Building Large-Scale Coverage Maps of Posidonia Oceanica using an Autonomous Underwater Vehicle

Antoni Burguera, Francisco Bonin-Font and Emilio Garcia-Fidalgo
Departament de Matemàtiques i Informàtica
Universitat de les Illes Balears
Ctra. Valldemossa Km 7.5, 07122 Palma, Spain
Email: {antoni.burguera,francisco.bonin,emilio.garcia}@uib.es

***Abstract*—This paper proposes a method to build large scale coverage maps of Posidonia Oceanica (P.O.) using the imagery provided by an AUV endowed with a bottom-looking camera. The process begins by detecting the regions containing P.O. in each individual image. Afterwards, the overlap between images is searched and the P.O. detection data in these overlapping regions is fused to build a global detection map. The paper proposes three different descriptors to detect P.O. in individual images and five strategies to fuse the overlapping regions. The experiments, performed in coastal areas of Mallorca, evaluate and compare the proposed descriptors and fusion techniques, showing that our approach surpasses the 91% detection rate.**

I. Introduction

Nowadays, several industrial and touristic activities depend on the preservation of the marine biodiversity. This preservation, which constitutes an important goal of the Horizon 2020, involves tasks such as monitoring certain species of seaweed.

In particular, the *Posidonia Oceanica* (P.O.) forms large underwater meadows that protect the shoreline against erosion, provides food and shelter for many organisms and, by absorbing large amounts of carbon, increases the water quality and transparency [1]. For these reasons, the European Commission's directive 92/43/CEE defines the P.O. as a priority natural habitat. Consequently, monitoring and mapping P.O. is becoming increasingly important, especially in the Mediterranean where intrusive species such as *Caulerpa Taxifolia* are threatening the P.O. survival.

Most strategies aimed at monitorning P.O. rely on photographs taken by human divers. The extension of the meadows is measured by means of markers and gauges placed in the seabed. In some cases, acoustic localization is used to recover the diver position [2]. Nonetheless, these approaches are slow, inaccurate and limited in time by the scuba air tanks capacity.

Recently, multi-spectral satellite imagery [3] and acoustic bathymetry [4] are becoming popular. However, these approaches present important flaws when trying to accurately discriminate P.O. from other algae types. Other researchers [5] use unmanned underwater vehicles endowed with cameras. However, at the extent of the authors knowledge, a fully autonomous detection and mapping of P.O. has not been achieved.

The *Augmented Reality Sub-sea Exploration Assistant* (ARSEA) is a funded Spanish project aimed at facilitating underwater monitoring using *Autonomous Underwater Vehicles* (AUV) and *Remotely Operated Vehicles* (ROV). One of the project goals is to autonomously or semi-autonomously build coverage maps of the P.O. meadows using image sequences obtained by a bottom-looking camera attached to the vehicle. Thanks to this, both the state and the evolution of the meadows can be measured.

This process involves three main steps. The first step, presented in Section II, consists on detecting P.O. regions in each of the gathered images [6]. The second step, discussed in Section III, involves AUV self-localization in order to detect the regions where the gathered images overlap. Finally, the third step, described in Section IV, is aimed at improving the P.O. classification by combining the individual detections corresponding to overlapping images.

Thanks to this, a P.O. coverage map is constructed, making it possible to quantify the extent of the P.O. meadows and their growth with time. Section V shows the experimental results validating our proposal.

II. Detection

In order to detect P.O., each gathered image is divided into a set of $M \times N$ sub-images or patches. A descriptor is computed for each patch and this descriptor is subsequently used to classify it as P.O. or not by means of a supervised learning schema.

In this paper, we propose three different descriptors. The first one, d_{RGB}, consists of three componentes, which are the mean of the red, green and blue color channels respectively within the patch under consideration.

The other two descriptors are based on 2D Gabor filters. These filters, which approximate some characteristics of the primary visual cortex of mammals, have been successfully used to perform image segmentation and object classification [7]. Moreover, they have predominant orientations, similarly to the actual P.O. leaves and, thus, they are likely to provide a strong response in front of this particular seaweed. As a matter of fact, they have successfully been used to detect P.O. in color enhanced imagery [6].

A Gabor filter is a sinusoidal plane wave modulating a Gaussian kernel function and can be formulated as follows:

$$h(x,y) = e^{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)} e^{-j2\pi(u_0 x + v_0 y)} \quad (1)$$

where (u_0, v_0) defines the central frequency and (σ_x, σ_y) denotes the standard deviation of the Gaussian kernel.

Our proposal is to generate a bank of 40 Gabor filters, involving 8 different orientations and 5 different scales and discretize each filter to an 8×8 matrix.

In order to compute the second descriptor, d_{CG}, the red, green and blue channels of each patch are convolved with all the discretized filters in the filter bank. From each convolution, two significant values are extracted. On the one hand, the local energy, defined as follows:

$$E = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} c(i,j)^2 \tag{2}$$

where m and n are the number of rows and columns, respectively, of c, which is the result of the convolution.

On the other hand, the amplitude is also computed for each color channel of each patch as follows:

$$A = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} |c(i,j)| \tag{3}$$

Being each color channel convolved with 40 Gabor filters, a patch descriptor is composed of 240 values.

To build the third descriptor, d_{GG}, each patch is first converted to grayscale. To this end, each pixel value in the grayscale image is computed as $0.2989 \cdot R + 0.5870 \cdot G + 0.1140 \cdot B$. The idea behind this weighted sum is to remove hue and saturation while retaining the luminance. Afterwards, the aforedescribed Gabor filter approach is applied. Thus, a patch descriptor in this case consists of 80 values.

Overall, d_{RGB} is solely based on color information, d_{GG} only takes into account texture information and d_{CG} considers both.

Our proposal to classify the patches is to use a *Support Vector Machine* (SVM) [8] although other classifiers could be used. The effects of using each of the three aforementioned descriptors to feed the SVM will be experimentally assessed in Section V.

III. Overlap detection

In this study, the AUV motion between pairs of consecutively gathered images is computed by means of the multi-threaded mosaicking algorithm *Binary Descriptor-based Image Mosaicking* (BIMOS) [9]. As these estimates are the result of a global optimization process, the drift is almost neglectable.

Let X^i_{i+1} be the obtained motion estimate from the reference frame of image i to the reference frame of image $i+1$. The pose of an arbitrary image j with respect to another image i can be computed as follows:

$$X^i_j = \begin{cases} \bigoplus_{k=i}^{j-1} X^k_{k+1} & j > i \\ 0 & j = i \\ \bigoplus_{k=1}^{i-j} \ominus X^{i-k}_{i-k+1} & j < i \end{cases} \tag{4}$$

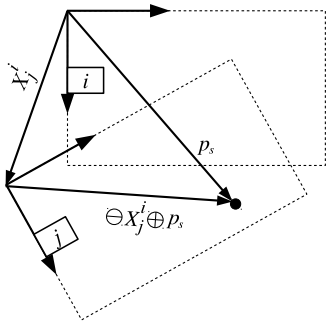

Fig. 1. Coordinate transformation

where $\oplus$ and $\ominus$ denote the composition and the inversion of transformations [10]. Thus, a point p in the coordinate frame of an image j can be expressed in the frame of an image i as $q = X^i_j \oplus p$.

Let $B = [b0, b1, b2, b3]$ be the set of points defining the boundaries of an image with respect of its own coordinate frame expressed in pixels. As the coordinate frames used by BIMOS are located at the top-left corner of each image, $B = [[0,0]^T, [w,0]^T, [w,h]^T, [0,h]^T]$, where w and h are the image width and height respectively.

Having all the images the same resolution, the boundary polygon of image j with respect to image i can be computed as $B^i_j = X^i_j \oplus B$. Let bx^i_j and by^i_j denote the x and y coordinates, respectively, of the four points in B^i_j.

Our proposal is to select one of the gathered images, namely i, as a global reference frame and then computing B^i_j for each of the other images. In this way, the bounding box of the whole observed area with respect to i, $B^i_{all} = [x_{min}, y_{min}, x_{max}, y_{max}]$ can be easily computed from the coordinates of all the resulting image boundaries.

Next step is to sample the whole bounding box at a desired sampling resolution δ and, for each sampling point, check if it lies within each of the individual boundary polygons. The coordinates of a sampled point p_s can be expressed with respect to the frame of an arbitrary image j as $\ominus X^i_j \oplus p_s$, as illustrated in Figure 1. Thus, if a sampled point lies within one or more images, the corresponding pixel intensities can be computed and stored.

After applying this process, a collection of pixel intensities $V_{x,y}$ is available for each sampled point (x,y). If the input images are the result of the classification described in Section II, the value list $V_{x,y}$ holds information of the P.O. presence according to each image that observed the corresponding sampled point. Figure 2 summarizes the process.

It is important to emphasize that determining whether a sampled point lies within an image or not could also be achieved by simply checking if $\ominus X^i_j \oplus p_s$ is inside B. However, by pre-computing B^i_j and using a fast *Point In Polygon* (PIP) algorithm such as the *crossing number algorithm* [11] the time consumption is significantly reduced.

Input: $[w, h]$: Image resolution
δ: Sampling resolution
i: Index of the reference image
$X^i_{0..n-1}$: Relative poses
$I_{0..n-1}$: P.O. classified images
Output: $V_{x,y}$: Set of P.O. detections per sampled point
for $j = 0$ **to** $n-1$ **do**
  $B^i_j \leftarrow X^i_j \oplus [[0,0]^T, [w,0]^T, [w,h]^T, [0,h]^T]$
end
$[x_{min}, x_{max}] \leftarrow [\min_{j=0}^{n-1} bx^i_j, \max_{j=0}^{n-1} bx^i_j]$
$[y_{min}, y_{max}] \leftarrow [\min_{j=0}^{n-1} by^i_j, \max_{j=0}^{n-1} by^i_j]$
for $x = x_{min}$ **to** x_{max} **step** δ **do**
  for $y = y_{min}$ **to** y_{max} **step** δ **do**
    for $j = 0$ **to** $n-1$ **do**
      if (x, y) **inside** B^i_j **then**
        $p = [p_x, p_y]^T \leftarrow \ominus X^i_j \oplus (x, y)$
        $V_{x,y} \leftarrow V_{x,y} \cup \{I_j([p_x], [p_y]\}$
      end
    end
  end
end

Fig. 2. Building the value list.

IV. Data fusion

The goal of the data fusion is to properly aggregate the values in $V_{x,y} = [v^0_{x,y}, v^1_{x,y}, ..., v^{n-1}_{x,y}]^T$ in order to obtain a single value for each sampled point stating the likelihood of P.O. at these coordinates.

We propose five different aggregation strategies. The first four ones, named $A^{mean}_{x,y}$, $A^{median}_{x,y}$, $A^{max}_{x,y}$ and $A^{min}_{x,y}$ consist on computing the mean, the median, the maximum and the minimum of $V_{x,y}$ respectively.

As, in our particular implementation the regions classified as P.O. are labelled as zero and the regions not containing P.O. are labelled as one, a single P.O. detection in $V_{x,y}$ leads to a P.O. result in $A^{min}_{x,y}$. Similarly, a single value in $V_{x,y}$ stating that no P.O. was present will result in no P.O. in $A^{max}_{x,y}$. These are extremely conservative approaches that will be evaluated experimentally. As for $A^{mean}_{x,y}$, the result is a number between 0 and 1 that must be thresholded. Regarding $A^{median}_{x,y}$, a value of zero or one is directly provided.

The fifth method, named $A^W_{x,y}$ weights each P.O. detection according the the image classifier itself. Being our proposal based on an SVM, further description assumes this particular classifier. However, extending the approach to other classifiers is straightforward.

Roughly speaking, an SVM considers each descriptor as a point in an n-dimensional space, being n the descriptor size. Training our SVM consists on finding the hyperplane P that divides the space in two regions, one containing the descriptors labelled as P.O. and another one containing the descriptors not labelled as P.O. Afterwards, classifying a new descriptor is achieved by determining in which side of the hyperplane lies.

However, additional information can be obtained during the classification. In particular, the distance $D(d, P)$ from each descriptor d to the hyperplane P can be easily computed. Large distances mean that the descriptor is likely to be well classified, as it is clearly sepparated of the threshold plane P. Similarly, short distances suggest a possible wrong classification. In other words, the likelihood of an image patch to be wrongly classified is inversely proportional to the distance from its descriptor to P.

Our proposal is to compute the inverse $D^{-1}(d, P)$ of the aforementioned distance for all the descriptors being classified and normalizing the values to the interval [0,1]. Let $U_{x,y} = [u^0_{x,y}, u^1_{x,y}, ..., u^{n-1}_{x,y}]^T$ be the set of these normalized values corresponding to each of the classification values in $V_{x,y}$.

In order to properly weight each detection according to its uncertainty, a Kalman filter (KF) approach is adopted by considering that $V_{x,y}$ is the measurement. As a result, $A^W_{x,y}$ can be written as follows:

$$\begin{aligned} A^W_{x,y} &= A_0 + (P_0 \cdot H^T \cdot (H \cdot P_0 \cdot H^T + R_{x,y})^{-1}) \cdot \\ &\cdot (V_{x,y} - H \cdot A_0) \end{aligned} \quad (5)$$

where A_0, P_0, H and $R_{x,y}$ are defined as follows.

$A_0 \in [0, 1]$ is the mean of the prior estimate. In absence of any other information, A_0 can be set to 0.5, meaning that any point in the space has equal probability of containing P.O. If the region to explore is known to contain large amounts of P.O., A_0 could be increased, or reduced if low amounts of P.O. are expected.

P_0 is the variance of the prior estimate. We propose the following heuristic to compute its value:

$$P_0 = \left(\frac{\min(1 - A_0, A_0)}{2} \right)^2 \quad (6)$$

As P_0 represents the uncertainty of A_0, the closer A_0 is to 0.5, the larger is the uncertainty. Also, by means of this Equation, the 2σ bound is always inside the [0,1] interval.

The parameter H is in charge of projecting the prior A_0 into the space of P.O. detections $V_{x,y}$. Thus, $H = [1]_{n \times 1}$ is a column vector of ones of the same size that $V_{x,y}$.

As for $R_{x,y}$, it denotes the covariance of the $V_{x,y}$. Our proposal is to consider each item in $U_{x,y}$ as a 2σ bound as follows:

$$R_{x,y} = \begin{bmatrix} \left(\frac{u^0_{x,y}}{2}\right)^2 & 0 & \cdots & 0 \\ 0 & \left(\frac{u^1_{x,y}}{2}\right)^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \left(\frac{u^{n-1}_{x,y}}{2}\right)^2 \end{bmatrix} \quad (7)$$

Let us define A^{mean}, A^{median}, A^{max}, A^{min} and A^W as the result of applying the mentioned aggregation criteria to all the sampled points (x, y). Both A^{mean} and A^W provide

TABLE I
QUALITY INDICATORS FOR THE THREE DESCRIPTORS ACCORDING TO MONTE CARLO CROSS VALIDATION

Indicator	d_{RGB}	d_{GG}	d_{CG}
Hit ratio (μ)	90.95%	91.62%	95.39%
Hit ratio (σ)	2.9	1.84	2.24
True positives	48.32%	48.73%	49.27%
True negatives	42.63%	42.89%	46.12%
False positives	5.82%	6.2%	2.84%
False negatives	3.23%	2.18%	1.77%

values between 0 and 1 and, thus, they must be thresholded. Our proposal is to use the Otsu method [12] to this end.

V. EXPERIMENTAL RESULTS

In order to experimentally assess the validity of our approach, two data sets have been used. The first one will be referred to as *base dataset* and is composed of 69 images of different resolutions gathered by an AUV with a bottom looking camera in several coastal areas of Mallorca. These images involve different illumination conditions and different types of P.O. textures. One third of the images in the dataset has only P.O. Another third has no P.O. at all, and the last third contains patches with P.O. and patches without it. A hand labelled ground truth has been built for each of these images.

The second one, called the *mission dataset*, is composed of 333 images with a resolution of 960x650 gathered during a single mission by the bottom looking camera of our AUV in Port de Valldemossa (Mallorca).

A. Training

In order to evaluate the training process, a Monte Carlo cross validation schema is used on the base dataset as follows. First, a 20% of the images has been randomly selected as the training set and the remaining 80% as the test set.

Second, the descriptors d_{RGB}, d_{CG} and d_{GG}, described in Section II have been computed for the training set and used to train the SVM together with the ground truth. To this end, each image has been divided into a set of 20×20 patches. Taking into account that in some ground truth images it was difficult to clearly state the P.O. perimeter, the SVM is trained under the assumption that a 5% of the training samples are outliers

Third, the descriptors have also been computed for the test set and classified using the trained SVM. The quality of the classification has been assessed thanks to the ground truth.

These steps have been repeated 500 times, randomly building the training and test sets each time. In other words, each of the 500 tests involves classifying a random set containint an 80% of the images using an SVM trained with the remaining 20% of the images.

Table I summarizes the obtained results. In the context of this study, true positives and negatives denote properly classified patches containing P.O. and not containing P.O. respectively. False positives and negatives denote the situations in which the classifier output (P.O. and not P.O. respectively) is wrong.

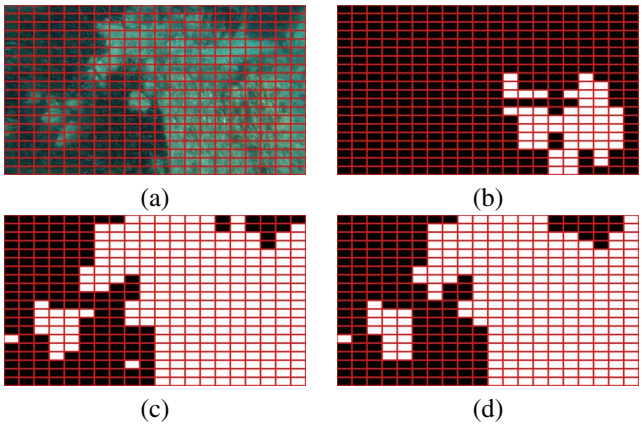

Fig. 3. An example of P.O. detection using the mission dataset, showing the 20×20 patch divisions. (a) Raw image and classifications according to (b) d_{RGB}, (c) d_{GG} and (d) d_{CG}.

The results show that the first two descriptors provide similar results, being d_{GG} slightly more stable as the standard deviation is smaller. To the contrary, a significant improvement appears when using d_{CG}, which surpasses a 95% hit ratio.

Being the percentage of P.O. in the dataset approximately 50%, the differences between positives and negatives for all the descriptors show that detecting P.O. is slightly more accurate than detecting regions without P.O. This is likely to be due to the fact that all P.O. patches are similar whilst those not containing P.O. may be very different as they may contain other types of vegetation, sand or rocks among many other.

B. Coverage map

In order to evaluate the coverage map building, the SVM has been trained using all the images in the base dataset and the three descriptors. The trained SVM was then used to classify the images in the mission dataset.

As an example, Figure 3 shows one of the images in the mission dataset classified with each of the descriptors. As it can be observed, d_{RGB} performs poorly in this case. Being the main difference between the base and the mission datasets the illumination conditions, this suggests that color-only descriptors are not appropppriate.

A set of coverage maps for this dataset was built using the overlap detection approach presented in Section III and the five aggregation methods described in IV applied to the three descriptors presented in Section II. To build the maps, a sampling resolution $\delta = 1$ was used and $A_0 = 0.5$ was chosen for the A^W aggregation method. A hand labelled coverage map was used as ground truth.

The obtained results when using d_{RGB} to describe the patches are shown in Table II. In this case, the aggregation results are significantly below those obtained with indivisual images. As suggested by the example in Figure 3, a change in the illumination with respect to the training dataset renders the d_{RGB} descriptor unreliable. Moreover, being A^{min} the aggregation method providing the best results in this case

TABLE II
QUALITY INDICATORS FOR EACH AGGREGATION METHOD USING THE DESCRIPTOR d_{RGB}

Indicator	A^{mean}	A^{min}	A^{max}	A^{median}	A^W
Hit ratio	60.86%	67.38%	55.94%	60.86%	61.25%
True positives	52.50%	51.49%	52.39%	52.51%	52.39%
True negatives	8.35%	15.89%	3.56%	8.35%	8.86%
False positives	39.14%	31.73%	44.06%	39.14%	38.75%
False negatives	0.00%	0.89%	0.00%	0.00%	0.00%

TABLE III
QUALITY INDICATORS FOR EACH AGGREGATION METHOD USING THE DESCRIPTOR d_{GG}

Indicator	A^{mean}	A^{min}	A^{max}	A^{median}	A^W
Hit ratio	90.04%	62.26%	82.20%	90.05%	90.79%
True positives	42.26%	14.70%	50.32%	42.26%	43.47%
True negatives	47.78%	47.56%	31.87%	47.79%	47.32%
False positives	0.27%	0.05%	15.74%	0.26%	0.29%
False negatives	9.69%	37.69%	2.06%	9.69%	8.92%

shows that d_{RGB} is more prone to classify patches as not containing P.O.

The use of d_{GG} significantly increases the detection rate, as shown in Table III. In this case, A^{min} is clearly surpassed by the other aggregators, reaching a 90.79% hit rate when using A^W.

The results when using d_{CG}, shown in Table IV, slightly improve the previous ones. In this case, A^{mean}, A^{median} and A^W surpass the 91% hit rate.

Overall, these results show that color alone is not enough to properly detect P.O. and that texture information is needed. Moreover, switching from d_{GG} to d_{CG} do not lead to major improvements, suggesting that adding color information to the texture descriptor is not significant.

As for time consumption, the data shown in Table V correspond to the total time spent to aggregate the data while building the whole coverage map. The time, which corresponds to a Matlab implementation running on a quad-core i7 laptop at 3.1 GHz, does not depend on the specific descriptor because, in all cases, the amount of values to aggregate is the same.

It can be observed that, even though A^W is slightly better

TABLE IV
QUALITY INDICATORS FOR EACH AGGREGATION METHOD USING THE DESCRIPTOR d_{CG}

Indicator	A^{mean}	A^{min}	A^{max}	A^{median}	A^W
Hit ratio	91.21%	62.22%	82.09%	91.22%	91.38%
True positives	43.20%	14.62%	50.40%	43.20%	43.93%
True negatives	48.01%	47.60%	31.69%	48.02%	47.44%
False positives	0.12%	0.01%	15.92%	0.12%	0.17%
False negatives	8.66%	37.77%	1.99%	8.66%	8.45%

TABLE V
TIME SPENT AGGREGATING THE DATA TO BUILD THE WHOLE COVERAGE MAP

Method	A^{mean}	A^{min}	A^{max}	A^{median}	A^W
Time	10.92s	5.44s	5.60s	12.04s	21.19s

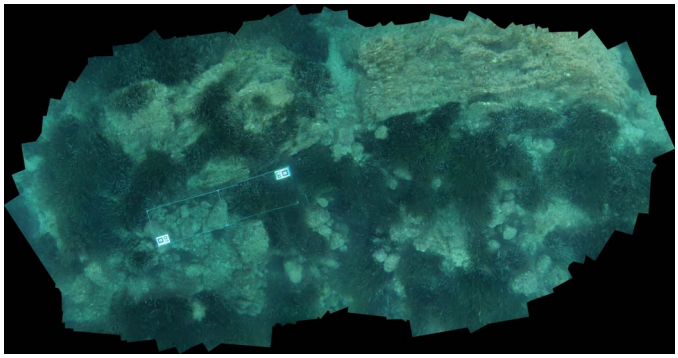

Fig. 4. Mosaic of the mission dataset built using BIMOS.

that A^{mean} and A^{median}, the time consumption is almost twice the one of these methods.

Summarizing, if time consumption is not important, the combination of d_{CG} and A^W provides the best results, reaching 91.38% hit rate. However, if time consumption is relevant, d_{GG} and A^{mean} lead to a similar hit rate of 90.04% with a significantly lower execution time.

To illustrate the results, Figure 4 shows the BIMOS mosaic corresponding to the mission dataset.

The fifteen coverage maps corresponding to the combinations of the three descriptors and the five aggregators are shown in Figure 5. The ones corresponding to A^{mean} and A^W are depicted prior to the thresholding process.

VI. CONCLUSION

A method to build large scale coverage maps of P.O. from underwater imagery has been presented. The proposal is to first detect P.O. in each of the gathered images and then fuse all the detections into a single, globally consistent, coverage map.

Three different approaches to P.O. detection and five different methods to perform the data fusion have been described and experimentally evaluated. As for P.O. detection, the results suggest that, although color provides useful information, it is texture the truly discriminant feature. Regarding data fusion, the KF approach slightly increases the detection rate at the cost of larger computational requirements.

Overall, our proposal is able to build large-scale coverage maps of P.O. with a hit ratio surpassing the 90%.

ACKNOWLEDGMENT

This work is partially supported by the Spanish Ministry of Economy and Competitiveness under contracts TIN2014-58662-R (AEI,FEDER,UE) and DPI2014-57746-C3-2-R (AEI,FEDER,UE).

REFERENCES

[1] E. Diaz-Almela and C. Duarte, "Management of Natura 2000 Habitats 1120, (Posidonia Oceanicae)." European Commission, Tech. Rep., 2008.

[2] D. Scaradozzi, G. Conte, G. de Capua, L. Sorbi, C. Luciani, P. de Cecco, and A. Sorci, "Innovative Technology for Studying Growth Areas of Posidonia Oceanica," in *Proceedings of the IEEE WorkShop on Environmental, Energy and Structural Monitoring Systems.*, 2009, pp. 71–75.

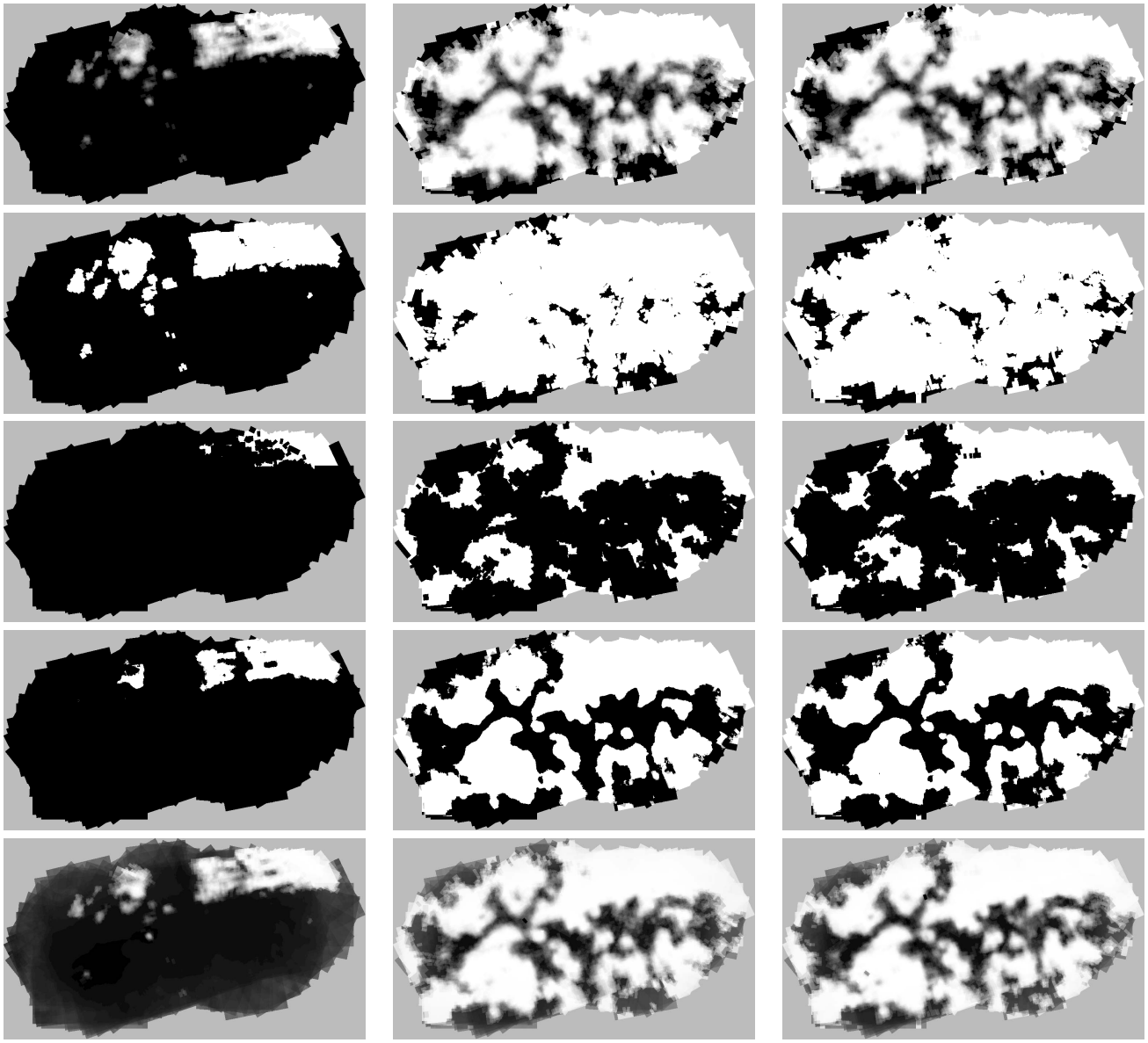

Fig. 5. Obtained coverage maps. The rows show the results for A^{mean}, A^{min}, A^{max}, A^{median} and A^{W} respectively. The columns correspond to the descriptors d_{RGB}, d_{GG} and d_{CG} respectively.

[3] R. Matarrese, M. Acquaro, A. Morea, K. Tijani, and M. Chiaradia, "Applications of Remote Sensing Techniques for Mapping Posidonia Oceanica Meadows," in *Proceedings of IEEE International Geoscience and Remote Sensing Symposium*, 2008, pp. 906–909.

[4] M. Montefalcone, A. Rovere, V. Parravicini, G. Albertelli, C. Morri, and C. N. Bianchi, "Evaluating Change in Seagrass Meadows: A time-framed Comparison of Side Scan Sonar Maps," *Aquatic Botany*, vol. 104, pp. 204–212, 2013.

[5] A. Vasilijevic, N. Miskovic, Z. Vukic, and F. Mandic, "Monitoring of Seagrass by Lightweight AUV: A Posidonia Oceanica Case Study Surrounding Murter Island of Croatia." in *Mediterranean Conference on Control and Automation (MED)*, June 2014, pp. 758–763.

[6] A. Burguera, F. Bonin-Font, J. Lisani, A. Petro, and G. Oliver, "Towards automatic visual sea grass detection in underwater areas of ecological interest," in *Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. Berlin, Germany: IEEE, 2016.

[7] A. Jain, N. Ratha, and S. Lakshmanan, "Object Detection Using Gabor Filters," *Pattern Recongnition*, vol. 30, no. 2, pp. 295–309, 1997.

[8] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*. Cambridge University Press, 2000.

[9] E. García-Fidalgo, A. Ortiz, F. Bonnín-Pascual, and J. Company, "Fast image mosaicing using incremental bags of binary words," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 1174–1180.

[10] R. Smith, P. Cheeseman, and M. Self, "A stochastic map for uncertain spatial relationships." in *Proceedings of International Symposium on Robotic Research, MIT Press*, 1987, pp. 467–474.

[11] M. Shimrat, "Algorithm 112: Position of point relative to polygon," *Communications of the ACM*, vol. 5, no. 8, 1962.

[12] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 1, no. 9, pp. 62–66, 1979.