Autonomous Mapping of Natural and Unstructured Environments

Maria João Rendas, Stefan Rolfes and Albert Tenas

Abstract—In this paper we assess the problem of using autonomous robots for mapping natural environments. We present new solutions to two related problems: automatic classification of benthic habitats using video images, and robot navigation. An information-based segmentation algorithm is proposed to tackle the first problem. Robot navigation in this type of environments is cast in a new formalism, grounded on the notion of Random Closed Set models. We study in detail the problem of estimating from the segmented images the features relevant for this type of models, and their use for robot navigation.

Index Terms—autonomous robots, habitat mapping, unstructured environments, non-linear filtering, robot guidance.

I. INTRODUCTION

THE paper describes on-going work at I3S on the use of autonomous robots to autonomously map regions of the sea bottom. The work presented has been developed in the context of the European project SUMARE,[1] with the goal of producing maps of the maerl habitat in the Orkney Islands (Scotland) using an underwater robot equipped of vision and sonar sensors. The major goal of these surveys is to evaluate the amount of living and dead maerl in the region, and characterizing its spatial distribution. Precise spatial registration is thus not a major concern for this study.

One of the major challenges presented by this application was the requirement of producing maps of large regions without having to install artificial beacons for localization purposes, which would imply prohibitive costs and work load. In order not no loose the vehicle, i.e., to guarantee that it will return at the end of each survey to a pre-specified recovery area, the robot must iteratively map the regions surveyed, and use this information to navigate inside the operating area. Several approaches have been presented for simultaneous navigation and mapping, which are appropriate for human-made environments, where a dense set of outstanding features can be identified and mapped by the robot [1-4]. However, these methods are not adequate for navigation in natural areas, where, in general, the set of features that can be used for positioning is reduced and sparse, preventing a continuous update of the robot's position. Based on the assumption that natural environments, such as the sea bed, are fundamentally random-like in appearance (which is an equivalent way to say that they have a high complexity, in the sense of Kolmogorov), we formulated a new approach to the navigation and mapping based on local statistical descriptions of the workspace. The formal tool on which our approach is based are Random Closed Set models [5], which have been used in the context of image analysis in biology and physics studies. These models are doubly stochastic processes: a first spatial (point) processes determines the (random) set of points at which realizations of a second (shape) process place randomly shaped objects. They are commonly named "germ-grain models", reflecting this two-step construction process. Our maps are thus, not a "photographic" representation of some individual regions which have outstanding characteristics that make then useful for the (re)positioning of the sensor, but rather a distributed representation of the random structure of each observed region.

More formally, the environment representation that the robot builds and uses for localization is a collection of vector fields and labels defined over a partition of the observed space, which for the sea-bed mapping applications, we identify with a manifold O in $\Re^2 \times \Re^+$, each point being identified by its coordinates in the horizontal plane and its depth (always positive). Let $\{O_i\}_{i=1}^M$ be a partition of O:

$$O = \bigcup_{i=1}^M O_i, O_i \cap O_j = \varnothing, i \neq j$$

These subsets correspond to the homogenous regions of the observed space. Their boundaries code, thus, the contours of the macroscopic objects present in the workspace, as well as the limits of regions occupied by different habitats. Inside each subset, the environment characteristics are coded by a vector field and a label. The label ℓ_i establishes the type of Random Closed Set model (families of the point and shape processes) and the continuous vector field $\boldsymbol{q}_i(p) \in \Theta_i \subset \Re^{d_i}$, $p \in O_i$ defined at each point of each subset specifies the vector of parameters (of dimension d_i depending on the model type) of the Random Closed Set model of which the neighborhood of the point is a typical realization.

Basing the robot navigation on this kind of maps presents several advantages, compared to feature-based methods:

Maria-João Rendas, Stefan Rolfes and Albert Tenas are with the Laboratoire d'Informatique, Signaux et Systèmes de Sophia Antipolis (I3S), affiliated with the Université de Nice Sophia Antipolis and the CNRS (Centre National de la Recherche Scientifique), Sophia Antipolis, France (telephone: +33-4-92942714, e-mail: rendas@ i3s.unice.fr).

- It does not require a dense set of outstanding features. It can thus provide guidance information in workspaces where the other approach would breakdown;
- It is robust with respect to small variations (displacements induced by currents, for instance) of the individual microscopic objects present.
- It implicitly codes the outstanding features of the environment (boundaries of the homogenous regions), which are important for decision-making processes such as path planning.

Several new signal/image processing problems must be solved to make this approach operational:

- Map building: How to identify the partition $\{O_i\}$, i.e., the boundaries of the "homogenous" regions and their labels (i.e., the model type), and how to estimate, inside each region, the vector fields $\boldsymbol{q}_i : O_i \rightarrow \mathfrak{R}^{d_i}$ that locally provide information with respect to the appearance of the environment ?
- Map navigation: How to use such a map for navigation, i.e., how to update the estimate of the robot position given its perceptual data (vision and sonar) at each position ?

For map building, the first problem is a detection/segmentation problem, between models of distinct complexities. Naïve use of Maximum likelihood detectors will lead to over-estimation of the model complexity, and must be corrected using additional penalty terms, such as the Akaike or the MDL criteria. The second problem is a parameter estimation problem that we tackle using a maximum likelihood approach and the fundamental result of Mathéron stating the equivalence between a Random Closed Set model and the set of hitting probabilities that it determines over the set of compact shapes. The work we present is based on segmented video images, and uses an innovative algorithm for image segmentation based on information theory tools, which is appropriate for sea-bed images.

Map navigation is a filtering problem. Since the workspace description is not continuous, use of the common Kalman or Extended Kalman approaches requires a two-step procedure: a first association phase provides indication of which homogenous region the robot is perceiving, and a subsequent step effectively processes the new data using the (continuous) map of the relevant region. This approach is highly sensitive to errors in the association step. We rely on an approximation to the optimal non-linear filter based on Gaussian mixtures that we proposed previously for terrestrial applications [6], which does not suffer from this problem, and can accommodate non-linear dynamics (which is the case for underwater platforms) and non-Gaussian observations (which is the case for small observation windows, e.g., when the robot is close to the sea bottom).

In the paper, we present in detail our solutions to the above problems, illustrated by simulations that demonstrate the validity of our approach to the navigation of robots in natural environments and the effectiveness of our algorithms for map building and robot positioning.

We point out that, for the application under study, our map provides more than guidance information, enabling the determination of the ultimate information being sought: an indication of the spatial distribution of a given species (maerl) and its total amount (directly derived from the parameters of the random closed set models inside each region). For this purpose, a (supervised) classification must be done, indicating which random closed set models indeed correspond to maerl.

II. Data processing

In this section we present the pre-processing stage of the mapping and navigation system, which, from the raw image and proprioceptive robot data produces the input data to navigation filtering and map building. The section is organized into two parts. The first is dedicated to the image segmentation problem, which produces segmented versions of the acquired images. For the application under consideration (mapping of a given habitat) a supervised approach is required, since we must assign a label to each identified homogenous region of the sea-bed. The second assesses the determination of the information relevant for update of (and matching with) the statistical maps.

A. Image segmentation

The image segmentation algorithm that we use in this study is a combination of supervised and unsupervised techniques, which help us gaining robustness against the large variability that can be expected in images of an natural sea bed, produced under varying lighting conditions, see figure below. A first step of unsupervised segmentation, based on the comparison of local estimates of the intensity distribution produces an initial segmentation of the image into its large macroscopic regions. A second step assigns labels to these areas, using as a staring point the result of the unsupervised segmentation. This second step is supervised, using a data base of possible intensity distributions for the classes that may appear in the region of study. This data base has been learned by manual labeling of a representative set of unsupervised segmented images, in collaboration with members of the International Centre for Island Technology, of the Heriot Watt University, Scotland.

The algorithms used in each step are briefly presented below.

1) Unsupervised images segmentation
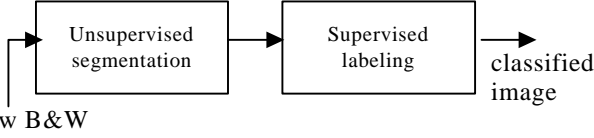
Initial image segmentation is obtained by using a novel

Fig. 1 Image classification.

algorithm that partitions the raw image into regions of *homogeneous* local distributions of the intensity level of the pixels. It is an unsupervised algorithm that automatically adjusts to the complexity of the observed scene. The algorithm is based on the analysis of the probability distribution of image intensity (gray level) over small neighborhoods, and uses formal decision theory tools to iteratively learn the distributions of the classes present in the image.

It is grounded on fundamental results from type theory [refs type theory], which is a branch of statistics that studies repeated realizations of a basic random variable, and on the Minimum Description Length, which establishes a consistent approach to model selection under varying model complexity. We present here the major result on which our algorithm is based, referring the interested reader to [Albert Oceans] where the algorithm is fully motivated.

We first introduce some nomenclature and notation. Let X be a discrete random variable (*rv*) with probability space (Ω, A, P) where $\Omega = \{a_1, a_2, \cdots, a_L\}$ is the (finite discrete) realization space, A is a sigma-field of subsets of Ω and P is a probability measure. We denote by lower-case letters x the realizations of X. Consider a sequence

$$x^{(n)} = \{x_1, x_2, \cdots, x_n\} \in \Omega^n$$ of n independent realizations

of *X*. The **type** of $x^{(n)}$, which we denote by $\mathbf{n}_{x^{(n)}} : \Omega \mapsto [0,1]$ is the empirical estimate of the probability distribution (*pd*) of *X*, and is given by:

$$v_{x^{(n)}}(a_j) = \frac{1}{n} \sum_{i=1}^{n} 1_{a_j}(x_i), \quad j = 1, \ldots, L$$

where $1_{a_j}(x_i) = \begin{cases} 1, & x_i = a_j \\ 0, & x_i \neq a_j \end{cases}$

Our unsupervised algorithm is based on the following result:

Lemma

Consider that we are given two sequences of length *n*

$$x_1^{(n)} = \left(x_{1_1}, \cdots, x_{1_n}\right)$$

$$x_2^{(n)} = \left(x_{2_1}, \cdots, x_{2_n}\right)$$

of *iid* discrete *rv*'s taking values in alphabet $\Omega = \{a_1, \cdots, a_L\}$

of size $|\Omega| = L$. The MDL (Minimum Description Length, see [1] for a deep discussion of this approach to model selection) test for choosing between the two hypothesis

$$H_0 : x_1^{(n)} \propto p_{\mathbf{m}}^{\ n} \quad x_2^{(n)} \propto p_{\mathbf{m}}^{\ n}$$

$$H_1 : x_1^{(n)} \propto p_{\mathbf{m}_1}^{\ n} \quad x_2^{(n)} \propto p_{\mathbf{m}_2}^{\ n}, \quad \mathbf{m}_1 \neq \mathbf{m}_2$$

where the probability distributions $\mathbf{m}, \mathbf{m}_1$ and $\mathbf{m}_2$ are unknown, is given by

$$\mathbf{g} \equiv \frac{(L-1)}{L} (2\log(n+1) - \log(2n+1)) \overset{H_0}{\underset{H_1}{\gtrless}} D(\mathbf{n}_1 \| \hat{\mathbf{m}}) + D(\mathbf{n}_2 \| \hat{\mathbf{m}}) \equiv T$$

where v_1, v_2 are the types of the sequences $x_1^{(n)}, x_2^{(n)}$, respectively, and

$$\hat{\mathbf{m}} = \frac{1}{2}\left(\mathbf{n}_1 + \mathbf{n}_2\right)$$

is the empirical estimate of the distribution law under H_1, which coincides with the balanced mixture of the two types. I(n the previous expression, $D(\mathbf{n} \| \mathbf{m})$ is the Kullback-Liebler directed divergence (also called relative entropy) between ν and μ:

$$D(\mathbf{n} \| \mathbf{m}) = \sum_{j=1}^{L} v(a_j) \ln \frac{\mathbf{n}(a_j)}{\mathbf{m}(a_j)}.$$

The proof of this result can be found in [3]. It shows that the Kulback-Liebler divergence is the relevant metric to decide whether two sequences are realizations of the same *rv* or of distinct ones. It also indicates that we must compare the two individual types to their mixture, and not directly to each other, as intuition could suggest. The decision test is always well defined, since the original types are necessarily absolutely continuous with respect to their mixture, even if they may not be mutually absolutely continuous.

The previous Lemma indicates which statistic T we should use to decide whether two small neighborhoods of the image have (or have not) the same intensity distribution. It also indicates how the threshold of the test ($\mathbf{g}$) varies with the size of the sample sequence *n*. Our algorithm is an anisotropic diffusion algorithm, which iteratively learns the underlying distribution of the intensity for each region. The algorithm works over a square grid of fixed size defined over the image, each point of the image describing the (intensity) distribution associated to the corresponding neighborhood. In the first step, each node is initialized with the type of the pixels of the corresponding image region. At each subsequent step, the distribution at each node *i,j* is updated as a weighted mixture of the distributions $\mathbf{n}_{p,q}^k$ of the nodes in its neighborhood $V(i, j)$:

$$\mathbf{n}_{i,j}^{k+1} = \sum_{p,q \in V(i,j)} \mathbf{a}_{k+1}(p, q \mid i, j) \mathbf{n}_{p,q}^k, \quad k = 0,1,2,\ldots$$

$$\sum_{p,q \in V(i,j)} \mathbf{a}_{k+1}(p, q \mid i, j) = 1$$

where the weights $a_k(p,q\,|\,i,j)$ depend on the test statistic.

$$T^k(p,q\,|\,i,j) = D\left(n_{p,q}^0\,\|\,\hat{m}\right) + D\left(n_{i,j}^k\,\|\,\hat{m}\right)$$

where

$$\hat{m} = \frac{1}{2}\left(n_{p,q}^0 + n_{i,j}^k\right)$$

that is optimal to decide whether the image window corresponding to node *i,j* follows the statistical model associated to node *p,q*. The weights must be close to one if the nodes (i,j) and (p,q) correspond to the same distribution, and close to zero otherwise. If we assume that the size of the sequences (n) is large, $n \gg 1$, the threshold g can be approximated by

$$g \cong \frac{(L-1)}{L}\log\frac{n}{2} \cong \log\frac{n}{2},$$

where the last approximations assumes that *L>>1* also. In this case, our test is equivalent to

$$c \underset{\substack{< \\ H_1}}{\overset{\substack{H_0 \\ >}}{}} \frac{e^T}{n}$$

where *c*=1/2. This shows that the relevant statistic depends on the size of the observed sequence: as *n* grows, a larger value of the distance are tolerated for deciding that the sequences come from the same statistical model. For this reason, the weights $a_k(p,q\,|\,i,j)$ increase with iteration number:

$$a_k(p,q\,|\,i,j) = e^{-b_k T}, \quad b_k \le b_{k-1}.$$

The algorithm gradually diffuses the local histograms, converging to a grid $n_{i,j}^\infty$ partitioned in homogeneous regions inside which the types are the same.

We point out that, as said before, our algorithm differentiates the different image regions by comparing local estimates of the intensity level. Thus, it will not be able to distinguish regions that may have distinct texture (granularity) if they result in the same gray level distributions. Although this possibility exists, the extensive tests that we conducted on real underwater images show that the algorithm is able to separate the different sea-bed habitats present in all practical cases found until now.

2) Classification

Association of a label to the homogenous regions of the solution $n_{i,j}^\infty$ of the segmentation algorithm uses a data base of histograms that has been built by manual classification of segmented images. This data base contains, for each class *n* (in our case five major classes have been learned: dead maerl, living maerl, macroalgae, algae balls and sand) a set of K_n examples of histograms that have been obtained using the segmentation algorithm $\{v_m^n\}_{m=1,\ldots,K_n}$. Association of each type $v_{i,j}^\infty$ to a class *n* is made by choosing the class that minimizes the distance to the classes' representatives: $D_n^*\left(n_{i,j}\right) = \min_m D\left(n_{i,j}^\infty\,\|\,n_m^n\right)$.

We present below one example of a classified image, where four classes are present: macroalgae, living maerl, dead maerl and sand. Fig. 2 show the original image and Fig. 3 the result of classification.
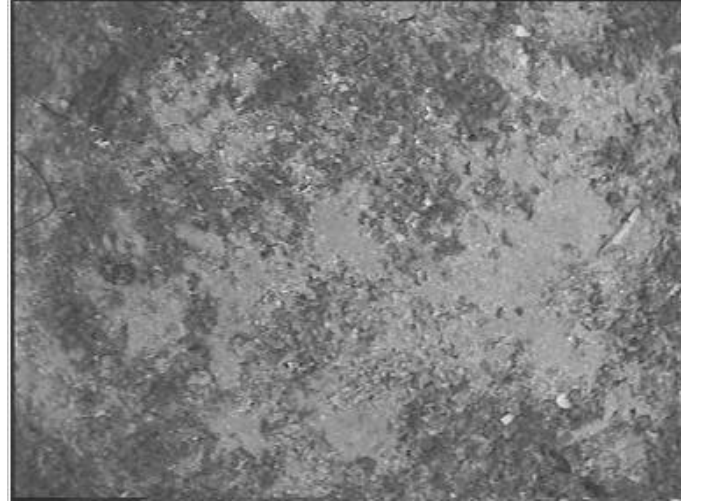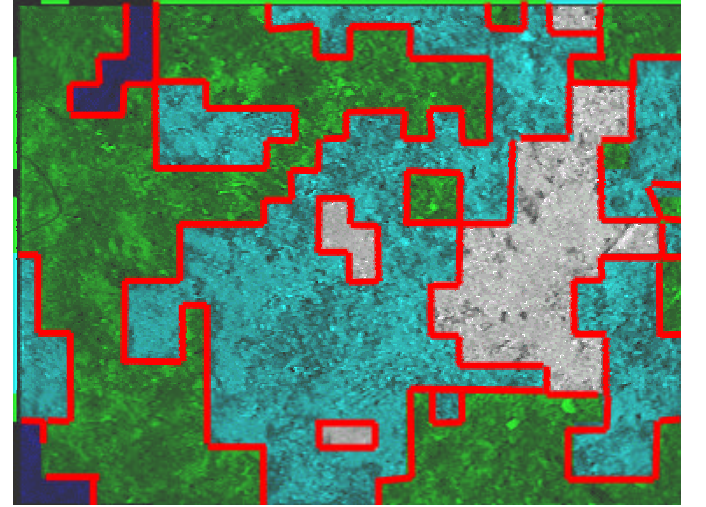

Fig. 2 Original image.

Fig. 3 Classified image.

The darker region corresponds to macroalgae, the dark gray to living maerl, light gray to dead maerl and the lightest region is sand.

B. Local statistics of spatial distribution and shape

1) Random Closed Set Modeling

Without loss of generality we can formally describe an environment by associating to each point $p \in \Re^2$ a mark $m \in M$, describing the type of object (in our case; maerl, sand, alga,etc).,at that location (*M* is the markspace). The determination of the mark associated to each point is a result of the image segmentation algorithm previously described together with knowledge of the robot state. The workspace

can thus be represented as a collection of compact sets, to which a mark is associated (all points inside the compact set belong to the same type of object):

$$\Xi=\bigcup_{i=1}^{\infty}(\Xi_i+p_i,m_i); \Xi_i\in\mathbf{K};m_i\in\mathbf{M},$$

where, $\mathbf{K}$ is the family of compact sets, and the sum Ξ_i+p_i denotes the compact set Ξ_i translated by the vector p_i . Note that this formalism assumes that one of the classes plays the role of background, against which the objects are placed. The Ξ_i describe thus the morphological characteristics of the individual objects and the vector of locations their spatial distribution. We can consider that the patches form a random pattern and that the environment is a realization of a random process. If we consider a large number of local observations (segmented images) we remark that there is some regularity in the occurrence of the random field, meaning that some 'occurrences' are more likely than other, restricting e.g. the size and the form of the individual compact sets to a strict subset of the family of compact sets $\mathbf{K}$.

Random Closed Set (RCS) models are a mathematical model that has been widely used in order to analyze random patterns We recall that a RCS is a doubly stochastic process. A first process (point process) determines the locations of the objects and a second process (shape process) the morphological characteristics of the objects placed at each location.
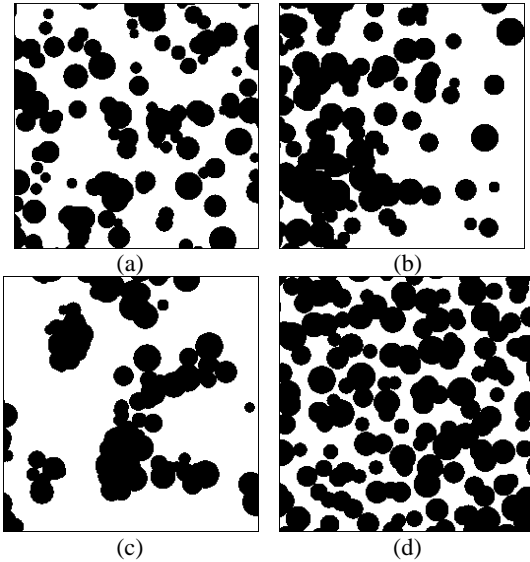

(a) (b)
(c) (d)

Fig. 4 Example of RCS models: (a) isotropic boolean model, (b) anisotropic boolean model, (c) clustered distribution and (d) regular distribution of the grains.

We consider that these two processes are independent, which is not always the case for natural environments. Under the assumption of independency we can construct a family of RCS model types $\ell=\{\ell_i\}_{i=1}^N$ as a set of pairs of one point process model and one shape process model. Examples of point processes are: Poisson point process, regular

distribution or clustered distribution (see Fig. 4). A particular model is obtained by a model parameter $\mathbf{q}_i=(\mathbf{l}_i,\mathbf{g}_i)$, where $\mathbf{l}_i$ determines the point process and $\mathbf{g}_i$ the shape process, means the measure defined on the shape space. (In the sequel we denote by $\mathbf{q}_i$ the RCS model type and the particular model parameter). The shape process is chosen in order to restrict the possible shapes to simple basic shapes, such as line-segments of random length and orientation or compact discs of random radius. This is an approximation of generally very complex shaped objects, which are impossible to model by low dimensional parameter vectors.

Direct estimation of the spatial distribution (count measure) and the morphological characteristics is impossible. However, it is known, [7], that knowledge of the RCS model is equivalent to knowledge of the hitting capacities of the random field for all compact sets. The hitting capacity is defined as the probability that the intersection of the random field with a compact set $K\in\mathbf{K}$ is non-empty:

$$T_{\Xi}(K)=P(\Xi\cap K\neq\varnothing).$$

2) Estimation of model parameters

While we are not able to directly estimate the model parameter, we can obtain an estimate of the hitting capacities from the segmented images, under the assumption that the random field is locally isotropic which implies that $T_{\Xi}(K)=T_{\Xi}(K+p)$. The equivalence between the hitting capacities and the random closed set model mentioned above can be exploited to estimate the model parameters. It is thus more convenient to write $T_{\mathbf{q}_i}(K):=T_{\Xi}(K)$, knowing that the model type is ℓ_i . For boolean models (the point process is a Poisson point process and the grains are i.i.d. in the workspace) the hitting capacity can be written as a function of the compact set K shape and the model parameter $\mathbf{q}_i$ [5]:

$$T_q(K)=1-\exp(-\mathbf{l}E\{\mathbf{n}(\Xi_0\oplus\breve{K})\})$$

where $\oplus$ is the Minkowski-addition $A\oplus B=\{a+b,\forall a\in A,b\in B\}$, $E(\cdot)$ is the statistical expectation operator with respect to the measure of the shape process, $\breve{K}=\{-x,x\in K\}$, $\mathbf{n}(.)$ is the Lebesgue-measure and Ξ_0 is a random shape. The basic requirement for our approach to the modelization of natural scenes and to mobile robot navigation is the ability to determine the hitting capacities as a function of the RCS model type and parameters. For the moment we are restricted to the consideration of boolean models. In order to identify more complex models, such as regular or clustered distributions of the objects, additional efforts need to be spent.

Of course we cannot observe the hitting capacities for all compact sets, but just for a limited number of compact sets $K^n=\{K_1,\ldots,K_n\}$, which we call the structuring elements, since the whole information of the structure of the random field will be captured by the hitting capacities for these sets. The aim of the theory of random closed sets is to estimate the

model type and the model parameter such that the observed scene is a typical realization of the RCS.

3) The likelihood function

Estimates $\hat{T}(K_j)$ of the hitting capacities can be obtained by placing each structuring element in K^n at N (sampling number) positions $\{p_i\}_{i=1}^N$ inside the observation window, each time evaluating the event: K_j *hits* (or not) *the random field* Ξ:

$$\hat{T}(K_j) = \frac{k_j}{N}.$$

In order to estimate the model parameter, assuming the model type to be known, we must find the likelihood of observing a given set of hitting capacities, given the model parameter: $p(\hat{T}(K^n)|\boldsymbol{q}_i)$. The model parameter $\boldsymbol{q}_i$ can then be obtained as the maximum likelihood estimate:

$$\hat{\boldsymbol{q}}_i : \left.\frac{\partial \ln(p(\hat{T}(K^n)|\boldsymbol{q}_i))}{\partial \boldsymbol{q}_i}\right|_{\boldsymbol{q}_i=\hat{\boldsymbol{q}}_i} = 0.$$

(1)

I the N events leading to the determination of the sample estimates $\hat{T}(K_j)$ are mutually independent, the probability of a given number of hits k_j (for a given structuring element K_j) follows a binomial distribution:

$$p(k_j|N) = \binom{N}{k_j} T(K_j)^{k_j}(1-T(K_j))^{N-k_j},$$

where $T(K_j)$ is the hitting capacity, dependent on the model parameter. The variance of $p(k_j|N)$ is $\boldsymbol{s}_{k_j}^2 = NT(K_j)(1-T(K_j))$. In order to guarantee that the individual hitting events are mutually independent it is important to choose an appropriate sampling number N and the positions at which the events (hit or not hit) are evaluated. This number depends on the size of the observation window, on the structuring element and on the RCS model. If the samples are taken on a (appropriately chosen) regular grid in the observation window we can for boolean models identify a lower bound to their optimal number. The worst consequence of small sampling numbers (below the optimum number) is that we do not exploit all the information that we can retrieve via the hitting capacities. We can thus characterize the likelihood of a hitting capacity (or equivalently the number of hits k_j given the model parameter as

$$p(k_j|\boldsymbol{q}_i,N) = \binom{N}{k_j} T_{\boldsymbol{q}_{ij}}(K_j)^{k_j}(1-T_{\boldsymbol{q}_i}(K_j))^{N-k_j}.$$

Under the assumption that the estimates for all hitting capacities are independent we obtain the joint likelihood as

$$p(k^n|\boldsymbol{q}_i) = p(k_1|\boldsymbol{q}_i)\ldots p(k_n|\boldsymbol{q}_i),$$

omitting the dependency on the sampling number N. The likelihood function $p(\hat{T}(K^n)|\boldsymbol{q}_i)$ is obtained from this density by a simple scaling/normalizing operation:

$$p(T(K^n)|\boldsymbol{q}_i) = \prod_j p\left(\frac{k_j}{N}|\boldsymbol{q}_i\right).$$

4) Choice of structuring elements

An important issue concerning the estimation of the model parameters is the choice of the set K^n. The problem is to determine *(i)* how many structuring elements should be used and *(ii)* their shape. We assume now that the maximum likelihood estimate given by equation (1) is unbiased. In this case a lower bound of the covariance of the estimate is given by the inverse of the Fisher information matrix:

$$J_{k^n} = -E\left\{\nabla_{\boldsymbol{q}}\left(\nabla_{\boldsymbol{q}}\left(\ln(p(k^n|\boldsymbol{q}))\right)\right)^T\right\}.$$

We searched for the structuring elements that maximize the determinant of J_{k^n}. For a simple boolean model, with $\boldsymbol{q}=(\boldsymbol{l},r)$, Poisson point process of intensity $\boldsymbol{l}$, and whose objects are compact discs of radius r, and restricting the analysis to structuring elements that are squares of varying side d, the hitting capacities can be easily shown to be given by

$$T_{\boldsymbol{q}}(K_j) = 1-\exp(-\boldsymbol{l}(d_j^2+4d_j r+\boldsymbol{p}r^2))$$

And the Fisher information matrix is obtained from using this expression on the binomial distribution found before, and computing

$$J_{k^n} = -E\left\{\begin{bmatrix}\frac{\partial^2 \ln(p(k^n|\boldsymbol{q},N))}{\partial^2\boldsymbol{l}} & \frac{\partial^2 \ln(p(k^n|\boldsymbol{q},N))}{\partial\boldsymbol{l}\partial r} \\ \frac{\partial^2 \ln(p(k^n|\boldsymbol{q},N))}{\partial\boldsymbol{l}\partial r} & \frac{\partial^2 \ln(p(k^n|\boldsymbol{q},N))}{\partial^2 r}\end{bmatrix}\right\} = J_{k_1}+\ldots+J_{k_n}$$

It is trivial to show that a single structuring element is not sufficient in order to estimate this RCS model, since it leads to a singular Fisher information matrix.

We assume that the first structuring element k_1 is a single point, yielding

$$\left|J_{k_1,k_2}\right| = c_1 a_1^2 c_2 b_2^2 + c_1 b_1^2 c_2 a_2^2 - 2c_1 a_1 b_1 c_2 a_2 b_2,$$

where

$$c_i = \frac{N\exp(-\boldsymbol{l}a_i)}{1-\exp(-\boldsymbol{l}a_i)}; a_i = d_i^2+4d_i r+\boldsymbol{p}r^2; b_i = 4d_i\boldsymbol{l}+2\boldsymbol{p}\boldsymbol{l}r.$$

The other structuring element k_2 is a square of side d_2^* which is the solution of:

$$\left. \frac{\partial \left\| J_{k_1 k_2} \right\|}{\partial d_2} \right|_{d_2 = d_2^*} = 0 \, .$$

It is easy to show that an unique positive solution $d_2^* > 0$ exists, which can be easily found numerically.

We addressed next the following question: What is the information-gain if we add a third structuring element of side d_3? The Fisher information matrix is now:

$$J_{k^3} = J_{k^2} + J_{k_3} = G_{k^2}(I + G_{k^2}^{-1} J_{k_3} G_{k^2}^{-1}) G_{k^2} \, ,$$

where G_{k^2} is a symmetric square-root of J_{k^2}. The determinant of J_{k^3} is

$$\left| J_{k^3} \right| = \left| J_{k^2} \right| \left| I + G_{k^2}^{-1} J_{k_3} G_{k^2}^{-1} \right|$$

$$= \left| J_{k^2} \right| (1 + \left\| G_{k^2}^{-1} \sqrt{c_3} \begin{bmatrix} a_3 \\ b_3 \end{bmatrix} \right\|^2) \, .$$

In the above equation, $G_{k^2}^{-1} J_{k_3} G_{k^2}^{-1}$ is the information gain provided by the third structuring element. Study of this gain as a function of the d_3 indicates that the optimal third structuring element always coincides with either $K_1 = sq(d_1 = 0)$ or $K_2 = sq(d_2^*)$, depending on which is of the two corresponding hitting probabilities is affected by a larger uncertainty, see Fig. 5 showing a plot of the gain for a specific Boolean model. This study indicates that for the type of Boolean models two distinct structuring elements provide all required information, that grows with the number of samples that are used to estimate the corresponding hitting capacities. For this reason, in all numerical studies presented in subsequent sections of this paper, we use $n = 2$.
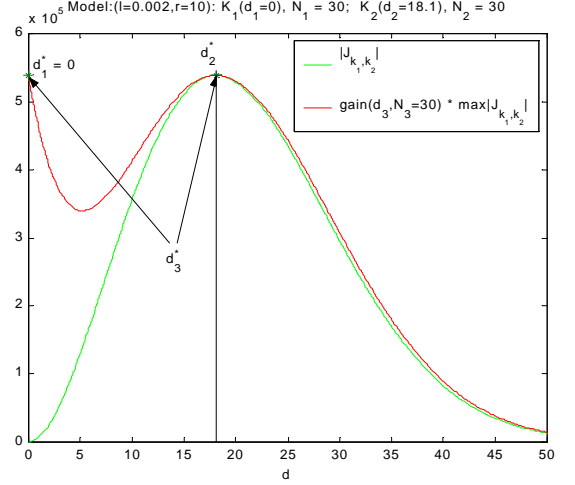
Fig. 5 Determinant of Fisher matrix for n=2 and the information gain of an additional third shape, showing that its optimal size is either $d_1 = 0$ or d_2^*.

III. MAP NAVIGATION

Assume that the Random Closed Set map, i.e., the partition of the workspace, and the associated model type is known. We address now the problem of using this map in an efficient way to estimate the robot's position. If the RCS model is anisotropic inside a given region O_i, meaning that the model parameter is a smooth function of the point considered, observation of the environment characteristics provides useful information with respect to the robot localization at all points inside O_i. If the map contains isotropic regions the localization is possible only when the robot reaches the boundary of adjacent areas, corresponding to an abrupt change of either the model parameter or the RCS model type. Inside each isotropic area perceptual information cannot be used for localization, and the robot must rely entirely on its odometry in order to estimate its pose. The uncertainty affecting its position estimates when it reaches a region boundary may be very large, resulting in a large set of possible true locations along the boundary (large ambiguity). Most approaches to position estimation for mobile robots use Extended Kalman filters, and must thus assume that the observations are differentiable with respect to the robot's state. Navigation between adjacent areas requires in this case a first symbolic association step, prior to actual observations filtering, in order to determine inside which region the robot is located. We propose a method that does not rely on this two step decomposition.

We first formulate the general framework of the Bayesian approach to localization. Assume that the dynamic model of the robot's state X_k and the observation model are known:

$$X_{k+1} = f(X_k, u_k) + w_k \, , \qquad (2)$$
$$Y_k = h(X_k, \Xi) + v_k \, ,$$

(3)

where $f(.,.)$ and $h(\cdot,\cdot)$ are known (eventually non-linear) functions, $f:C\times U\rightarrow C$, $h:C\times\{\Xi\}\rightarrow E$, where C is the configuration space of the robot's state, $\dot{U}$ is the space of the control input, u_k, $\{\Xi\}$ is the space of possible environment realizations, E is the space of the perceptual observations and w_{k-1},$\mathbf{n}_k$ are uncorrelated, zero-mean white noises. The optimal MMSE estimate of the robot's state given the past observations $Y^k=\{Y_1,\dots,Y_k\}$, is given by the conditional mean

$$\hat{X}_k=\int_C X_k\, p(X_k|Y^k)dX_k,$$

where $p(X_k|Y^k)$ is the posterior density, which can be recursively updated by alternating prediction and filtering steps:

$$p(X_{k-1}|Y^{k-1})\underset{\text{Pred.}}{\rightarrow}p(X_k|Y^{k-1})\underset{\text{Filt.}}{\rightarrow}p(X_k|Y^k).$$
(4)

The prediction step (convolution) propagates the probability distribution in the state space according to the dynamic model. If Y_k is the output of a memoryless observer, the filtering step (pointwise multiplication) computes

$$p(X_k|Y^k)\propto p(X_k|Y^{k-1})p(Y_k|X_k).$$

The observations $Y_k=(D_k,Z_k)$ contain proprioceptive observations D_k (velocity, heading,…) and measures Z_k, obtained using perceptual sensors (vision, sonar,…). The measures Z_k are in our case estimates of the hitting capacities for a set of structuring elements K^n. These estimates are obtained directly from the classified images: $Z_k=\hat{T}_k=\hat{T}_k(K_1),\dots,\hat{T}_k(K_n)\}$. If we assume that, given the robot's state, the proprioceptive and the perceptual observations are uncorrelated we obtain:

$$p(Y_k|X_k)=p(D_k|X_k)p(\hat{T}_k|\mathbf{q}(X_k)),$$

since the observations Z_k depend on X_k only through the parameters of the RCS model at that point. The likelihood of the uncertainty of the perceptual observations $p(T_k|\mathbf{q}(X_k))$ was found in the previous section to be binomial distributed. If the sampling number N for the estimation of the hitting capacities is large enough the binomial distribution is well approximated by a Gaussian (see Fig. 6):

$$p(\hat{T}|\mathbf{q}(X_k))\cong N(T_q,T_q(1-T_q)/N),$$

where $N(x,A)$ is the normal law with mean x and covariance A. This allows us to transfer the randomness due to the environment into an equivalent additive Gaussian "observation noise."
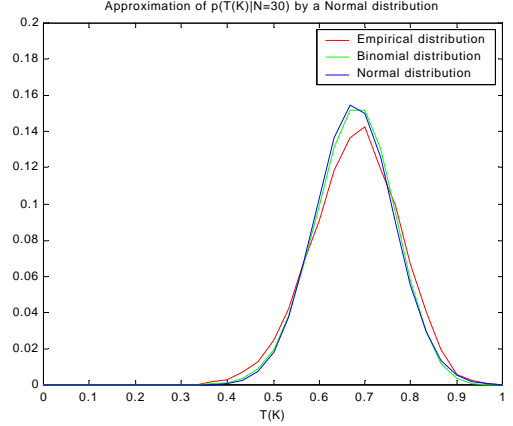

Fig. 6 **Binomial distribution of the number of hits and its approximation by a normal distribution.**

A. IMPLEMENTATION OF THE OPTIMAL FILTER

Direct computation of the prediction (convolution) and filtering (pointwise multiplication) steps is in practice not feasible. We saw in the previous section that the observation noise (for locally isotropic models) is well approximated by a normal distribution. Under the assumption that the noises w_k,v_k of equations (2) and (3) are zero mean and Gaussian with covariance Q and R, respectively, and linearising the state and observation model around the current estimate of the robot's state an approximation of the optimal non-linear filter is given by the standard Extended Kalman Filter (EKF). The linearised state and observation models are given by:

$$X_k\cong f(X_{k-1|k-1},u_{k-1})+\dot{F}_k(X_{k-1}-X_{k-1|k-1})+w_{k-1},$$

$$Y_k\cong h(X_{k|k-1})+\dot{H}_k(X_k-X_{k|k-1})+v_k,$$

where $\dot{F}_k=\left.\dfrac{\partial f(.,.)}{\partial X}\right|_{X_{k-1|k-1},u_{k-1}}$ and $\dot{H}_k=\left.\dfrac{\partial h}{\partial X}\right|_{X_{x|k-1}}$.

This approximation is based on the assumption that at each step the estimation error is small. To prevent filter divergence, the linearisation must be a good approximation over the entire uncertainty domain, which is the case when the robot is moving in informative anisotropic areas. However, navigation in isotropic areas leads to a considerable growth of the uncertainty of the robot's state and linearisation of the non-linear model may artificially shrink the estimated uncertainty of the position estimates, creating the possibility that the EKF diverges. As we mentioned before, this is particularly important when the robot reaches the boundary of adjacent areas after having crossed an isotropic area where consideration of the linearised model leads to a null gain for perceptual information.

In [7] we proposed an approximation of the optimal non-linear filter by a Gaussian Mixture Model (GMM). We assume that the posterior density of the robot's state at time $k-1$ is Gaussian. We approximate the prediction density of equation (4) by a Gauss mixture:

$$p(X_k|Y^{k-1}) \cong \sum_{i=1}^{N_k} s_{k-1}^i p(X_k|Y^{k-1}), \quad \sum_{i=1}^{N_k} s_{k-1}^i = 1 ,$$

where the prediction density for each terms is normal distributed with mean $x_{k|k-1}^i$ and covariance $\Sigma_{k|k-1}^i$. Each term is multiplied by a scaling parameter s_{k-1}^i.

We assume that the number N_k and the mean and variance of the terms are chosen such that linearisation of the state space model around each term is valid inside the principal support of each component. Application of the filtering equation results then in a Gaussian mixture with the conditional mean

$$X_{k|k} = \sum_{i=1}^{N_k} s_k^i X_{k|k}^i .$$

The updated scaling parameter after filtering, s_k^i, are given by:

$$s_k^i = s_{k-1}^i \frac{C_k^i}{\sum_{l=1}^{N_k} s_{k-1}^l C_k^l} ; \quad C_k^i = \frac{\left|\Sigma_{k|k}^i\right|}{|R|\left|\Sigma_{k|k-1}^i\right|} \exp\left(-\frac{1}{2} z_k^{iT} S_k^{i-1} z_k^i\right),$$

where $z_k^i = Y_k - h(X_{k|k-1}^i)$ are the innovations of each term and

$$S_k^i = \dot{H}_k^{iT} \sum_{k|k-1}^i \dot{H}_k^i + R$$

their correlation matrices. The principal result is that each component can be propagated by an EKF, and repeated application of the prediction and filtering step results always in a Gaussian mixture. The scaling parameter depends strongly on the innovations and as a consequence, terms for which the predicted observations (hitting capacities) correspond well to the true observations (small innovations) are reinforced, otherwise they loose importance and will not contribute significantly to the posterior density. When crossing a boundary between adjacent areas, the terms on the 'correct' side are reinforced while all other loose importance, resulting in a concentration of the density mass around the true location of the vehicle.

In the current implementation of the filter we trigger the mixture model when the linearisation is no longer valid over the entire domain of uncertainty. For navigation between isotropic areas this is the case when a boundary lies inside the significant support of the uncertainty domain. In order to reduce the computational complexity of the filter (depending linearly on the number of terms), we eliminate spurious terms (the scaling parameter falls below a given threshold) and fuse close terms. If all but one term are eliminated we return to the simple EKF.

IV. OPTIMAL OBSERVATION STRATEGY

An important issue on autonomous navigation is to choose a trajectory that reduces the uncertainty of the robot at its destination [11]. This corresponds to the problem of defining a sequence of control inputs $U_k^{k+M} = \{u_k, \ldots, u_{k+M}\}$ that moves the robot through M intermediate positions between the current position and the destination position, such that the uncertainty at the destination is minimized. Such a path can be obtained by a recursive $M-$ step predictor with the constraint that the final position must be the destination. In [12],[13] path planning is performed by determining a set of regions (dependent on the informativeness of the map) forming a graph where the arc indicate the transitions between those areas. In this case path planning reduces to a graph search, whose complexity depends on the number of identified positions. Here we limit the problem to a $1-$ step ahead predictor that drives the robot to the next position where the information gain is maximum.

It is known [10] that the estimate of the EKF, $X_{k|k}$, (assuming the validity of the linearisation) minimizes the mean square error for a given control input u_{k-1}:

$$J_k(u_{k-1}) = \int_C (X_{k|k} - X_k)^T (X_{k|k} - X_k) p(X_k|Y^k, u_{k-1}) dX_k .$$

We want to find the control input u_{k-1}^* that minimizes the above function:

$$u_{k-1}^* = \arg \min_{u_{k-1} \in U} (J_k(u_{k-1})) .$$

For the Gaussian mixture,

$$J_k(u_{k-1}) = \sum_{i=1}^{N_k} s_k^i \int_C \left\| X_{k|k} - X_k^i \right\|^2 p_i(X_k^i|Y^k, u_{k-1}) dX_k^i$$

$$= \sum_{i=1}^{N_k} s_k^i \left\| b_k^i \right\|^2 + \sum_{i=1}^{N_k} s_k^i \int_C \left\| X_{k|k}^i - X_{k|k} \right\|^2 p_i(X_k^i|Y^k, u_{k-1}) dX_k^i ,$$

where $b_k^i = X_{k|k}^i - X_{k|k}$.

The first term of the previous equation reflects the error due to the distribution of the terms and the second the uncertainty of each term, given by covariance matrix $\Sigma_{k|k}^i$.

In our implementation we defined a finite set of control inputs that guide the robot to a dense set of positions in the region ahead of it. For each control input we evaluate the mean square error and choose the control input that leads to the minimum error. The computation of the mean square error depends strongly on the innovations, and thus on the perceptual observations (not know), via the scaling parameters and the estimates of each term. For each mixture term, we predict the hitting capacities assuming that the robot's nominal dynamic model is perfect, and that the observation are equal to their expected value given the predicted position.

V. SIMULATION RESULTS

In [9] we demonstrated navigation of a mobile robot inside anisotropic areas. The approximation of the optimal filter by an EKF was valid, since the uncertainty of the robot was

maintained small during the entire trajectory. The linearisation around the estimated state was thus valid over the principal uncertainty support, enabling application of the EKF.

We concentrate here on navigation between isotropic areas. As was said before localization can in this case be performed when the robot reaches the boundary of adjacent areas, indicated by a jump of the RCS model (either by a change of the model type or by a jump of the model parameter). Since permanent location is not possible during the crossing of an isotropic area the accumulated error of the position estimate is quite large. We simulated the navigation of an underwater robot equipped with proprioceptive sensors (compass and speed sensor), used for dead-reckoning, and with a camera pointing at the sea bottom. We assume that the robot moves at a constant altitude above the sea bed. Scaling of the images is thus not necessary in order to preserve the metric of the observations. We simulated a workspace that was divided into two areas $O = \bigcup_{i=1}^{2} O_i$. Inside each area, we have chosen the RCS model to be an **isotropic boolean model**, where the grains are compact discs of constant radius. The model parameter is thus $q_i = (I_i, r_i)$. The realized workspace (just the locations of the grains are indicated) is shown in Fig. 8. The initial EKF pose estimate is indicated by the big cross along with the uncertainty indicated by the ellipse (blue). The true robot location is centered at the square, indicating the area that is observed by the camera. Some typical images obtained in area O_1, O_2 and at the boundary are illustrated in Fig. 7.
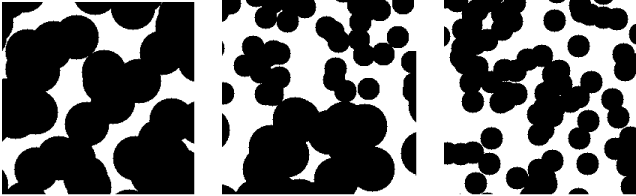

Fig. 7 Typical images obtained (a) inside area 1, (b) on the boundary, (c) inside area 2.

While the RCS model is isotropic inside the areas this is not the case at the boundary. We said that the estimation of the hitting capacities is based on the assumption that the model is at least locally isotropic. In order to guarantee that the observation noise is unbiased (which is due to the anisotropy not the case at the boundary) we predict the hitting capacity by integrating the RCS model all over the area observed by the camera.
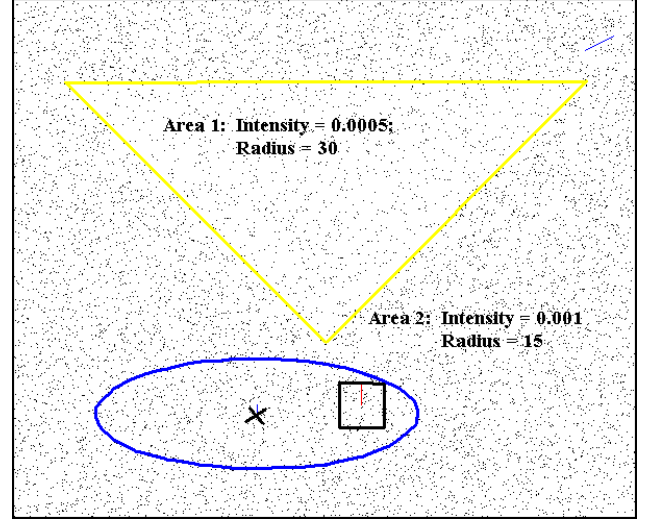

Fig. 8 Realization of a random field. The workspace is segmented in two areas, with distinct RCS model parameters. The initial position of the robot is indicated by a large dot.

A simulated ocean current disturbs the nominal trajectory of the robot, resulting in a non observed drift between the true position and the Bayesian estimate. Throughout the trajectory images are acquired at regular time intervals. The perceptual observations are empirical estimates of the hitting capacities $Z_k = \{\hat{T}_k(K_1)\}$. For these simulations a single structuring element (a point) provides enough information allowing the discrimination of the two RCS models.

The Gaussian mixture is triggered (**Fig. 9**) when the boundary lies inside the significant support (99%) of the uncertainty of the EKF estimate. The terms of the Gaussian mixture are indicated by the plus signs and the boundary of the principal support (coverage 99%) of the posterior density by the thick line. The mean of the mixture (the Minimum Mean Square Error estimate) is shown as a large circle and the trajectory (in the subsequent figures) as a dashed line.

At each iteration the robot searches for the optimal control input, restricted to those driving the robot inside a cone in front of the robot. The angular aperture of the cone was restricted to 90 degrees and the most distant position was at 70 (the size of the observation window, indicated by the large square is 260x260). It should be noted that due to the stochastic nature of the observations (see Fig. 6) for the uncertainty of the observed hitting capacities), the effective uncertainty reduction does not coincide exactly with the predicted one. Fig. 10 shows, however, that the effective mean square error follows well the predicted error throughout the whole trajectory (the iteration number of the consecutive images correspond to those indicated in Fig. 10 and Fig. 11).
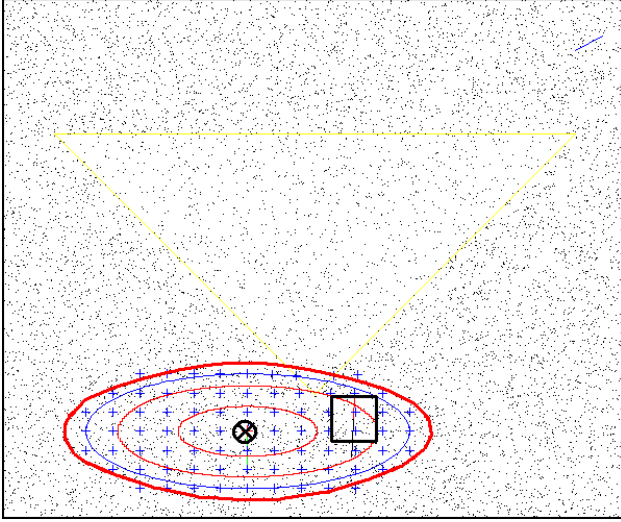
Fig. 9 (Iteration 10) Creation of the Gaussian Mixture Model. The terms are indicated by the plus signs. The principal support of the density (99%) is indicated by the thick (red) line.
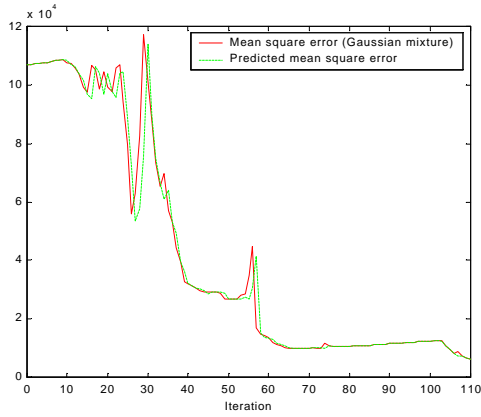
Fig. 10 The effective mean square error after application of the optimal action follows well the predicted mean square error.
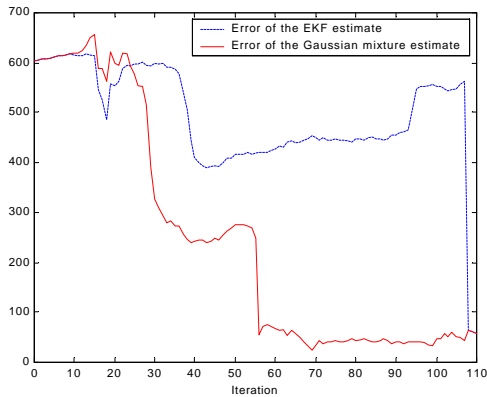

Fig. 11 The error of the estimated pose (distance to the true location) of the EKF estimate and the GME.

Initially the robot is driven in direction to the boundary, where localization can be performed. While the Gaussian mixture estimate (GME) can already reduce its uncertainty, some of the terms lying on or beyond the boundary, the EKF estimate has a null gain for the perceptual observation, see **Fig. 12**. When the EKF estimate finally reaches the boundary,

the non-zero gain results in a drastic reduction of its uncertainty, as shown in Fig. 13, such that the true robot location lies outside the principal support of the posterior density. The consequence is that the error of the EKF estimate cannot be significantly reduced in the future (see Fig. 11). The GME on the contrary preserves probability mass at the true robot location, maintained by those terms whose predicted observations correspond well to the real ones. The chosen control input maintains the trajectory of the robot at the boundary, and the ambiguity of the GME is slowly reduced. In **Fig. 14** its density is concentrated on two principal nodes. However the terms of the density mode at the left will be eliminated when they cross the boundary (to area 1), where the predicted hitting capacities do not correspond any longer to the observations. In Fig. 15 the density is already concentrated in a single mode all along the boundary. A significant reduction of the uncertainty is achieved when the robot reaches the second corner of area 2 (corresponding to iterations 55-60 in Fig. 10).

Throughout the remaining trajectory the robot continues to follow the boundary and is able to maintain its positioning error small (Iterations 70-110). Note that the number of remaining terms is very small, due to elimination of spurious terms and fusion to close ones. The principal support of uncertainty is small and linearisation of the observation model around these estimates is thus valid, such that adding new terms to the mixture is not necessary. At iteration 108 a single term of the mixture remains, replacing the previous EKF estimate.
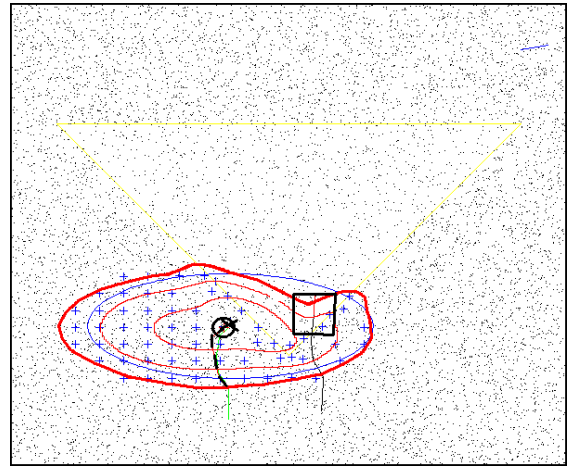
Fig. 12 (Iteration 17) The robot crosses the boundary. The EKF estimate has still null gain for the perceptual observations.
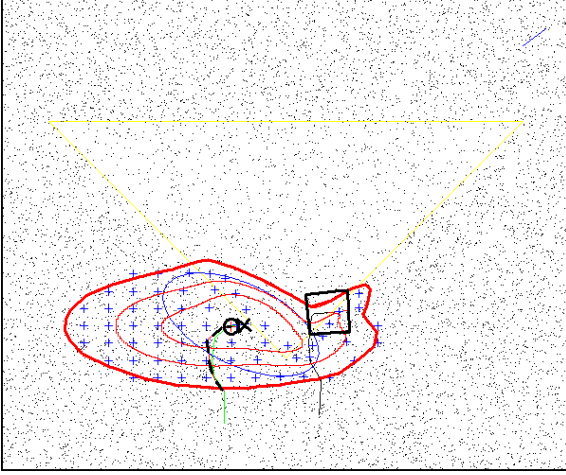
Fig. 13 (Iteration 20) The EKF estimate has a non-zero gain for the perceptual observation, resulting in a drastic reduction of its uncertainty (overconfident)
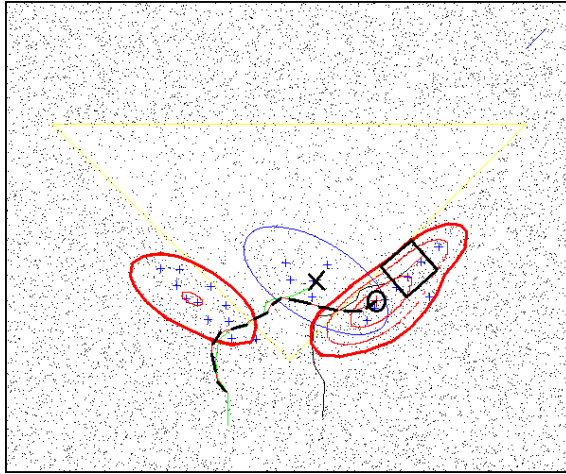

Fig. 14 (Iteration 33) The density of the GME has two nodes, representing well the uncertainty of the workspace. The EKF estimate is already lost.
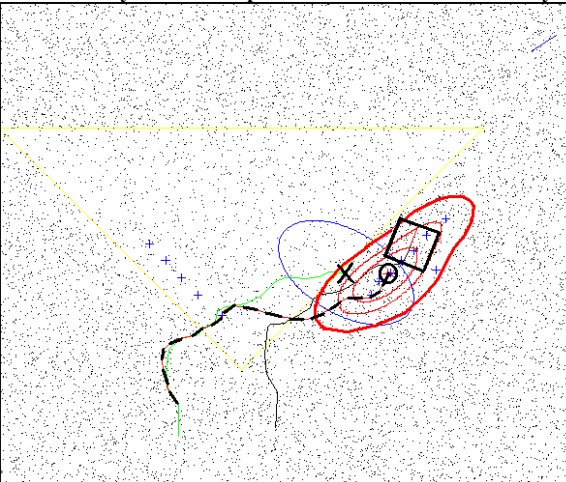

Fig. 15 (Iteration 40) The density of the GME is now concentrated at a single node and the robot is driven along the trajectory.
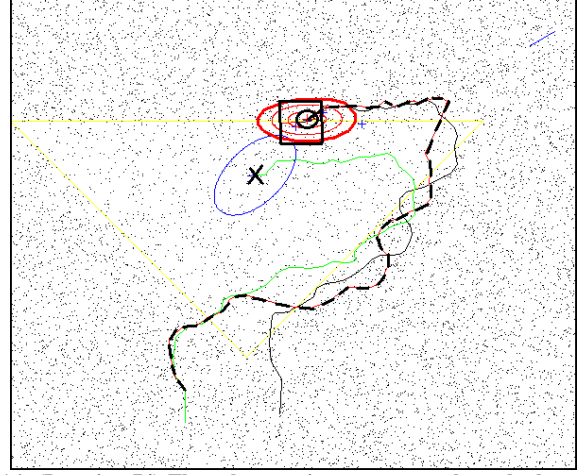

Fig. 16 (Iteration 74) The robot continues to move along the boundary. The mixture model contains now a few number of terms (all other terms are eliminated or fused). The linearisation of the observation model is valid, since the uncertainty support is small.

VI. CONCLUSIONS

In this paper we discussed two steps required for mapping regions of the sea bottom, and the use of statistical maps of their spatial characteristics for robot navigation. Modelization of the sea bottom as RCS's, describing the spatial distribution and morphological characteristics of labeled objects, requires a stable segmentation of the visual information. An algorithm, performing unsupervised segmentation has been presented. This algorithm is based on an analysis of the distribution of the intensities of neighboring pixels. We assessed the determination of the maximum likelihood estimates of the RCS model parameters. This approach is until now restricted to boolean models. The perceptual features used are the estimates of the hitting capacities for a limited number of structuring elements, whose choice was also discussed. The interest of this kind of environment modeling is two-folded. It provides biologists with a complete model of the distribution of the natural phenomena under study. They can also be used for the localisation of mobile robots presenting the advantage that they do not rely on the identification of individual outstanding features, which do not always exist. The consequence is an increased robustness with respect to small environment changes.

We present expressions that approximate the Bayesian estimator of the robot state using RCS maps. We addressed the problem of ambiguity in the workspace, which precludes the use of a simple EKF, when uncertainty of the estimated pose is very large. An approach that is related to multiple hypothesis, the Gaussian mixture model, is proposed and its feasibility is demonstrated by simulation results. In order to use the RCS map in an efficient way we propose an observation strategy that drives the robot to locations inside the workspace where the information provided by the perceptual observations results in the most significant reduction of uncertainty. Simulation results illustrates that RCS maps can effectively be used for localization.

REFERENCES

[1] Thrun, Dieter Fox and Wolfram Burgard, "Probabilistic Mapping of an Environment by a Mobile Robot," Sebastian Proc. IEEE Int. Conf. on Robotics and Automation (ICRA),Leuven, Belgium, May 1998.

[2] R. Smith, M. Self and P. Cheeseman, "Estimating Uncertain Spatial Relationships in Robotic, " *in* I. Cox and G. Wilfong, Eds, Autonomous Robot Vehicles, Springer-Verlag, 1990.

[3] P. Mourtalier and R. Chatila, "An Experimental System for Incremental Environment Modeling by an Autonomous Mobile Robot,". 1st Int. Symp. On Experimental Robotics, Montreal, June 1989.

[4] C.M. Smith and J. Leonard, "A Multiple hypothesis approach to concurrent mapping and localisation for autonomous underwater vehicles," Int. Conf. on Field and Service Robotics, Sydney, Australia, 1997.

[5] Dietrich Stoyan and Wilfrid Kendall and Joseph Mecke, *Stochastic Geometry and its Applications*, 2nd edition Wiley Series in Probability and Statistics, John Wiley & Sons, 1985.

[6] Maria-João Rendas, Stefan Rolfes, "Using Non-linear Filtering Techniques for Robot Localisation in Sparse Environments Using Perception Sensors," 7th Int. Symp. on Intelligent Robotic Systems, SIRS99, Coimbra, Portugal, July 1999.

[7] G. Matheron, "Random Sets and Integral Geometry", John Wiley & Sons, New York 1982.

[8] J.M. Rendas, S. Rolfes, "Using Non-linear Filtering Techniques for Robot Localisation in Sparse Environments using Perceptual Sensors", Proc. Of the 7th Int. Symposium on Intelligent Robotic Systems, Coimbra, Portugal, July 1999, pp 429-438.

[9] S. Rolfes, J.M. Rendas, "Statistical environment representation for navigation in natural environments", Proc. Of the 9th Int. Symposium on Intelligent Robotic Systems, Toulouse, France, July 2001.

[10] A. Gelb, "Applied Optimal Estimation", Mit Press, 1974.

[11] H. Takeda, C. Facchinetti, J.C. Latombe, "Planning the Motion of a Mobile Robot in a Sensory Uncertainty Field", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 16, October 1994.

[12] R. Bauer, "Active manoewers for supporting the localisation process of an autonomous mobile robot", Robotics and Autonomous Systems, vol. 16, 1995.

[13] J.M. Rendas, S. Rolfes, "Learning safe navigation in uncertain environments", Robotics and Autonomous Systems, vol. 31, 2000.