Water region and multiple ship detection for port surveillance

Document status and date:
Published: 01/01/2012

Document Version:
Accepted manuscript including changes made at the peer-review stage

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

Water Region and Multiple Ship Detection for Port Surveillance

X. Bao, S. Zinger, P. H. N. de With

VCA group, EE Faculty

Eindhoven Univ. of Technol.

5600 MB Eindhoven

the Netherlands

R. Wijnhoven

ViNotion B.V.

Horsten 1

5600 CH Eindhoven

the Netherlands

J. Han

C.W.I

Science Park 123

Amsterdam

the Netherlands

Abstract

In this paper, we present a robust and accurate multiple ship detection system for port surveillance. First, water region is detected using a region-based technique. Second, ships are located by a cabin detector for the same port surveillance sequences. Third, a verification process is performed to remove the false detections of ships using the detected water region as contextual cues. We have analyzed our water region detection algorithm by experimenting on 5 sequences and we have found that it achieves an average pixel classification precision of 96.9% and a recall of 91.8%. The multiple ship detection system is tested on 3 different surveillance sequences. We successfully detect 133 ships out of 150 ships with a precision of 87.5% and a recall of 88.7%.

1 Introduction

Automatic port surveillance is an emerging area for monitoring ship traffic, autonomous ship identification and traffic control for port security systems. Radar technology is commonly used in such systems to detect and track large ships coming from or leaving for the sea. Although radar technology gives accurate detection results, echoes returned from other targets such as the buildings, harbor infrastructure or other ships lead to difficulties for reliable ship detection. Furthermore, small and non-metal ships cannot be detected by a radar system, which causes potential threats to the traffic control and security. In this paper, we develop a multiple ship detection system based on video cameras as a complementary tool for port security systems.

Multiple ship detection is a complex and challenging problem. Techniques developed for road traffic surveillance [5] are not applicable, due to the various environmental influences such as weather conditions and the high variability of the types of ships passing by. Figure 1 shows some examples of the large variation of ships in port surveillance videos. Recently, several algorithms are developed for ship detection [1][3]. Fefilatyev *et al.* combine the segmentation and image registration technique to detect ships. However, the algorithm is largely based on a specific horizon line detection [3] and limited to the surveillance videos captured from a camera mounted on a untethered buoy at open sea. A more general approach is developed based on background registration and morphological operations in [1]. Their approach has no assumptions on the geometric structure of the surveillance video sequences, however, it cannot perform robust detections when there are sudden changes in the background, such as severe illumination changes and disturbances of the water surface.

In our ship detection system, we make use of the fact that ships always travel within the water area in a port surveillance scenario. It is expected that false detections of

Figure 1: Examples of ships with different appearances in surveillance videos.
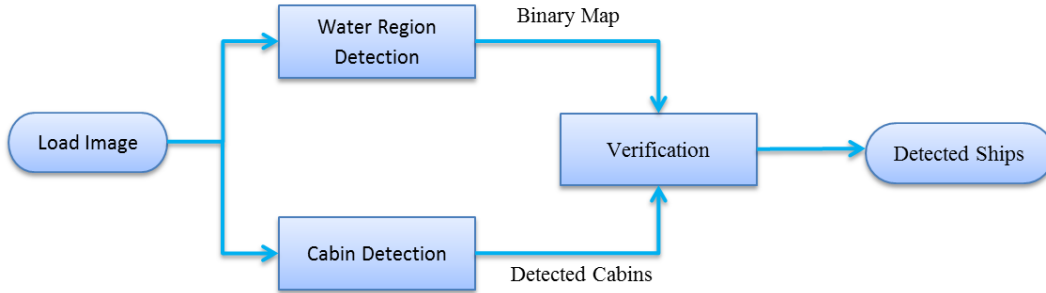

Figure 2: Flowchart of our multiple ship detection system.

ships can be significantly reduced if the water region is *a-priori* known and provided as contextual information. For this reason, our detection system consists of two specific detectors: (1) region-based water area detector, and (2) Histogram Oriented Gradients (HOG) based cabin detector [2][7]. The water region detector explores the appearance model of water at a region level, using RGB color features and generates a binary water map for a single surveillance frame, based on machine learning. For the same frame, a cabin detector is applied to detect the possible regions containing ship cabins. Then, a verification process is performed to remove the false detections produced by the cabin detector, using the pre-detected water region as contextual information. The whole process is illustrated in Figure 2.

The paper is organized as follows. Section 2 presents the main techniques used in the water region and multiple ship detection system. Section 3 presents the results for both water and ship detection. Section 4 draws conclusions and discusses future work.

2 Water Region and Multiple Ship Detection

In this section, we discuss the three main steps in our ship detection system: (1) creating a binary map indicating the water region, based on a two-step water detection algorithm; (2) detecting the ship cabins using an offline-trained cabin detector; (3) verifying the cabin detection results by removing the false detections based on the binary map obtained in (1). The details of the techniques in each step will be explored in the following subsections.

2.1 Water Region Detector

Considering that the appearance of water varies significantly in different situations, we design a region-based water detector, instead of labeling the water directly at pixel
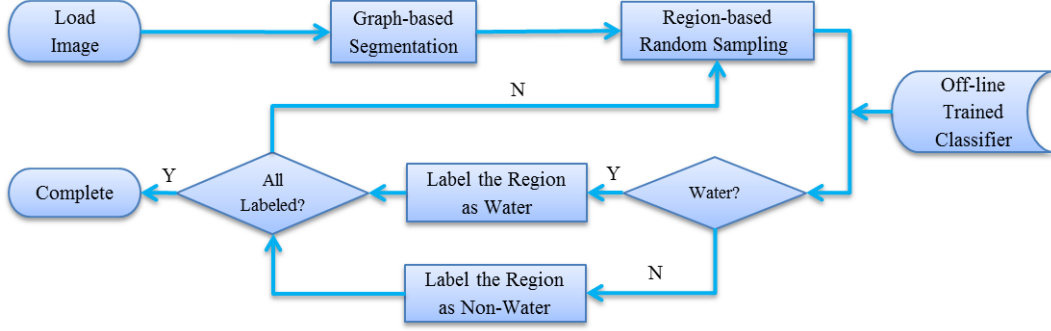
Figure 3: Flowchart of our water region detection algorithm.

level. The algorithm combines a graph-based segmentation with a sampling-based Support Vector Machine (SVM) classification, and involves two stages. First, a graph-based segmentation is applied to segment the surveillance images into perceptually meaningful segments. Second, random sampling is performed to select a certain amount of pixels from each segment. The sampled pixels are classified as water or non-water using an off-line trained SVM. If most of the sampled pixels inside a segment are labeled as water, then the complete segment is labeled as water. The algorithm stops when all segments are processed. The flowchart of the algorithm is depicted in Figure 3.

Efficient graph-based segmentation [4] is employed as the first step in our water region detector to achieve two objectives: (a) distinguish the water region from other objects (sky, vegetation, ships etc.) while preserving the water region as a complete area without over-segmentation; (b) perform fast segmentation to support real-time application in port surveillance systems.

Algorithm. In graph-based segmentation, a key element that defines the criterion of segmentation is the edge weight $w_{i,j}$, which measures the difference between two neighboring pixels i and j in a specified feature space. Considering color as the most important cue to distinguish different objects, the weight $w_{i,j}$ in our approach is defined as:

$$w_{i,j} = \sqrt{\left(\frac{R_i}{L_i} - \frac{R_j}{L_j}\right)^2 + \left(\frac{G_i}{L_i} - \frac{G_j}{L_j}\right)^2 + \left(\frac{B_i}{L_i} - \frac{B_j}{L_j}\right)^2}, \tag{1}$$

where R_i, G_i, B_i are the R, G, B color values of the pixel i (or j when indicated). Likewise, L_i represents the brightness of the pixel i, defined as:

$$L_i = \begin{cases} (R_i + G_i + B_i)/3, & \text{if } R_i + G_i + B_i \neq 0; \\ 1, & \text{otherwise.} \end{cases} \tag{2}$$

In Equation (1), the RGB values are normalized by the corresponding brightness to reduce the influence of brightness. The motivation for this normalization is that parts of the water region with strong disturbances or reflections usually differ considerably from other parts of the water region in terms of brightness [6]. The segmentation based on such normalized color differences will preserve the overall nature of the water region so that it is not over-segmented. This will not only ensure a faster classification in the next step, but also reduce the probability of erroneous labeling of water segments containing water pixels with high brightness values.

After the segmentation, we need to find all segments representing water. In our algorithm, we employ a supervised learning approach to train an SVM classifier offline,

based on a set of representative surveillance images. Again, the RGB values are chosen to construct the feature vector that can discriminate the water and non-water regions.

For each segment C of the image, we randomly select a group of pixels $G(C)$, according to the following criterion:

$$G(C) = \begin{cases} \{\text{randomly sampled pixels, 5\% of the total}\}, & \text{if } N(C) > 2000; \\ \{\text{randomly sampled set of 100 pixels}\}, & \text{otherwise.} \end{cases} \quad (3)$$

In the Equation (3), $N(C)$ is the total number of pixels in segment C. The off-line trained SVM is then applied to each pixel in $G(C)$ and the number of pixels labeled as water are counted as $N_W(G(C))$. We define the probability $P_W(C)$ that the segment C is a part of the water region, giving:

$$P_W(C) = \frac{N_W(G(C))}{N(G(C))}. \quad (4)$$

In this equation, $N(G(C))$ is the total number of sampled pixels in segment C. The segment is then labeled with label $L(C)$ as follows:

$$L(C) = \begin{cases} 1, & \text{if } P_W(C) > 0.6; \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

The binary map indicating water region is generated after all segments are labeled and can be used as contextual information supporting the verification of ship detection.

2.2 Cabin Detector

To perform the initial ship detection, the HOG-based cabin detector from [2] [7] is applied to the images. First, the image is divided in cells of $N \times N$ pixels and an orientation histogram is created for each cell. The gradient is calculated for each pixel and the gradient magnitude is stored in the histogram bin corresponding to the gradient orientation. Each histogram is then normalized to become invariant to contrast changes. To train an ship object detector, the training images of ships and background are first converted to HOG descriptions. Next, a classifier is learned to distinguish ship object samples from background samples. Ship object detection is performed by sliding a detection window over the image and classifying each image position. To obtain invariance to the ship object size, the image is processed at several scales. The output score of the cabin classifier is interpreted as a confidence score for the ship detection. Since a verification will be performed in the next step, the threshold on this confidence score is set with a low value to enable a sensitive ship detection so as to detect as many presented ships as possible, at the cost of introducing false ship detections.

2.3 Verification for Multiple Ship Detection

To design a reliable ship detection system, a verification is performed to remove the false detections generated by the cabin detector. The verification process is based on the intuitive fact that the detected ship regions should contain only a small portion of water pixels. The binary map obtained by the water region detector is applied to the ship detection results to count the number of water pixels $N_W(D)$ within each detected

ship region D. Then, the probability of false detection of a ship $P_F(D)$ in D, is defined as follows:
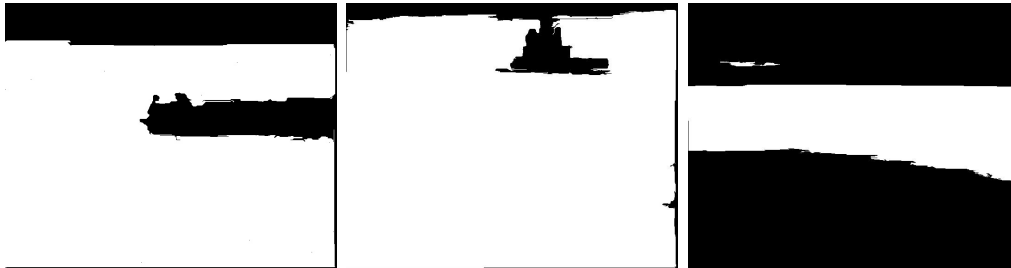
$$P_F(D) = \frac{N_W(D)}{N(D)}, \tag{6}$$

where $N(D)$ denotes the total amount of pixels inside the detected ship region D. The region D is recognized as a false detection if $P_F(D) > 0.65$. The final ship detection results are then obtained by removing all found false detections using the previous criterion for $P_F(D)$.

(a) Original images
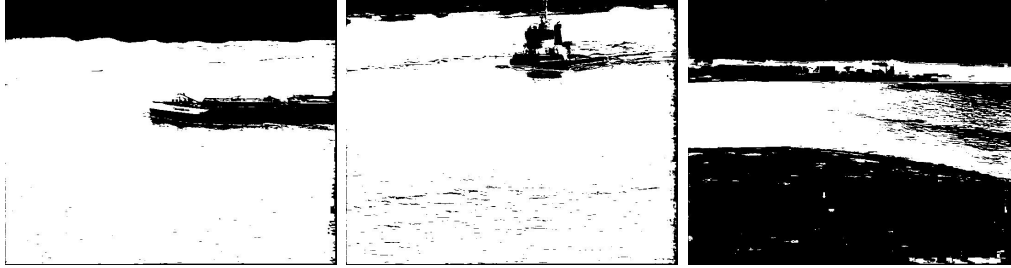
(b) Results of the pixel-based approach

(c) Results of our region-based detection

Figure 4: Water detection results: white color represents the water regions and black indicates non-water regions.

3 Implementation and Experiments

The proposed system is tested for both the performance of water region detection and multiple ship detection. In the test, we have used 8 video sequences recorded in the harbor of Rotterdam, the Netherlands, during daytime. The video sequences are recorded with a PTZ camera, and the camera position and zoom-in factor differ per sequence. The captured video sequences have a Standard-Definition (SD) resolution of 720×576 pixels, and contain between 40 frames and 260 frames each. We select
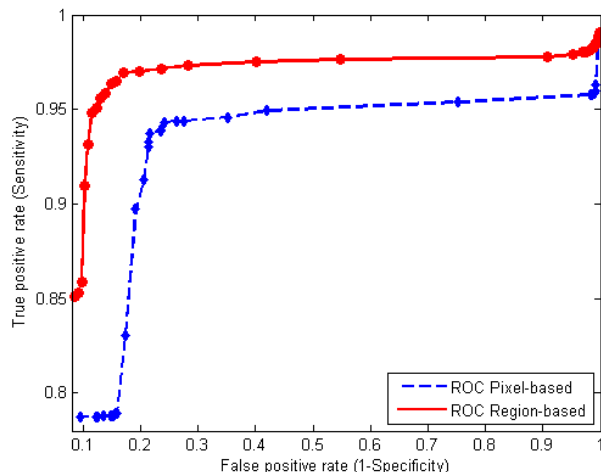
Figure 5: ROC curve depicting the performance of region-based and pixel-based water detection approaches.

60 typical frames from 3 sequences to train a two-cluster SVM classifier (water and non-water) and 100 frames from the other 5 sequences to compose the test set.
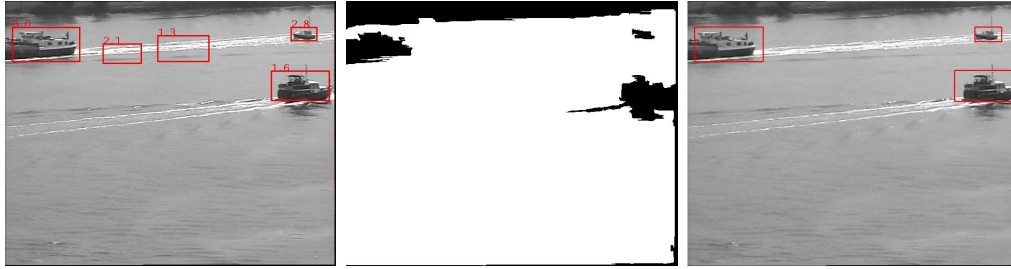
3.1 Results of Water Region Detection

We first measure the performance of water region detection. To indicate the performance of our region-based approach, we have also implemented a pixel-based detection using the same offline trained SVM for comparison. This is reasonable because the trained SVM classifier itself can perform water region detection without pre-processed segmentation. Figure 4 shows some binary maps generated by the two approaches.

When the water region is relatively smooth, as in the left image of Figure 4(a), both approaches can find the correct water region, although the pixel-based approach cannot produce coherent regions. When parts of the water regions have strong disturbances and reflections, as in the middle and right images of Figure 4(b), our region-based algorithm is still able to label those highly disturbed and reflected parts as water. As expected, the pixel-based detection incorrectly labels those parts as non-water. The ROC curve in Figure 5 also illustrates and confirms the better performance of our region-based water detection. Finally, we have measured the recall-precision performance of our region-based water detection by calculating the pixel recall and precision rate for 100 frames from the 5 sequences. We have found that it achieves an average precision of 91.8% and a recall of 96.9%.

3.2 Results of Multiple Ship Detection and Verification

To measure the performance of the multiple ship detection system, we have selected 96 frames from 3 video sequences, which contain 6 different ships. The ships travel across the water regions and appear differently in various subsets of the frames. As a result, the test frames actually contain 150 ships in total. We test the ship detection with only applying the sensitive cabin detector and compare it with the detection when using the contextual water map to perform the verification. Table 1 shows the test results with measurements of true positives, false positives, precision and recall.

(a)

(b)

Figure 6: Examples of multiple ship detection: left column contains the results generated by cabin detector, middle column - binary maps generated by water region detection and right column - results after verification process. Bounding boxes delimit the areas of detected ships.

Table 1: Multi-ship detection results of the algorithm "Cabin" (only cabin detector) and algorithm "Cabin-Water" (cabin detector combined with water region detector).

Algorithm	True Pos.	False Pos.	Precision	Recall
Cabin	140	56	71.43%	93.33%
Cabin-Water	133	19	87.50%	88.67%

The results show that about 67% of the false detections are successfully removed by employing the water map as a contextual cue. Although the recall slightly decreases with about 5%, the precision is largely improved by nearly 17%. Some visual results of multiple ship detection are shown in Figure 6.

4   Conclusion and Future Work

In this paper, we have presented a multiple ship detection system designed for harbor surveillance. Our system combines water region detection with a cabin detector to achieve robust and accurate results. The water region is detected using a technique combining segmentation with region-based classification. Segmentation is performed to separate water regions from other objects in the image while preserving the overall characterization of it. Then, a random sampling is applied for each segment and an SVM-based classification is performed to generate a binary map indicating water regions. In our experiments, water detection at region level has proven to be robust

and accurate, with a precision of 91.8% and a recall of 96.9%.

The results of water region detection are then provided as contextual information to a cabin detector. A verification step has been designed to reduce the false positive rate of the cabin detector by removing the detected regions that contain mostly water pixels. The experiments show that using detected water regions as contextual information further improves the reliability of multiple ship detection by removing 67% of the false detections at the cost of 5% decrease in the recall.

In the future, we will explore the fusion of multiple features for both segmentation and SVM classification in a water region detector. Since the appearance of water region varies significantly in different conditions, a single cue (color in our approach) is not sufficiently robust to segment and classify the water area. It is expected that the robustness of water detection will be increased if we apply multiple cues so that the performance of ship detection will be improved simultaneously.

References

[1] N. Arshad, K. Moon, and J. Kim, *Multiple ship detection and tracking using background registration and morphological operations*, FGIT-SIP/MulGraB, vol. 123, Springer Berlin Heidelberg, 2010, pp. 121–126.

[2] N. Dalal and B. Triggs, *Histograms of oriented gradients for human detection*, in Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (San Diego, CA, USA), vol. 1, june 2005, pp. 886 –893 vol. 1.

[3] S. Fefilatyev, V. Smarodzinava, L.O. Hall, and D.B. Goldgof, *Horizon detection using machine learning techniques*, International Conference on Machine Learning and Applications (Orlando, FL, USA), 2006, pp. 17–21.

[4] P. F. Felzenszwalb and D. P. Huttenlocher, *Efficient graph-based image segmentation*, International Journal of Computer Vision **59** (2004), 167–181.

[5] F. Porikli, *Achieving real-time object detection and tracking under extreme conditions*, Journal of RealTime Image Processing **1** (2006), no. 1, 33–40.

[6] A. Rankin, L. Matthies, and A. Huertas, *Daytime water detection by fusing multiple cues for autonomous off-road navigation*, in Proc. Army Science Conference (Orlando, FL, USA), 2004.

[7] R. Wijnhoven, K. van Rens, E. G. T. Jaspers, and P. H. N. de With, *Online learning for ship detection in maritime surveillance*, Symposium on Information Theory in the Benelux (Rotterdam, The Netherlands), 2010, pp. 73–80.