Computer Vision in Underwater Environments: a Multiscale Graph Segmentation Approach

Fabjan Kallasi*, Dario Lodi Rizzini*, Fabio Oleari*, Jacopo Aleotti*
*RIMLab - Robotics and Intelligent Machines Laboratory,
Dipartimento di Ingegneria dell'Informazione, University of Parma, Italy,
Email: {kallasi,dlr,oleari,aleotti}@ce.unipr.it

***Abstract*—In this paper, we propose a novel object detection algorithm for underwater environments exploiting multiscale graph-based segmentation. The graph-based approach to image segmentation is fairly independent from distortion, color alteration and other peculiar effects arising with light propagation in water medium. The algorithm is executed at different scales in order to capture both the contour and the general shape of the target cylindrical object. Next, the candidate regions extracted at different scales are merged together. Finally, the candidate region is validated by a shape regularity test. The proposed algorithm has been compared with a color clustering method on an underwater dataset and has achieved precise and accurate detection.**

I. Introduction

A novel frontier of underwater robotics is the execution of manipulation tasks. Stereo vision enables object detection and pose estimation which are essential for object manipulation and grasping, but has to cope with challenging operating conditions and with distortion and attenuation effects arising when light propagates in water. In particular, light propagation in underwater environments suffers from phenomena such as absorption and scattering which strongly affect visual perception. These issues limit the available image invariants that can be exploited in object detection. In particular, robust identification of detailed patterns and features is difficult on blurred and attenuated frames acquired by underwater cameras. Moreover, the brightness and colors of the scene significantly change with the observer depth and with the distance between the object and the camera.

In underwater environments artificial objects are more easily recognized by their shape regularity or their pattern uniformity (at least in some parts). Salient color uniformity and sharp contours are exploited by the *pixel-feature clustering* (PFC) method described in [11], [13] and implemented to find human-made artifacts in natural backgrounds. However, the hypothesis about patternless object is too strong in many applications and may hold for single parts of artifacts. Moreover, local color or brightness changes due to backscattering usually result into oversegmentation.

In this paper, we propose a novel object detection algorithm based on *multiscale graph-based segmentation* (MGS) of underwater images and we compare its performance with PFC method. The approach is designed to reduce the dependence of detection from color. The graph-cut approach enables to segment the image into regions according to the local differences between adjacent pixels without any global features. Moreover, the resulting regions are connected components of a graph and better fit to object parts or areas with homogenous features. Unfortunately, the number of partitions obtained by graph-segmentation may depend on the observable details and image granularity. Hence, the region-of-interest (ROI) corresponding to an object may be split into several partitions. To overcome this problem, the graph-cut segmentation is performed at different scales, i.e. by executing the algorithm on blurred and downscaled instances of the input images. Scaling enables removal of the ephemeral image details and of specific problem affecting images acquired in underwater environments. Thus, the candidate ROI is found by matching segmented regions at different scales and testing the shape regularity of the ROI. The two detection algorithms PFC and MGS have been experimentally compared on a underwater dataset.

The paper is organized as follows. Section II reviews the state of the art in vision-based object detection for underwater environments. Section III describes the two object detection algorithms. Section IV illustrates the pose estimation of the detected object. Section V illustrates the results on object detection and pose estimation in underwater environments. Section VI provides some final remarks and observations.

II. Related Work

Computer vision is a major perception modality in robotics for object detection tasks. In underwater environments, however, vision is not as widely used due to the problems arising with light transmission in water. Artificial vision applications in underwater environments include detection and tracking of submerged artifacts [14], seabed mapping with image mosaicing [15], and underwater SLAM [7]. Garcia et al. [9] compare popular feature descriptors extracted from underwater images with high turbidity. Stereo vision systems have been only recently introduced in underwater applications due to the difficulty of calibration and the computational performance required by stereo processing [19], [3].

Underwater object recognition using computer vision is rather difficult due to the lighting condition of such environment. Kim et al. [12] present a vision-based object detection method based on template matching and tracking for underwater robots using artificial objects. All the tests are performed in a swimming pool. Several works perform object detection by segmenting the scene according to color and, then, by performing a more accurate assessment on the found region of interest. Bazeille et al. [2] discuss the color modification occuring in underwater environments and experimentally assess the performance of object detection based on color. Since underwater imaging suffers from short range, low contrast and

non-uniform illumination, simple color segmentation is one of the few viable approaches. In [18] the underwater stereo vision system used in Trident European project is described. Object detection is performed by constructing a color histogram in HSV space of the target object. In the performed experiments, there is an intermediate step between inspection and intervention where real images of the site to manipulate are available and used for acquiring the underwater target object appearance [10].

III. Object Detection

The aim of object detection is the identification of a region in the image containing the target object in order to estimate its pose. The input data of the algorithm consist of a set of image pairs acquired by an underwater stereo vision system like the one described in [17]. If the pose estimation is based on dense stereo estimation, then the convenient output of the detection algorithm is a selection mask computed on the stereo reference image (usually the left one). Otherwise, the detection procedure may also extract the features required for a sparse stereo matching from both stereo images, for example the straight object contours.

In this paper, two main properties typical of human-made artifacts are used: the relative color uniformity and the regular shape. The underwater lighting conditions make it difficult to fully exploit the first hypothesis. The water medium distorts colors so that even with a priori information about the target color, color segmentation is often unreliable. Moreover, the object contours are often ambiguous due to the fading luminance. Taking advantage from the regular shape condition, the target contours are searched by fitting lines. In our experiment, the object to be found is a pipe, although this information is not explicitly used until pose estimation. Two detection algorithms have been proposed to take advantage of these properties. PFC algorithm performs clustering on the pixels of the image according to local pixel features and next selects the connected components according to the shape. MGS algorithm belongs to the popular graph-cut approach representing the image as a grid graph. It first exploits color uniformity to enable better partitioning of the image into homogeneous regions, and then finds the shape searching for regular contours.

The image processing pipeline includes pre-processing, object detection and pose estimation. Pre-processing consists of image rectification and color restoration. Next, object detection and recognition are performed according to MFC or CFC methods described below.

A. Pre-processing

Pre-processing consists of two main operations related to the effect of light propagation in water. The first one is the de-distortion and rectification of the input images acquired by the stereo. While this is a standard operation, it has to be carefully performed to achieve accurate line extraction. The values of intrinsic parameters of a camera are affected by the water medium and have to be computed accordingly, as discussed in [17].

The second relevant operation is the restoration of the color modified by water. Figure 1(left) shows an example of underwater image with altered color. The pixel color shift is

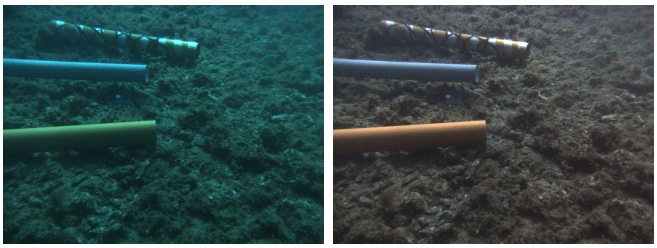


Fig. 1. An example of color restoration: the originally acquired image from the dataset [17] (left) and after color restoration according to grey world hypothesis(right).

apparent in all image, but it is most noticeable for the grey pipe that appears in cyan shades. In order to facilitate detection, it would be convenient to restore the color shades to their original color in air. Several approaches have been proposed for color restoration [1]. In our case, a color constancy method based on *grey-world hypothesis*, which assumes the average edge difference in the scene to be achromatic, sufficed. The results are illustrated by the example in Figure 1(right).

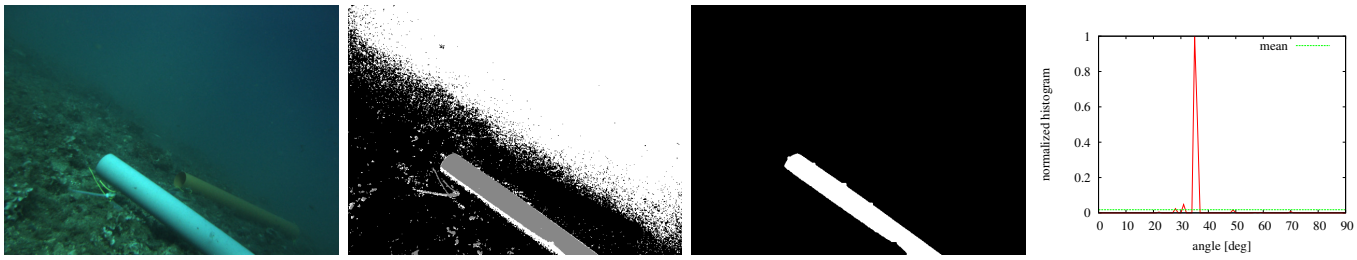
B. Pixel-Feature Clustering

The PFC algorithm extracts a vector of local features from each pixel like the HSV color values and the response to an edge extraction filter. The input image can be rescaled to a lower size to remove unnecessary details and to reduce the computational complexity of object detection. The scaling operation also acts as a low-pass filter in the image. The initial classification of each pixel p_i is independent from the classification of other pixels. In particular, the feature vector computed for p_i consists of the color channels of HSV space, respectively hue h_i, saturation s_i and value v_i, and of the gradient response to a *Sobel* filter g_i. Next, the item vectors $f_i = [h_i, s_i, v_i, g_i]^\top$ are clustered according to k-means algorithm [6]. The number of clusters used in the experiments is $k = 3$ and is independent from the number of objects in the scene.

The connected regions of each cluster are classified according to the corresponding angular histogram computed on its contour. The angular histogram of artifacts is concentrated on one or few peaks, while the histograms corresponding to the blobs extracted from natural seabed elements are usually more distributed. Hence, a set of segments is extracted from the contour. Each segment j is described by its length l_j and by its supporting line with equation $x\cos\alpha_j + y\sin\alpha_j = r_j$ (coordinates are expressed w.r.t. the image origin). Detection of line direction is allowed by an angular histogram $\mathcal{H}$ with bin counters $h_s \in \mathbb{N}$ and intervals $[s\Delta\theta, (s+1)\Delta\theta[$, $s = 0, \ldots, n_h - 1$ and $\Delta\theta = \pi/(2n_h)$. In particular, the segment j increments the corresponding angle bin h_k with a contribution proportional to the square of its length l_j as

$$s = \left\lfloor \frac{(\alpha_j + \pi) \bmod \frac{\pi}{2}}{\Delta\theta} \right\rfloor \tag{1}$$

$$h_s \leftarrow h_s + \left(\frac{l_j}{\max_i l_i} \right)^2 \tag{2}$$



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

Fig. 2. An image with a uniformly colored object from [17] (a), the output of the initial k-mean clustering (b), one connected component of a k-mean cluster (c) and its corresponding angular histogram (d).

The square of the normalized length reduces the influence of the smaller segments resulting from the potential oversegmentation of the contour. Finally, a cluster is classified as an object with regular shape if the histogram is "peaked", i.e. it is distributed along few principal directions. In particular, the validation condition of clusters is

$$\bar{h} = \frac{1}{n_h} \sum_{s=0}^{n_h-1} h_s < \sigma_{th} \max_{0 \leqslant s < n_h} h_s \tag{3}$$

The condition of equation 3 is satisfied when there is a dominant orientation in the histogram w.r.t. the average bin value $\bar{h}$. This occurs for objects with regular shapes and segment contours like the pipe considered in this paper.

C. Multiscale Graph-based Segmentation

The MGS method combines the detection of uniform color regions and the multiscale paradigm to identify the stable image partitions. The MGS method, like all graph-cut algorithms, enables to find contiguous regions of the image with homogeneous features. The borders of such regions correspond to high luminance or color gradients. Hence, unlike the segmentation achieved by PFC, the partitions computed with MGS consist of contiguous pixels by construction at each scale. Algorithm 1 illustrates the steps of MGS that are presented in the following.

Let $\mathcal{I}^s$ be the images at different scales $s = 0, \ldots, n_s$ with $\mathcal{I}^0$ is the input image with size $w^0 \times h^0$. Each image $\mathcal{I}^s$ at scale $s > 0$ is obtained by successively applying a Gaussian blur filter and downsampling $\mathcal{I}^{s-1}$. The scale factor of $\mathcal{I}^s$ w.r.t. $\mathcal{I}^{s-1}$ is equal to $1:2$. The multi-scale operations are shown at lines 2-7 of Algorithm 1: the segmentation algorithm $MinForestSegmentation$ and the edge extraction.

The unsupervised graph-cut algorithm $MinForestSegmentation$ proposed in [8] is used to partition each image $\mathcal{I}^s$. The method is closely related to Kruskal algorithm for the construction of minimum spanning tree [5]. The algorithm handles each pixel $p_i^s \in \mathcal{I}^s$ as a node of a weighted undirected grid graph connected to its 8-neighbors. The weights of the edges are computed as the color distance between adjacent pixels. In particular, the weight w_{ij}^s between two adjacent pixels p_i^s and p_j^s is computed as Manhattan norm of RGB component vectors of p_i^s and p_j^s. Henceafter, the apex ^{s} is omitted when the scale s is clear from the context. Initially, the image $\mathcal{I}^s$ is partitioned into

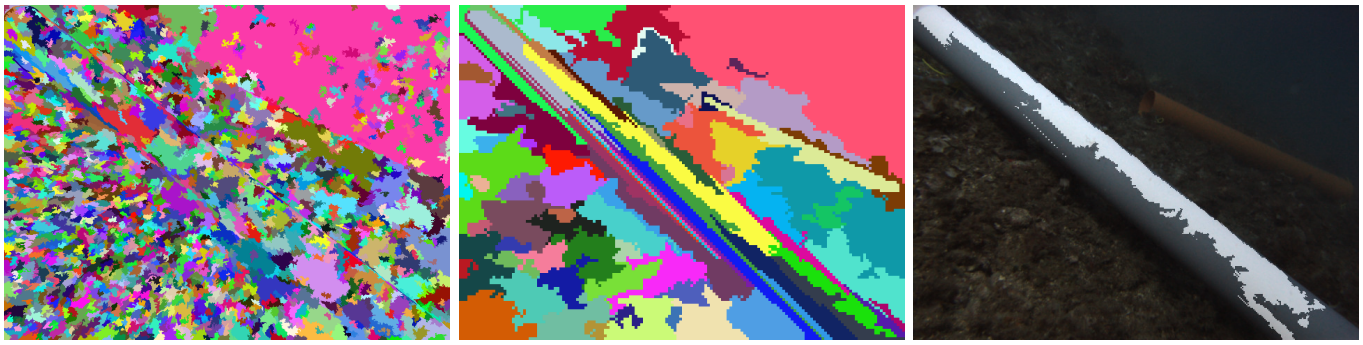
Algorithm 1: MultiscaleGraphSegmentation

Data: $\mathcal{I}^0$: input image. Parameters: n_s maximum scale; K_τ initial connection threshold; s_{min} minimum size of partition; d_{th} distance threshold; γ_{th} histogram ratio threshold.
Result: $\mathcal{O}$: image mask corresponding to the image.

1 $\mathcal{E} \leftarrow \{black\}$;
2 **for** $s = 0, \ldots, n_s$ **do**
3 $\quad sm \leftarrow \min\{1, s_{min}\ 2^{-s}\}$;
4 $\quad \{\mathcal{S}_i^s\}_{i=1,\ldots,k_s} \leftarrow MinForestSegmentation(\mathcal{I}^s, K_\tau, sm)$;
5 $\quad \mathcal{E} \leftarrow \mathcal{E} + Contour(\cup_{i=1}^{k_s} \mathcal{S}_i^s)$;
6 $\quad \mathcal{I}^{s+1} \leftarrow Downscale(\mathcal{I}^s)$;
7 **end**
8 $l = (\theta_l, \rho_l) \leftarrow FindDominantLine(RosinThres(\mathcal{E}))$;
9 $\mathcal{U}_l \leftarrow \{black\}$, $\mathcal{D}_l \leftarrow \{black\}$;
10 **for** $s = 0, \ldots, n_s$ **do**
11 $\quad$ **for** $i = 0, \ldots, k_s$ **do**
12 $\quad\quad c \leftarrow Mean(\mathcal{S}_i^s, l)$;
13 $\quad\quad d \leftarrow \min_{p \in \mathcal{S}_i^s} \|p_x \cos\theta_l + p_y \sin\theta_l - \rho_l\|$;
14 $\quad\quad$ **if** $d < d_{th}$ *and* $c_x \cos\theta_l + c_y \sin\theta_l - \rho_l \geq 0$ **then**
15 $\quad\quad\quad \mathcal{U}_l \leftarrow \mathcal{U}_l + Mask(\mathcal{S}_i^s)$;
16 $\quad\quad$ **else if** $d < d_{th}$ *and* $c_x \cos\theta_l + c_y \sin\theta_l - \rho_l < 0$ **then**
17 $\quad\quad\quad \mathcal{D}_l \leftarrow \mathcal{D}_l + Mask(\mathcal{S}_i^s)$;
18 $\quad\quad$ **end**
19 $\quad$ **end**
20 **end**
21 $\mathcal{H}_U \leftarrow HoughSpetrum(Contour(RosinThres(\mathcal{U}_l)))$, $\gamma_U \leftarrow \frac{\text{mean}\mathcal{H}_U}{\max \mathcal{H}_U}$;
22 $\mathcal{H}_D \leftarrow HoughSpetrum(Contour(RosinThres(\mathcal{D}_l)))$, $\gamma_D \leftarrow \frac{\text{mean}\mathcal{H}_D}{\max \mathcal{H}_D}$;
23 **if** $\gamma_U \leq \gamma_D$ *and* $\gamma_U < \gamma_{th}$ **then**
24 $\quad \mathcal{O} \leftarrow \mathcal{H}_U$;
25 **else if** $\gamma_U > \gamma_D$ *and* $\gamma_D < \gamma_{th}$ **then**
26 $\quad \mathcal{O} \leftarrow \mathcal{H}_D$;
27 **else**
28 $\quad \mathcal{O} \leftarrow \{black\}$;
29 **end**

segment regions $\mathcal{S}_l^s \subset \mathcal{I}^s$, each consisting of exactly one pixel. All the edges are sorted in increasing order according to their weights. At each iteration, the algorithm visits each edge (p_i, p_j) with weight w_{ij}. If (p_i, p_j) connects two pixels belonging to different regions, i.e. $p_i \in \mathcal{S}_{l_i}^s$ and $p_j \in \mathcal{S}_{l_j}^s$ with $\mathcal{S}_{l_i}^s \cap \mathcal{S}_{l_j}^s = \emptyset$, then the segments $\mathcal{S}_{l_i}^s$ and $\mathcal{S}_{l_j}^s$ could be joined in a unique segment. Such decision depends on the thresholds τ_{l_i} and τ_{l_j} associated respectively to $\mathcal{S}_{l_i}^s$ and $\mathcal{S}_{l_j}^s$: the two segments are joined into segment $\mathcal{S}_{\bar{l}} = \mathcal{S}_{l_i}^s \cup \mathcal{S}_{l_j}^s$, if $w_{ij} \leq \tau_{l_i}$ and $w_{ij} \leq \tau_{l_j}$. The threshold of the joined segment is equal to

$$\tau_{\bar{l}} = w_{ij} + \frac{K_\tau}{|\mathcal{S}_{\bar{l}}|} \tag{4}$$



(a) (b) (c)

Fig. 3. Graph-based segmentation of image in Figure 2(a) at the original resolution (a), at the 3-times rescaled version (b) and the detected region (c).

where K_τ is a parameter of the algorithm. The thresholds τ_l represent the sum of the minimal internal difference of segment $\mathcal{S}_l^s$ and of a tolerance dependent on K_τ and decreasing with the size of the segment. The parameter K_τ is the most critical for the resulting segmentation, since it affects the size and the number of segments. If K_τ is too low, the result is an over-segmented graph due to the local color and luminance changes. Otherwise, if it is too high, the contour of the object may be lost. To overcome this problem, segmentation is repeated on downscaled versions of the input image.

Scaling enables removal of the ephemeral image details associated to light propagation in water and highlighting of the object shapes. Object shape recognition is easier on images with a coarser segmentation. Figure 3 shows the segmentation output at two different scales. The overdetails at the lowest scale are limited and the general shape is easier to find. The target object used in the experiments is recognizable from its longer borders. Thus, the contours of segment regions $\mathcal{S}_l^s$ are extracted and accumulated into an edge image $\mathcal{E}$ (line 5), which is used to find the dominant line l (line 8). The dominant line l is a peculiar feature that enables to find the segment regions corresponding to the target object. Such dominant line divides the object area from the background, but without further analysis it is unclear which one of the two half-planes defined by l contains the object. Thus, two candidate ROIs $\mathcal{U}_l$ and $\mathcal{D}_l$ are built using the segment regions. In lines 10-20, each segment region $\mathcal{S}_i^s$, which is close to l (i.e. with distance d less than threshold d_{th}), is associated either to the ROI $\mathcal{U}_l$ above the line or to the ROI $\mathcal{D}_l$ below the line according to the position of its centroid c. The two candidate ROIs $\mathcal{U}_l$ and $\mathcal{D}_l$ are binarized using Rosin threshold [20]. The contour of the binarized regions is used to build their Hough Spectrum histograms $\mathcal{H}_U$ and $\mathcal{H}_D$ [4]. The two histograms provide the same information of the angular histogram in equation (2). Thus, the target object is detected if either $\mathcal{H}_U$ or $\mathcal{H}_D$ satisfies the "peak" test in equation (3), after changing the respective histogram and by using threshold γ_{th} instead of σ_{th}. This operation is illustrated at lines 21-29 of Algorithm 1 and Figure 3(c) shows an example of algorithm output.

IV. Pose Estimation

The pose estimation of an object is possible once the geometry of the object is known. Its pose is defined by a reference frame conveniently placed on the object. The frame coordinates are extracted from its recognizable parts in the image. In our case, the target object has a cylindrical shape identified by a pair of straight contour lines. The target may completely lie inside the field-of-view (FoV) or be only partially visible. In particular, one or both far ends of the cylinder may be occluded. Due to the symmetry of the cylinder, when no end-point is visible, the information is insufficient to properly estimate the reference frame.

Pose computation of the cylinder can be performed from a single frame or from two frames acquired by the stereo vision system. In the first case, the radius of the cylinder must be known in order to correctly find the cylinder symmetry axis [16]. If the length of the cylinder c_l is also known, then the cylinder position is know completely. In particular, a cylinder is defined given the radius c_r and its axis, a line in the form $\mathbf{c}(t) = \mathbf{c}_p + \mathbf{c}_d\ t$. In the image plane, the cylinder contour is delimited by two lines with equations $\mathbf{l}_i^\top \mathbf{u} = 0$, where $\mathbf{u} = [u_x, u_y, 1]^\top$ is the pixel coordinate vector and $\mathbf{l}_1$, $\mathbf{l}_2$ are the line coefficients. The sign of each line coefficient vector l_i is conventionally chosen s.t. the cylinder lies in the positive half-plane $\mathbf{l}_i^\top \mathbf{u} > 0$. Let $\mathbf{P}_c$ be the projection matrix which projects 3D points in the camera coordinate frame to 2D pixel coordinates using intrinsic and distortion camera parameters. Each contour line is projected in the 3D space as a plane π_i with equation $\mathbf{n}_i^\top \mathbf{p} + c_i = 0$ where $[\mathbf{n}_i^\top | c_i]^\top = \mathbf{P}_c^\top \mathbf{l}_i$ and $\mathbf{p}$ a generic point in camera reference frame coordinates. Since $\mathbf{n}_i$ points orthogonally through the center of the cylinder, the axis line can be obtained by intersecting both translated planes of the radius c_r in $\hat{\mathbf{n}}_i$ direction, where $\hat{\mathbf{n}}_i = \mathbf{n}_i / \|\mathbf{n}_i\|$. Thus, the direction of the cylinder axis is given by direction vector $\mathbf{c}_d = \hat{\mathbf{n}}_1 \times \hat{\mathbf{n}}_2$. Let π_{ep} be the plane $\mathbf{n}_{ep}^\top \mathbf{p} + c_{ep} = 0$ passing through one of the cylinder end-point with normal vector $\mathbf{n}_{ep}$ parallel to $\mathbf{c}_d$. The cylinder center point $\mathbf{c}_c$ is computed as

$$\begin{bmatrix} \mathbf{n}_1^\top \\ \mathbf{n}_2^\top \\ \mathbf{n}_{ep}^\top \end{bmatrix} \cdot \mathbf{p}_{ep} = \begin{bmatrix} -c_1 \\ -c_2 \\ -c_{ep} \end{bmatrix} \tag{5}$$

$$\mathbf{c}_c = \mathbf{p}_{ep} + \hat{\mathbf{n}}_{ep} \frac{c_l}{2} \tag{6}$$

The accuracy of this approach entirely depends on the detection of object edges in the image and on the camera calibration parameters.

Stereo processing can be exploited in different ways. For

example, a dense point cloud can be obtained by computing the disparity image from the two frames and the cylinder could be found through shape fitting techniques. Although this approach is rather general and can be applied to arbitrary shapes, it has several drawbacks. First, the point cloud density depends on the availability of reliable homologous points and, thus, on the color and pattern of the scene. Color uniformity facilitates the detection of target object, in particular due to the blurred and poor light conditions of underwater environments, but it may result into empty regions in the disparity image. Second, the accuracy of stereo 3D estimation depends on the accuracy of camera calibration. An approach similar to the single-frame method can be performed on a stereo image pair. The same cylindrical contour is projected in two different image planes resulting in four major contour lines. Each line is then reprojected in the 3D space resulting in four planes tangent to the cylinder. Let $\mathbf{P}_L$ and $\mathbf{P}_R$ be the projection matrices of the left and the right cameras respectively, π_{Li} and π_{Ri} the planes tangent to the cylinder obtained projecting in the 3D space the lines contour computed in both stereo images. Matrices $\mathbf{P}_L$ and $\mathbf{P}_R$ are referred to the same reference frame. The direction of the cylinder axis vector $\mathbf{c}_d$ must be orthogonal to each plane normal vectors $\hat{\mathbf{n}}_{Li}$ and $\hat{\mathbf{n}}_{Ri}$ respectively. Let $\mathbf{N}$ be the matrix whose rows are the normal vectors of planes through the cylinder axis

$$\mathbf{N} = \begin{bmatrix} \mathbf{n}_{L1}^\top \\ \mathbf{n}_{L2}^\top \\ \mathbf{n}_{R1}^\top \\ \mathbf{n}_{R2}^\top \end{bmatrix} \tag{7}$$

The cylinder axis direction $\mathbf{c}_d$ is estimated as the normalized vector that satisfies homogeneous linear equation:

$$\mathbf{N} \cdot \mathbf{c}_d = 0 \tag{8}$$

Clearly, this method is consistent with the single frame case which requires just a cross product to compute the cylinder axis vector. Moreover, solving (8) is more robust in presence of noisy data as can be the projection of lines in 3D space. This method does not require all the projected planes. If there are only two planes with normals $\hat{\mathbf{n}}_1$ and $\hat{\mathbf{n}}_2$, the cylinder axis is estimated as in the single frame case without loss of generality. As for the single frame method, let π_{ep} be the plane passing through one of the cylinder end-point detected in one of the stereo images. Then, the cylinder center point is obtained as

$$\begin{bmatrix} \mathbf{n}_{L1}^\top \\ \mathbf{n}_{L2}^\top \\ \mathbf{n}_{R1}^\top \\ \mathbf{n}_{R2}^\top \\ \mathbf{n}_{ep}^\top \end{bmatrix} \cdot \mathbf{p}_{ep} = \begin{bmatrix} -c_{L1} \\ -c_{L2} \\ -c_{R1} \\ -c_{R2} \\ -c_{ep} \end{bmatrix} \tag{9}$$

$$\mathbf{c}_c = \mathbf{p}_{ep} + \hat{\mathbf{n}}_{ep} \frac{c_l}{2} \tag{10}$$

The cylinder reference frame cannot be estimated without ambiguity due to the intrinsic symmetry of the cylinder.

V. Experiments

The performance of MGS object detection has been compared with the PFC algorithm on a dataset acquired using a stereo vision system near Portofino (Italy) [17]. The underwater stereo vision system was submerged at a depth of 10 m together with a set of cylindrical pipes of different colors. The dataset provides pairs of camera frames acquired by stereo cameras, but the proposed object detection algorithms operate on a single frame. Hence, a subset of 305 frames acquired by the left camera has been selected and manually annotated. The chosen frames may contain one or two cylindrical objects, but there is always only one object in the foreground that is considered as target object. For example, in Figure 2(a) there are two pipes, but the orange one is in the background. The proposed detection algorithms have been designed to search exactly one target object and, when multiple candidate objects appear in the frame, it selects the one with prominent features (image area, regular borders, etc.).

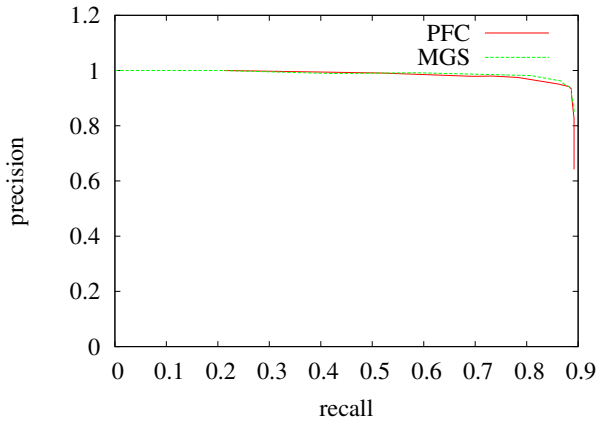


Fig. 4. PR curves of PFC and MGS algorithms for Portofino dataset.

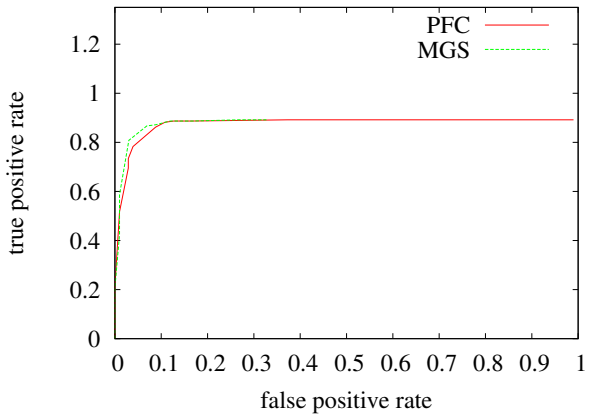


Fig. 5. ROC of PFC and MGS algorithms for Portofino dataset.

	PFC	MGS
Frame number	305	
Threshold	$\sigma_{th} = 0.04$	$\gamma_{th} = 0.48$
TP	179	177
TN	91	94
FP	11	7
FN	24	27
Precision	94.2%	96.2%
Recall	88.2%	86.8%
Accuracy	88.5%	88.9%
1-FPRate	89.2%	93.1%
F-Measure	91.1%	91.2%

TABLE I. Detection results for PFC and MGS algorithms on the dataset.

The detection algorithm output and the annotated groundtruth consist of binary images: the white color pixels corresponds to the positive region (i.e. occupied by the target object) and the black ones to the negative. The aim of the detection algorithms is the identification of the image region containing a significant part of the target object. Let $\mathcal{D}$ be the pixel sets classified as target object and $\mathcal{G}$ the groundtruth. A correct detection occurs if $|\mathcal{D} \cap \mathcal{G}|/|\mathcal{D}| > 0.5$.

Figures 4 and 5 respectively show the precision-recall (PR) and receiving operating curve (ROC) for the two proposed algorithms PFC and MGS. Table I illustrates the classification values obtained for a specific value of acceptance thresholds σ_{th} and γ_{th}. Observe that the recall value of PFC and MGS never reaches the 100% due to the filtering in the segmentation phase: even after increasing the thresholds σ_{th} and γ_{th}, clearly negative images are not classified as positive and the precision is not compromised. An alteration of the initial algorithm steps would be required for an indiscriminate acceptance of all input images as positive. Although the PR and the ROC curves of PFC and MGS tend to overlap, MGS achieves slightly better results. In particular, the precision of MGS is higher than PFC: the regions obtained from graph cut better fit the object than those based on color. The values in Table I confirm the previous observation: performance parameters, in general, are better for MGS than for PFC. The only exception is the recall value, which is slightly better for PCF. Both proposed algorithms exhibit detection performance suitable for the execution of manipulation tasks, which usually allow the repeated observation of the object to be grasped.

The pose estimation method has not been quantitatively assessed in this paper. The qualitative assessment showed that the robustness and stability of pose estimation highly depends on object detection. In future works, we expect to investigate

VI. Conclusion

In this paper, we have presented a novel object detection algorithm for underwater environments relying on multiscale graph-based segmentation. The algorithm exploits the refined image segmentation obtained from the graph-based approach and which consists of connected regions. The segmentation procedure is applied at different scales to capture both the accurate contour in high resolution images and the general shape of the object at higher scales. High scale images are free from ephemeral image details, since they are obtained by iteratively blurring and downsampling the original image. The candidate ROI containing the object is computed by merging the overlapping image segments at different scales. The ROI is classified as target object according to a shape regularity test.

The proposed MGS algorithm has been compared with a pixel-feature clustering algorithm. The detection performance of both algorithm are comparable w.r.t. both precision and recall values with a slight advantage for the multiscale graph-based segmentation. Both algorithms achieve performance suitable for the execution of manipulation tasks.

Acknowledgment

The system has been developed within the Italian national Project MARIS (PRIN 2010FBLHRJ_007). The authors would like to thank the divers of Federazione Italiana Attività Subacquee of Parma (FIAS, http://www.fiasparma.it/) for their valuable support in realizing the underwater experiments in Portofino.

References

[1] C. Ancuti, C.O. Ancuti, T. Tom Haber, and P. Bekaert. Enhancing underwater images and videos by fusion. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[2] S. Bazeille, I. Quidou, and L. Jaulin. Color-based underwater object recognition using water light attenuation. *Intel Serv Robotics*, 5:109–118, 2012.

[3] R. Campos, R. Garcia, and T. Nicosevici. Surface reconstruction methods for the recovery of 3D models from underwater interest areas. In *Proc. of the IEEE/MTS OCEANS*, pages 1–10, 2011.

[4] A. Censi, L. Iocchi, and G. Grisetti. Scan Matching in the Hough Domain. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2005.

[5] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, Cambridge, MA, 2001.

[6] R.O. Duda, P. E Hart, and D.G. Stork. *Pattern classification*. John Wiley & Sons, 2004.

[7] R. Eustice, H. Singh, J. Leonard, M. Walter, and R. Ballard. Visually navigating the rms titanic with slam information filters. In *Proceedings of Robotics: Science and Systems*, Cambridge, USA, June 2005.

[8] P.F. Felzenszwalb and D.P. Huttenlocher. Efficient Graph-Based Image Segmentation. *Int. J. Comput. Vision*, 59(2):167–181, sep 2004.

[9] R. Garcia and N. Gracias. Detection of interest points in turbid underwater images. In *Proc. of the IEEE/MTS OCEANS*, pages 1–9, 2011.

[10] J. J. Javier Fernandez, M. Prats, P.J. Sanz, J. C. Garcia, R. Marin, M. Robinson, D. Ribas, and P. Ridao. Grasping for the seabed: Developing a new underwater robot arm for shallow-water intervention. *IEEE Robot. Automat. Mag.*, 20(4):121–130, 2013.

[11] F Kallasi, F. Oleari, M. Bottioni, D. Lodi Rizzini, and S. Caselli. Object detection and pose estimation algorithms for underwater manipulation. In *Advances in Marine Robotics Applications (AMRA) - in International Conference on Autonomous Intelligent Systems (IAS)*, pages 1–7, Jul 2014.

[12] D. Kim, D. Lee, H. Myung, and H.-T. Choi. Object detection and tracking for autonomous underwater robots using weighted template matching. In *Proc. of the IEEE/MTS OCEANS*, pages 1–5, 2012.

[13] D. Lodi Rizzini, F. Kallasi, F. Oleari, and S. Caselli. Investigation of vision-based underwater object detection with multiple datasets. *International Journal of Advanced Robotic Systems (IJARS)*, -(-):–, 2015 (in press).

[14] M. Narimani, S. Nazem, and M. Loueipour. Robotics vision-based system for an underwater pipeline and cable tracker. In *Proc. of the IEEE/MTS OCEANS*, pages 1–6, 2009.

[15] T. Nicosevici, N. Gracias, S. Negahdaripour, and R. Garcia. Efficient three-dimensional scene modeling and mosaicing. *Journal of Field Robotics*, 26(10), 2009.

[16] F. Oleari, F. Kallasi, D. Lodi Rizzini, J. Aleotti, and S. Caselli. Performance Evaluation of a Low-Cost Stereo Vision System for Underwater Object Detection. In *Proc. of the World Congr. of the International Federation of Automatic Control (IFAC)*, 2014.

[17] F. Oleari, F. Kallasi, D. Lodi Rizzini, J. Aleotti, and S. Caselli. An underwater stereo vision system: from design to deployment and dataset acquisition. In *Proc. of the IEEE/MTS OCEANS*, pages 1–5, 2015.

[18] M. Prats, J.C. Garcia, S. Wirth, D. Ribas, P.J. Sanz, P. Ridao, N. Gracias, and G. Oliver. Multipurpose autonomous underwater intervention: A systems integration perspective. In *Mediterranean Conference on Control & Automation*, pages 1379–1384, July 2012.

[19] J.P. Queiroz-Neto, R. Carceroni, W. Barros, and M. Campos. Underwater stereo. In *Computer Graphics and Image Processing, 2004. Proceedings. 17th Brazilian Symposium on*, pages 170–177, 2004.

[20] P.L. Rosin. Unimodal thresholding. *Patter Recognition*, 34(11):2083–2096, 2001.