

Benthic boundary tracking using a profiler sonar

Christian Barat

Laboratoire I3S (Informatique, Signaux et Systèmes de
Sophia Antipolis), CNRS-UNSA
2000 route des Lucioles, BP 121, 06903 Sophia Antipolis
cedex, FRANCE

barat@i3s.unice.fr

Maria João Rendas

Laboratoire I3S (Informatique, Signaux et Systèmes de
Sophia Antipolis), CNRS-UNSA
2000 route des Lucioles, BP 121, 06903 Sophia Antipolis
cedex, FRANCE

rendas@i3s.unice.fr

Abstract - The paper presents work on autonomous tracking of boundaries between distinct benthic regions using an AUV equipped of a mechanically scanning profiler sonar. By exploiting sonar scans of the region below the robot, a classical control loop is closed around the sonar data, using a feedback signal that is robust with respect to classification “noise.” Results obtained in a real experiment of tracking a boundary between sand and Posidonia are presented.

I. INTRODUCTION

The ability to track natural boundaries defined in the ocean floor by distinct habitats is useful in several applications, either military (e.g. avoidance of dangerous operational regions) or civilian (physical oceanography, study of the evolution of biological species,...). In the past, we assessed the problem of automatic boundary tracking using visual information [1]. However, use of video data in the ocean can be often compromised by lack of ambient light, or by water turbulence. A more robust alternative is the use of acoustic sensors. In this communication, we present signal processing and automatic control algorithms to perform contour-tracking based on the information provided by a mechanically scanning profiler sonar.

The paper is organized as follows. The next section briefly describes the platform used for this study, and identifies the main processing blocks involved in the definition of the tracking behavior. The two subsequent sections, III and IV, are dedicated to the sonar signal processing and to controller definition, respectively. Finally, section V presents results of real experiments at sea. Section VI summarizes our conclusions, presenting some directions for future improvements.

II. ARCHITECTURE

The underwater platform used in this study is the ROV Phantom,¹ shown in Figure 2. This robot is equipped of three thrusters, two allowing control in the horizontal plane

¹ Phantom is a Remotely Operated Vehicle produced by Deep Ocean Engineering, USA, which has been made available for research in underwater robotics at I3S through a special educational arrangement.

(forward, reverse, turning) and another controlling the motion in the vertical plane (up/down motions), and of the following navigation and perception sensors: a magnetic compass, a rate gyro, a pressure (depth) gauge, an altimeter, a profiler sonar mounted on a tilt platform and a video camera. Moreover, each axis has been equipped of sensors allowing the measurement of the rotation speed of the corresponding motor shaft. The vehicle is linked to a dry-end operational station through an umbilical cable of about 120 meters, which allows remote automatic control of the robot.

The implementation of a sonar-based tracking behavior is based on two separate software modules: acoustic data processing, whose goal is to extract from the received data an indication \mathcal{E}_k of the *offset of the robot with respect to the tracked contour*, and the acoustic controller, whose responsibility is to generate the actual control signals r_k that guide the robot along the contour between the distinct classes. The next two sections describe each of these two blocks, shown in bold in Figure 1 below.

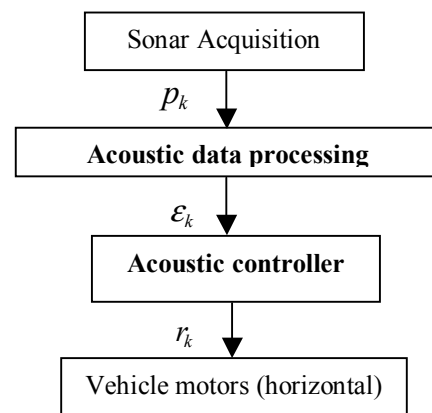


Figure 1. Signal processing and control architecture.

We assume that an independent control loop maintains the robot at constant altitude above the sea bottom, and address here only the control of the platform in the horizontal plane (surge speed and yaw).

III. SONAR CLASSIFICATION

In this section we describe the sonar signal processing algorithms. We first describe the sensing equipment on which the work is based, in subsection *A*. Subsection *B* addresses the problem of (unsupervised) learning of the probabilistic models of the observed data on which the algorithm presented in subsection *C* is based.

A The sensor

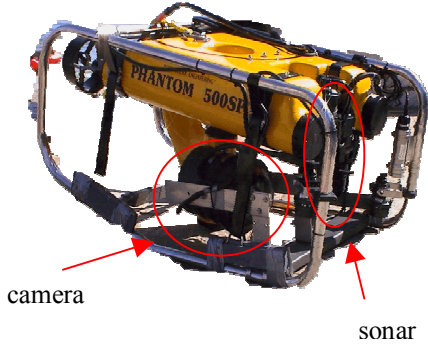


Figure 2 The ROV Phantom and its perception sensors

The sensor used to scan the sea bed is a dual frequency Tritech Seaking profiler Sonar, mounted in the ROV Phantom. During our experimentation, the following configuration has been used :

- the sonar is mounted in the front of the ROV (see Figure 2), oriented towards the sea bottom, and scanning the angular sector between $+30^\circ$ and -30° ,
- mechanical step size: 0.9° ,
- depth (range) resolution: 0.04 m,
- frequency of the emitted signal: 1.2 MHz,
- beamwidth: 1.4° (conical).

B Learning the classes models

The unsupervised segmentation algorithm exploits the fact that the sonar profiles corresponding to sea floor regions occupied by distinct species have *distinct shapes*. Each complete profile p_k is first reduced to a small set of features $\{f_1^k\}_{l=1}^L$. For the experiments presented in this paper, $L=1$, simply the energy of the profile inside a fixed length window centered on the detected maximum (shown in blue in Figure 3):

$$E_n = \frac{1}{32} \sum_{i=j-16}^{i+15} x_i^2. \quad (1)$$

The segmentation algorithm is based on a probabilistic framework, and associates to each individual class C_i a probability distribution of the extracted features, $p(\{f_1^k\}_{l=1}^L | C_i)$, $i=1,2$. These probability distributions are initially unknown, and are learned dynamically by the algorithm described below.

We introduce first some nomenclature and notation. Let X be a discrete random variable (rv) with probability space (Ω, A, P) where $\Omega = \{a_1, a_2, \dots, a_M\}$, is the (finite) 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 $x^{(N)} = \{x_1, x_2, \dots, x_N\} \in \Omega^N$ of N independent realizations of X . The **type** of $x^{(N)}$, which we denote by $v_{x^{(N)}} : \Omega \rightarrow [0,1]$ is the empirical estimate of the probability distribution (*pd*) of X , and is given by:

$$v_{x^{(N)}}(a_j) = \frac{1}{N} \sum_{i=1}^N 1_{a_j}(x_i), j=1, \dots, M \quad (2)$$

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

Consider that we are given two sequences of length N : $x_1^{(N)} = (x_{2_1}, \dots, x_{2_N})$ and $x_2^{(N)} = (x_{1_1}, \dots, x_{1_N})$. Then, the MDL (Minimum Description Length, see [2]) test for choosing between the two following composite hypotheses:

$$\begin{aligned} H_0: & x_1^{(N)} \propto p_0^N, x_2^{(N)} \propto p_0^N \\ H_1: & x_1^{(N)} \propto p_1^N, x_2^{(N)} \propto p_2^N, p_1^N \neq p_2^N \end{aligned} \quad (3)$$

where the probability laws p_0^N , p_1^N and p_2^N are unknown, i.e., for deciding whether the two sequences were generated by the *same* probability law or if they are samples from *distinct* distributions, is

$$\frac{(M-1)}{M} [2 \log(N+1) - \log(2N+1)] \underset{H_1}{\overset{H_0}{>}} D(v_1 \| \hat{\mu}) + D(v_2 \| \hat{\mu}) \quad (4)$$

In the previous expression, $\hat{\mu}$ is the balanced mixture of the types of the two observed sequences, and $D(\cdot \| \cdot)$ is the Kullback-Leibler divergence between probability laws:

$$\hat{\mu} = \frac{1}{2}(v_1 + v_2), \text{ and } D(v \| \mu) = \sum_{j=1}^M v(a_j) \ln \frac{v(a_j)}{\mu(a_j)}, \quad (5)$$

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

Eqs. (4) and (5) show that under the hypothesis that the individual samples (in our case, the set of features extracted

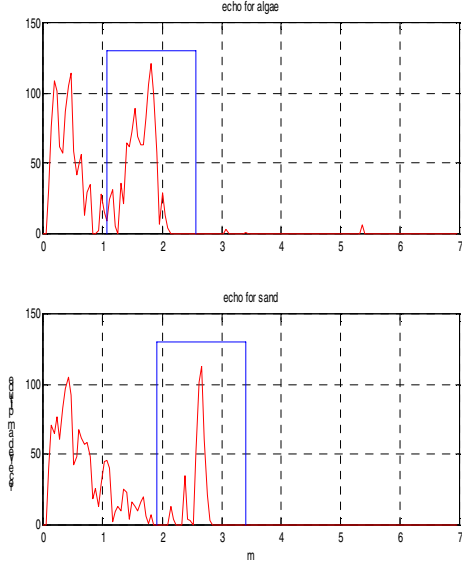


Figure 3: Profiles received from Posidonia (top) and sand (bottom) during one experiment at sea.

from each profile) are statistically independent, the types of the observed sequences are sufficient statistics for the decision problem formulated above.

To learn the classes models, we initialize the probability law of the first class with the type of the sequence of the first N measures: $\hat{p}_1^N = v_1$. We then use test (4) to decide if the types of the subsequently observed sequences, $v_k, k > 1$, correspond to the same distribution (\hat{p}_1^N). This test is repeated until hypothesis H_1 is accepted, for a given $k = k^*$. The learning phase is then stopped and we set the estimate of the second class probability law equal to the corresponding type: $\hat{p}_2^N = v_{k^*}$.

C Classification/Estimation (tracking) phase

We assume in this step that the probability laws $p\left(\left\{f_i^k\right\}_{i=1}^L \middle| C_i\right)$ associated to each class have been learned in a previous step using the method outlined above. Consider a set of N consecutive profiles acquired by the robot, $p^k = \{p_k, p_{k-1}, \dots, p_{k-N+1}\}$ and denote by f^k the corresponding set of $(N \times L)$ extracted features.

If during the acquisition of all these N profiles the robot observed the same sea-bed type, then, according to our hypothesis, the type of the sequence f^k should be close (in the Kullback-Leibler “metric”) to the corresponding probability law, and the optimal test to decide which class has been observed would choose the class m^* that minimizes the Kullback-Leibler divergence between the type of the observed sequence, V_k and the classes’ representatives, p_1^n and p_2^n :

$$m_k^* = \arg \min_{m=1,2} D(V_k \| p_m^n). \quad (6)$$

However, the assumption that the observed class is constant during the set of N consecutive profiles is not realistic, and the test above too simplistic. In general, the N successive sonar beams will hit sea-bed regions occupied by distinct habitats, such that a more realistic model for the type of the sequence of length N observed at time k is a *mixture* of the two basic pd corresponding to each of the two classes present:

$$v_k = \pi_k p_1^n + (1 - \pi_k) p_2^n \equiv p^n(\pi_k), \quad \pi_k \in [0,1],$$

where we defined the notation $p^n(\pi)$. The unknown mixture coefficient π_k indicates *the relative percentage of the two classes* in the observed sequence. The maximum likelihood estimate of π_k is obtained by solving the following minimization problem:

$$\hat{\pi}_k = \arg \min_{\pi} D(v_k \| p^n(\pi)). \quad (7)$$

We have verified numerically, for a large number of class models, that the criterion $D(v_k \| p^n(\pi))$ presents a dependency on π which is well approximated by a quadratic:

$$D(v_k \| p^n(\pi)) \equiv c_o^k + c_1^k \pi + c_2^k \pi^2, \quad (8)$$

where the unknown coefficients $\{c_i^k\}_{i=0}^2$ depend on the observed type and on the classes models. If these coefficients were known, $\hat{\pi}_k$ could be determined by the following analytical expression:

$$\hat{\pi}_k = -\frac{c_1^k}{2c_2^k}, \quad (9)$$

if the minimum is inside the interval $[0,1]$, and on one of its extrema (0 or 1) otherwise, indicating in this last two cases a “pure type”.

In order estimate the coefficients $\{c_i^k\}_{i=0}^2$, we evaluate the criterion $D(v_k \| p^n(\pi))$ for three distinct values of π (0.25, 0.5 and 0.75). We finally use the estimated coefficients in Eq. (9) to obtain the relative frequency of the classes present in the N most recently acquired profiles.

IV. CONTOUR TRACKING

The role of the acoustic tracker is to use the information yield by the sonar processing algorithm to generate control signals for the robot lower level control loops. To minimize problems due to variability induced by changing observation conditions, we impose that the detected contours be observed at *constant altitude*. This reduces the control problem to guidance in the horizontal plane. Ideally, we want the robot’s center of mass to describe a curve that is the parallel translation of the observed contour, whose shape is unknown. We have assessed

the problem of contour tracking with an autonomous platform equipped of a single point sensor in [3], where it is shown that a classic proportional-derivative controller with suitably defined gains, and using an error signal that is the distance of the robot to the tracked contour in the direction orthogonal to the tracked line, achieves the control objective of driving this distance to zero. In the application

cons
platf
erro
betw
altiti
sens
dete

mec
cont
that
[-Δ
verti
refer
moti
pres
whic
N=2
whe
cent
cont
form
to th

as a
the i

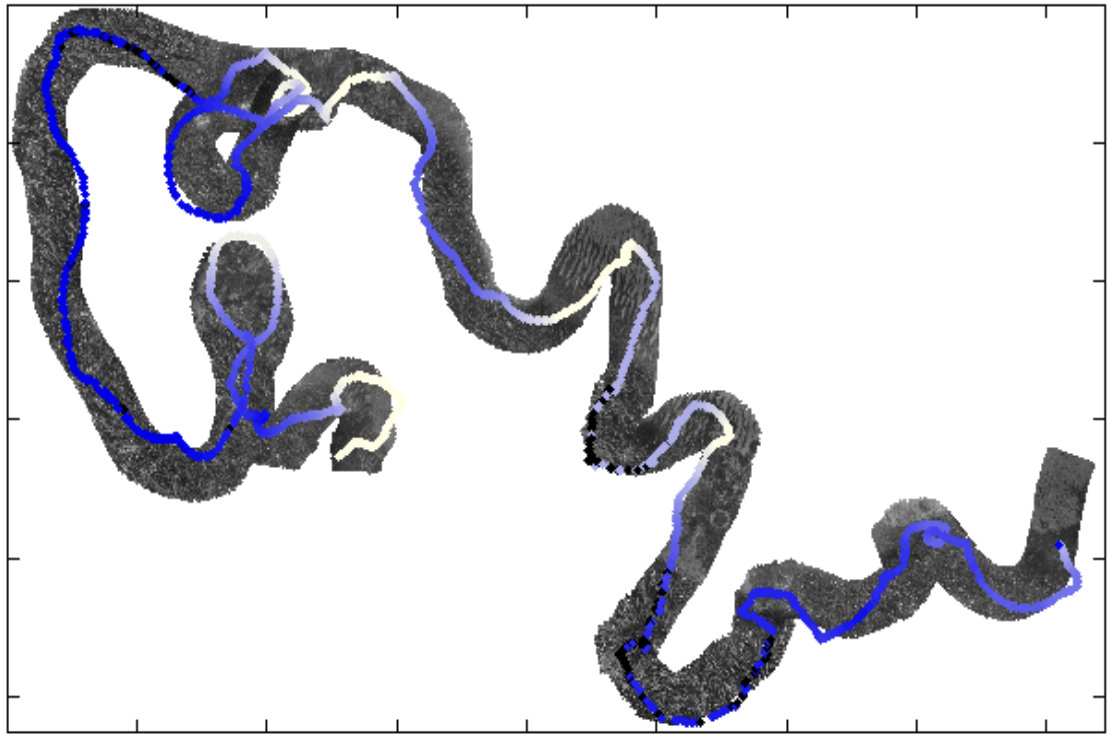
para
the r
cont

K_p and K_d depend on whether class 1 is on the left or on the right side of the boundary during tracking.

Note that basing the tracking task on this error signal, which is based on the set of $2N_s$ most recently received sonar profiles, improves the robustness of the algorithm with respect to outliers in the acquired signals. This controller is completely driven by the perception-derived error signal \mathcal{E}_k , and thus independent of the assumed dynamic and kinematic platform models, increasing its robustness with respect to modeling errors.

A simple implementation of this controller would consider constant surge velocity $u_k = u_0$, and directly control the trajectory curvature by imposing an instantaneous yaw rate equal to

$$r_k = u_0 K_k. \quad (11)$$



compared, see Figure 7.

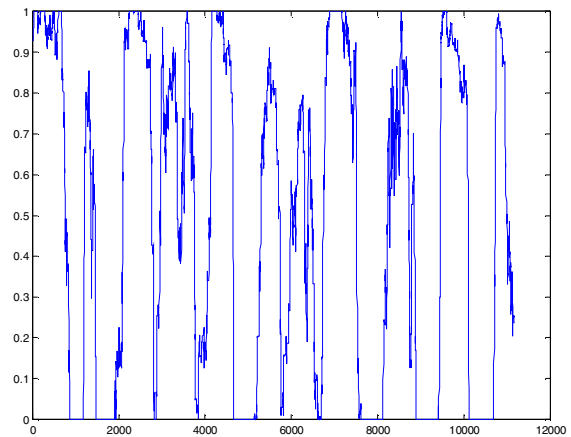
The sonar scans are processed using the method presented in section III. Sonar scans too close to the sea bottom (profile maximum occurs at a distance less than 0.4m) are not processed, since they correspond to returns from the ROV crash frame. They are represented in black in the figures.

We first learned, using the algorithm outlined in section III.B, the probability distributions of the two classes (sand and *Posidonia*). Using the geometrical model that relates the origins of the visual and acoustic sensors we co-registered the profiles classification onto the video mosaic, using the following color

code: white indicates pure sand ($\hat{\pi}_k = 0$), blue indicates pure *Posidonia* ($\hat{\pi}_k = 1$), while intermediate values are coded by varying intensities of blue. The result is shown in Figure 4, for an observation window of length equal to twice the length of the sonar scans (in the configuration of this experiments this leads to $N=134$), as explained in section IV.

B Contour Tracking

Figure 5 shows the evolution of the mixture signal π_k over a tracking experience at Villefranche-sur-mer (tracking of the boundary of a *Posidonia* patch). As we can see, the mixture coefficient oscillates between the extreme values of 0 and 1, indicating that the robot's path is oscillating between sand and *Posidonia* regions. The average value of the coefficient along the entire experience is 0.53, demonstrating the efficiency of the tracking methodology proposed in the paper. Figure 6 shows a small section of the complete tracking experience. Along the projected trajectory of the robot's center, we plotted, using a color code (white indicates *Posidonia* and blue indicates sand), the estimated mixture coefficient. The strong oscillations observed are due to the rapidly varying curvature of this region of the tracked contour.



VI. CONCLUSIONS

We presented algorithms for contour tracking using data provided by a profiler sonar. The main innovation of the approach is the proposal of basing tracking control on the estimation of the parameter of a mixture model that is fit to the most recently received sonar returns. A convenient error signal is obtained by centering the estimated mixture coefficient, which drives a simple discrete classical proportional-derivative controller. Results of this data processing algorithm on real data collected at sea for the boundary between two natural sea bed types demonstrate the adequacy of the approach

ACKNOWLEDGMENTS

This work has been partially funded by the European Union through the IST program, project SUMARE (*Survey of Marine Resources*), contract IST-1999-10836. See <http://www.mumm.ac.be/SUMARE> for more details.

REFERENCES

Figure 5: Evolution of mixture coefficient during a tracking sequence.

- [1]. Albert Tenas, Maria-João Rendas, Jean-Pierre Folcher, *Image Segmentation by Unsupervised Adaptive Clustering in the Distribution Space for AUV guidance along sea-bed boundaries using Vision*, Proc. OCEANS 2001, Honolulu, Hawaii, USA, November 2001.
- [2]. Jorma Rissanen, *Stochastic Complexity in Statistical Inquiry*, World Scientific, Series in Computer Science—Vol. 15, 1989.
- [3]. Maria-João Rendas, Isabel Lourtie, Georges Pichot, *Adaptive Sampling for sand bank mapping using an autonomous underwater vehicle equipped of an altimeter*, ISESS 2003, Vienna, Austria, May 2003.

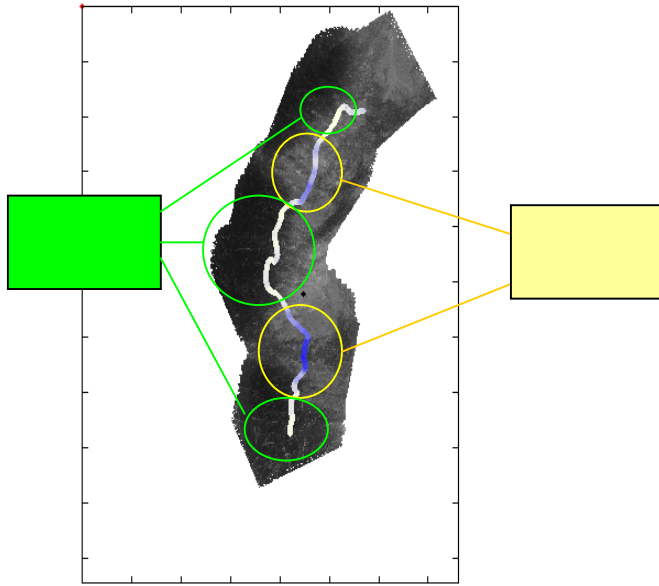


Figure 5: Detail of tracking.

Figure 2: detail of tracking.