

Article

# Coarse-Fine-Stitched: A Robust Maritime Horizon Line Detection Method for Unmanned Surface Vehicle Applications

Yuan Sun <sup>1,\*</sup> and Li Fu <sup>2</sup>

<sup>1</sup> School of Electronics Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

<sup>2</sup> Beijing Institute of Aerospace Control Devices, Beijing 100094, China; li.fu@buaa.edu.cn

\* Correspondence: sunyuan@bupt.edu.cn; Tel.: +86-188-1031-1695

Received: 17 July 2018; Accepted: 24 August 2018; Published: 27 August 2018



**Abstract:** The horizon line has numerous applications for an unmanned surface vehicles (USV), such as autonomous navigation, attitude estimation, obstacle detection and target tracking. However, maritime horizon line detection is quite a challenging problem. The pixel points of the horizon line features are far fewer than the pixel points of the entire image, on the one hand. Conversely, the detection results might be impacted negatively by the complex maritime environment, waves, light changing, and partial occlusions due to maritime vessels or islands, for example. To solve these problems, a robust horizon line detection method named coarse-fine-stitched (CFS) is proposed in this paper. First, in the coarse step of CFS, a line segment detection approach using gradient features is applied to build a line candidate pool, which probably contains many false detection results. Then, hybrid feature filtering is designed to pick the horizon line segments from the pool in the fine step. Finally, the fine line segments are stitched to obtain the whole horizon line based on random sample consensus (RANSAC). Using real data in the maritime environment, the experimental results demonstrate the effectiveness of CFS, compared to the existing methods in terms of accuracy and robustness.

**Keywords:** horizon line detection; unmanned surface vehicle; maritime application; robust detection; vision sensor

## 1. Introduction

Currently, with the rapid development of artificial intelligence technology, unmanned surface vehicles (USVs) have become more and more important in maritime systems [1–3]. Compared with manned surface vehicles, USVs are more easily manipulated and are more adaptable to the environment with lower risk to staff. USVs have been applied in various military and civilian missions, such as science, mapping, defense and general robotics research [4–6]. The vision sensors equipped on the USVs usually are used to perceive surrounding information, which plays an important role in guaranteeing the safety and efficiency of the USVs without human intervention.

The horizon line is an important information source for USVs. When considering real applications, the maritime horizon line can be utilized for autonomous navigation, attitude estimation, obstacle detection, and target tracking, for example [7,8]. However, maritime horizon line detection is quite challenging due to the following two points. First, the pixel points of the horizon line features are far fewer than the pixel points of the entire image; and second, horizon line detection suffers from the complex maritime environment, waves, light changing and partial occlusions by maritime vessels or islands, for example. Sometimes, the horizon line in maritime application is blurry since the image of the sky and the sea is similar.

During the last few decades, horizon line detection has been attracting more attention in the community of computer vision and USV applications. The existing research can be divided into methods using local features and methods using global features.

The local features methods usually regard the horizon line as a line segment using local features, such as edge detection. Kim et al. proposed a horizon line detection method with the combination of Canny edge detection and the Hough transform [9]. However, the Hough transform is time-consuming, which might not meet the real-time requirement in maritime applications. Tang et al. used the Radon transform for horizon detection for infrared ship detection applications [10], but the result might be influenced by false line segments as the Radon transform cannot determine the endpoints of line segments. A detailed comparison of methods using local features for horizon line detection in maritime images is shown in Libe's work [11]. Prasad et al. presented a horizon line detection method named MuSCoWERT [12]. First, their method detects the long linear features consistent over multiple scales using multi-scale median filtering of the image. Then, the Radon transform on a weighted edge map is applied to detect linear features. Prasad et al. presented another multi-scale approach, MSCM-LiFe [13]. Using that method, the Hough transform, and intensity gradients are used to find the line feature candidates. Although these methods using local features are effective for line segment extraction, the major shortcoming is that they are unable to distinguish the horizon line from the similar line segments in the sea image.

The methods in the other category use global features to calculate an optimization criterion for different candidate line positions and orientations using the features of the whole image. Wang et al. proposed a sea-sky line detection method based on global gradient saliency [14], but it still suffers from false detection when some straight lines are in the image, such as waves. Gershikov analyzed the performance of horizon line detection using color features and showed that color information can be justified when smaller angular and height deviations are of primary importance [15]. However, there is little discussion of the problem of partial occlusion. Segmentation methods are also used to obtain the horizon line by segmenting the image into the sea/water and the sky/ground. Scherer et al. proposed a water detection method from a flying robot [16] where the onboard inertial measurement unit (IMU) is used to obtain the horizon line in order to obtain a color distribution of the regions below and above the horizon. The image is then segmented to extract the water region using the classifier. Fefilyatyev et al. proposed a horizon line detector for the recognition and tracking of marine vehicles in videos [17]. Prior to trying to find the horizon line, their method generates a preliminary segmentation of objects by creating binary maps where the location of a possible target object is defined by non-zero elements. A statistical horizon line detection algorithm is then proposed, which attempts to minimize the intra-class variance of the sea/ground and sky distributions. However, the fundamental drawback of that method is that it approximates the edge of water by a horizon line and cannot handle situations in coastal waters [18]. Kristan et al. addressed the problem of online detection by constrained unsupervised segmentation [18]. To achieve this, a semantic segmentation model (SSM) is proposed for structurally constrained semantic segmentation with application to USV obstacle-map estimation.

More recently, a deep neural network has been applied for horizon line detection. Jeong et al. proposed a horizon detection method for maritime scenes using a scene parsing network [19]. They also proposed a horizon detection method that combines a multi-scale approach and a convolutional neural network [20]. However, the generalization of these methods still needs to be further investigated.

The aim of this paper is to propose a robust maritime horizon line detection method, coarse-fine-stitched (CFS), for USV applications. CFS is mainly composed of three steps. The first step of CFS is the coarse detection, in which a line segment detection approach in which gradient features are applied to extract all the line segments in the image to build a line candidate pool. Although there are a few missing detections in the candidate pool, it probably contains many false detection results due to the background environment. To solve this problem, using the fine step of CFS, hybrid feature filtering is designed to select the horizon line segments from the pool. To improve the performance

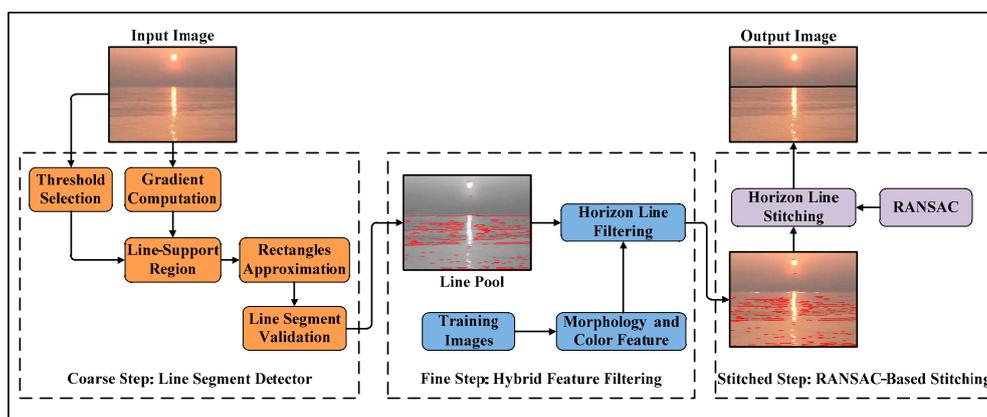
of the fine step, the distinguished features of the horizon line segments are modeled to build hybrid feature filtering, including color and morphology. Finally, a line segment stitching method based on random sample consensus (RANSAC) is proposed to obtain the whole horizon line. Considering the linear relationship of the horizon line segments, the RANSAC method is used to stitch the true results and remove the negative line segments, which can improve the robustness of this approach in a complex maritime environment.

The remainder of this paper is organized as follows. The details of the proposed CFS method are presented in Section 2. Section 3 discusses the experiments used to demonstrate the practical utility of this approach. Finally, the conclusions are shown in Section 4.

## 2. Coarse-Fine-Stitched Method

### 2.1. Framework

The framework of the proposed coarse-fine-stitched (CFS) method for unmanned surface vehicle (USV) applications is presented in Figure 1. It operates in a three-step way. During the coarse step, the image gradient features are exploited to obtain all the lines of the image, which are all candidates for horizon line segments. In this step, a problem-specific design line segment detection (LSD) algorithm is proposed to extract all the candidates with a short processing time, generating a line pool. Although the coarse detection method can obtain the horizon line candidates of the image, some non-horizon line segments could be selected into the pool. Therefore, in the fine second step of this method, a horizon line filter is designed to eliminate the non-horizon lines from the pool by using hybrid feature filtering, which was designed based on the morphological and color features of the horizon line. Finally, to achieve the accurate identification of the horizon line, the horizon line segments are stitched in this step. Specifically, a random sample consensus-based (RANSAC-based) line segment stitching method is proposed to obtain the whole horizon line, which can improve the robustness of this approach in a complex maritime environment.



**Figure 1.** The framework of the coarse-fine-stitched (CFS) method for unmanned surface vehicle (USV) applications. The first step of the method is a problem-specific design line segment detector, the second step is a horizon line filter, and the third step is a random sample consensus-based (RANSAC-based) line segment stitching method.

### 2.2. Coarse Detection

Line features are one of the most important features that determine the effectiveness of the coarse-fine-stitched (CFS) method. von Gioi et al. presented a line segment detection (LSD) algorithm which gave accurate results with fast computational speed [21]. The LSD algorithm extracts line segments from an image in three steps: (1) Line-support region: Group connected pixels with gradient values exceeding a threshold into line-support regions by applying a region-growing algorithm;

- (2) Rectangle approximation: Find the rectangle that best approximates each line-support region;
- (3) Line segment validation: Validate each rectangle based on a contrary method, which counts the number of aligned points (points with a gradient direction approximately orthogonal to the line segment) and finds the line segments as outliers.

However, the direct application of the LSD algorithm is not appropriate for the horizon line detection problem for the following two reasons.

(1) LSD cannot distinguish the horizon line from other line segments because it just focuses on detecting line segments [22–24]. A line that is not the horizon line also will be obtained. Figure 2b gives the result of using LSD for a given image, lines of the island, the clouds and the waves in the image are also extracted, causing a false positive. Therefore, to accurately extract the horizon line from a given image, a filter method should be considered to eliminate non-horizon line segments from all line segments.



**Figure 2.** The direct application of the line segment detection (LSD) algorithm with false positive: (a) A given image with a small island, clouds and waves; (b) Line segments of the island, the clouds and the waves in the image also are extracted by the LSD algorithm, causing a false positive.

(2) LSD ignores the horizon line that is background-like, which might cause a significant false negative. Figure 3b shows the horizon line is totally ignored. During the first step of the LSD algorithm, to improve the exactness of line segment detection, a gradient threshold is selected to eliminate pixels with a small gradient magnitude because the orientation error cannot be ignored for a small gradient magnitude [25], which can create a linear pattern and thereby cause false detections. The threshold on the gradient magnitude in the LSD algorithm is set as [21]:

$$T'_g = \frac{2}{\sin 22.5^\circ} = 5.3 \quad (1)$$

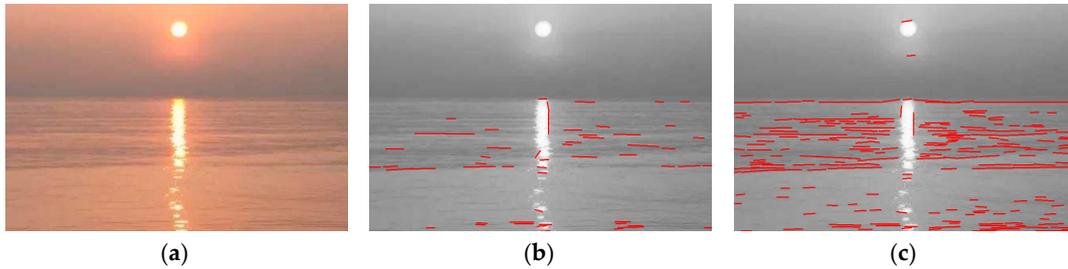
Pixels with a gradient magnitude smaller than the threshold will not be regarded as part of any line segment. However, in this horizon line detection task, the horizon line obtained by the USV might be very background-like, due to vapor or illumination, for example. Their corresponding pixels with gradient magnitude might be under the threshold often. This leads to the horizon line being ignored and causes high false negatives. Figure 3b shows the horizon line is ignored totally in this step, since the gradient magnitude of the pixels do not exceed  $T'_g$ .

Considering that false positive can be processed further in the next step, a smaller threshold is set to avoid ignoring pixels with background-like horizon lines in this step. Demonstrated in the analysis above and inspired by the LSD algorithm, the LSD framework is applied and then the current method focuses on how to redesign the gradient threshold in the first step of the algorithm. A problem-specific LSD design then is proposed to obtain all the lines of a given image.

Regarding the gradient threshold selection problem, an intensity-based method is applied to set the threshold [26]:

$$T_g = \frac{\mu_{mean}}{\mu_{max}} \quad (2)$$

where  $\mu_{mean}$  is the mean gradient value of the image, and  $\mu_{max}$  is the maximum of the mean gradient value of the grid images. The authors divided the image into an  $8 \times 8$  grid. It represents an adaptive threshold that depends on the gradient value of the image. It is evident that  $T_g \leq 1 \leq T'_g$ , thus, with the smaller threshold, pixels of the background-like horizon line segments will not be ignored in the first step of this method, as shown in Figure 3c.



**Figure 3.** The direct application of the LSD algorithm with a false negative: (a) A given image with a horizon line that is very background-like due to vapor and illumination; (b) The horizon line is ignored totally by the LSD algorithm, causing a false negative; (c) The results of this problem-specific design LSD for coarse detection.

The problem-specific LSD design is able to extract all the lines of a given image and generate a line pool,  $C_h$ , which is composed of all candidates for horizon line segments, denoted as  $l_1, l_2, \dots, l_{N_c}$ , where  $N_c$  is the total number of line segments obtained by the coarse detection method, resulting in  $\{l_i | l_i \in C_h, i \in [1, N_c]\}$ .

### 2.3. Fine Detection

Subsequent to the process of the first step, a line pool  $C_h$  is obtained. However, some elements of  $C_h$  might be background noise causing a sufficient false positive. To avoid the interference of the non-horizon line, a horizon line filter is applied to suppress the noise and extract horizon line segments from the line pool  $C_h$ . The non-horizon line is filtered out from the pool according to the distinguished features of the horizon line segments, including color and morphology.

#### 2.3.1. Morphology Feature

The morphology feature of the horizon line includes the length and direction properties, as well as others [7]. However, in some environments, the horizon line detection might be obscured partially by maritime vessels or islands, as shown in Figure 2a. Thus, the length property would not perform well for horizon line detection. A new direction property is used for fine detection in this paper.

Denote the equation of the line horizon segment  $l_i$  as:

$$l_i : y - k_i x - b_i = 0 \quad (3)$$

where  $k_i$  and  $b_i$  are the gradient and the intercept of the line horizon segment  $l_i$ , respectively.

Denoting the direction of the horizon line segment  $l_i$  as  $d_i$ :

$$d_i = \tan^{-1} k_i \quad (4)$$

Since the camera system is usually fixed on the unmanned surface vehicle (USV), the direction of the horizon line obtained by the camera system is approximately equal to the roll angle of the USV  $\beta$ . When used in real applications, the angle error between the roll angle of the USV and the direction of the horizon line is the alignment error, which can be measured earlier, denoted as  $e$ . Subsequently, the result is:

$$|d_i - \beta - e| \leq \delta \quad (5)$$

where  $\delta$  is a very small angle value, which is caused by other factors, such as structural deformation.

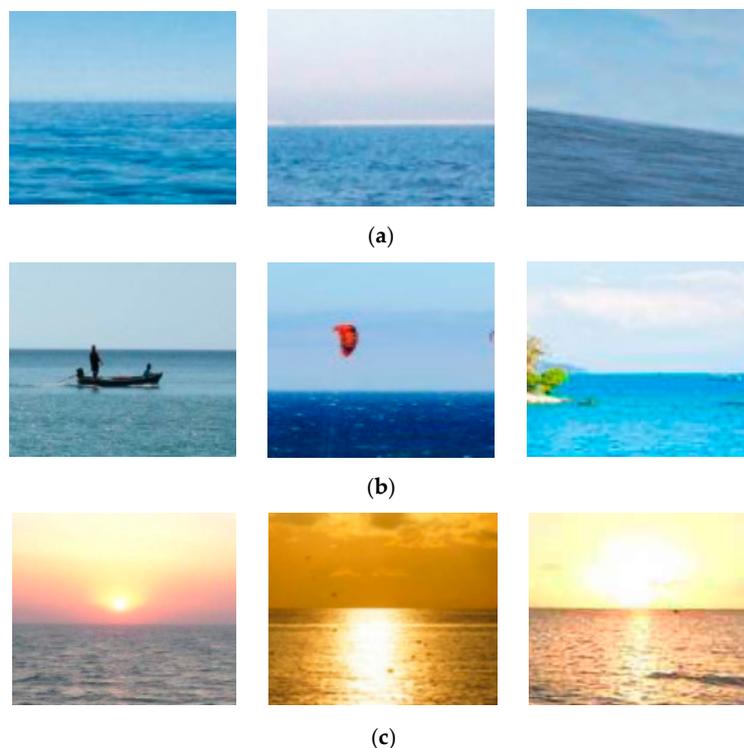
Denote the maximum value of the roll angle as  $\beta_{\max}$ :

$$-\delta - \beta_{\max} + e \leq d_i \leq \delta + \beta_{\max} + e \quad (6)$$

$\beta_{\max}$  is set as  $30^\circ$ , and the USV does not risk capsizing.

### 2.3.2. Color Feature

The color feature is another essential feature to distinguish the horizon line segments from other line segments. Since the horizon line is the apparent edge that separates sea from sky, the greater the color feature difference of the sea and the sky, the easier it is to distinguish the horizon line segments and other line segments. Figure 4 shows the image examples containing the color feature of the horizon line segments. The color feature between the two sides of the horizon line segments is distinguished.

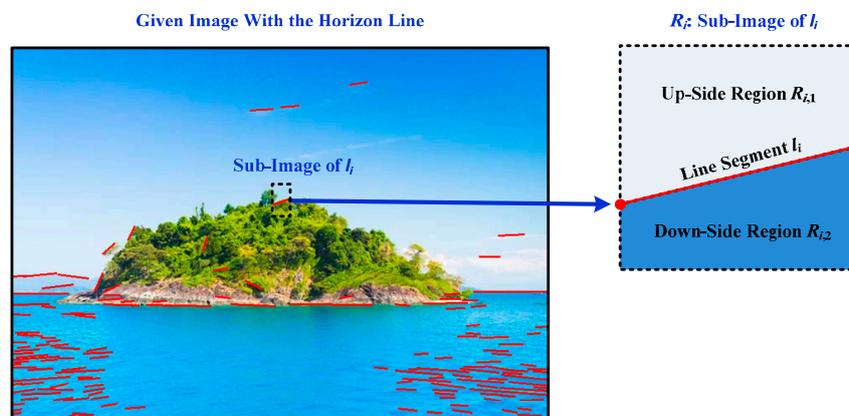


**Figure 4.** The color feature of the horizon line segments: (a) The image examples contain the horizon line with blue sea and white/blue sky; (b) The image examples contain the horizon line with blue sea, white/blue sky and other objects, such as boat, bird, and island; (c) The image examples contain the horizon line with the setting sun scene.

First, there is a notable color difference between the two sides of the horizon line segments. The greater the color feature variance of the two sides of a horizon line segment, the easier it is to distinguish the horizon line segments and other line segments. Second, the color of each side of the horizon line segment is composed of several categories, for example, the color of the up-side of a horizon line segment is probably blue, white, gray, red, yellow, and more. The color of the down-side of the horizon line segment is probably blue, gray, etc. Third, on the same side of the horizon line segment, the color and texture are roughly the same and, the closer to the horizon line segment, the more important it is to distinguish the result.

Given an image with a horizon line, the line segment pool is obtained through the coarse detection step. The sub-image of line segment  $l_i$  is extracted for fine detection, denoted as  $R_i$ .

The sub-image with a line segment detected by the coarse detection step is shown in Figure 5. The up-side region and down-side region of line segment  $l_i$  is denoted as  $R_{i,1}$  and  $R_{i,2}$ , respectively, with the sub-image of  $l_i$ , the color feature is calculated to identify whether it is the horizon line segment.



**Figure 5.** Given an image with a horizon line, the line segment pool is obtained through the coarse detection step. The sub-image of line segment then is extracted for fine detection, whose color feature is calculated to identify whether it is the horizon line segment.

Concerning the current method, the first three color moments of the sub-image of  $l_i$  are used for fine detection, which characterize the color distribution of sub-images. It is scaling and rotation invariant. Moreover, since color moments encode both shape and color information, they are a good feature to use under changing lighting conditions [27]. The first three color moments of sub-image  $R_i$  are obtained as follows:

Mean

The first color moment of region  $R_{i,1}$  and region  $R_{i,2}$  can be calculated as the average color in the image region:

$$\mu_{k,1}^i = \frac{1}{N_{i,1}} \sum_{j=1}^{N_{i,1}} p_{k,1,j}^i \quad (7)$$

$$\mu_{k,2}^i = \frac{1}{N_{i,2}} \sum_{j=1}^{N_{i,2}} p_{k,2,j}^i \quad (8)$$

where  $N_{i,1}$  and  $N_{i,2}$  are the number of pixels of region  $R_{i,1}$  and region  $R_{i,2}$ , respectively.  $p_{k,1,j}^i$  and  $p_{k,2,j}^i$  are the value of the  $j$ th pixel of region  $R_{i,1}$  and region  $R_{i,2}$  at the  $k$ th color channel, respectively. The color image is processed and  $k = 1, 2, 3$ , corresponding to the three color channels of the image.

Standard Deviation

The second color moment is the standard deviation. The second color moment of region  $R_{i,1}$  and region  $R_{i,2}$  can be obtained by taking the square root of the variance of the color distribution:

$$\sigma_{k,1}^i = \sqrt{\frac{1}{N_{i,1}} \sum_{j=1}^{N_{i,1}} (p_{k,1,j}^i - \mu_{k,1}^i)^2} \quad (9)$$

$$\sigma_{k,2}^i = \sqrt{\frac{1}{N_{i,2}} \sum_{j=1}^{N_{i,2}} (p_{k,2,j}^i - \mu_{k,2}^i)^2} \quad (10)$$

### Skewness

The third color moment is the skewness. It measures how asymmetric the color distribution is and, thus, it gives information about the shape of the color distribution. Skewness of region  $R_{i,1}$  and region  $R_{i,2}$  can be computed with the following formula:

$$s_{k,1}^i = \sqrt[3]{\frac{1}{N_{i,1}} \sum_{j=1}^{N_{i,1}} (p_{k,1,j}^i - \mu_{k,1}^i)^3} \quad (11)$$

$$s_{k,2}^i = \sqrt[3]{\frac{1}{N_{i,2}} \sum_{j=1}^{N_{i,2}} (p_{k,2,j}^i - \mu_{k,2}^i)^3} \quad (12)$$

Using Equation (7) to Equation (12), the color feature vector of region  $R_{i,1}$  and region  $R_{i,2}$  can be obtained as the nine-dimension vectors:

$$c_{i,1} = [\mu_{1,1}^i, \mu_{2,1}^i, \mu_{3,1}^i, \sigma_{1,1}^i, \sigma_{2,1}^i, \sigma_{3,1}^i, s_{1,1}^i, s_{2,1}^i, s_{3,1}^i]^T \quad (13)$$

$$c_{i,2} = [\mu_{1,2}^i, \mu_{2,2}^i, \mu_{3,2}^i, \sigma_{1,2}^i, \sigma_{2,2}^i, \sigma_{3,2}^i, s_{1,2}^i, s_{2,2}^i, s_{3,2}^i]^T \quad (14)$$

Using the color feature vectors, the color feature difference between region  $R_{i,1}$  and region  $R_{i,2}$  can be calculated as:

$$d(c_{i,1}, c_{i,2}) = \sum_{k=1}^3 (w_\mu |\mu_{k,1}^i - \mu_{k,2}^i| + w_\sigma |\sigma_{k,1}^i - \sigma_{k,2}^i| + w_s |s_{k,1}^i - s_{k,2}^i|) \quad (15)$$

where  $w_\mu$ ,  $w_\sigma$  and  $w_s$  are the weights for each of the three color moments used.

The color feature difference between region  $R_{i,1}$  and region  $R_{i,2}$  will be used to compute a difference score. Considering that the horizon line is the apparent edge that separates sea from sky, the value of  $d(c_{i,1}, c_{i,2})$  should be larger than threshold  $T_{d,1}$ :

$$d(c_{i,1}, c_{i,2}) \geq T_{d,1} \quad (16)$$

Since the color of each side of the horizon line segment is composed of several categories, the color feature difference between region  $R_{i,1}$  (or region  $R_{i,2}$ ) and the training data is calculated, which should be smaller than threshold  $T_{d,2}$ :

$$d(c_{i,m}, c_{i,m,t}) \leq T_{d,2} \quad (17)$$

where  $m = 1, 2$ , corresponding to the up-side region  $R_{i,1}$  and down-side region  $R_{i,2}$ , respectively.

The training data is composed of 1000 images including the horizon line with different illumination, rain, fog, and occlusion, for example. The authors have  $T_{d,1} = 1.5$ , and  $T_{d,2} = 1$  used in these experiments.

To improve the effectiveness of the coarse-fine-stitched (CFS) method, the up-side region  $R_{i,1}$  and down-side region  $R_{i,2}$  are divided into a series of sub-regions, as shown in Figure 6.

Similar to the color moments of region  $R_{i,1}$  and region  $R_{i,2}$ , the color feature vector of each sub-region can be calculated as:

$$c_{i,1,p} = [\mu_{1,1,p}^i, \mu_{2,1,p}^i, \mu_{3,1,p}^i, \sigma_{1,1,p}^i, \sigma_{2,1,p}^i, \sigma_{3,1,p}^i, s_{1,1,p}^i, s_{2,1,p}^i, s_{3,1,p}^i]^T \quad (18)$$

$$c_{i,2,p} = [\mu_{1,2,p}^i, \mu_{2,2,p}^i, \mu_{3,2,p}^i, \sigma_{1,2,p}^i, \sigma_{2,2,p}^i, \sigma_{3,2,p}^i, s_{1,2,p}^i, s_{2,2,p}^i, s_{3,2,p}^i]^T \quad (19)$$

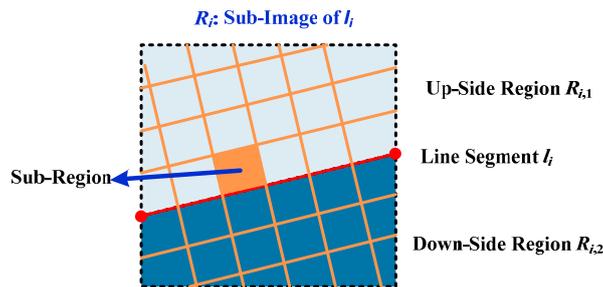
where  $c_{i,1,p}$  and  $c_{i,2,p}$  are  $p$ th sub-regions of region  $R_{i,1}$  and region  $R_{i,2}$ , respectively.

Denote the color feature difference of two sub-regions on the same side as  $d(c_{i,m,p}, c_{i,m,q})$ . As on the same side of the horizon line segment the colors are roughly the same, then the color feature differences of the two sub-regions on the same side should satisfy:

$$\text{var}(d(c_{i,m,p}, c_{i,m,q})) \leq T_{d,3} \quad (20)$$

where  $\text{var}(d(c_{i,m,p}, c_{i,m,q}))$  is the variance of  $d(c_{i,m,p}, c_{i,m,q})$  on the same side of the horizon line segment. The threshold  $T_{d,3}$  is set as 1.

Corresponding to the discussion in this section, hybrid feature filtering is designed and the line segments of the line pool that satisfy Equation (6), Equation (16), Equation (17) and Equation (20) are picked as the horizon line segments for the next step of CFS.



**Figure 6.** The up-side region  $R_{i,1}$  and down-side region  $R_{i,2}$  are divided into a series of sub-regions.

#### 2.4. Robust Stitching

Through the fine detection step of the coarse-fine-stitched (CFS) method, line segments are picked from the line pool as the horizon line segments, denoted as  $\tilde{l}_1, \tilde{l}_2, \dots, \tilde{l}_{N_f}$ , where  $N_f$  is the total number of line segments obtained by the fine detection method. Since these candidate horizon line segments have already passed the hybrid feature filtering, most of them have the typical characteristics of the horizon line. However, there are still two problems that need to be resolved to obtain the horizon line result for unmanned surface vehicle (USV) applications.

First, some parts of the horizon line might be missing due to the occlusion of islands and vessels, illumination, or fog, for example. Using this scenario, the horizon line is divided into several pieces of horizon line segments, rather than a complete line.

Second, although most of the line segments are the horizon line segments through fine detection, there inevitably might be some non-horizon line segments, which will have a negative impact on the final result.

To address these challenges, a random sample consensus-based (RANSAC-based) line segment stitching method is proposed to obtain the whole horizon line. Found in real applications, the horizon line is a straight line. Even if it is divided into several pieces, there is still a linear relationship between them. Therefore, a natural idea is that the horizon line can be obtained through straight line fitting, which is used to estimate the best line passing through the candidate line segments after fine line detection. Considering the fine step, it is difficult to completely exclude a false positive, therefore a RANSAC algorithm is applied in this paper for robust line fitting. The advantage of RANSAC is the ability to use redundant consistency to fit the correct results and to reduce the effect of a false positive on the final results [28].

Denote the equation of the horizon line  $L$  as:

$$L : y - kx - b = 0 \quad (21)$$

where  $k$  and  $b$  are the gradient and the intercept of the line horizon  $L$ , respectively the parameters  $k$  and  $b$  are to be determined as follows.

Denote the equation of the line segment  $\tilde{l}_i$  as:

$$\tilde{l}_i : y - \tilde{k}_i x - \tilde{b}_i = 0 \quad (22)$$

where  $i \in [1, N_f]$ ,  $\tilde{k}_i$  and  $\tilde{b}_i$  are the gradient and the intercept of the line horizon segment  $\tilde{l}_i$ , respectively. The two end points of the line segment  $\tilde{l}_i$  are  $[x_{i1}, y_{i1}]^T$  and  $[x_{i2}, y_{i2}]^T$  in image coordinates, and the position of its midpoint is  $[\bar{x}_i, \bar{y}_i]^T = [(x_{i1} + x_{i2})/2, (y_{i1} + y_{i2})/2]^T$ .

The stitching method is used to solve the two parameters in Equation (21) by minimizing the total mean distance between the line segment  $\tilde{l}_1, \tilde{l}_2, \dots, \tilde{l}_{N_f}$  and the horizon line results:

$$\min_{k,b} \left\{ \sum_{i=1}^{N_f} \|l_i\| d_i \right\} \quad (23)$$

where  $d_i$  is the distance between the midpoint  $[\bar{x}_i, \bar{y}_i]^T$  of line segment  $\tilde{l}_i$  and the horizon line results:

$$d_i = \frac{\bar{y}_i - k\bar{x}_i - b}{\sqrt{k^2 + 1}} \quad (24)$$

and  $\|l_i\| = \sqrt{(x_{i1} - x_{i2})^2 + (y_{i1} - y_{i2})^2}$  is the weight of the distance, which means the result is intended to choose the long-line segment candidates as the final results.

This illustrated that the optimal function is a continuous convex function, which can be solved easily by the gradient descent method [29]. Using all line segment candidates, the horizon line detection result can be solved, denoted as  $L_0$ .

Although most of the line segments are horizon line segments through fine detection, there inevitably might be some non-horizon line segments, which will have a negative impact on the final result. To solve this problem, RANSAC is applied for robust stitching. Batches of line segment candidates are randomly selected to solve the equation of the horizon line, denoted as  $L_j$ , which is the  $j$ th iteration of RANSAC.

The principle of the RANSAC-based line segment stitching method is shown in Figure 7.  $L_0$  is the line fitting result using all line segment candidates and  $L_j$  is the RANSAC-based line fitting. The distance between the midpoint of the line segment candidate  $\tilde{l}_j$  and  $L_j$  is larger than the threshold, which is marked as an outlier and excluded for stitching.

Looking at the batch, each distance between the midpoint of the line segment candidate and  $L_j$  is calculated. When the distance is smaller than a threshold, the line segment is marked as an in-line candidate; otherwise, the line segment is marked as an outlier candidate. The batches of line segment candidates are randomly selected to obtain a batch with the largest number of inline candidates.

The main procedures of the CFS method are summarized as follows:

#### Step 1. Coarse detection

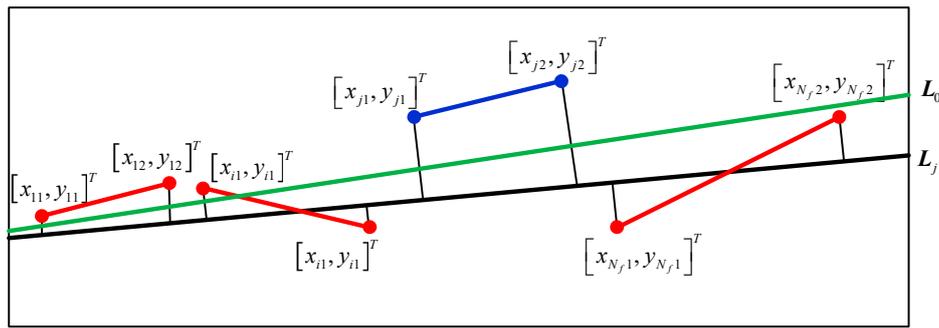
- 1.1 Calculate gradient threshold  $T_g$  using Equation (2)
- 1.2 Implement LSD algorithm with  $T_g$  and finding the line pool  $C_h$

#### Step 2. Fine detection

- 2.1 Calculate the direction of the line segments in  $C_h$  using Equation (4)
- 2.2 Calculate the color feature vector of the sub-images using Equations (13)–(14)
- 2.3 Calculate the color feature vector of the sub-regions using Equations (18)–(19)
- 2.4 Select the line segments of  $C_h$  that satisfy Equations (6), (16), (17) and (20)

#### Step 3. Robust stitching

- 3.1 Execute RANSAC and solve Equation (23)



**Figure 7.** RANSAC-based line segment stitching method.  $L_0$  is the line fitting results using all line segment candidates and  $L_j$  is the RANSAC-based line fitting.

### 3. Results and Discussion

Three separate experiments were designed to evaluate the practical utility of the coarse-fine-stitched (CSF) method. The first experiment was to evaluate the performance of the CFS method for images with different backgrounds. The second experiment was the parameter sensitivity analysis. The third experiment was to test the performance of the method compared with existing methods in terms of accuracy and time-consumption.

The experiments were conducted using a PC with Intel Core i3-6006U CPU 2.00 GHz and 4 GB memory. The testing data was composed of 500 testing images, the public dataset Singapore Maritime Dataset (SMD) [13], and the public dataset Marine Obstacle Detection Dataset (MODD) [18]. The 500 testing images were acquired using a digital camera (Sony-DSC-T70) in Weihai City, China, in July, 2017. The resolution was  $640 \times 800$  pixels, with different backgrounds. Half of the images contained mist or fog and the other half contained relatively clear horizon lines. Additionally, to test the performance of this method, the public dataset SMD and MODD were used for comparison. The training images were composed of more than 1000 images with the horizon line downloaded from <http://image.baidu.com/>, with different illumination, rain, fog, and occlusion, etc.

#### 3.1. Coarse-Fine-Stitched Detection Results with Different Backgrounds

High visual complexity made it difficult to detect the horizon line that was not salient. The authors divided the testing image based on its clutter measure into  $K$  sub-images [30]. The measure was based on the mean and standard deviation ( $\mu, \sigma$ ) of the intensity distribution of the pixels:

$$\mu_n = \frac{1}{N} \sum_{(x,y) \in S_n} I(x,y) \quad (25)$$

where  $I$  was a given image,  $(x,y)$  indicated the pixel index.  $N$  was the total number of pixels in sub-image  $S_n$ .

$$\sigma_n = \sqrt{\frac{1}{N-1} \sum_{(x,y) \in S_n} (I(x,y) - \mu_n)^2} \quad (26)$$

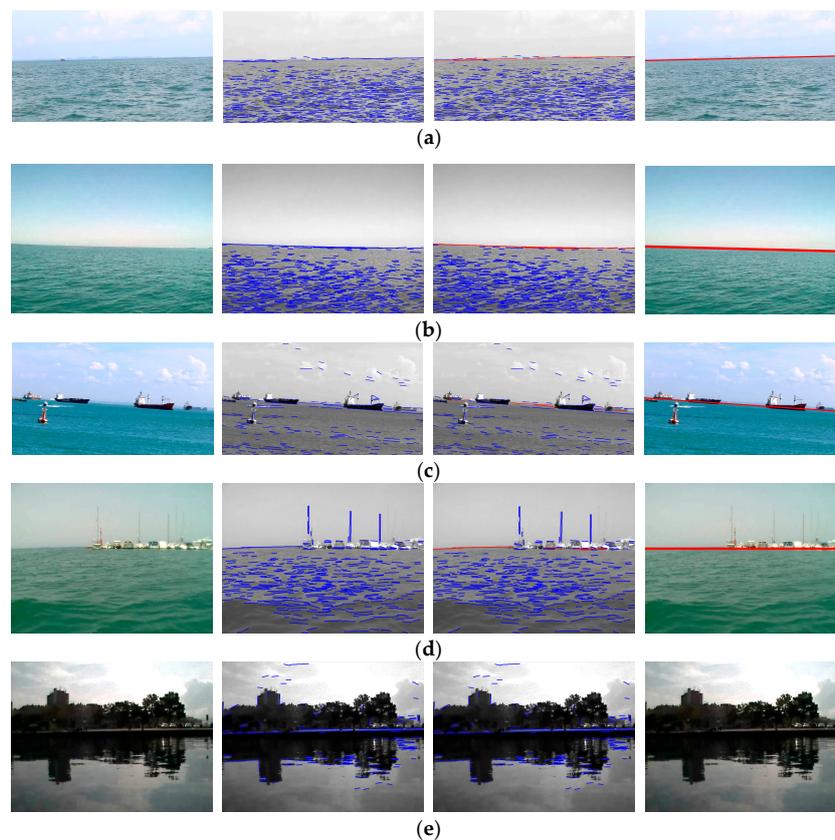
The clutter measure of an image was given by the average of the intensity variance of the sub-images:

$$clutter = \sqrt{\frac{1}{K} \sum_{n=1}^K \sigma_n^2} \quad (27)$$

The authors divided the image into  $4 \times 4$  sub-images with reference to Candamo's work [31]. The mean clutter of the database was denoted as  $\mu$ . When the clutter of an image was less than  $\mu$ , it was considered as low clutter. To test the validation of this approach, the authors used the public datasets

Singapore Maritime Dataset (SMD) and Marine Obstacle Detection Dataset (MODD) for benchmarking. The clutter of these images (or frames in video) varied from 1 to 45, and  $\mu = 14$ . The processing time for an image was a fraction of a second, which permitted the authors to test the method on thousands of images.

Coarse-fine-stitched (CFS) detection results with a low clutter image ( $clutter < \mu$ ) and high clutter image ( $clutter \geq \mu$ ) are shown in Figure 8. The intermediate process of the CFS method is shown to illustrate their effects. The second, third, and fourth column were the output of the coarse detection, fine detection, and robust stitching of CFS, respectively. Figure 8a–d show that the experimental results demonstrated that the method was able to extract the horizon line accurately with different backgrounds. The experiment also showed that there were few false positives for images without the horizon line, even though the background noise was large (Figure 8e).

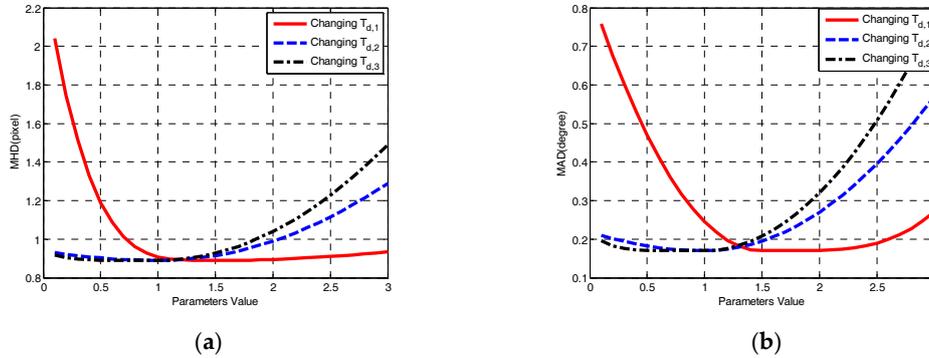


**Figure 8.** CFS detection results for the images from Singapore Maritime Dataset (SMD) and Marine Obstacle Detection Dataset (MODD) with different backgrounds. The first column is the original images and the next three columns are the output of the coarse detection, fine detection, and robust stitching of CFS, respectively: (a) Test results for a low clutter image ( $clutter = 11$ ) with the horizon line from SMD; (b) Test results for a low clutter image ( $clutter = 11$ ) with the horizon line from MODD; (c) Test results for a high clutter image ( $clutter = 17$ ) with the horizon line from SMD; (d) Test results for a high clutter image ( $clutter = 17$ ) with the horizon line from MODD; (e) Test results for a high clutter image ( $clutter = 41$ ) without the horizon line from MODD.

### 3.2. Parameter Sensitivity Analysis

There were three main control parameters in the fine detection of the coarse-fine-stitched (CFS) method, such as threshold  $T_{d,1} = 1.5$ , threshold  $T_{d,2} = 1$  and threshold  $T_{d,3} = 1$ . By fixing two parameters and changing another parameter, the mean height deviation (MHD, pixels) and mean angle deviation (MAD, degrees) of CFS performed on the testing data are shown in Figure 9. Setting a smaller  $T_{d,1}$

resulted in a larger probability of selecting the line segments, even if the difference between the regions was not distinct enough. Setting a larger  $T_{d,2}$  or  $T_{d,3}$  resulted in a larger probability of picking the line segments, even if the similarity between the sub-regions was not enough. Both of these caused false positives and decreased the detection accuracy.

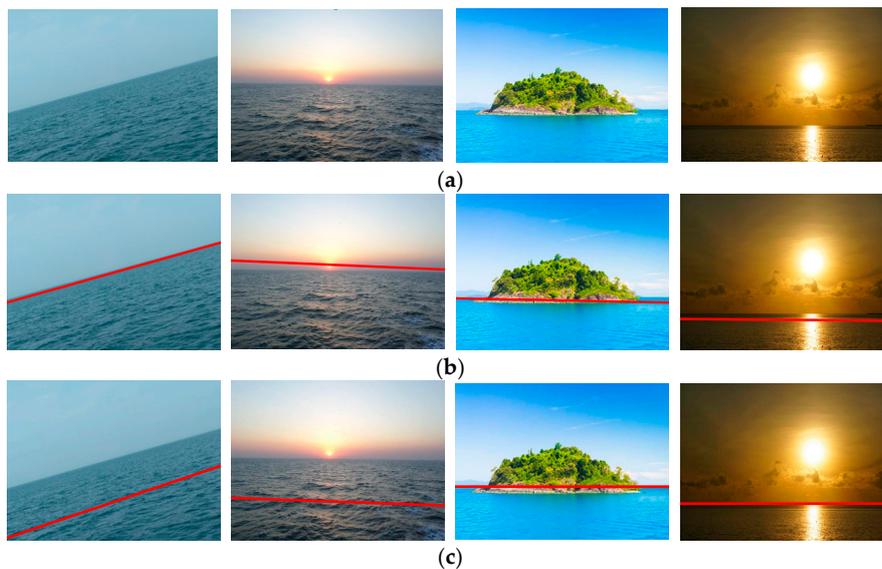


**Figure 9.** Parameter sensitivity analysis: (a) Mean height deviation (MHD) with changing control parameters. (b) Mean angle deviation (MAD) with changing control parameters.

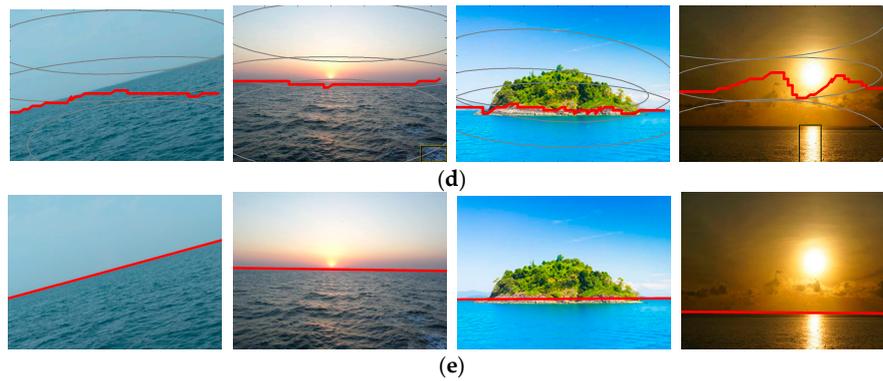
### 3.3. Methods Comparison

To test the performance of this method compared with the state-of-the-art methods, horizon line detection methods with a processing time for an image that was less than one second was applied. To verify this method, the authors used Wang’s algorithm [14], MSCM-LiFe [13], and SSM [18] as the baseline comparisons for the work. The dataset, the Singapore Maritime Dataset (SMD), and Marine Obstacle Detection Dataset (MODD), were applied for comparison in this section.

The performance of the baseline methods and this approach using the authors’ dataset in a maritime environment is shown in Figure 10. The figure illustrates that the proposed coarse-fine-stitched (CFS) method can detect the horizon line more accurately and robustly in the presence of various interfering factors, when compared with the other baseline methods. When using the testing image with a relatively simpler background, such as the first column in Figure 10, most of the methods obtained the horizon line accurately. However, when there were several interfering factors, such as illumination changing, clouds, or islands, the CFS method performs best when compared with the other methods.

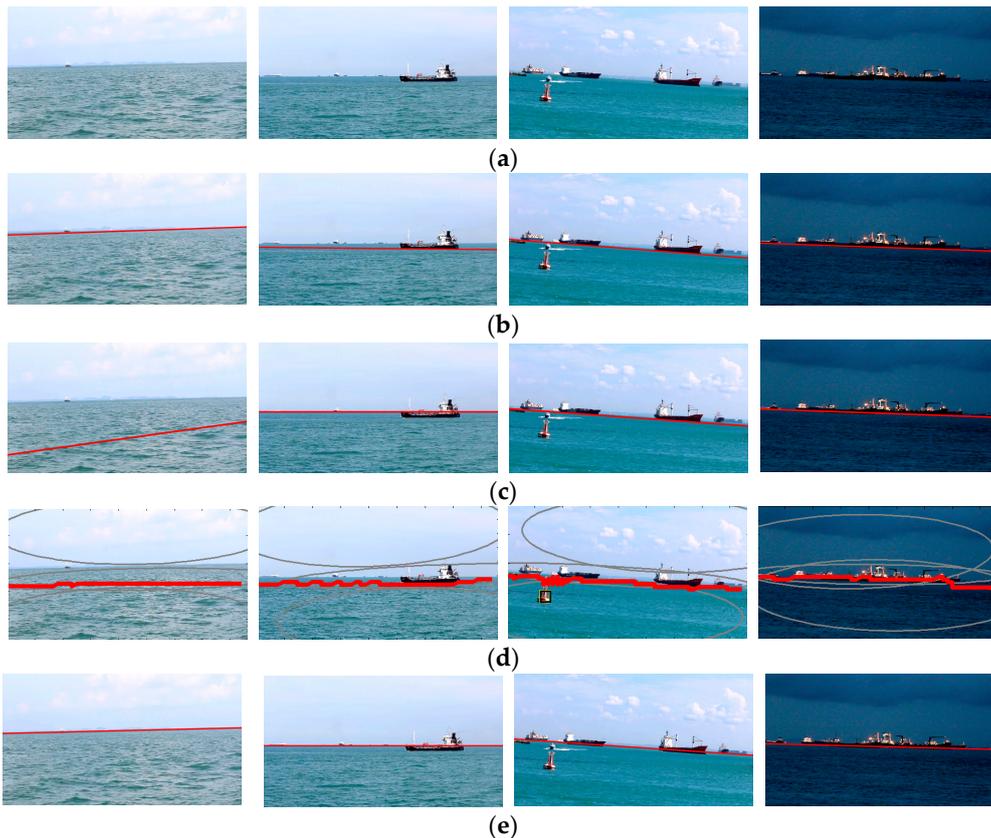


**Figure 10.** Cont.



**Figure 10.** A comparison of various horizon line detection methods using the authors' testing data: (a) Original image; (b) Wang's algorithm; (c) MSCM-LiFe; (d) SSM; (e) CFS method.

The performance of the baseline methods and this approach using SMD is shown in Figure 11. Wang's algorithm was affected easily by the bottom of the boats. SSM was very sensitive to obstacles when using the segmentation method. MSCM-LiFe was affected easily by waves, as shown in the first column of Figures 10c and 11c. Compared with the other methods, the CFS method performed accurately and robustly for the testing data.



**Figure 11.** A comparison of various horizon line detection methods using SMD: (a) Original image; (b) Wang's algorithm; (c) MSCM-LiFe; (d) SSM; (e) CFS method.

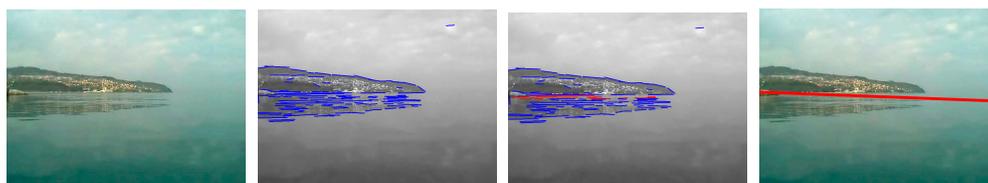
The performance of the baseline methods and this approach using MODD is shown in Figure 12. When using the testing image with a relatively simpler background, such as the first to third columns in Figure 12, most of the methods obtained the horizon line accurately. Compared with other methods,

Wang's algorithm and SSM were more sensitive to obstacles or shadow. MSCM-LiFe and this CFS method performed accurately and robustly for the testing data.



**Figure 12.** A comparison of various horizon line detection methods using MODD: (a) Original image; (b) Wang's algorithm; (c) MSCM-LiFe; (d) SSM; (e) CFS method.

However, when the horizon line was very blurry, such as in the fourth column in Figure 12, this coarse detection method cannot detect the horizon feature and the result was affected entirely by the ground and its shadow. The intermediate process of the CFS method on the very blurry horizon line is shown Figure 13.



**Figure 13.** The performance of this CFS method on the very blurry horizon line. The first column is the original image, and the next three columns are the output of the coarse detection, fine detection, and robust stitching of CFS, respectively.

Table 1 shows the comparison of the horizon line detection performance on this test data. The detection errors consisted of mean height deviation (MHD) and mean angle deviation (MAD). The average computing time for each testing was counted for comparison. Table 1 shows that the MHD and MAD of this method was the smallest when compared with the three baseline methods. The MHD of this approach was 49.7%, 82.4% and 39.7% of the results of Wang’s algorithm, MSCM-LiFe, and SSM, respectively. The MAD of this approach was less than  $0.2^\circ$ , which was also the most accurate of all of the methods. Since SSM is the segmentation method, the results were not straight lines. Thus, the MAD of SSM was not calculated in Table 1. The experimental results show that the color feature and RANSAC of this approach played an important role in obtaining low false positives and accurate results. Due to the iteration in RANSAC, the average computing time for this method was 94 ms. The computing time of this method was a little longer than Wang’s algorithm and SSM, and was shorter than MSCM-LiFe. Since it included the time for the Hough transform and intensity calculation for the multi-scale median filtering of the image, the computing time of MSCM-LiFe was relatively longer than other methods.

**Table 1.** Horizon line detection results using the current test data.

Algorithm	MHD	MAD	Average Computing Time
Wang’s algorithm	1.79	$0.38^\circ$	57 ms
MSCM-LiFe	1.08	$0.23^\circ$	231 ms
SSM	2.24	–	27 ms
CFS	0.89	$0.19^\circ$	94 ms

#### 4. Conclusions

To meet the requirement of horizon line detection for unmanned surface vehicle (USV) applications, a robust horizon line detection method, coarse-fine-stitched (CFS), was proposed in this paper. This CFS method operated in a three-step way. First, in the coarse step, a problem-specific design line segment detection (LSD) algorithm was presented to extract all the candidates, creating a line pool. Second, in the fine step, hybrid feature filtering was designed to select the horizon line segments from the pool. Finally, a random sample consensus-based (RANSAC-based) line segment stitching method was presented to obtain the whole horizon line. Experimental results demonstrate that the CFS outperformed the other state-of-the-art methods in terms of accuracy and robustness. Regarding the authors’ future work, a segmentation algorithm will be investigated further for horizon line and obstacle detection in USV applications.

**Author Contributions:** Each co-author made substantive intellectual contributions to this paper. Y.S. implemented the algorithm, analyzed the data, performed the experiments and wrote the paper. L.F. contributed to the experiment analysis and scientific writing.

**Funding:** The presented research work is supported by the National Natural Science Foundation of China (61803037).

**Conflicts of Interest:** The authors declare no conflict of interest.

#### References

1. The Navy Unmanned Surface Vehicle (USV) Master Plan. Available online: <http://www.navy.mil/navydata/technology/usvmppr.pdf> (accessed on 17 July 2018).
2. Manley, J.E. Unmanned Surface Vehicles, 15 Years of Development. In Proceedings of the OCEANS 2008, Quebec, QC, Canada, 15–18 September 2008.
3. Liu, Z.; Zhang, Y.; Yu, X.; Yuan, C. Unmanned Surface Vehicles: An Overview of Developments and Challenges. *Anal. Rev. Control* **2016**, *41*, 71–93. [CrossRef]
4. Wei, Y.; Zhang, Y. Effective Waterline Detection of Unmanned Surface Vehicles Based on Optical Images. *Sensors* **2016**, *16*, 1590. [CrossRef] [PubMed]

5. Giordano, F.; Mattei, G.; Parente, C.; Peluso, F.; Santamaria, R. Integrating Sensors into a Marine Drone for Bathymetric 3D Surveys in Shallow Waters. *Sensors* **2016**, *16*, 41. [[CrossRef](#)] [[PubMed](#)]
6. Villa, J.L.; Paez, J.; Quintero, C.; Yime, E.; Cabrera, J. Design and Control of an Unmanned Surface Vehicle for Environmental Monitoring Applications. In Proceedings of the IEEE Colombian Conference on Robotics and Automation, Bogota, Colombia, 29–30 September 2016.
7. Fu, L.; Hu, C.Q.; Kong, L.B. A Sea-Sky Line Detection Aided GNSS/INS Integration Method for Unmanned Surface Vehicle Navigation. In Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2017), Portland, OR, USA, 25–29 September 2017; pp. 1809–1815.
8. Fefilyatyev, S.; Goldgof, D.B.; Langebrake, L. Toward Detection of Marine Vehicles on Horizon from Buoy Camera. *Proc. SPIE* **2007**. [[CrossRef](#)]
9. Kim, S. Sea-based Infrared Scene Interpretation by Background Type Classification and Coastal Region Detection for Small Target Detection. *Sensors* **2015**, *15*, 24487–24513. [[CrossRef](#)] [[PubMed](#)]
10. Tang, D.; Sun, G.; Wang, D.H.; Niu, Z.D.; Chen, Z.P. Research on Infrared Ship Detection Method in Sea-Sky Background. In Proceedings of the Fifth International Symposium on Photoelectronic Detection and Imaging, Beijing, China, 25–27 June 2013.
11. Libe, T.; Gershikov, E.; Kosolapov, S. Comparison of Methods for Horizon Line Detection in Sea Images. In Proceedings of the Content 2012, Nice, France, 22–27 July 2011; pp. 79–85.
12. Prasad, D.K.; Rajan, D.; Rachmawati, L.; Rajabally, E.; Quek, C. MuSCoWERT: Multi-Scale Consistence of Weighted Edge Radon Transform for Horizon Detection in Maritime Images. *J. Opt. Soc. Am. A* **2016**, *33*, 2491–2500. [[CrossRef](#)] [[PubMed](#)]
13. Prasad, D.K.; Rajan, D.; Prasath, C.K.; Rachmawati, L.; Rajabally, E.; Quek, C. MSCM-LiFe: Multi-Scale Cross Modal Linear Feature for Horizon Detection in Maritime Images. In Proceedings of the 2016 IEEE Region 10 Conference (IEEE TENCON), Singapore, 22–25 November 2016; pp. 1366–1370.
14. Wang, B.; Su, Y.; Wan, L. A Sea-Sky Line Detection Method for Unmanned Surface Vehicles Based on Gradient Saliency. *Sensors* **2016**, *16*, 543. [[CrossRef](#)] [[PubMed](#)]
15. Gershikov, E. Is Color Important for Horizon Line Detection? In Proceedings of the 2014 International Conference on Advanced Technologies for Communications, Hanoi, Vietnam, 15–17 October 2014; pp. 262–267.
16. Scherer, S.; Rehder, J.; Achar, S.; Cover, H.; Chambers, A.; Nuske, S.; Singh, S. River Mapping from a Flying Robot: State Estimation, River Detection, and Obstacle Mapping. *Auton. Rob.* **2012**, *33*, 189–214. [[CrossRef](#)]
17. Fefilyatyev, S.; Goldgof, D. Detection and Tracking of Marine Vehicles in Video. In Proceedings of the IEEE International Conference on International Conference on Pattern Recognition, Tampa, FL, USA, 8–11 December 2008; pp. 1–4.
18. Kristan, M.; Kenk, V.S. Kovacic, S.; Pers, J. Fast Image-based Obstacle Detection from Unmanned Surface Vehicles. *IEEE Trans. Cybern.* **2016**, *46*, 641–654. [[CrossRef](#)] [[PubMed](#)]
19. Jeong, C.; Yang, H.S.; Moon, K.D. A Novel Approach for Detecting the Horizon Using a Convolutional Neural Network and Multi-Scale Edge Detection. *Multidimens. Syst. Signal Process* **2018**. [[CrossRef](#)]
20. Jeong, C.; Yang, H.S.; Moon, K.D. Horizon Detection in Maritime Images Using Scene Parsing Network. *Electron. Lett.* **2018**, *54*, 760–762. [[CrossRef](#)]
21. Von Gioi, R.G.; Jakubowicz, J.; Morel, J.M.; Randall, G. LSD: A Fast Line Segment Detector with a False Detection Control. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *32*, 722–732. [[CrossRef](#)] [[PubMed](#)]
22. Fu, L.; Lu, S. Obstacle Detection Algorithms for Aviation. In Proceedings of the IEEE International Conference on Computer Science and Automation Engineering, Shanghai, China, 10–12 June 2011; pp. 710–714.
23. Luo, X.; Zhang, J.; Cao, X.; Yan, P.; Li, X. Object-aware Power Line Detection Using Color and Near-infrared Images. *IEEE Trans. Aerosp. Electron. Syst.* **2014**, *50*, 1374–1389. [[CrossRef](#)]
24. Zhang, J.; Shan, H.; Cao, X.; Yan, P.; Li, X. Pylon Line Spatial Correlation Assisted Transmission Line Detection. *IEEE Trans. Aerosp. Electron. Syst.* **2014**, *50*, 2890–2905. [[CrossRef](#)]
25. Desolneux, A.; Ladjal, S.; Moisan, L.; Morel, J.M. Dequantizing Image Orientation. *IEEE Trans. Image Process.* **2002**, *11*, 1129–1140. [[CrossRef](#)] [[PubMed](#)]
26. Gonzalez, R.C.; Woods, R.E.; Eddins, S.L. *Digital Image Processing Using MATLAB: AND Mathworks, MATLAB Sim SV 07*; Prentice Hall Press: Upper Saddle River, NJ, USA, 2007.

27. Afifi, A.J.; Ashour, W. *Image Retrieval Based on Content Using Color Feature: Color Image Processing and Retrieving*; LAP Lambert Academic Publishing: Saarbrücken, Germany, 2001.
28. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **1981**, *24*, 381–395. [[CrossRef](#)]
29. Tseng, P.; Yun, S. A Coordinate Gradient Descent Method for Nonsmooth Separable Minimization. *Math. Programm.* **2009**, *117*, 387–423. [[CrossRef](#)]
30. Schmieder, E.; Weathersby, M.R. Detection Performance in Clutter with Variable Resolution. *IEEE Trans. Aerosp. Electron. Syst.* **1983**, *19*, 622–630. [[CrossRef](#)]
31. Candamo, J.; Kasturi, R.; Goldgof, D.; Sarkar, S. Detection of Thin Lines Using Low Quality Video from Low Altitude Aircraft in Urban Settings. *IEEE Trans. Aerosp. Electron. Syst.* **2009**, *45*, 937–949. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).