Exploring Information Theory for Vision-Based Volumetric Mapping

Rui Rocha*†, Jorge Dias* and Adriano Carvalho†

{*rprocha* | *jorge*}*@isr.uc.pt, asc@fe.up.pt*

*Institute of Systems and Robotics
Faculty of Sciences and Technology, Univ. of Coimbra
Pole II, 3030-290 Coimbra, PORTUGAL

†Department of Electrical and Computer Engineering
Faculty of Engineering, University of Porto
Rua Dr. Roberto Frias, 4200-465 Porto, PORTUGAL

***Abstract*— This article presents an innovative probabilistic approach for building volumetric maps of unknown environments with autonomous mobile robots, which is based on information theory. Each mobile robot uses an entropy gradient-based exploration strategy, with the aim of maximizing information gain when building and improving a 3-D map upon measurements yielded by an on-board stereo-vision sensor. The proposed framework was validated through experiments with a real mobile robot equipped with stereo-vision, in order to be further used on cooperative volumetric mapping with teams of mobile robots.**

***Index Terms*— 3-D volumetric mapping, entropy, probabilistic maps, mapping and exploration, stereo-vision sensors.**

I. Introduction

Building a volumetric map of an unknown environment is one of the application fields of autonomous mobile robots. Robotic mapping addresses the problem of acquiring spatial models of physical environments using mobile robots and range sensors, such as cameras or laser range finders [1]. As sensors have always limited range, are subject to occlusions and yield measurements with noise, mobile robots have to navigate through the environment and build the map iteratively. Robots can be used for building fastidious maps of indoor environments [2], but they are particularly useful on mapping missions of hazardous environments for human beings, such as underground mines [3] or nuclear facilities [4].

Grid-based maps [5], [6] are widely used to intuitively represent distributed spatial information, such as occupancy or, closely related, traversability. In [5], Moravec *et al.* built 2-D occupancy grids by using a robot with sonars. In [7], they extended the occupancy grid technique for environment mapping of 3-D grids using stereo-vision. In [2], the notion of occupancy grid was refined to model the cell's occupancy as a continuous value between 0 and 1, denoted as coverage. They used 2-D coverage maps to perform indoor exploration tasks with a robot equipped with sonars. In this article, we propose grid-based 3-D coverage maps with two important improvements: a more compact representation of each cell's state than using histograms [2]; and an efficient Bayesian update procedure. We present a straightforward method to update the map upon new data yielded by range sensors.

When a robot or a team of robots explore an unknown environment to build a map, the objective is to acquire as much new information as possible with every sensing cycle, so as to minimize the time needed to completely explore it. Bourgault *et al.* [8] used occupancy grids to address the single robot exploration problem as a balance of alternative motion actions from the point of view of information gain, localization quality and navigation cost. Yamauchi *et al.* proposed frontier-based exploration [9] whereby robots are driven towards boundaries between open space and unexplored regions. Burgard *et al.* used the frontier-cell concept to develop a technique for coordinating a team of robots when building 2-D occupancy grid [10]. In a seminal work [11], they used entropy minimization to actively localize a robot by minimizing the expected future uncertainty. Extensive research has been devoted to SLAM (*e.g.* [3], [12]), which provides an integrated solution of localization and mapping for applications where a global positioning system is not available. In this article, we does not address SLAM and assume that robots are externally localized. Our approach to multi-robot exploration is closely related with frontier-based exploration [9], [10], with two important contributions. Firstly, we use entropy to explicitly represent uncertainty in the 3-D map, as a means to define a formal information-theoretic background to reason about the mapping and exploration process. Secondly, we propose an entropy gradient-based exploration strategy that drives the robot to frontier cells and maximizes information gain.

Section 2 defines probabilistic 3-D maps and presents how to update a map upon new sensory information from range sensors, based on a Gaussian sensor model. Section 3 defines map's entropy and presents the entropy gradient-based exploration strategy. Section 4 presents experimental results obtained with a real robot equipped with stereo-vision, and demonstrates the validity of the proposed framework. The article ends with conclusions and future work.

II. Probabilistic volumetric maps

This section proposes *probabilistic maps* as a means to represent how uncertain the robots' knowledge about the environment is. The proposed framework can be used to model any phenomena spatially distributed but, since we have validated it

using stereo-vision sensors providing distance measurements[1], hereafter a map is denoted as a *coverage map*, which is a 3-D representation of the environment occupancy with obstacles.

A. Volumetric model

One of the most popular space representation models are *occupancy grids* [5], [6], which are discretised random fields where the probability of occupancy of each independent cell is maintained. Our definition of probabilistic map was firstly introduced in [2], wherein the notion of *occupancy grid* was refined in order to avoid a strictly binary representation of each cell's occupancy (free or occupied), through the notions of *coverage* and *coverage map*. The *coverage* of a cell is the portion of the the cell that is covered by obstacles (a value between 0 and 1). A *coverage map* stores for each cell a probabilistic belief about its coverage. Our main contribution is a more compact representation of the voxel's belief than using histograms [2], and an efficient Bayesian update procedure.

Our volumetric model assumes that we define a 3-D grid $\mathcal{Y}$, which divides the robotic team workspace into equally sized voxels (cubes) with edge $\epsilon \in \mathbb{R}$ and volume ϵ^3. Fig. 1 shows a geometric representation of our model. Any edge of any voxel is assumed to be aligned with one of the axes of a global coordinates frame $\{W\}$. The portion of the volume of a voxel $l \in \mathcal{Y}$ that is covered (occupied) by obstacles is modeled through the continuous random variable C_l, taking values c_l in the interval $0 \leq c_l \leq 1$, and having $p(c_l)$ as its probability density function (pdf). The objective of building a map is to obtain for each voxel $l \in \mathcal{Y}$ an estimate as accurate as possible about its coverage C_l. Let

$$M_k = (\mathbf{x}_k, \mathcal{V}_k) : k \in \mathbb{N}, \tag{1}$$

be the k-th batch of measurements, being $\mathbf{x}_k$ the sensor's position from where measurements are obtained and $\mathcal{V}_k$ the set of measurements belonging to the batch, provided by the robot's sensor at $t = t_k,\ t_k \in \mathbb{R},\ k \in \mathbb{N}$. Let also

$$\mathcal{M}_k = \{M_i : i \in \mathbb{N},\ i \leq k\} \tag{2}$$

be a sequence of k batches of measurements, gathered in the time interval $t_0 \leq t \leq t_k$, being t_0 the initial time before any batch of measurements. Before any batch of measurements, *i.e.* for $k = 0$, the sequence of batches is the empty set $\mathcal{M}_0 = \emptyset$. The knowledge about the voxel's coverage C_l, after k batches of measurements, is modeled through the pdf

$$p(c_l \mid \mathcal{M}_k),\ 0 \leq c_l \leq 1. \tag{3}$$

We define the *3-D probabilistic map*, after k batches of measurements, as the set of random variables

$$\mathcal{C} = \{C_l : l \in \mathcal{Y}\}, \tag{4}$$

containing a coverage random variable for each voxel l of the grid $\mathcal{Y}$. These random variables are described statistically through the set of coverage probability density functions (pdf):

$$\mathcal{P}(\mathcal{C} \mid \mathcal{M}_k) = \{p(c_l \mid \mathcal{M}_k) :\ l \in \mathcal{Y}\}. \tag{5}$$

[1]The framework can also be used with other range sensors, such as laser range, sonar, *etc*., including combinations of different range sensors.

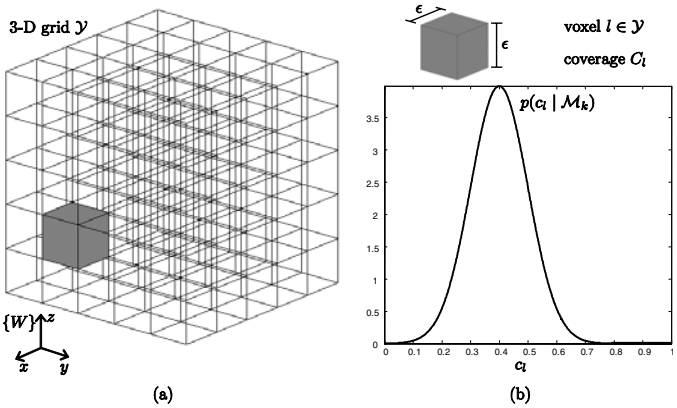

Fig. 1. 3-D grid map: (a) the grid divides the workspace into equally sized voxels, whose edges are aligned with one of the axes of the world coordinates frame $\{W\}$; (b) the coverage C_l of each voxel $l \in \mathcal{Y}$ with edge ϵ, given the sequence of batches of measurements $\mathcal{M}_k$, is modeled through a pdf $p(c_l \mid \mathcal{M}_k)$. The example is the Gaussian pdf $N(\mu_l = 0.4, \sigma_l = 0.1)$.

The coverage of each individual voxel is assumed to be independent from the other voxels' coverage and thus $\mathcal{C}$ is a set of statistically independent random variables.

B. Measurements influencing the voxel's coverage belief

A range sensor typically provides batches of distance measurements from each point where it is located. Consider a batch of measurements $M_k = (\mathbf{x}_k, \mathcal{V}_k)$, being $\mathbf{x}_k \in \mathbb{R}^3$ the sensor's position from where measurements are obtained (shared by all measurements in the batch), and a set

$$\mathcal{V}_k = \{\overrightarrow{\mathbf{v}}_{k,i} \in \mathbb{R}^3 : i \in \mathbb{N},\ i \leq m_k\} \tag{6}$$

of m_k applied vectors (measurements) connecting $\mathbf{x}_k$ to the set of points $\{\mathbf{x}_k + \overrightarrow{\mathbf{v}}_{k,i} : i \in \mathbb{N},\ i \leq m_k\}$ where obstacles are detected. For each measurement $\overrightarrow{\mathbf{v}}_{k,i} \in \mathcal{V}_k$, obtained from the sensor's location $\mathbf{x}_k$, we need to determine the set of voxels $\mathcal{Z}_{k,i} \subset \mathcal{Y}$ whose coverage is influenced by that measurement.

Consider an applied vector $\overrightarrow{\mathbf{u}} \in \mathbb{R}^3$ applied at point $\mathbf{a}$, connecting point $\mathbf{a}$ to point $\mathbf{a} + \overrightarrow{\mathbf{u}}$. The set of voxels $\mathcal{Z}(\overrightarrow{\mathbf{u}}, \mathbf{a})$ traversed by $\overrightarrow{\mathbf{u}}$ can be determined by sampling the vector so that at least one sample per traversed voxel is gathered in a set of w 3-D points $\mathcal{Q}(\overrightarrow{\mathbf{u}}, \mathbf{a})$. To guarantee this minimum sampling, vector $\overrightarrow{\mathbf{u}}$ is divided into segments with maximum length equal to the voxel's edge ϵ. Let $v : \mathbb{R}^3 \rightarrow \mathcal{Y}$ be a function that computes what grid's voxel a given point belongs to. The set of voxels traversed by vector $\overrightarrow{\mathbf{u}}$ applied in $\mathbf{a}$ is

$$\mathcal{Z}(\overrightarrow{\mathbf{u}}, \mathbf{a}) = \{v(\mathbf{q}_i) : \mathbf{q}_i \in \mathcal{Q}(\overrightarrow{\mathbf{u}}, \mathbf{a})\} \subset \mathcal{Y}. \tag{7}$$

Thus, given a measurement $\overrightarrow{\mathbf{v}}_{k,i}$, which is an applied vector in point $\mathbf{x}_k$, the respective set of influenced voxels is

$$\mathcal{Z}_{k,i} = \mathcal{Z}(\overrightarrow{\mathbf{v}}_{k,i}, \mathbf{x}_k) \cup \{l'\}. \tag{8}$$

It includes the set $\mathcal{Z}(\overrightarrow{\mathbf{v}}_{k,i}, \mathbf{x}_k)$ of voxels traversed by $\overrightarrow{\mathbf{v}}_{k,i}$ plus the voxel l' that is immediately behind the obstacle and is more likely to be fully occupied.

Let $\mathbf{w}(l) \in \mathbb{R}^3$ denote the center coordinates of a voxel $l \in \mathcal{Y}$. Let the tuple $D_j^l = (d_j, d_j^l)$ be an individual measurement influencing the coverage estimate of l, being $d_j = \|\overrightarrow{\mathbf{v}}_{k,i}\|$ the distance between the sensor and the detected obstacle, and $d_j^l = \|(\mathbf{w}(l) - \mathbf{x}_k)\|$ the distance between the sensor and the voxel's center. The set of $n_k(l)$ measurements influencing the coverage estimate of l, after k batches of measurements, is

$$\mathcal{D}_k^l = \{D_j^l : j \in \mathbb{N},\ j \leq n_k(l)\} = \{D_1^l, \ldots, D_{n_k(l)}^l\}, \quad (9)$$

having cardinality

$$n_k(l) < \sum_{a=1}^{k} m_a,\ n_k(l) \in \mathbb{N}_0, \quad (10)$$

because not all measurements yielded by the sensor necessarily influence the voxel's coverage. For this reason, we have

$$p(c_l \mid \mathcal{M}_k) = p(c_l \mid \mathcal{D}_k^l),\ \forall_{l \in \mathcal{Y},\ k \in \mathbb{N}_0}. \quad (11)$$

C. Sensor model

The pdf $p(c_l \mid D_j^l)$ represents a sensor model whereby measurements $D_j^l = (d_j, d_j^l)$ are converted in estimates of coverage values $C_l = c_l$ of a voxel l. We generally don't know the exact model of the distribution $p(c_l \mid D_j^l)$. However, since localization errors and sensor errors can be usually assumed to follow a Gaussian model, we represent the voxel's coverage belief through a Gaussian model

$$p(c_l \mid D_j^l) = N(\mu(d_j, d_j^l), \sigma(d_j, d_j^l), c_l). \quad (12)$$

The Gaussian's mean is given by

$$\mu(d_j, d_j^l) = \begin{cases} 0, & (d_j^l - d_j) \leq -\frac{\epsilon}{2} \\ \frac{1}{2} + \frac{d_j^l - d_j}{\epsilon}, & |d_j^l - d_j| < \frac{\epsilon}{2} \\ 1, & (d_j^l - d_j) \geq \frac{\epsilon}{2} \end{cases}. \quad (13)$$

This equation distinguishes three situations: in the first case, the measured distance does not end in the voxel l, with $d_j^l < d_j$, and thus it is more likely that the voxel is fully empty (null coverage); in the second case, the measured distance ends in l and its coverage is inverse proportional to the amount of the voxel covered by d_j (a value between 0 and 1); in the third case, which is only applicable to the voxel l' in equation (8), the measured distance does not end in the voxel l, with $d_j^l > d_j$, and thus it is more likely that the voxel is fully occupied. The standard deviation is given by

$$\sigma(d_j, d_j^l) = \begin{cases} \frac{\sigma_s(d_j)}{\epsilon}, & |d_j^l - d_j| \leq \frac{\epsilon}{2} \\ \frac{\sigma_s(d_j)}{\epsilon} \exp\left(-\frac{|d_j^l - d_j| - \frac{\epsilon}{2}}{\tau}\right), & \text{otherwise} \end{cases}, \quad (14)$$

wherein $\sigma_s(d_j) = \sigma_{min} + \zeta.d_j$, which is a typical behavior of range sensors because accuracy decreases with distance. Equation (14) states that $\sigma(d_j, d_j^l)$ is maximum near the detected obstacle and that, given a damping ratio τ, it decays with $|d_j^l - d_j|$ for voxels farther from the obstacle, which have intuitively less uncertain coverage estimates. Accordingly with equation (3), the Gaussian sensor model has to be truncated so that the cumulative probability over the coverage domain sums up to one, *i.e.* $P(0 \leq C_l \leq 1) = 1$. In [2], it is proposed a sensor model based on a mixture of a Gaussian and an uniform distribution, wherein the latter distribution adds some white noise to ensure a correct normalization when truncating the Gaussian to the range $[0, 1]$. We claim that a better way of normalizing a normal distribution truncated to that interval is to multiply the pdf by a normalization factor

$$\gamma(\mu, \sigma) = \left(\int_0^1 N(\mu, \sigma, x).dx\right)^{-1}, \quad (15)$$

which preserves the normal distribution instead of summing white noise. As we shall see later, preserving the normal distribution makes the coverage update upon new measures quite simple. Fig. 2 shows an example of the sensor model.

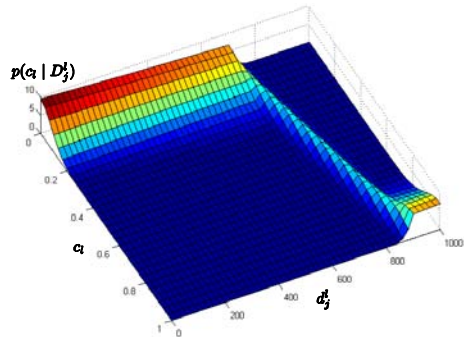

Fig. 2. Example of a sensor model's probability density function: $d_j = 0.8\ m$, $\sigma_{min} = 16\ mm$, $\zeta = 1 \times 10^{-2}$, $\tau = 2\ m$, $\epsilon = 0.2\ m$.

D. Updating the map

Given the 3-D map described by equation (5), updating it upon a new batch $M_k = (\mathbf{x}_k, \mathcal{V}_k)$ means updating the coverage pdf (3) of voxels $l \in \mathcal{Z}_{k,i} \subset \mathcal{Y}$, which are influenced by the new measurements. The previous set of influencing measurements for those voxels is joined with the new $n_k(l) \leq m_k + n_{k-1}(l)$ influencing measurements as $\mathcal{D}_k^l = \mathcal{D}_{k-1}^l \cup \{D_j^l : n_{k-1}(l) + 1 \leq j \leq n_k(l)\}$. We are going to state how to specify the initial voxel's coverage belief $p(c_l \mid \mathcal{D}_0^l) = p(c_l \mid \mathcal{M}_0)$ and how to update the map upon new coverage estimates.

1) Initial map: The initial belief $p(c_l \mid \mathcal{D}_0^l) = p(c_l \mid \mathcal{M}_0)$ represents prior knowledge about the voxel's coverage, before any batch of measurements. Unless there is a previous map of the environment being mapped, it is usually chosen to be the less informative, *i.e.* a pdf with maximum uncertainty.

Entropy is a general measure for the uncertainty of a belief [11], [2]. Although it is defined for either discrete and continuous random variables, the discrete definition is generally preferable because it is an absolute measure of uncertainty and is always non-negative. Being X a discrete random variable over a discrete sample space $\mathcal{S}$ with probability distribution $p(x) = P(X = x)$, entropy is defined as:

$$H(X) = -\sum_{x \in \mathcal{S}} p(x) \log p(x). \quad (16)$$

Note that $H(X) \geq 0$, being assumed the continuity convention $0 \log 0 = 0$. The logarithm's base determines the information unit whereby entropy is measured. Hereafter, we use the base 2 for the logarithm and, in this case, entropy is measured in *bits*.

In order to use the discrete definition of entropy, we use a quantized version of the coverage pdf to compute discrete entropy. Thus, we discretise the coverage continuous random variable $C_l,\ l \in \mathcal{Y}$, with a discrete random variable $C_l^{\triangle}$ having b possible outcomes $c_l^{\triangle} \in \{1, \ldots, b\}$ and a relative frequency histogram $p(c_l^{\triangle})$. Using the entropy definition given by equation (16), the voxel's discrete entropy is

$$H(C_l) \equiv \sum_{i=1}^{b} p(c_l^{\triangle} = i) \log p(c_l^{\triangle} = i). \tag{17}$$

Hereafter, we always use $b = 128$ bins in the computation of equation (17), which means that $0 \leq H(C_l) < 7\ bits$.

It can be shown that the normal distribution is the maximum entropy pdf given the first two moments. A convenient initial belief $p(c_l \mid \mathcal{D}_0^l)$ is thus a normal distribution with $\sigma \to +\infty$, *i.e.* an uniform distribution. In practice, this means choosing a Gaussian with σ much larger (*e.g.* ten times greater) than the sensor standard deviation given by equation (14).

2) Updating the coverage belief of a voxel: Consider a given voxel $l \in \mathcal{Y}$, the set $\mathcal{D}_{n-1}^l$ containing $n-1$ measurements influencing its coverage, and its current coverage probabilistic belief $p(c_l \mid \mathcal{D}_{n-1}^l)$. Given a new influencing measurement D_n^l and a new coverage estimate $p(c_l \mid D_n^l)$, given by equation (12), the new voxel's belief is

$$p(c_l \mid \mathcal{D}_n^l) = \frac{p(\mathcal{D}_n^l \mid c_l).p(c_l)}{p(\mathcal{D}_n^l)} = \beta_1.p(c_l).p(\mathcal{D}_n^l \mid c_l) \tag{18}$$

$$= \beta_1.p(c_l).\prod_{j=1}^{n} p(D_j^l \mid c_l) \tag{19}$$

$$= \beta_1.p(c_l).\prod_{j=1}^{n} \frac{p(c_l \mid D_j^l).p(D_j^l)}{p(c_l)} \tag{20}$$

$$= \beta_1.\beta_2.\prod_{j=1}^{n} p(c_l \mid D_j^l) = \beta_1.\beta_2.p(c_l \mid D_n^l).p(c_l \mid \mathcal{D}_{n-1}^l). \tag{21}$$

Applying Bayes rule, we obtain (18). Then, if we assume that consecutive measurements are independent given the voxel's coverage, we obtain (19). Applying again Bayes rule, we obtain (20). If we assume that $p(D_j^l)$ is constant with j, we finally obtain (21). The constants β_1 and β_2 ensure that the left-hand side sums up to one over all c_l. Note that for $n = 1$ equation (21) uses the initial belief $p(c_l \mid \mathcal{D}_0^l),\ D_0^l = \emptyset$.

Consider a set of $n_{k-1}(l)$ measurements $\mathcal{D}_{k-1}^l = \{D_1^l, \ldots, D_{n_{k-1}(l)}^l\}$, influencing the coverage estimate of a voxel $l \in \mathcal{Y}$ until the $(k-1)$-th batch of measurements, and the respective voxel's belief $p(c_l \mid \mathcal{D}_{k-1}^l)$. When the sensor provides the k-th batch of measurements M_k, some measurements are eventually appended to the set $\mathcal{D}_{k-1}^l$, which yields a new set of measurements $\mathcal{D}_k^l = \mathcal{D}_{k-1}^l \cup \{D_{n_{k-1}(l)+1}^l, \ldots, D_{n_k(l)}^l\}$ having cardinality $n_k(l) > n_{k-1}(l)$. Using recursively equation (21), the voxel's coverage belief after the k-th batch of measurements can be computed as

$$p(c_l \mid \mathcal{D}_k^l) = \beta_3.\Big[\prod_{j=n_{k-1}(l)+1}^{n_k(l)} p(c_l \mid D_j^l)\Big].p(c_l \mid \mathcal{D}_{k-1}^l), \tag{22}$$

wherein β_3 is a normalization constant.

3) Special case of updating Gaussians: At the beginning of the mapping process, each voxel has a Gaussian coverage belief, typically having high entropy. As our sensor model (12) is also Gaussian, when the first influencing measurement D_1^l comes, equation (21) involves the multiplication of two Gaussians. It can be easily shown that given two Gaussians $p(c_l \mid \mathcal{D}_{n-1}^l) = N(\mu_1, \sigma_1)$ and $p(c_l \mid D_n^l) = N(\mu_2, \sigma_2)$, their product yields a Gaussian multiplied by a constant:

$$p(c_l \mid \mathcal{D}_{n-1}^l).p(c_l \mid D_n^l) = \frac{1}{\beta}.N(\mu, \sigma), \tag{23}$$

$$\mu = \frac{\mu_1\sigma_2^2 + \mu_2\sigma_1^2}{\sigma_1^2 + \sigma_2^2}, \tag{24}$$

$$\sigma = \frac{\sigma_1\sigma_2}{\sqrt{\sigma_1^2 + \sigma_2^2}}, \tag{25}$$

$$\beta = \sqrt{2\pi(\sigma_1^2 + \sigma_2^2)}\exp\left[\frac{(\mu_1 - \mu_2)^2}{2(\sigma_1^2 + \sigma_2^2)}\right]. \tag{26}$$

Comparing equations (21) and (23) we conclude that: updating the coverage belief of a voxel between consecutive influencing measurements is as simple as computing the parameters of a new Gaussian through equations (24) and (25); and the normalization constant is $\beta_1.\beta_2 = \beta$, with β given by equation (26). This simplicity of computation is a consequence of the Gaussian nature of sensor model and our careful choice of an initial coverage belief. While in [2] the coverage belief of a cell was represented through histograms with b bins (b is typically more than 10), in our case we represent the voxel's coverage belief as a Gaussian, which is fully characterized by just *two* parameters: μ_l and σ_l. Thus, in the map (5) we have to store only two values for each voxel, which is a much more compact representation than a histogram. Moreover, our voxel's update procedure is very efficient and we can still build histograms upon the pdf with an arbitrary number of bins.

III. Entropy gradient-based exploration strategy

In an exploration mission, the objective is to acquire as much new information about the environment as possible with every sensing cycle. When a robot has to select a new viewpoint for acquiring data through its sensor, we claim that the robot's sensor should be directed to regions having higher magnitudes of entropy gradient and low expected coverage, in the neighborhood of the robot. This strategy drives the robot's sensor to frontier voxels between more explored and less explored regions, so as to maximize the information gain of new acquired data.

Although our method can be applied to a 6 DOF robot, we've been mainly interested on using it on ground mobile

robots with 3 DOF, whose sensor's motion is instantaneously restricted to a plane Γ parallel to the robot's motion plane (*e.g.* the floor plane). For this reason, voxels near to that plane are preferable to be explored. Consider the current robot's pose $Y = (\mathbf{x}, \mathbf{a})$, being $\mathbf{x}$ its current position and $\mathbf{a} = \{\theta, \phi, \psi\}$ its orientation. Given a robot's coordinates frame $\{R\}$, which is obtained from the global (absolute) coordinates frame $\{W\}$ after translation and rotation, the robot's motion plane Γ is defined by two orthogonal axes: a longitudinal axis $\hat{\mathbf{p}}' = [1, 0, 0]^T$, which is the unitary vector along xx axis, and a transverse axis $\hat{\mathbf{q}}' = [0, 1, 0]^T$, which is the unitary vector along yy axis; for example, for an UAV, $\hat{\mathbf{p}}$ would be the axis between tail and head, and $\hat{\mathbf{q}}$ would be the axis connecting the wings. It can be shown that the robot's axes can be expressed in the global coordinates frame $\{W\}$ as

$$\hat{\mathbf{p}} = [\cos\theta.\cos\phi,\ \sin\theta.\cos\phi,\ -\sin\phi]^T, \tag{27}$$

$$\hat{\mathbf{q}} = \begin{bmatrix} \cos\theta.\sin\phi.\sin\psi - \sin\theta.\cos\psi \\ \sin\theta.\sin\phi.\sin\psi + \cos\theta.\cos\psi \\ \cos\phi.\sin\psi \end{bmatrix}. \tag{28}$$

The angles θ, ϕ and ψ are the yaw angle, the pitch angle and the roll angle, respectively, and are assumed to be positive in the counterclockwise direction. Note that axis $\hat{\mathbf{p}}$ can also be viewed as the robot's sensor gaze direction. Any vector $\overrightarrow{\mathbf{u}}$ can be projected on the robot's motion plane Γ as

$$\underset{\Gamma}{\text{proj}}\ \overrightarrow{\mathbf{u}} = (\overrightarrow{\mathbf{u}} \cdot \hat{\mathbf{p}})\hat{\mathbf{p}} + (\overrightarrow{\mathbf{u}} \cdot \hat{\mathbf{q}})\hat{\mathbf{q}}, \tag{29}$$

wherein $(\cdot)$ denotes the internal product of two vectors.

Let denote the applied vector connecting the robot's sensor position $\mathbf{x} \in \mathbb{R}^3$ to the center of voxel l as $\overrightarrow{\mathbf{u}}(\mathbf{x}, l) = \mathbf{w}(l) - \mathbf{x}$. Given a neighborhood around the current robot's sensor position with radius ξ, its new position is selected as the center of a voxel belonging to the set of voxels

$$\mathcal{N}_\Gamma(\mathbf{x}, \xi) = \{l \in \mathcal{Y},\ \|\overrightarrow{\mathbf{u}}(\mathbf{x}, l)\| \leq \xi,\ l = v(\underset{\Gamma}{\text{proj}}\ \mathbf{w}(l))\}. \tag{30}$$

Let define the *map's entropy* after the k-th batch of measurements as

$$H(\mathcal{C} \mid \mathcal{M}_k) = \sum_{l \in \mathcal{Y}} H(C_l \mid \mathcal{M}_k). \tag{31}$$

This is a measure of the map's uncertainty, wherein $H(C_l \mid \mathcal{M}_k)$ denotes the discrete entropy of each voxel l at $t = t_k$. The 3-D grid $\mathcal{Y}$ discretises the 3-D space $\mathbb{R}^3$ at discrete points $\mathbf{w}(l),\ l \in \mathcal{Y}$, equally spaced by ϵ (the voxel's edge). The 3-D map enables us to associate with each of these points an entropy $H(l) = H(C_l)$ given by equation (17), therefore we might say that a continuous entropy field $H : \mathbb{R}^3 \rightarrow \mathbb{R}$ is sampled along the centers of the voxels belonging to the grid $\mathcal{Y}$. The exploration strategy that we propose claims that the robot's sensor should be directed to regions in the neighborhood of the robot, having higher magnitudes of entropy gradient $\overrightarrow{\nabla} H$, and that are more likely unoccupied. Let $l_{\Theta-}$ denote the contiguous voxel to l in the negative direction of axis Θ. A reasonable (first order) approximation to the entropy gradient at the center of a voxel l is

$$\overrightarrow{\nabla} H(l) \approx \frac{1}{\epsilon}[H(l) - H(l_{x-}), H(l) - H(l_{y-}), H(l) - H(l_{z-})]^T. \tag{32}$$

The projection of the voxel's entropy gradient on the robot's sensor motion plane Γ is

$$\overrightarrow{\nabla} H_\Gamma(l) = \underset{\Gamma}{\text{proj}}\ \overrightarrow{\nabla} H(l), \tag{33}$$

with magnitude $\left\|\overrightarrow{\nabla} H_\Gamma(l)\right\|$. If the center of a voxel $l \in \mathcal{N}_\Gamma(\mathbf{x}, \xi)$ is selected to be the next robot's selected position $\mathbf{x}^s$, our method claims that the robot should select the gaze direction $\mathbf{a}(l)$ defined by the unitary vector

$$\hat{\mathbf{p}}(l) = \frac{\overrightarrow{\nabla} H_\Gamma(l)}{\left\|\overrightarrow{\nabla} H_\Gamma(l)\right\|},\ \overrightarrow{\nabla} H_\Gamma(l) \neq \overrightarrow{0}. \tag{34}$$

Accordingly with our exploration strategy, being $E(C_l)$ the expected coverage of a voxel $l \in \mathcal{Y}$, and given the set of voxels $\mathcal{N}_\Gamma(\mathbf{x}, \xi)$ in the robot's neighborhood, the robot's sensor is directed to the voxel

$$l^s = \underset{l \in \mathcal{N}_\Gamma(\mathbf{x}, \xi)}{\text{argmax}} \left(\left\|\overrightarrow{\nabla} H_\Gamma(l)\right\| . [1 - E(C_l)]\right), \tag{35}$$

with a gaze on arrival defined by the unitary vector $\hat{\mathbf{p}}(l^s)$. If the gradient-based criteria is not conclusive, the robot should wander randomly until that condition is not verified.

IV. Experiments with a real robot

The 3-D mapping framework presented in previous sections was used for carrying out experiments with a real mobile robot in our lab. The experiments were performed until the map's entropy, computed through equation (31), was reduced below a given predefined threshold H_{th}. This stopping criteria has an important associated performance measure for the 3-D mapping mission, which is the time instant $t_{k_{max}}$ when it is achieved. This time instant verifies the proposition

$$H(\mathcal{C} \mid \mathcal{M}_{k_{max}}) \leq H_{th} \wedge \forall_{k < k_{max}},\ H(\mathcal{C} \mid \mathcal{M}_k) > H_{th}, \tag{36}$$

which states that the k_{max}-th batch of measurements is the first one for which the map's entropy falls below the threshold.

The mobile robot (see Fig. 3-a) is a Scout robot equipped with sonars, stereo-vision and wireless communication. Sonars are used for detecting obstacles when moving the platform, and for preventing the robot to acquire stereo image pairs below a given distance threshold to obstacles. The stereo-vision sensor (see bottom of Fig. 3-a) is a small, compact, low-cost analog stereo rig, with resolution 160x120 pixels. For computing range data from stereo images, we use the Small-Vision System (SVS) v2.3c, a stereo engine from SRI International (see Fig. 3-b for an example of a depth map yielded by the SVS engine). After calibrating the stereo-vision sensor, we determined the sensor model parameters: $\sigma_{min} = -0.06\ mm$, $\zeta = 3.75 \times 10^{-3}$ and $\tau = 2\ m$. The robot is localized through a RGB camera covering the

robot's workspace within our lab. The colored markers on the top of the robot's platform are used for determining the robot's pose in the global reference frame $\{W\}$ through a color segmentation algorithm, which runs in a remote PC.

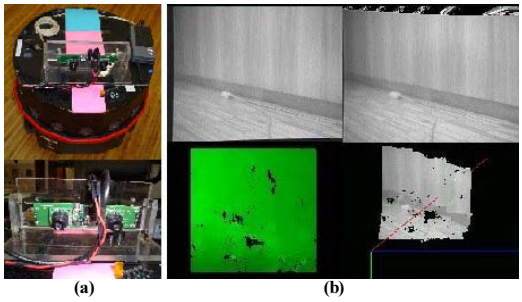

Fig. 3. Example of a depth map yielded by stereo-vision sensors: (a) the Scout robot (top) and its stereo-vision sensor (bottom); (b) an example of a stereo image pair (top) and the respective disparity map (bottom, left) and depth map (bottom, right).

Our robot was used to build a 3-D map of a volume in our lab., covering approximately $12.2\ m^3$ with a resolution $\epsilon = 0.1\ m$. The initial map had entropy $H(\mathcal{C}_l \mid \mathcal{M}_l) = 11.167 \times 10^4\ bits$, wherein every voxels $l \in \mathcal{Y}$ had the maximum entropy (7 $bits$, for $b = 128$). We used the stopping criteria $H_{th} = 3 \times 10^4\ bits$. Fig. 4-a shows the map at three different instant times, showing that the map improved gradually as long as the map's entropy decreased until the final (best) 3-D map (bottom of the figure). The robot needed $t_{k_{max}} = 9289\ s$ to accomplish the mission without any human intervention, and had to travel a distance of about $72\ m$. A significant part of the final map's entropy (44%) is due to unexplored voxels located behind the walls, which the robot would never be able to sense. If we discount this entropy bias, the final map yields an entropy decrease of 84% when compared with the initial map. Fig. 4-b presents the final map from different viewpoints.

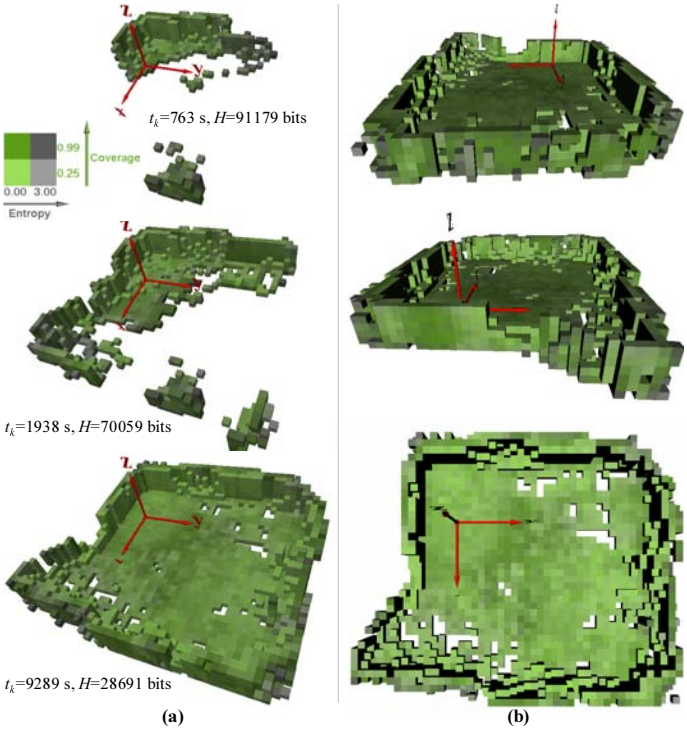

Fig. 4. Snapshots of the 3-D coverage map obtained by the robot: (a) evolution of the map along the mission until the final map with lower entropy (denoted in the figure as H); (b) the final map viewed from three additional viewpoints. The world reference frame $\{W\}$ is represented through the red axes. The scale of the pictures is such that each arrow of $\{W\}$ is equivalent to a real length of $1\ m$. The voxels' edge for the presented maps is $\epsilon = 0.1\ m$.

V. Conclusion

Most of the previous research on building maps has been restricted to models without an explicit representation of the map's uncertainty. This article presented innovative work related with developing a probabilistic model for 3-D mapping, which uses information theory to formally represent uncertainty. We presented a straightforward method to build a 3-D grid map, a sensor model for a range sensor providing distance information, a Bayesian inference procedure to update the coverage belief of each cell, and an entropy gradient-based survey strategy. Experimental results obtained with a real robot and stereo-vision successfully validated the proposed framework. We are currently extending it to perform volumetric mapping with teams of robots (more than one), as a means to drastically reduce the mission time through cooperation and coordination.

References

[1] S. Thrun. Robotic mapping: a survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artif. Intell. in New Millenium*. M. Kaufmann, 2002.

[2] C. Stachniss and W. Burgard. Mapping and exploration with mobile robots using coverage maps. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'2003)*, pages 467–472, 2003.

[3] S. Thrun, D. Hahnel, D. Ferguson, M. Montermelo, R. Riebwel, W. Burgard, C. Baker, Z. Omohundro, S. Thayer, and W. Whittaker. A system for volumetric robotic mapping of underground mines. In *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA'03)*, 2003.

[4] M. Maimone, L. Matthies, J. Osborn, E. Rollins, J. Teza, and S. Thayer. A photo-realistic 3-D mapping system for extreme nuclear environments: Chernobyl. In *Proc. of IEEE/RSJ Int. Workshop on Intelligent Robots and Systems (IROS'98)*, volume 3, pages 1521–1527, 1998.

[5] H. Moravec and A. Elfes. High resolution maps from wide angle sonar. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1985.

[6] D. Pagac, E. Nebot, and H. Durrant-Whyte. An evidential approach to map-building for autonomous vehicles. *IEEE Trans. on Robotics and Automation*, 14(4):623–629, 1998.

[7] M. Martin and H. Moravec. Robot evidence grids. Technical Report CMU-RI-TR-96-06, Robotics Institute, CMU, Pittsburgh, USA, 1996.

[8] F. Bourgault, A. Makarenko, S. Williams, B. Grocholsky, and H. Durrant-Whyte. Information based adaptive robotic exploration. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'02)*, pages 540–545, 2002.

[9] B. Yamauchi. Frontier-based exploration using multiple robots. In *Proc. of 2^{nd} Int. Conf. on Autonomous Agents*, pages 47–53, 1998.

[10] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA'00)*, volume 1, pages 476–481, 2000.

[11] W. Burgard, D. Fox, and S. Thrun. Active mobile robot localization by entropy minimization. In *Proc. of 2^{nd} Euromicro Workshop on Advanced Mobile Robots (EUROBOT'97)*, 1997.

[12] D. Hahnel, W. Burgard, D. Fox, and S. Thrun. An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'03)*, 2003.