Texture Segmentation Applied to P. Oceanica Beds' Upper Boundary Tracking by ROVs ⋆

Matko Barisic * **Dula Nad** * **Antonio Vasilijevic** *

* *University of Zagreb, Faculty of Electrical Engineering and Computing, Laboratory for Underwater Systems and Technologies. Unska 3, HR-10000 Zagreb, Croatia (e-mail: {matko.barisic, dula.nad, antonio.vasilijevic}@fer.hr).*

Abstract: The paper deals with an image processing technique for texture segmentation. By way of texture segmentation, a binary image is constructed and used to fit a line of the dominant direction of propagation of the texture in the image plane. The texture is captured by an oblique angle that positions the image plane in a near vertical orientation to the sea bottom plane. The texture being recognized in the experiment is that of *Posidonion Oceanicæ*, the benthic community (meadow or bed) of the Neptune grass Mediterranean endemic. It is the motivation of this paper for the dominant direction of propagation of the texture segment in the image plane to be used as a measure of the true direction of propagation of the upper border of *Posidonion Oceanicæ* in a feedback control loop that will enable the vehicle to autonomize the task of upper border tracking. The algorithm for extraction features four distinct phases: multi-resolution analysis using wavelets, vector quantization, post-processing of the obtained binary image and the extraction of the line parameters. The classification and line-fitting procedure are computationally optimized and made more robust by using weights in the Least Squares fitting procedure, and using nonlinear binary-image domain processing.

Keywords: image segmentation, texture segmentation, texture classification, wavelet transform, vector quantization, nonlinear binary-image transforms.

1. INTRODUCTION

Today, a marked challenge in successfully handling the issues of sustainable, nature-friendly and ecologically responsible maritime economic growth attuned with EC regulations (European Council and European Parliament, 2008, 2006) inter alia, is an implied unprecedented level of capability of collecting broad-spectrum, varied and time-actual biological and ecological data in the sea. Marine robotics are an important facilitator of this data-collection capability. Various types of marine robots can be used to support, or wholly autonomously carry out complex data-acquisition tasks underwater. An example of such a use of off-the-shelf, commercial micro- and mini-scale remotely operated vehicles (ROVs) (VideoRay LLC, 2011; SeaBotix Ltd., 2011; Seamor Marine LLC, 2011) is autonomous tracking of *Posidonion Oceanicæ* beds. When ROVs' hydroacoustic localisation systems (Tritech International, 2011; Sonardyne, 2011) are used to resolve the georeferenced position of the ROV along the tracked border, geo-tagged contours of this important angiosperm bio-indicator habitat, constructed by Neptune grass, a prolific Mediterranean endemic, are obtained.

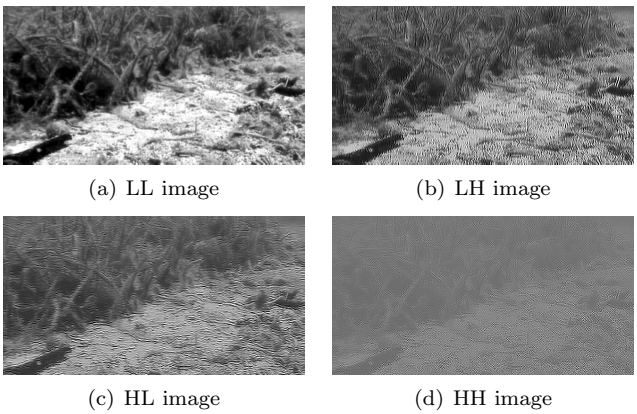
⋆ This research was made possible by the EU 2008-1-REGPOT grant for the "Developing the Croatian Underwater Robotics Research Potential" project, grant agreement no. 229553, and by the non-government organization Center for Underwater Systems and Technologies (CUST).

This paper presents a foundation for the eventual design of a closed loop controller with the above-mentioned capability. The paper deals with autonomous image analysis and recognition of the dominant line of division in between the *Posidonion* and the surrounding seabed. The images analyzed feature a view from a PAL analogue TV-camera (digitized by an off-the-shelf DVR to a 640 × 480 RGB image) of the Seamor 300F ROV (Seamor Marine LLC, 2011; Barisic et al., 2010) being piloted close to the ground (ca. 0.5 m – 1.5 m), with a slight deflection of the panning joint (ca. –10° – –15°). The presented analysis was performed off-line. The majority of the method, barring some newly introduced processing techniques such as the *melting* operation, are based on (Chandler, 2003).

Section 2 presents the *multiresolution analysis* that is the foundation of classifying and segmenting the image according to the dominant texture motifs that can be expected in the *Posidonion*. Section 3 presents the vector quantization technique that is used to compress the amount of data while retaining the analytical power of the data required to segment the image properly, and the histogram-based segmentation mechanism. Section 4 presents a chain of binary image morphological operations that are necessary to make the segmentation robust and useful for line-fitting. Section 5 presents the operations used to fit a line through the longest, dominant, mostly near-vertical or diagonal edge of the *Posidonion* w.r.t. the remainder of the sea bottom, in the *image plane*. Section 6 concludes the paper.

2. MULTIRESOLUTION ANALYSIS

The first block in the presented scheme of the overall system is the multiresolution analysis (MRA). The multiresolution analysis is concerned with extracting the features of an image that appear in different spatial scales (levels of resolution). The one reproduced here, using the Discrete wavelet transform (DWT) is concerned with extracting the graininess of the textures in the image. This is extracted by any procedure that awards high values to pixels of a neighborhood of an image wherein sharp rises and falls in the level of intensity occur. Alternatively, a separate layer can keep track of the pixels (by awarding them high values) wherein no such sharp rises and falls are present, but either more gradual dynamics of intensity or a flat level throughout is present. In a DWT-based MRA, each row, and alternatively, column of the image is subjected to a convolution with a scaling or wavelet function. This procedure, applied once, results in 4 (2^2 – two directions: row-wise and column-wise vs. two functions, scaling and wavelet) resultant 2D arrays of real coefficients for each of the pixels of the original image. The images are marked (LL, LH, HL, HH) where the first letter signifies the "lowpass" or "highpass" nature of the convolution across rows, and the second one the nature of the convolution across columns. The "lowpass" vs. "highpass" marks whether the convolution attenuates or accentuates, respectively, the graininess of textures. In the case of row-wise convolution, this differentiates between vertically grainy (or vertically homogeneous) textures, and in the case of column-wise convolution, it differentiates between horizontally grainy (or homogeneous) textures. The LL array (interpreted as an image by fixing the dynamic range of values to 0 – 255) is just the blurred image. The HL image (accentuating graininess in the vertical and attenuating graininess in the horizontal) will tend to award higher coefficients to the patches of textures that feature predominantly vertical motifs. The LH image vice versa. The HH image awards higher coefficient values to textures composed of diagonal motifs. A demonstration of such a first level MRA of a stock image is displayed in figure 1. Each of the reproduced arrays can be sequentially interpreted further as an image, to which this basic step of MRA is re-applied. This process, resulting in a power-of-four, i.e. 4^n for n the number of levels of analysis, multiplication of images, produces images wherein textures are accentuated w.r.t. the dominant direction of their motifs at varying spatial scales. Only with the introduction of levels beyond the 1st one can it be said that the procedure possesses a truly *multiresolution* nature.



(a) LL image (b) LH image

(c) HL image (d) HH image

Fig. 1. Discrete wavelet transform-based Multiresolution analysis

2.1 Choice of Wavelets and MRA Coefficients

Due to practical and near real-time processing considerations, MRA will only be applied at level 2. The wavelet and scaling functions used are *coiflets*, (Beylkin et al., 1991), and listed in (1, 2).

$$\begin{aligned} s(k) &= \sum_{t=0}^{11} s_t \cdot \delta(k-t) \\ &= 0.0231751934774337 \cdot \delta(k) \\ &\quad + 0.0586402759669371 \cdot \delta(k-1) \\ &\quad - 0.0952791806220162 \cdot \delta(k-2) \\ &\quad + 0.5460420930695330 \cdot \delta(k-3) \\ &\quad + 1.1493647877137300 \cdot \delta(k-4) \\ &\quad + 0.5897343873912380 \cdot \delta(k-5) \\ &\quad - 0.1081712141834230 \cdot \delta(k-6) \\ &\quad - 0.0840529609215432 \cdot \delta(k-7) \\ &\quad + 0.0334888203265590 \cdot \delta(k-8) \\ &\quad + 0.0079357672259240 \cdot \delta(k-9) \\ &\quad - 0.0025784067122813 \cdot \delta(k-10) \\ &\quad - 0.0010190107982153 \cdot \delta(k-11) \end{aligned} \tag{1}$$

$$\begin{aligned} \psi(k) &= \sum_{t=0}^{11} \psi_t \cdot \delta(k-t) \\ &= -1^{12-t} s_{11-t} \cdot \delta(k-t) \end{aligned} \tag{2}$$

$$\begin{aligned} I_{rL}(i,j) &= s(\lambda) * I(i,\lambda) \\ &= \sum_{\iota=-5}^{6} s_{5+\iota} I(i, j+\iota) \, \forall i = 1 \ldots m \end{aligned} \tag{3}$$

$$\begin{aligned} I_{rH}(i,j) &= \psi(\lambda) * I(i,\lambda) \\ &= \sum_{\iota=-5}^{6} \psi_{5+\iota} I(i, j+\iota) \, \forall i = 1 \ldots m \end{aligned} \tag{4}$$

$$\begin{aligned} I_{cL}(i,j) &= s(\lambda) * I(\lambda,j) \\ &= \sum_{\iota=-5}^{6} s_{5+\iota} I(i+\iota, j) \, \forall j = 1 \ldots n \end{aligned} \tag{5}$$

$$\begin{aligned} I_{cH}(i,j) &= \psi(\lambda) * I(\lambda,j) \\ &= \sum_{\iota=-5}^{6} \psi_{5+\iota} I(i+\iota, j) \, \forall j = 1 \ldots n \end{aligned} \tag{6}$$

Using (2 – 6), the images, where the original is a 2D array of unsigned 8-bit integers denoting intensities 0 – 255 (a black and white image) $\mathbf{I} = [I(i,j)]$, and the daughter-images are 2D arrays of real coefficients $\mathcal{I}_{(LL,\,HL,\,LH,\,HH)}(i,j)$, with i being the index of rows and j of columns are convolved to ultimately produce the 2nd generation granddaughter images.

Each of the $4^2 = 16$ second-level (granddaughter) images is obtained by filtering using one of (3, 4) followed by one of

(5, 6) for the first level of MRA (producing $4^1 = 4$ daughter images of the original), and repeating such a choice at the second level. Even among those 16 images, only the select ones, most representative of the textures inherent in the recognition of the chosen benthic habitat, the *Posidonion oceanicæ*, will be actually calculated. Since the nature of the problem closely coincides with (Chandler, 2003), follow the author's lead in this respect, choosing 5 out of the 16 resultant granddaughter-images for further analysis, $k = \{2, 4, 6, 7, 8\}$ indexed by a binary scheme wherein "L" is equal to the 0 binary digit and "H" is equal to 1 (so "LLLL" = 0000x0 = 0, "HHHH" = 1111x0 = 15). The 5 chosen 2D arrays of reals can be stacked, whereby each pixel of the *stacked array* can be conceived as a 5D vector of reals $\mathbf{c}(i,j) = [c_2(i,j)c_4(i,j)c_6(i,j)c_7(i,j)c_8(i,j)]^{\mathrm{T}} \in \mathbb{R}^5$. This proposed concept of segmentation relies on the ability of $\mathbf{c}(i,j)$ to encapsulate well and robustly the nature of "*Posidonion-likeness*". If so, there exists some metric on $\mathbb{R}^5$, $d : \mathbb{R}^5 \to \mathbb{R}^+$ such that there exist a pair $(\mathbf{c}_P^{(rep)}, \mathbf{c}_{NP}^{(rep)})$ that for a significant majority, in the statistical sense, and in the practical sense, of $\mathbf{c}(i,j)$, $d(\mathbf{c}(i,j), \mathbf{c}_P^{(rep)}) \leq d(\mathbf{c}(i,j), \mathbf{c}_{NP}^{(rep)}) \Rightarrow (i,j)$ is a pixel representing sea bottom overgrown with *Posidonion*, and $d(\mathbf{c}(i,j), \mathbf{c}_{NP}^{(rep)}) \leq d(\mathbf{c}(i,j), \mathbf{c}_P^{(rep)}) \Rightarrow (i,j)$ is a pixel of something other than the sea bottom overgrown with *Posidonion*. We assume this to be true.

2.2 Envelope Detection

In order to make the classification more robust, the row- or column-wise waveforms obtained by (3 – 6) must further be processed. As demonstrated in figure 2, convolving with the wavelet function of an image row (or column) will result in strong dynamics of intensity wherever a pattern of matching spatial frequency is encountered.

The amount of expression of the particular texture motif is evident in the local peak-to-peak amplitude of the signal. In essence, the peak-to-peak amplitude expresses a qualitative measure of presence of the particular texture motif (vertical or horizontal, of a spatial frequency matching the possibly down-sampled original image w.r.t. the wavelet or scaling function) in the local region of the image. The dynamics of intensity in between the local peaks doesn't contribute textural information. Therefore, it is sensible to process the convolved waveform by running an envelope detection on it. There are many possible envelope detectors, and the best ones in terms of trade-off of complexity, lag and efficiency are the ones based on the Hilbert transform (Hahn, 1996). In this work, however, an envelope detector represented in fig. 2 in a dashed outline is a simple peak hold of the absolute value in between each pair of zero-crossings of the signal.

3. VECTOR QUANTIZATION

The steps covered by this stage present us with an issue – the sheer number of feature-relevant data that needs to be processed by the analyzer in or near real-time. The *vector quantization* (Gersho and Gray, 1991) (VQ) is used to decrease the amount of information being processed whilst still retaining key textural information. VQ presents a *coding algorithm* whereby finite-dimensional

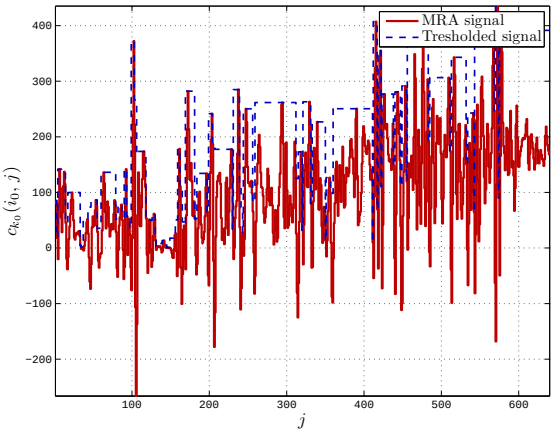


Fig. 2. Results of convolving a row of the image in fig. 1 with wavelet function (2).

vectors of reals are encoded into codewords (integer indices into a codebook). This coding uses the arguments of the minimum Euclidean distance between a vector and a codebook entry. The codebook is constituted out of a small set of representative data vectors that best capture the local variations of the entire universe of data, irrespective of a frame of reference. In the case of the problem at hands, this reduces the amount of information from $640 \times 480 \times 5 \times 8$ B $= 11.72M$ MiB to as little as 640×480 B $= 300$ kiB, for the total maximum compression ratio of 1:40. This assumes the additional existence of a codebook of at most $256 \times 5 \times 8$ B $= 10$ kiB. The operating assumption is that the codebook contains less than $2^8 = 256$ entries (which can be indexed by an unsigned integer stored in 1 B). An efficient and well-known algorithm (Dasarathy, 1991) of the k Nearest Neighbours (kNN) search can be applied to building up the codebook which has power-of-2 entries. The presented system uses a codebook of $2^4 = 16$ vectors, obtained by the kNN search algorithm pseudo-coded in table 1, performed on the data constituting a training set. Figure 3 displays the result of the training procedure, visualized in a plane spanned by the 2 principal components of the dataset used. When the system is

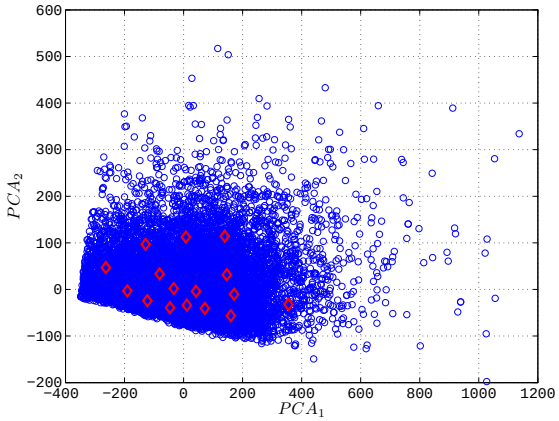


Fig. 3. The progression of the VQ kNN codebook-building algorithm, visualized in a plane spanned by the 2 principal components.

working in classification, rather than training mode, the 5D feature vectors are assigned a code κ_{ij} indexing the

Table 1. The kNN VQ codebook-building algorithm.

Initialization

1. Merge 2 sets of 5D vectors of reals specifying Posidonia and Non-Posidonia pixels (keeping them tagged)
2. Pick 10,000 random samples from the set $\mathcal{C} = \mathcal{C}_{NP} \cup \mathcal{C}_P$ into the training set $\mathcal{C}_{VQ}$
3. Calculate centroid $\boldsymbol{\gamma}_0$ of the entire set $\mathcal{C}_{VQ}$
4. Build the trivial, initial codebook set $\boldsymbol{\Gamma} = \{\boldsymbol{\gamma}_0\}$

Implementation

1. For $k = 1 \dots \log_2(16)$
2. For each member of $\boldsymbol{\Gamma}$, indexed by i
3. Randomize direction $\hat{\mathbf{n}}$ (a unitary vector in 5D)
4. Randomize deviation ε (a scalar)
5. Generate 2 5D points $\boldsymbol{\gamma}_i^{(r,\,l)} = \boldsymbol{\gamma}_i \pm \varepsilon\hat{\mathbf{n}}$
6. Partition $\mathcal{C}_i$, the part of $\mathcal{C}_{VQ}$ that the current codebook codes as symbol i w.r.t. $\boldsymbol{\gamma}_i \in \boldsymbol{\Gamma}$, into $(\mathcal{C}_i^{(r)}, \mathcal{C}_i^{(l)})$ according to $\arg\min\|\mathbf{c} - \boldsymbol{\gamma}_i^{(r,\,l)}\|$, $\forall \mathbf{c} \in \mathcal{C}_{VQ}$
7. Assign to the new value of $\boldsymbol{\gamma}_i$ the centroid of $\mathcal{C}_i^{(r)}$
8. Add to $\boldsymbol{\Gamma}$ a new vector $\boldsymbol{\gamma}_g$, $g = i + 2^{4-k}$, the centroid of $\mathcal{C}_i^{(l)}$
9. End for each
10. End for

codeword from the codebook $\boldsymbol{\Gamma}$ that is the nearest to each vector.

3.1 The Statistical Model

The set of codes $\mathcal{K}(\mathcal{C})$ of the entire training set $\mathcal{C} = \mathcal{C}_P \cup \mathcal{C}_{NP}$ (a random section of which was used in Algorithm in table 1), is used, together with the ground-truthed tags for each of the pixels contributing $\mathbf{c}(i,j) \in \mathcal{C}$ of ("Posidonia","Not Posidonia") to construct the *classification histogram*. The histogram resulting from dividing the number of occurrences of each individual $\kappa_{ij} \in \mathcal{K}(\mathcal{C})$, $\kappa_{ij} \in \{0, \dots, 15\}$ with the total number of pixels used to construct $\mathcal{C}$ is given in fig. 4.

Ideally, the statistical model would feature a large discrepancy between the likelihoods that a codebook index κ_{ij} denotes a pixel of Posidonia vs. Not Posidonia consistently throughout the codebook. However, that is not the case in a realistic scenario. As a consequence, the "raw" classification, presented in fig. 5, resulting in a binary image $\mathbf{B}_{raw} = [B_{raw}(i,j)]$ is non-ideal, featuring a spurious proliferation of both false positives and false negatives.

4. POST-PROCESSING

A far better quality of classification than the one displayed in fig. 5 is needed for robust fitting of a line in the image plane representing the border of the *Posidonion* upper border. Such classification may be achieved by a more involved statistical model, or a different paradigm altogether, used for segmentation. However, in the interest of processing speed and the optimal use of computing resources, we resort at this stage to *morphological operations* (Shih, 2010) on the binary image, however suboptimal it has turned out to be. First the image $\mathbf{B}_{raw}$ is morphologically opened using a default structural element

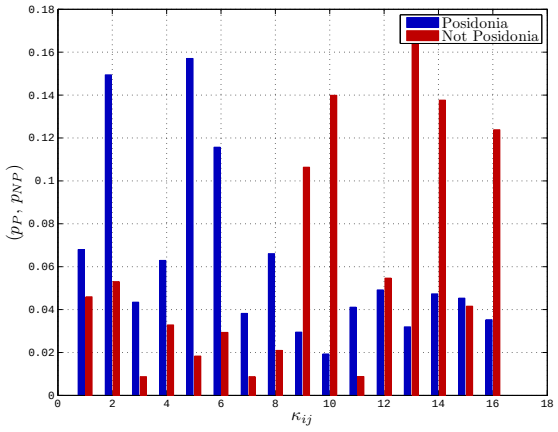


Fig. 4. The statistical model obtained from ground-truthed sets of VQ codes in hand-classified regions of interest of images from a training set.

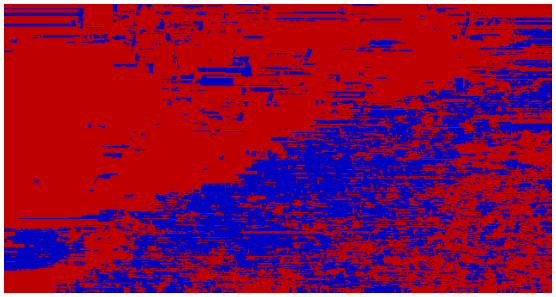


Fig. 5. The raw classification results based on the VQ codebook represented in 3 and the statistical model in 4.

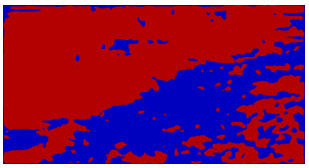
Table 2. The post-processing parameters used.

$\mu =$	0.35	$\nu_{obj} =$	0.5
$\nu_{hole} =$	0.2	$\mu_{WLS} =$	0.35

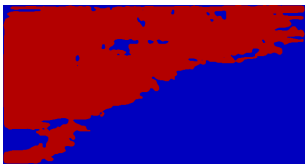
of a 3×3 matrix of 1s to remove sporadic misclassification. Next, the image is "melted" using a scalar normal parameter $\mu \in (0,1)$, cf. sec. 4.1, producing a melted image $\mathbf{B}_{mel} = [B_{mel}(i,j)]$. The obtained image is pruned of all 8-connected binary image objects with areas less than the ν_{obj} proportion of the total positive (1-valued) area of the image $A_1(\mathbf{B}_{mel}) = \sum_{i=1}^{m}\sum_{j=1}^{n} B_{mel}(i,j)$. This produces the image $\mathbf{B}_{pr} = [B_{pr}(i,j)]$ Finally, holes in the produced image are flooded in any remaining larger objects (intentionally it should be just the one object representing the captured expanse of the *Posidonion* on the sea bottom) that have survived the cull. The cutoff area for the size of holes to be closed is up to ν_{hole} of the total negative (0-valued) area of the original image, $A_0(\mathbf{B}_{pr}) = mn - A_1(\mathbf{B}_{pr}) = mn - \sum_{i=1}^{m}\sum_{j=1}^{n} B_{pr}(i,j)$. The resulting image is $\mathbf{B}_{fl} = [B_{fl}(i,j)]$. The resulting images in this chained post-processing operations are given figure 6. The parameters (μ, ν_{obj}, ν_{hole}) used are specified in table 2.

4.1 The Melting Operation

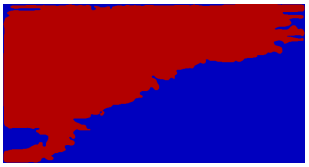
In the interest of clearing up the raw classified binary image $\mathbf{B}_{raw}$, we introduce a novel intermediary step in the post-processing of the binary, classified image, and call it



(a) Melted image, $\mathbf{B}_{mel}$.



(b) Eliminated small objects, $\mathbf{B}_{pr}$.



(c) Eliminated holes, $\mathbf{B}_{fl}$.

Fig. 6. The chain of post-processing binary image morphological transforms.

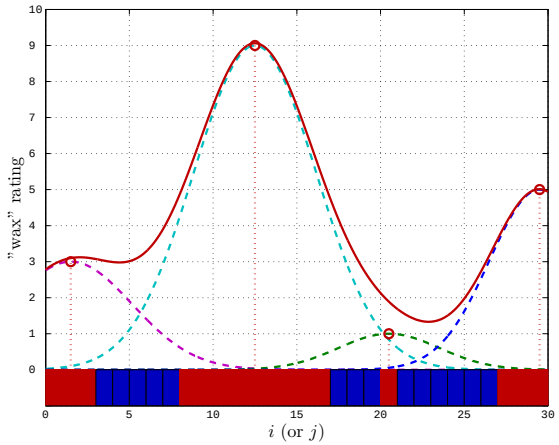
"melting". The visual appearance of the procedure is as if the positive (1-valued) areas of the binary image were agglomerations of wax to which heat was applied. The wax spreads viscously over surrounding areas, flattening out and covering more image area.

The procedure relies on a pair of 2D array accumulators of reals, inteded for row- and column-wise agglomerations of "wax". These 2 arrays can be construed as (color-map) pseudo-images containing a measure of row- or column-wise "clumpiness". Each row (or column) of the array(s) is excited by a progression of variously positioned Gaussian signals. These are centered over centroids of each separate aggregation of 1s (row-wise or column-wise, respectively). The height of each is equal to the area (i.e. a sum of 1s) of the aggregation. The standard deviation of each is one third of the larger of the two distances in between the current centroid and the preceding and following ones. This procedure is illustrated in figure 7. The finalized value, given by the full line in fig. 7.a) of the individual row (or column) of either pseudo-image is arrived at by summing over all excitations. A single finalized pseudo-image in Figure 8.a) is then produced by the RMS [1] of the row- to the column-wise pseudo-image. Before classification back to a binary image domain, a Gaussian 2D FIR filter is used for smoothing that eliminates the textile-like, gridded appearance of the total accumulator. The filter's impulse response is given in fig. 8.b). The resulting pseudo-image is displayed in fig. 8.c). The classification, resulting in the "melted" appearance in fig. 6.a) w.r.t. the original binary image in fig. 5 is obtained by tresholding, where the μ parameter specifies the top proportion of values in the pseudo-image that will be classified as 1s.

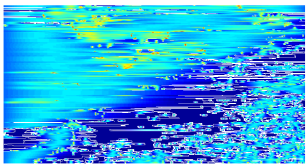
5. LINE-FITTING

The finalized binary image in fig. 6.c) is used to fit a line through the dominant edge in the image. The line-fitting consists of two intermediary steps: detecting an edge point $\mathbf{e} = [i_e \quad j_e]^{\mathrm{T}}$, $i_e = 1 \ldots m$ to each row of the image $\mathbf{B}_{fl}$, and using the Weighted Least Squares (WLS) algorithm (Björck, 1996) to fit a representative line to the thus constructed set of edge-points $\mathcal{E} = \{\mathbf{e}\}$.

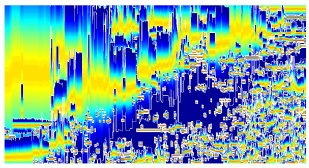
[1] The root of the mean of squares.



(a) Principle of excitation.



(b) Horizontally excited pseudo-image component.



(c) Vertically excited pseudo-image component.

Fig. 7. Illustration of the excitation of the two component pseudo-images of the melting procedure.

5.1 Edge detection

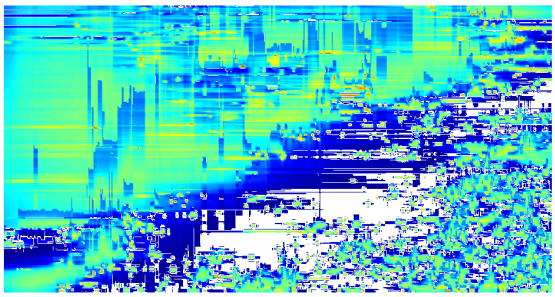
The edge-point fitting technique is taken from (Chandler, 2003) to be simple, human-readable in code and easy to implement. Memory and processing power of contemporary computers allow for efficient performance of this rather naïve fitting. Each row of the image, consisting of n columns, is compared against two families of step functions $S(j - j_0)$ and $1 - S(j - j_0)$, where $S(j)$ is the Heaviside function (across the columns, in any single row of the image). For the given row-index $i_e = 1 \ldots m$, an edge-point j_e is specified by the column-index j that is the minimum of the two minimizations in (7).

$$\begin{cases} j_e^{(r)} & = \arg\min_{j_0=0\ldots n+1}(\sum_{j=1}^{n} |I(i,j) - S(j - j0)|) \\ j_e^{(l)} & = \arg\min_{j_0=0\ldots n+1}(\sum_{j=1}^{n} |I(i,j) - 1 + S(j - j0)|) \end{cases} \tag{7}$$

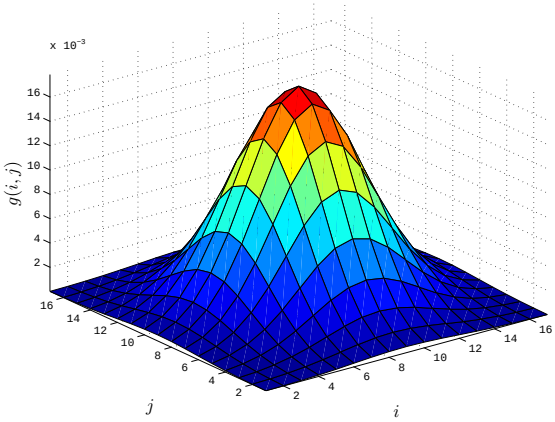
If the result of the minimization yields $j \in \{0, n+1\}$, the edge-point with the current row-index i_e is omitted from the set $\mathcal{E}$. Such an i_e-th row is best approximated by a 0-valued or 1-valued function throughout.

5.2 WLS Line-fitting

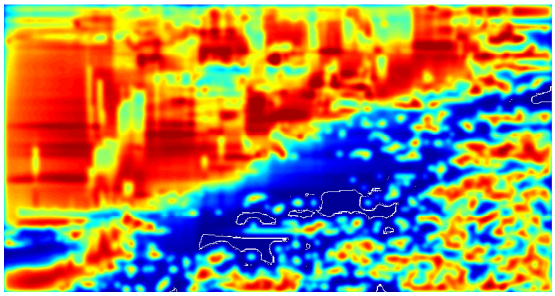
Finally, the analysis for each frame of the ROV pilot camera is concluded by fitting to the image a line that captures the direction of the border of the *Posidonion oceanicæ*. The stated goal of this research is that the automatic line-fitting procedure be used to provide feedback to the control and guidance of the ROV. Therefore, the algorithm should be influenced more by the geometry of the habitat's boundary nearer the camera. Taking into account the typical view-point of the ROVs camera with respect to the



(a) Final melting pseudo-image (the total accumulator).



(b) Impulse response of the smoothing filter.



(c) Filtered pseudo-image before classification.

Fig. 8. Illustration the final steps of the melting procedure.

scene and the sea bed (as evident from presented images), pixels closer to the ROV are situated at the bottom of the image. As a consequence, weights that accentuate the edge-points nearer the bottom of the image are used in (9, 10). With those, a WLS fitting of the line is performed by (11), obtaining the parameters (k_{line}, j_{line}), and thus fully specifying the line (8).

$$j_{line} = k_{line} \cdot i_{line} + b_{line} \tag{8}$$

$$\omega_i = \frac{2 \cdot (n + 1 - i)}{n \cdot (n + 1)} \tag{9}$$

$$w_k = \frac{\omega_{\arg_i \mathcal{E}}}{\sum_{\{\arg_i \mathcal{E}\}} \omega_i}, \; k = 1 \ldots \operatorname{card} \mathcal{E} \tag{10}$$

Let :
$$\mathbf{X} = [[i_e] \; \mathbf{1}_{\operatorname{card}\mathcal{E} \times 1}]$$
$$\mathbf{y} = [j_e]$$
$$\mathbf{W} = \operatorname{diag}([w_k])$$

Then :
$$\begin{bmatrix} k_{line} \\ b_{line} \end{bmatrix} = (\mathbf{X}^{\mathrm{T}} \mathbf{W}^{-1} \mathbf{X})^{-1} \mathbf{X}^{\mathrm{T}} \mathbf{W}^{-1} \mathbf{y} \tag{11}$$

In order to make the fit more robust, outlying edge-points are then disregarded and WLS re-applied to a reduced set $\mathcal{E}' = \mathcal{E} \setminus \mathcal{E}_{out}$. Outlying edge-points making up the set $\mathcal{E}_{out}$ are determined by tresholding the histogram of their distance from the obtained line. μ_{WLS} specifies the upper proportion of the distances that are far enough away to be considered outliers, and is listed in table 2. The distance of an edge-point to the fitted line, on which this tresholding is based, is specified by (12).

$$d_k = \frac{k_{line} i_e^{(k)} + j_e^{(k)} - b_{line}}{\sqrt{k_{line}^2 - 1}}, \; k = 1 \ldots \operatorname{card} \mathcal{E} \tag{12}$$

The finalized extracted and fitted line is represented in figure 9.

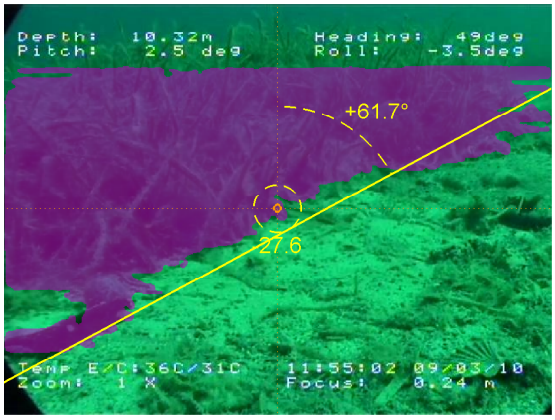


Fig. 9. The result of the texture-based detection of a well defined upper boundary of *Posidonion oceanicæ*.

5.3 Re-Parametrization for Navigation and Guidance Purposes

The line parameters obtained by the WLS procedure in the preceding subsection don't represent the best parameter set for the use of this system in the feedback of a heading and forward speed controller of a guided ROV. Considering the substance of the guidance problem, the distinguishing feature of the ROV being on proper course w.r.t. the border of the *P. oceanica* bed is that the line is vertical and nearly in the middle of the image. So, the introduced line, parameterized by (k_{line}, j_{line}) will be re-parameterized into (Φ, l). Here, Φ will be the angle offset from the vertical (with positive offset in the clockwise direction), defined in (14) and l a signed distance of the line from the image center defined in (13). In that respect, both are closed to zero for a vertical line in the middle of the image, so the controller based on them would also be at a static operating point. These parameters are very similar to the Hough parameters (Hough and Powell, 1960) of the fitted line. The sign of l will be provided by whether the line passes underneath or above the image center. The absolute value of l, i.e. the distance of the line from the image center d_0 can be obtained by inserting (320,240) into (12).

$$l = \begin{cases} k_{line} \cdot 320 + b_{line} > 240 : & -d_{cen} \\ k_{line} \cdot 320 + b_{line} \leq 240 : & d_{cen} \end{cases} \tag{13}$$

$$\Phi = \operatorname{sgn}(k_{line}) \cdot \left(|\operatorname{arctg}(k_{line})| - \frac{\pi}{2} \right) \tag{14}$$

6. CONCLUSION

A system for the automatic recognition of a well-defined and well-structured border of *Posidonion oceanicæ*, the

habitat of Neptune grass (*Posidonia oceanica*), a Mediterranean endemic of high environmental value, with surrounding bedrock, sediment or various other ground-covering habitats was presented. The system identifies the border as a best-fitted line. The angle of this line in the image plane of the ROV camera, and the distance of the line to the midpoint of the image can be used for feedback, to allow the ROV to autonomously follow such a border. The system uses digitized images from a PAL-TV analogue low-fidelity color camera. The system uses a processing chain in which the most computationally expensive step is the *multiresolution analysis* (MRA) using wavelets. For each pixel, 5 coefficients (chosen out of the 16 possible resulting coefficient arrays of a 2-level MRA) are stacked into coefficient vectors and subjected to vector quantization. A simple statistical model is then used to assign each code from the VQ codebook to being either Posidonia or Not Posidonia. Subsequently, morphological operations on the binary image are supplemented by a novel *melting* operation in order to make the classification more robust.

Future work will include the realization of the algorithm in optimized C++ code, using up-to-date software engineering techniques and available fast and resource-optimal SDKs (such as OpenCV), thus bringing the performance to real-time. This will allow the code to fit the line along the border of the observed *Posidonion* within the typical sampling time of ROV control systems – 0.1s. In that respect, a feedback loop will be closed using the entire described chain of image processing operations as a *vision-based software sensor*. A research into how well self-organizing maps (SOMs) can handle the classification step instead of the VQ approach will be undertaken. Various other wavelets, in addition to the coiflets specified by the scaling and wavelet functions (1, 2) will be tested.

ACKNOWLEDGEMENTS

The authors would like to thank Dr. Tatjana Bakran-Petricioli of the University of Zagreb, Faculty of Sciences, Division of Biology, Chair of Marine Biology, and Dr. Claudia Kruschel, University of Zadar, the Maritime Department, Marine Biology Section for their professional and scientific contribution in pinpointing the importance, and the contingent issues in tracking *Posidonion Oceanicæ* upper border, and helping the authors formulate this direction of research.

REFERENCES

Barisic, M., Vukic, Z., Miskovic, N., and Nad, D. (2010). Developing the Croatian underwater robotics research potential. In G. Parlangeli, A.M. Pascoal, and G. Indiveri (eds.), *Proceedings of the 7th IFAC Symposium on Intelligent Autonomous Vehicles 2010*. Universita del Salento, via Monteroni 73100, Lecce, Italy.

Beylkin, G., Coifman, R., and Rokhlin, V. (1991). Fast wavelet transforms and numerical algorithms i. *Communications on Pure and Applied Mathematics*, 44(2), 141–183. doi:10.1002/cpa.3160440202.

Björck, Å. (1996). *Numerical methods for least squares problems*. SIAM.

Chandler, R.C. (2003). *Autonomous Agent Navigation Based on Textural Analysis*. Ph.D. thesis, University of Florida, Gainesville, FL, USA.

Dasarathy, B. (1991). *Nearest neighbor (NN) norms: NN pattern classification techniques*. IEEE Computer Society Press tutorial. IEEE Computer Society Press.

European Council and European Parliament (2006). Towards a future Maritime Policy for the Union: A European vision for the oceans and seas.

European Council and European Parliament (2008). Directive 2008/56/EC of 17 June 2008, establishing a framework for community action in the field of marine environmental policy (Marine Strategy Framework Directive). *Official Journal of the European Union*, 164, 19 – 40. URL `http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2008:164:0019:0040:EN:PDF`.

Gersho, A. and Gray, R.M. (1991). *Vector quantization and signal compression*. Kluwer Academic Publishers, Norwell, MA, USA.

Hahn, S. (1996). *Hilbert transforms in signal processing*. Artech House signal processing library. Artech House.

Hough, P. and Powell, B. (1960). A method for faster analysis of bubble chamber photographs. *Il Nuovo Cimento (1955 – 1965)*, 18(6), 1184 – 1191.

SeaBotix Ltd. (2011). The LBV150-4 ROV system. URL `http://www.seabotix.com/products/lbv150-4.htm`. Product datasheet.

Seamor Marine LLC (2011). Seamor 300F ROV system. URL `http://www.seamor.com/products.html`. Product datasheet.

Shih, F. (2010). *Image Processing and Pattern Recognition: Fundamentals and Techniques*. John Wiley & Sons.

Sonardyne (2011). Ranger 2 USBL. URL `http://www.sonardyne.com/products/positioning/ranger2.html`. Product datasheet.

Tritech International (2011). Tritech Micron Nav - USBL Tracking System. URL `http://www.tritech.co.uk/products/products-micron_nav.htm`. Product datasheet.

VideoRay LLC (2011). The P4 CD 300BASE commercial ROV system. URL `http://www.videoray.com/products/42-p4-cd-300base`. Product datasheet.