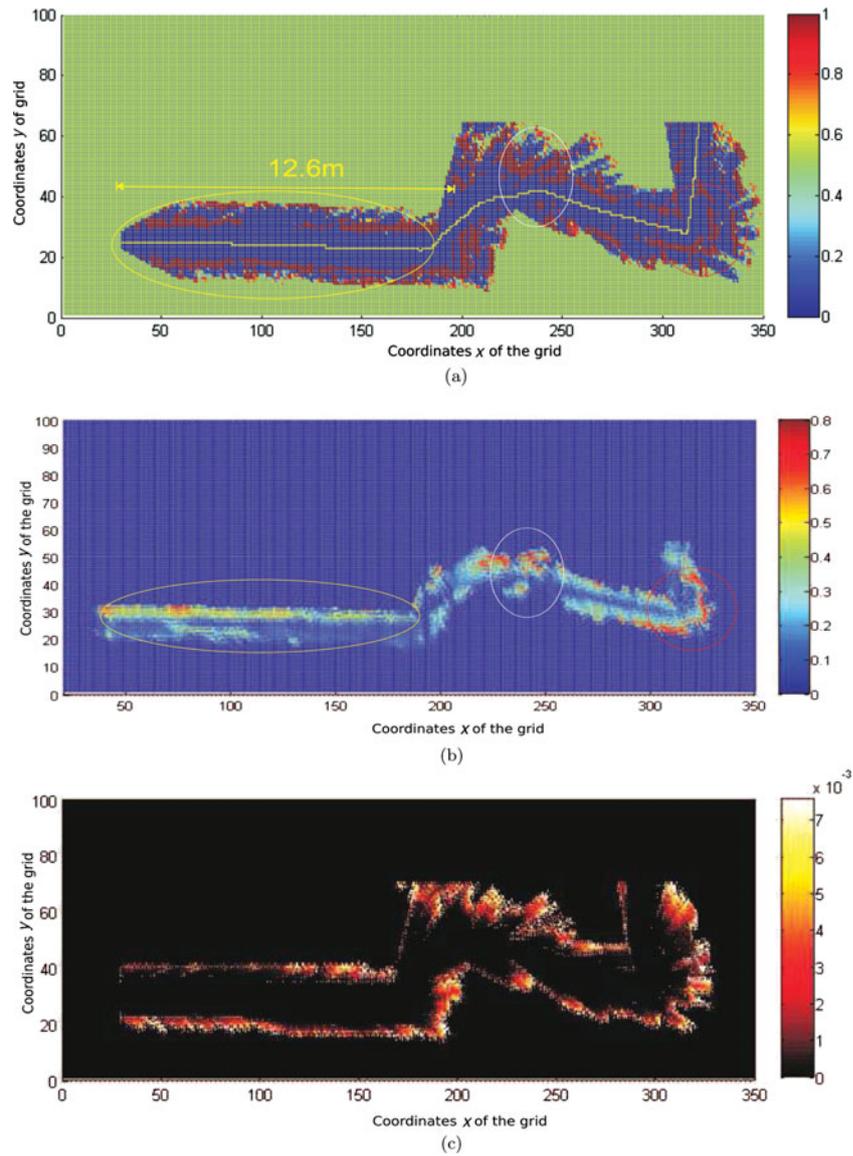


Fig 9. Estimated OEG map for outdoor environment. The values of each parameter are identified through the color bars: (a) occupancy, (b) elevation and (c) variance of the elevation.



Fig 10. Aerial view of the mapped area.

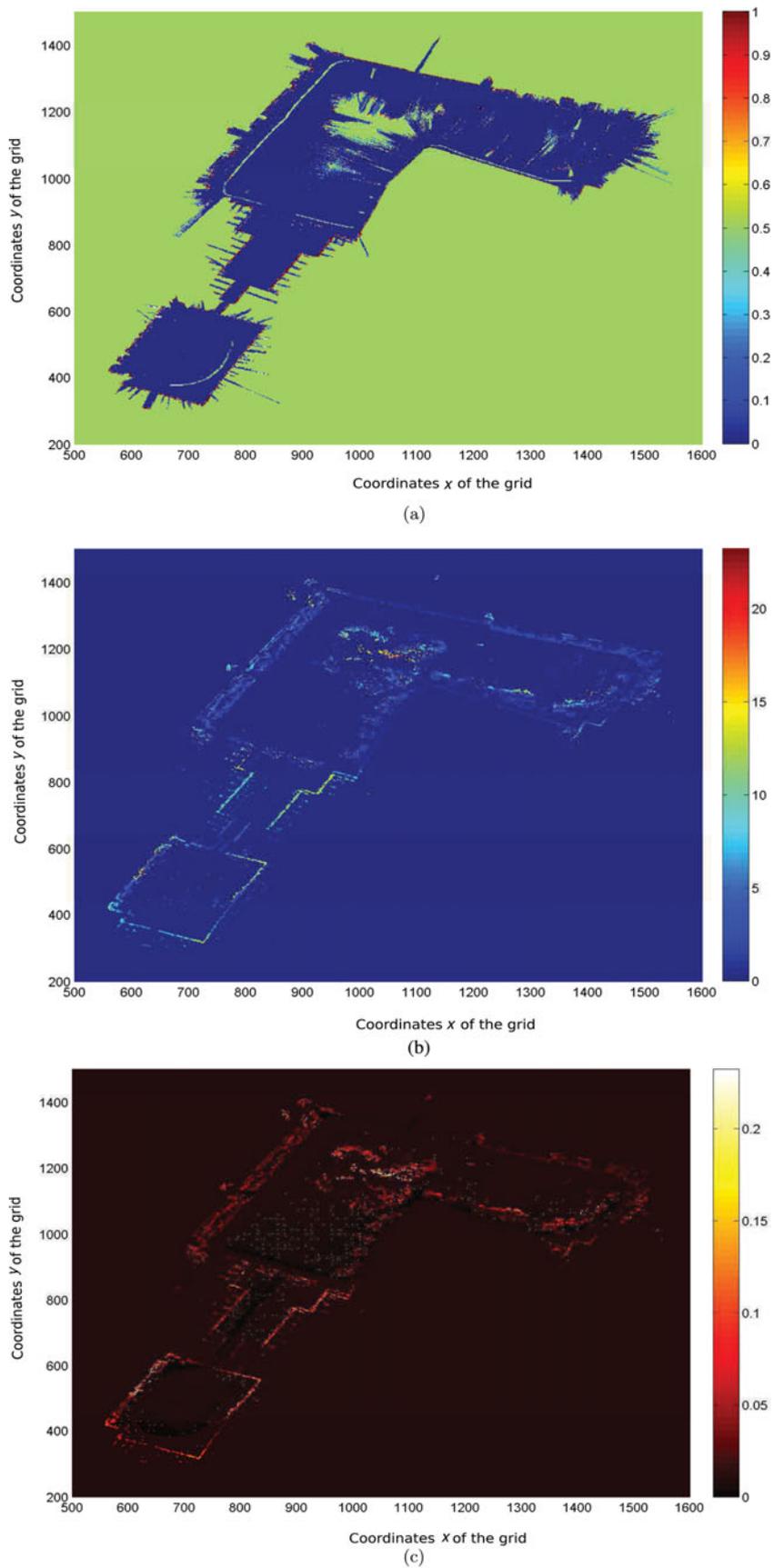


Fig 11. Estimated OEG map for outdoor environment. The values of each parameter are identified through the color bars: (a) occupancy, (b) elevation and (c) variance of the elevation.

Table I. Memory consumption analysis.

Map data set	Mapped		Memory consumption [MB]			
	Area [m ³]	Res. [m]	Full Grid	MVOG	OctoMap	OEG
FR-079 corridor	43.8 × 18.2 × 3.3	0.05	80.54	–	41.70	3.83
		0.1	10.42	–	7.25	0.96
Freiburg outdoor	292 × 167 × 28	0.2	654.42	–	130.39	14.63
		0.8	10.96	–	4.13	0.91
New college (Epoch C)	250 × 161 × 33	0.2	637.48	–	50.70	12.07
		0.8	10.21	–	1.81	0.75
New college (Epoch A)	80 × 100 × 33	0.1	390.75	16.44	56.31	9.6
		0.5	3.24	0.58	0.78	0.38

adapted to deal with these situations, without difficulties, by setting a height threshold. It is important to highlight that one of the main contributions of the proposed mapping approach is the generation of a useful map for robot navigation. In order to do that, we can limit the mapping process to build the map considering only structures a bit higher than the robot height. The other ones which are much higher than the robot and with underpassages can be ignored by the mapping since the robot can pass under them. That is, they will not affect the robot navigation. Thus, when the robot visualizes an overhang tree branch in a higher position than its height, for example, it will be ignored. Figure 12 illustrates an experiment dealing with this situation. Figure 12(a) shows the robot view with some trees and their branches. Our robot is 0.46 m tall, thus, it is able to pass under the trees branches. If it is used the mapping without height thresholding, the generated elevation values would indicate that the robot could not pass under the trees (Fig. 12(b)). However, if it is employed the mapping with height thresholding the produced elevation values demonstrate that it is possible to pass under the trees (Fig. 12(c)).

It is important to say that the mapping of thin obstacles depends on the sensor quality. However, in our approach, if the sensor can capture these obstacles they will be represented with its degree of certainty in the map. For example, it is possible that a thin board or wire can be strategically posted in front of the robot in such a way that the sensor cannot capture it, then the process would fail. Yet, sensors as the Kinect have a lower and higher distance limiting so obstacles outside this range would not be captured as well. Of course, in these cases, there is no technique that can avoid the problems caused.

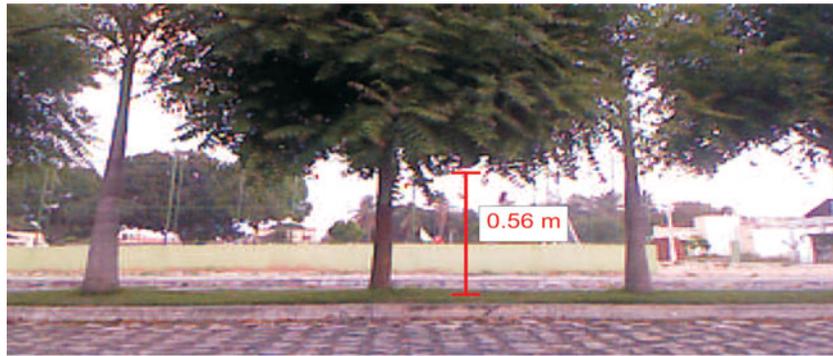
6.3. Analysis of memory usage

We have done an analysis of the memory usage by comparing our approach with the 3D occupancy grid mapping method presented by Andert,¹² MVOG introduced by Dryanovski *et al.*¹⁸ and OctoMap shown by Wurm *et al.*²⁵ The memory size provided for data structures assume single-precision floating point representations and is based on New College data set,²⁴ Freiburg¹ data set and results presented in refs. [18] and [25]. The results of this investigation, considering different resolutions (res.), are summarized in Table 6.3. Figure 13 shows the reduction achieved by OEG maps compared to Full 3D Grid, MVOG and OctoMap approaches for each data set.

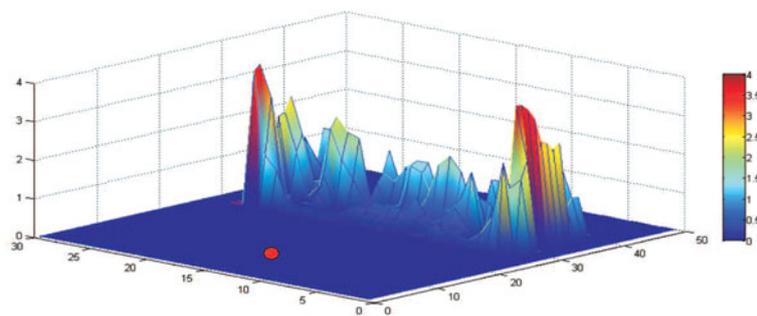
7. Conclusion

We have proposed a novel approach to perform environment mapping using a mobile robot that builds a map representation based on a probabilistic occupancy grid and an elevation map named OEG. Our approach can be used for robust autonomous navigation. To build the map, the system interprets data provided by a stereo camera system. However, as shown, the map representation introduced can also be used in conjunction with other kind of distance sensor, in fact it was also experimented with range finder data.

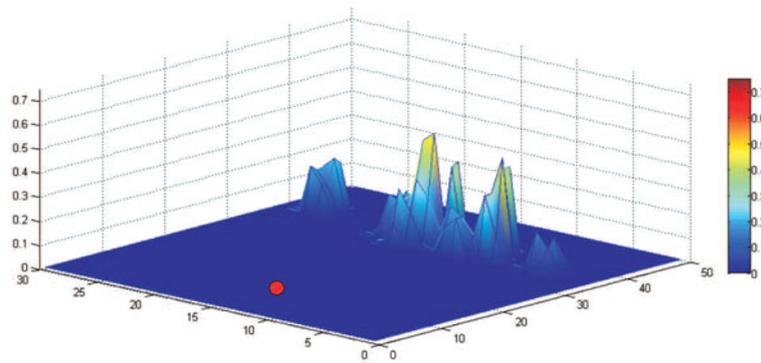
¹ Courtesy of B. Steder and R. Kümmerle, available at <http://ais.informatik.uni-freiburg.de/projects/datasets/fr360/>.



(a)



(b)



(c)

Fig 12. Overhang structures mapping. The red point in (a) and (b) represents the robot position. The color bars quantify the height of the considered obstacles in each case ((a) shows obstacle form 0 to around 3.5 m and (b) shows obstacle from 0 to around 0.5 m). (a) Robot view. (b) Elevation values without height thresholding. (c) Elevation values with height thresholding.

Our method allows a robot to perform mapping of environments actually coherent with sensory data provided by the perceptual system of the robot using less processing requirements and memory. During experimental evaluation, our method has been tested in indoor and outdoor situations, and the results show that the generated maps are consistent with the environments. The obtained maps can be used directly in tasks like path planning, obstacle avoidance and grid-based localization. Furthermore, the proposed OEG map can help the autonomous operation of a mobile robot system and its tasks execution considering traversable areas.

As the very first future work, we are already performing tests on more complex environments in order to validate the coherent construction of the proposed map. Further, we believe that other techniques can be developed on the top of this system for dealing with the several situations experimented in real scenarios where the robot can go ahead without problems or not, as ramps

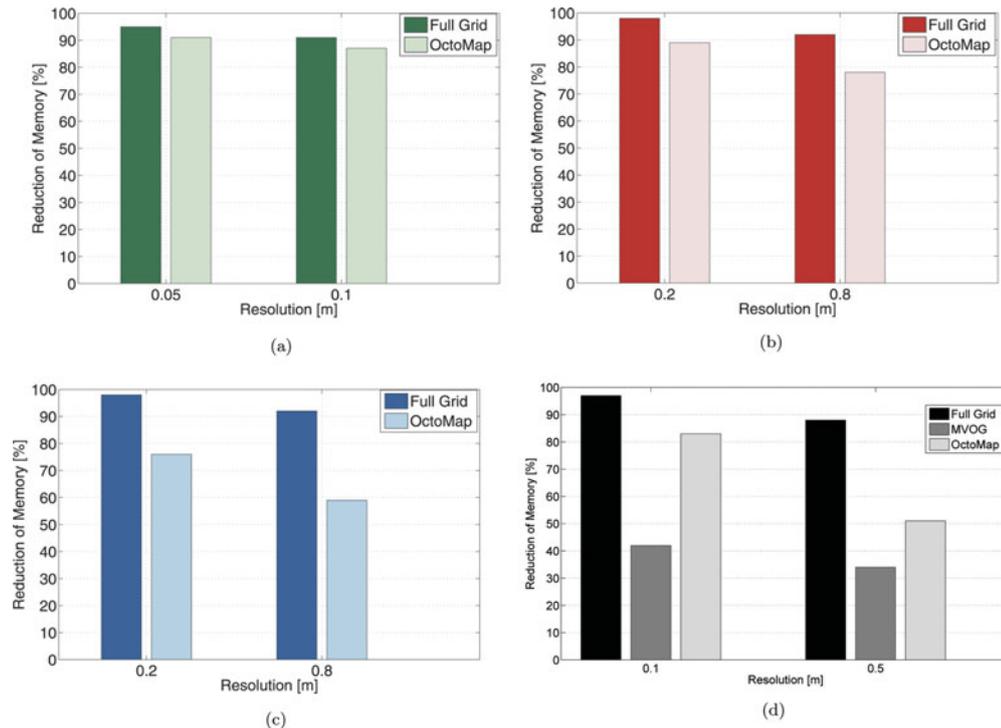


Fig 13. Reduction of memory. (a) Reduction achieved for FR-079 corridor mapping. (b) Reduction achieved for Freiburg mapping. (c) Reduction achieved for New College (Epoch C) mapping. (d) Reduction achieved for New College (Epoch A) mapping.

and other transposable (say small) obstacles. On situations of untransposable obstacles, techniques can be combined with the current one for learning that an obstacles cannot be passed, for example using reinforcement learning. Techniques based in image features as SURF (speeded up robust features) can also be combined to determine possible situations where the robot can transpose or not some region.

References

1. S.-W. Yang and C.-C. Wang, "Feasibility Grids for Localization and Mapping in Crowded Urban Scenes," *Proceedings of IEEE International Conference on Robotic and Automation*, Shanghai, China (2011) pp. 2322–2326.
2. S. Thrun, "Robotic Mapping: A Survey," *In: Exploring Artificial Intelligence in the New Millenium* (Morgan Kaufmann, San Francisco, CA, USA, 2009).
3. E. Einhorn, C. Schröter and H.-M. Gross, "Finding the Adequate Resolution for Grid Mapping - Cell Size Locally Adapting on-the-Fly," *Proceeding of IEEE International Conference on Robotic and Automation*, Shanghai, China (2011) pp. 1843–1848.
4. P. Pfaff, R. Triebel and W. Burgard, "An Efficient Extension to Elevation Maps for Outdoor Terrain Mapping and Loop Closing," *Int. J. Robot. Res. – IJRR* **26**(2), 217–230 (2007).
5. A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE J. Robot. Autom.* **3**(3), 249–265 (1987).
6. S. Thrun, W. Burgard and D. Fox, *Probabilistic Robotics*, 1st ed. (MIT Press, Cambridge, MA, USA, 2005).
7. Y. Liu, R. Emery, D. Chakrabarti, W. Burgard and S. Thrun, "Using EM to Learn 3D Models with Mobile Robots," *Proceedings of the International Conference on Machine Learning (ICML)*, Bellevue, Washington, USA (2001).
8. H. Moravec, "Sensor fusion in certainty grids for mobile robots," *Computer* **9**(2), 61–74 (1988).
9. H. P. Moravec, "Robot spacial perception by stereoscopic vision on 3d evidence grid," CMU Robotics Institute, Pittsburg, Pennsylvania, Technical Report CMU-RI-TR-96-34, 1996.
10. C. Stachniss and W. Burgard, "Mapping and Exploration with Mobile Robots using Coverage Maps," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA (2003) pp. 476–481.
11. H. Chen and Z. Xu, "3d Map Building based on Stereo Vision," *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control - ICNSC*, Ft. Lauderdale, FL, USA (2006) pp. 969–973.

12. F. Andert, "Drawing Stereo Disparity Images into Occupancy Grids: Measurement Model and Fast Implementation," *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, St. Louis, MO, USA (2009) pp. 5191–5197.
13. A. Azim and O. Aycard, "Detection, Classification and Tracking of Moving Objects in a 3d Environment," *Proceedings of 2012 IEEE Intelligent Vehicles Symposium (IV)*, Alcalá de Henares, Spain (2012) pp. 802–807.
14. A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss and W. Burgard, "Octomap: An efficient probabilistic 3D mapping framework based on octrees," *Auton. Robots* (2013), software available at <http://octomap.github.com>. [Online]. Available: <http://octomap.github.com>.
15. J. Bares, M. Hebert, T. Kanade, E. Krotkov, T. Mitchell, R. Simmons and W. R. L. Whittaker, "Ambler: An autonomous rover for planetary exploration," *IEEE Comput. Soc. Press* **22**(6), 18–22 (1989).
16. M. Hebert, C. Caillias, E. Krotkov, I. S. Kweon and T. Kanade, "Terrain Mapping for a Roving Planetary Explorer," *Proceedings of the IEEE International Conference on Robotics and Automation*, Scottsdale, AZ, USA (1989) pp. 997–1002.
17. T. Marks, A. Howard, M. Bajracharya, G. Cotrell and L. Mathies, "Gamma-slam: Visual slam in unstructured environments using variance grid maps," *J. Field Robot.* **26**(1), 26–51 (2009).
18. I. Dryanovski, W. Morris and J. Xiao, "Multi-Volume Occupancy Grids: An Efficient Probabilistic 3d Mapping Model for Micro Aerial Vehicles," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan (2010) pp. 1553–1559.
19. R. Triebel, P. Pfaff and W. Burgard, "Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Beijing, China (2006).
20. C. Rivadeneyra, I. Miller and M. Campbell, "Probabilistic Estimation of Multi-Level Terrain Maps," *Proceedings of IEEE International Conference on Robotics and Automation*, Kobe, Japan (2009) pp. 1643–1648.
21. N. Fairfield, G. Kantor and D. Wettergreen, "Real-time slam with octree evidence grids for exploration in underwater tunnels," *J. Field Robot.* **24**(1–2), 3–21 (2007).
22. H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations*, 1st ed. (MIT Press, Cambridge, MA, USA, 2005).
23. M. Yguel, O. Aycard and C. Laugier, "Update Policy of dense Maps: Efficient Algorithms and Sparse Representation," *Proceedings of the International Conference on Field Service Robotics (FSR)*, Chamonix, France (2007) pp. 23–33.
24. M. Smith, I. Baldwin, W. Churchill, R. Paul and P. Newman, "The new college vision and laser data set," *Int. J. Robot. Res. (IJRR)* **28**(5), 249–265 (2009).
25. K. M. Wurm, A. Hornung, M. Bennewitz, C. Stachniss and W. Burgard, "Octomap: A Probabilistic, Flexible, and Compact 3d Map Representation for Robotic Systems," *ICRA, Workshop on 3D Perception and Modeling*, Anchorage, Alaska, USA (2010).