

Article



# **Research on a Horizon Line Detection Method for Unmanned Surface Vehicles in Complex Environments**

Binghua Shi <sup>1</sup>, Chen Wang <sup>2</sup>, Yi Di <sup>1,3</sup>, Jia Guo <sup>1,3</sup>, Ziteng Zhang <sup>1</sup> and Yang Long <sup>4,\*</sup>

- <sup>1</sup> School of Information Engineering, Hubei University of Economics, Wuhan 433015, China; shibinghua1988@163.com (B.S.)
- <sup>2</sup> No. 722 Research Institute of CSSC, Wuhan 430000, China
- <sup>3</sup> Hubei Internet Finance Information Engineering Technology Research Center, Wuhan 433015, China
- <sup>4</sup> School of Intelligent Systems Science and Engineering, Hubei Minzu University, Enshi 445000, China
- Correspondence: longyang0718@foxmail.com

Abstract: A critical step in the visual navigation of unmanned surface vehicles (USVs) is horizon line detection, which can be used to adjust the altitude as well as for obstacle avoidance in complex environments. In this paper, a real-time and accurate detection method for the horizon line is proposed. Our approach first differentiates the complexity of navigational scenes using the angular second moment (ASM) parameters in the grey level co-occurrence matrix (GLCM). Then, the region of interest (ROI) is initially extracted using minimal human interaction for the complex navigation scenes, while subsequent frames are dynamically acquired using automatic feature point matching. The matched ROI can be maximally removed from the complex background, and the Zernike-moment-based edges are extracted from the obtained ROI. Finally, complete sea horizon information is obtained through a linear fitting of the lower edge points to the edge information. Through various experiments carried out on a classical dataset, our own datasets, and that of another previously published paper, we illustrate the significance and accuracy of this technique for various complex environments. The results show that the performance has potential applications for the autonomous navigation and control of USVs.

**Keywords:** horizon line detection; unmanned surface vehicles; intelligent navigation system; visual navigation technology; marine images

# 1. Introduction

Unmanned surface vehicles (USVs) can be deployed in complex natural environments, replacing manned ships in numerous applications, such as rescue, disaster relief, and environmental monitoring [1]. A closed-circuit television (CCTV)-system-based vision sensor is one of the conventional assemblies for USVs. Compared to laser rangefinder instruments, synthetic aperture radar vision sensors have extensive advantages, such as data richness, low cost, and good stability [2–4]. A USV with autonomous navigation capabilities can ensure self-safe and efficient execution to complete specific tasks; in particular, the rapid development of visual navigation technology has laid an important foundation for autonomous navigation [5,6]. For long voyages, early determination of the sea level relies on visual navigation technology to help maintain USV balance and smooth navigation [7,8].

The horizon line detection operation is an essential foundation of visual navigation technology. In general, the area above the horizon line is represented as a contour dividing the background and the water, which encompasses the navigable area for the USV. Horizon line detection greatly affects the performance of subsequent steps, such as intelligent autonomous navigation, situational awareness, dynamic positioning, obstacle detection, and target tracking, and the entire navigation system plays an important role [9]. Although many horizon line detection methods have been proposed for open-field environments, few methods are available for complex natural environments, such as harbours and inland



Citation: Shi, B.; Wang, C.; Di, Y.; Guo, J.; Zhang, Z.; Long, Y. Research on a Horizon Line Detection Method for Unmanned Surface Vehicles in Complex Environments. *J. Mar. Sci. Eng.* 2023, *11*, 1130. https://doi.org/ 10.3390/jmse11061130

Academic Editor: Rafael Morales

Received: 21 April 2023 Revised: 13 May 2023 Accepted: 25 May 2023 Published: 27 May 2023



**Copyright:** © 2023 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 (https:// creativecommons.org/licenses/by/ 4.0/). rivers, and often the robustness and accuracy do not address the practical needs [10]. How to extract the critical pixel points from the billions of raw pixels that make up a horizon line, whose accuracy and robustness in a natural navigation environment is unknown, still remains to be understood. For example, once there are reflections, obstacle occlusions, illumination changes, camera jitter, irregular waves, and other disturbances on the water surface, horizon line detection becomes quite a challenging problem [11]. The horizon line is the boundary that distinguishes the background area from the navigable area. Accordingly, the navigable area determination problem can be equated to a horizon line detection operation, which is a hot research topic to improve the autonomy of USVs [12]. Therefore, the problem of detecting horizon lines using intelligent CCTV systems for USVs is a topic of practical importance, which can help ships maintain balance, estimate their altitude, determine navigable areas, and identify obstacles to be avoided.

In the field of autonomous USV navigation, some typical approaches for horizon line detection have been successful in open waters [13,14]. However, real-time and accurate detection of the horizon line in complex environments also needs to address the following central issues: (1) There are challenges in distinguishing the sea and the sky with similar colours and low contrast, which is caused by the prevailing atmospheric and illumination conditions. (2) Horizon line detection methods are significantly different for single-frame images and dynamic videos. (3) A quick algorithm for horizon line detection needs to be explored for a moving USV in a cluttered environment. (4) Traditional horizon line detection approaches have poor generalization performances, and they often obtain unstable results. (5) When the horizon line is not completely straight or there are other straight lines that create interferences, low-resolution results may occur.

In this paper, we aim to address the above horizon line detection issues in a complex maritime scenario based on our previous work [15–17]. For this, the criteria for defining a complex scenario need to be clearly defined as a first step, which will also document the various challenging scenarios. Second, a novel and efficient horizon line detection algorithm is developed based on minimal manual interactions. Finally, some experiments are designed to evaluate the performance with other state-of-the-art networks on the Singapore maritime dataset (SMD) [18], maritime obstacle detection dataset (MODD) [19], and our self-collected Yangtze River navigation scene dataset (YRNSD). In summary, the main contributions of this paper are as follows:

- We propose criteria for classifying a complex scenario using the grey level co-occurrence matrix (GLCM) to cover various challenging scenarios.
- (2) We develop an efficient method based on a novel dynamic region of interest (ROI) approach to detect the horizon line in a challenging scenario for a moving USV.
- (3) We show that it is possible to use weak manual interactions and autonomous feature extraction techniques to detect the horizon line for intelligent visual navigation.

The remainder of this paper is organized as follows. A review of the related literature for horizon line detection is presented in Section 2. Section 3 provides design considerations and preliminaries, including the criteria for complex maritime scenarios. The details of our proposed method are presented in Section 4, which contains five steps, namely, classification of the scenario complexity, expanded ROI extraction, dynamic ROI matching, edge extraction based on Zernike moments and lower edge-point linear regression fitting. In Section 5, the experimental results obtained with our proposed method are presented and compared with other relevant state-of-the-art approaches. The subsequent conclusions and future work prospects are given in Section 6.

## 2. Related Work

Several approaches have been explored and reported for detecting horizon lines using onboard or onshore sensors. According to a review of the literature, the development of these approaches has progressed through three categories. The first category is the manual sifting of local features, which mainly uses colour, edge and texture information. For example, Ref.[20] first converts an RGB image into a binarized image using an Otsu threshold segmentation algorithm, and then combines a Hough transform to find the longest line and treats it as the horizon line. Ref. [21] extended Canny edge detection and a Hough transform to the field of horizon line detection. Similarly, Ref. [2] obtained some candidate horizons based on a Canny edge detector a Hough transform and then used a voting method and picked the horizon with the most votes as the true horizon. The inherent drawback of such methods based on local features is their instability in complex maritime environments, because the parameter selection relies on artificial empirical prior knowledge or underlying assumptions ([22,23]). For example, edge information easily suffers more in the presence of edge gaps, and texture information is easily blocked by local shadows and obstructions. Although edge gap filling processes have been proposed, it is also easy to get into trouble when the edge gap is larger than the search window.

The second category is adaptive global features. Most of the methods in this class are based on the overall features of an image and do not rely on prior knowledge, and these methods outperform the local feature methods ([24,25]). The extreme values of the gradient change are first sampled for each vertical column in the gradient image, and then the random sample consensus (RANSAC) method is used to fit the horizon line. Ref. [26] proposed a hierarchical horizon detection algorithm that combines a Canny edge detector with a Hough transform to adaptively find the longest line and then fine-tunes it to obtain the horizon line information. Ref.[27] extracted a rectangular region above a virtual horizon as a region-growing seed, and the region-growing algorithm helped obtain the final horizon. The overall features include colour distribution, texture information, and spatial context, and they can only be used as a basis for rough-level segmentation, as these features and principles may not be suitable for complex marine environments. Meanwhile, these methods still suffer from a homogeneous distribution problem in the water surface case, and there are local regions with abnormal changes in grey values that can reduce the accuracy, such as water surface shadows and obstacle occlusions.

The third category is intelligent detection methods, also called regression-based approaches. This class of approaches tends to use a coarse-to-fine strategy, i.e., initially focusing on the overall structural information and subsequently updating features using the finer details to provide more accurate predictions. Horizon line predictions use semantic segmentation based on deep learning ([28–30]). The key is transforming horizon line detection into clustering and classification. At the same time, logistic regression and polynomial spline modelling are less effective in treating the probability distribution and physical characteristics of a detected horizon line. Depending on the degree of manual labelling for the training data, these approaches can be subdivided into semisupervised and supervised learning ([31]). For example, Ref. [32] designed a semisupervised water region segmentation learning method for USVs in a changing unknown environment by using automatically labelled training data with the aid of LiDAR. Ref.[11] combined a multiscale approach and a convolutional neural network (CNN) to detect the horizon in maritime scenarios. In general, a horizon line extraction has two parts, a region of interest (ROI) extraction and horizon line estimation, where the important features are extracted first and then trained with intelligent detection methods. The vast majority of the above methods have been proven to be robust and accurate in general environments but have poor generalization performances for complex visual navigation environments, such as when mirror reflections exist on the water surface, or during water fogging or obstacle interference.

## 3. Design Considerations and Preliminaries

Before performing an accurate horizon line detection, we must differentiate the actual navigational environment captured using a CCTV system to determine whether the USV's navigational scenario is in open water or complex water. Open-water scenarios are usually defined as outer waters with a wide field of view, making the horizon line detection relatively simple. However, complex water scenarios are usually inland rivers and harbours with heavy traffic, where the detection of horizon lines for complex scenarios is susceptible to numbers factors.

First, the water surface is susceptible to extremely dark or bright areas due to direct sunlight or reflections. For example, object shadows are formed when the light shining on the surface of an object is partially or completely blocked ([33]). In addition to the effect of light projection, the reflection of the sun on the water surface can create large patches of extremely bright light that considerably block the continuity of the horizon line, as shown in Figure 1a. Second, complex scenarios are bound to have other ships, floating objects, navigation aids, and other types of cluttered background, as shown in Figure 1b, which disrupt the continuity of the horizon line features. When there are multiple obstacles blocking each other, horizon line detection will be even more challenging. Third, USVs are usually lightweight vessels, and when they are moving quickly, it is difficult to maintain the balance of the hull in a sustainable manner, resulting in a certain angle of tilt of the horizon line (as shown in Figure 1c), which may be random and unpredictable. In addition, special weather conditions, such as night, rain, snow, and fog, can also affect the accurate extraction of the horizon line. As shown in Figure 1d, foggy weather can cause the distinction between the water area and the background area to be blurred. As a result, the image information captured in complex scenarios is more complex and does not have a clear distinction between the water area and the background area compared to open water. Therefore, complex scenarios and their determination criteria are defined before conducting a horizon line detection to facilitate the coverage of a variety of challenging scenarios.



(a) Setting sun.





(b) Cluttered background.



(d) Scene tilting.

(d) Foggy weather.

Figure 1. Examples of typical complex elements.

# 4. Proposed Approach

#### 4.1. Classification of the Scenario Complexity

A spatial relationship is considered to be a function of the distance between two pixels, and the texture features extracted from a camera image using a grey level co-ocurrence matrix (GLCM) [34] can be used to differentiate between navigation scenario complexities. Before the construction of the GLCM, we need to transform the original navigation scenarios

into grey-level images. Assume an  $M \times N$  navigation scene *I* is transformed into a grey image  $I_{gray}$ , which is described as:

$$I_{grey} = f(x, y)_{M \times N} = 0.299 * R(x, y) + 0.587 * G(x, y) + 0.114 * B(x, y)$$
(1)

 $I_{grey}$  is an image with  $N_g$  grey grades, and  $(x_1, y_1)$  and  $(x_2, y_2)$  are two pixel points in scene  $I_{grey}$  with distance d in the direction of  $\theta$ . Then, the GLCM of this navigation scene is calculated as follows:

$$P(i, j, d, \theta) = \#\{(x_1, y_1), (x_2, y_2) \in M \times N \mid I(x_1, y_1) = i, I(x_2, y_2) = j\}$$
(2)

where # denotes the number of elements in the set.  $i, j = 0, 1, 2, ..., N_g - 1$  represents the grey levels of two pixels. The angular second moment (ASM) is one GLCM feature, which is often used to describe the uniformity of the greyscale distribution in images. The ASM is calculated as follows:

$$ASM = \sum_{i=0}^{N_g - 1} \sum_{j=0}^{N_g - 1} P(i, j, d, \theta)^2$$
(3)

When the distribution of elements in the GLCM is more concentrated around the main diagonal, a smaller ASM indicates a more uniform distribution of pixel greyscales and finer textures; conversely, it indicates an uneven distribution of pixel greyscales and coarser textures. Hence, we use the ASM value as a criterion to determine the complexity of the scenario in this paper.

## 4.2. Expanded Region-of-Interest (ROI) Extraction

As the valuable information containing the horizon line feature points is usually concentrated in the central region of the whole image, if all the pixels of the whole image are traversed for the search calculation, it is not only computationally time-consuming but also difficult to meet the real-time requirements, and it introduces a large amount of unnecessary noise interference, which increases the image processing difficulty. Therefore, region-of-interest (ROI) extraction is a necessary operation for image processing.

As shown in Figure 2, to simplify horizon line detection, an original ROI defining a bounding box (yellow) around the touched line must be drawn by the user. Due to the small size of the display and the jittering hands of the user, the ROI is properly expanded as the the red bounding box. The following relationship exists between the expanded and original ROI:

$$\begin{cases} x_e = x_o + \Delta x_1 + \Delta x_2 = M \\ y_e = y_o + \Delta y_1 + \Delta y_2 = N' \end{cases}$$
(4)

where  $x_e$  and  $y_e$  denote the width and height of the expanded ROI, and  $x_o$  and  $y_o$  denote the width and height of the original ROI, respectively.  $\Delta x$  and  $\Delta y$  represent the expansion of the ROI compared to the original ROI in the horizontal and vertical directions, respectively. The image captured by the vision camera is of a M \* N resolution, with three-channel RGB. In the actual processing, the width of the expanded ROI  $x_e$  is M, which is the width of the captured image. Usually, the values of  $\Delta x_1$  and  $\Delta x_2$  can be unequal, i.e., the user is not required to manually draw the original ROI as horizontally centred. However,  $\Delta y_1$  and  $\Delta y_2$  are generally equal and take values of 20 pixels unless the size of the expanded ROI exceeds the boundary of the captured image, in which case the upper/lower boundary of the captured image. The value of N' is approximately equal to one-fifth of N.



Figure 2. Schematic of interactive-based expanded ROI extraction.

## 4.3. Dynamic ROI Matching

Using an interactive-based expanded ROI extraction strategy, the expanded ROI to be selected for a horizon line detection can be reduced. Unfortunately, it is also impractical to perform interactive ROI extraction for every frame because the workload is undoubtedly huge for video images of at least 25 frames per second (FPS). Considering that video images captured by the shipboard camera are continuous sequences containing time-stamped information, there is strong spatial continuity between the sequence images over a short time interval. Therefore, we only select the first image frame for interactive ROI extraction in the initialization phase of the algorithm, which is a minimal interaction that is usually acceptable in a crowdsourcing approach. The specific process of dynamic ROI matching is as follows:

**Step 1**: Initialization of the master areas. The expanded ROI  $I_{eROI}$  in the first video frame of the shipboard camera acts as the master areas.  $I_{eROI}$  is coregistered until all the video images are registered.

$$I_{eROI}^{t=0}(i,j) \in I^{t=0}(x,y)$$
 (5)

**Step 2**: Coarse extraction of the keypoints. The keypoints  $\mathbf{P}^t$  of the master areas  $I_{eROI}^t$  and  $I^{t+1}(x, y)$  are initially extracted using the oriented features from a accelerated segment test (oFAST), which introduces the concept of feature orientation to achieve the feature point rotation invariance.

$$\mathbf{P}^t \in I^t_{eROI} \cap I^{t+1}(x, y) \tag{6}$$

**Step 3**: Fine extraction of the keypoints. Construct the Hessian matrix for the keypoints and finely extract  $\tilde{\mathbf{P}}^t$  again to select the keypoint with better traceability. For the keypoints  $\tilde{\mathbf{P}}^t$ , the following two conditions need to be met simultaneously:

$$\begin{cases} \det(\boldsymbol{H}_{P}) > \det(\boldsymbol{H}_{i}) & i = 1, 2, 3, \cdots, 8\\ \sum_{i=1}^{8} [\det(\boldsymbol{H}_{P}) - \det(\boldsymbol{H}_{i})] > \det_{\text{th}} \end{cases}$$
(7)

where det( $H_P$ ) and det( $H_i$ ) both represent the Hessian matrix discriminant. det<sub>th</sub> indicates the set threshold value. Equation (7) implies that while the Hessian matrix discriminant of  $\tilde{\mathbf{P}}^i$  is a local maximum, the sum of their differences needs to be greater than det<sub>th</sub>.

**Step 4**: The keypoint descriptor. After a fine extraction of the keypoints, we use the rotated binary robust independent elementary features (RBRIEF) operator to compute the feature descriptors. The RBRIEF descriptor is constructed from a set of binary intensity tests.

**Step 5**: ROI matching. Brute force (BF) descriptor matching assigns the closest descriptor of the slave areas to the master areas. The power of BF matching lies in its ability to retrieve the nearest neighbours with a high probability given enough hash tables.

The horizon line candidate regions can be obtained continuously after the above five processing steps. As shown in Figure 3, the spatiotemporal similarity between the upper and lower frames is used to dynamically locate the ROI.



Figure 3. Dynamic ROI matching process.

## 4.4. Edge Extraction Based on Zernike Moments

After the dynamic ROI area has been extracted, the navigational scene needs to be further processed to obtain the horizon line information. Considering the poor robustness of the traditional Sobel and Canny operator edge detection algorithms against noise and image rotation, this study uses Zernike moments to extract the image contour edge. Zernike moments are a type of convolutional integration method that is highly resistant to noise interference and is rotationally invariant.

First, the two-dimensional Zernike moments of the ROI region can be defined as:

$$Z_{nm} = \frac{n+1}{\pi} \sum_{x} \sum_{y} f(x,y) * V_{nm}^*(\rho,\theta)$$
(8)

where *m* and *n* are two integers,  $n \ge 0$ , and the value of n - |m| is an even number.  $\rho$  is the position of the edge of the image, while  $\rho = \sqrt{(x^2 + y^2)}$ .  $\theta$  is the angle between the X-axis and  $\rho$ . f(x, y) means the ROI region.  $V_{nm}$  is a Zernike polynomial of order *n*, defined as a function of  $\rho$  and  $\theta$  in the polar coordinate system.  $V_{nm}^*$  is the conjugate complex number of  $V_{nm}$ .

Second, the paper directly refers to the classical  $7 \times 7$  template factor calculated in [35]  $\{M_{00}, M_{11}, M_{20}, M_{31}, M_{40}\}$ . Let the ROI region f(x, y) = 1, and note that the template of  $Z_{nm}$  is  $M_{nm}$ ; then, we have:

$$M_{nm} = \int \int_{x^2 + y^2 \le 1} V_{nm}^*(\rho, \theta) \mathrm{d}x \, \mathrm{d}y \tag{9}$$

Then, Equation (9) is utilized to calculate the templates and each pixel point of the ROI region image for the convolution operation to obtain  $7 \times 7$  Zernike moments:  $\{Z_{00}, Z_{11}, Z_{20}, Z_{31}, Z_{40}\}$ .

Finally, using the rotational invariance principle of the Zernike moments and edge points, the spatial greyscale model of edge points can be solved to find the pixel positions of the edge points in the ROI region. Figure 4 shows the results of different edge extraction algorithms, where the ROI regions (Scene 1#, Scene 2#, Scene 3#, Scene 4#) come from Figure 1.





# 4.5. Lower Edge-Point Linear Regression Fitting

The ROI region can be roughly located using edge detection based on the Zernike moments; however, further linear fitting is required to obtain a more accurate horizon line. Due to the influence of water spots, obstacles, complex backgrounds, and foggy weather, edge areas are often uneven and haphazard, which poses a great challenge for fitting the horizon line directly. To this end, the analysis of many edge images shows that there is usually a clear differentiation between the background area and the navigable water surface area, i.e., edge noise is usually distributed in the background area, while there are no edge points in the navigable area. Therefore, in this paper, by performing lower edge-point tracking on the edge points in the background region and putting the tracked points into the set  $T_m$ , we obtain the specific tracking process in Algorithm 1.

# Algorithm 1 Lower edge-point tracking algorithm

**Require:** The binarized image  $I_{edge}$  after Zernike moments edge extraction **Require:** The number of rows of the ROI region M **Require:** The column number of the ROI region N **Ensure:** The set of  $T_m = \{(x_m, y_n) | , x_m \in [0, M], y_n \in [0, N] \}$ 1: while i < N do \\ i denotes the number of columns currently searched **while** j > 1 **do** \\ j denotes the number of rows currently searched 2: 3: if  $I_{edge}(i, j) == 1$  then 4: Put the point (i, j) into the set  $T_m$ 5: Break 6: end if 7: i = i - 18: end while 9: if  $I_{edge}(i, j) == 0$  then \\ denotes that no edge points exist in this column 10: Put the point (fix(i/2), j) into the set  $T_m$ 11: end if 12: i = i + 113: end while 14: Return  $T_m$ 

The linear regression method is used to fit the traced lower edge points  $T_m$ . A fitted linear equation y = kx + B is first constructed, and the fitted linear function f(x) is found so that the sum of squares of the errors with the actual value is minimized, and the objective function is:

$$J = \sum_{i=1}^{n} w_i [y_i - f(x_i)]^2 = \sum_{i=1}^{n} w_i [y_i - (kx_i + b)]^2$$
(10)

where  $w_i$  is the weight of the lower edge point, and the initial weight of each edge point is 1 in Equation (11).

Next, the weights of the edge points are updated according to the weight function, and the residual value  $r_i$  of the fitted line is calculated by fitting the equation of the line and the distance from the edge point to the fitted line. The weights of the lower edge points that deviate from the line should decrease as the residual  $r_i$  increases, and at the same time, the amount of weight function design computation is reduced, as follows:

$$w_i = \begin{cases} 1 & |r_i| = 0\\ 1/r_i & |r_i| \neq 0 \end{cases}$$
(11)

where the reciprocal of the residual value is used as the weighted value of the lower edge points when the residual value is not zero.

Finally, the updated weight values are substituted into Equation (10) using the least squares method to solve for the estimated horizon line information, and the specific effect is shown in Figure 5. The ROIs in Figure 5 are derived from Figure 1, where the blue points are the traced lower edge points and the yellow lines are the horizon lines.





(c) Scene tilting ROI.



(d) Foggy weather ROI.

Figure 5. Linear fit results for different navigation scenarios.

#### 5. Experiment and Results

# 5.1. Datasets and Evaluation Metric

To verify the effectiveness of the proposed method, experiments were conducted on the SMD [18], MODD [19], and YRNSD datasets [6], of which the first two are classical datasets and the third is a self-collected dataset. The SMD dataset was collected in Singapore waters from July 2015 to May 2016 under various environmental conditions, such as before sunrise

(40 min before sunrise), sunrise, midday, afternoon, evening, haze and rainfall, and after sunset (2 h after sunset). The MODD dataset was collected in Koper Bay, Slovenia, using a camera fixed to an unmanned boat over a period of approximately 15 months, with the camera capturing video at a given resolution at 10 frames per second. The self-collected YRNSD dataset in this paper was captured in the Wuhan section of the Yangtze River basin and consists of 64 videos, which cover a wide range of types of obstacles and meteorological conditions. The experimental environment in this paper is an Intel Core i7-8700K CPU 3.70 GHz\*12, NVIDIA GeForce GTX 1080Ti GPU, 32 GB RAM.

To quantitatively compare the performance of the methods, the ASM was employed to evaluate the complexity of the scene. In general, we consider that a larger ASM value represents a more complex navigational scene image and a greater difficulty in performing horizon line detection. The calculation of the ASM value can be used as a basis for choosing to use this method or a conventional edge detection before performing a specific horizon line detection. Table 1 shows the distribution of *ASM* values corresponding to the three datasets.

Table 1. Datasets and the parametric *ASM* distributions.

Dataset	SMD	MODD	YRNSD		
No. of videos	11	28	64		
No. of frames	4429	5084	14430		
$ASM_{min}$	0.0604	0.0144	0.0672		
$ASM_{max}$	0.0949	0.1180	0.1092		
$ASM_{S.D.}$	$3.5028 imes10^{-4}$	$7.0297  imes 10^{-6}$	$2.1278  imes 10^{-5}$		

The above ASM values are derived from random video sequences from the three datasets.

As seen from Table 1, the ASM value variation range for the same video scenes is small, as reflected by the small maximum ASM value variation ranges and their small standard deviation, indicating that the ASM values are relatively stable.

# 5.2. Impact of Different Complexity Levels

The complexity of the textures contained within each dataset also varies, so all the scene images are classified into low, medium, and high complexities, and specified as less than 25% (lower quartile), the 25% to 75% interval, and the greater than 75% (upper quartile) respectively. Figure 6 shows 9 images derived from the 3 datasets.



(g) YRNSD (ASM = 0.0696).

(h) YRNSD (ASM = 0.0813).

(i) YRNSD (ASM = 0.0914).

10 of 15

Figure 6. The real horizon line detection results.

To quantitatively verify the accuracy of the proposed method, we extracted the horizon lines, point by point, for each of the 9 images in Figure 6 by using manual annotations and used them as the real horizon line, and the extraction results are shown as the red straight lines in Figure 6. Meanwhile, the horizon line information is obtained using the method in this paper, and the effect is shown as a yellow straight line in Figure 7. Then, the average error *e* between the detection result and the real result is calculated as Equation (12), and the results are shown in Table 2.

$$e = \frac{\sum_{i=1}^{n} |L_{e,i} - L_{t,i}|}{n}$$
(12)

where  $L_{e,i}$  denotes the longitudinal coordinate of the *ith* horizon line points detected by the proposed algorithm.  $L_{t,i}$  denotes the real longitudinal coordinate of the *ith* horizon line points. *n* denotes the number of horizon line points.



Figure 7. Horizon line detection results using our method.

Datasets SMD		MODD			YRNSD				
ASM	0.0632	0.0835	0.0917	0.0216	0.0461	0.0975	0.0696	0.0813	0.0914
е	1.4005	1.5667	1.8161	1.2499	1.4036	1.3295	1.2996	1.2469	1.8167

Table 2. The error between our detection and the real results.

As seen from the results in Table 2, the average algorithm error for scenes of different complexity from the three different datasets is within 2 pixels, indicating that the horizon line detection results of our method are very close to the real results, and the impact of different complexity scenes on this method is small. This is because we detect the region of interest in advance, and the complex background interference is already removed. Moreover, except for the initial region, which needs to be calibrated the first time it is used, all subsequent detection operations can be carried out automatically.

#### 5.3. Comparative Analysis of Different Methods

To further verify the methods performance, we compared the same images using the proposed method, the conventional edge detection and threshold segmentation methods (EDTSM), and the semantic segmentation methods (SSM), and the results are shown in Figure 8. Figure 8a,d,g,j show the horizon line detection results obtained using the

proposed method; Figure 8b,e,h,k show the horizon line detection results obtained using the conventional EDTSM; and Figure 8c,f,i,l show the horizon line detection obtained using the SSM.

**Figure 8.** Qualitative comparison of our method with EDTSM and SSM. (**a**–**c**) represent the scenes of setting sun. (**d**–**f**) represent the scenes of cluttered background. (**g**–**i**) represent the scene titling. (**j**–**l**) represent the scenes of foggy weather.

From the above experimental results, it can be concluded that the proposed algorithm is able to detect the horizon line relatively accurately, even under scenarios such as a setting sun, cluttered background, scene tilting and foggy weather images. The horizon line detected by traditional EDTSM-like methods often consists of multiple discontinuous line segments, and horizon line detection will fail in scenarios with high background complexities, cluttered edge information, and bad weather (e.g., foggy days). Although the recent trend of semantic segmentation for extracting horizon lines is effective and has a certain ability to improve the anti-interference capability compared to the EDTSM-like methods, most of these methods need a large amount of manually labelled data, and it takes a long time from training and testing to application deployment.

We employed our method on the R2018b Matlab platform to process a 2 min video sequence (obtained from the YRNSD dataset). The horizon line detection time took approximately 40 s, which also included the selection of initial frames of ROI that took about 10 s. Excluding the manual interaction period, the average detection time per frame was roughly 0.213 s. On the other hand, EDTSM-like methods under similar conditions took about 15 s and an average processing time of 0.12 s per frame while providing subpar accuracy in complex environments. The SSM-based method may be applied directly to process the same video sequence without requiring prior training, which takes approximately 38 s of processing time. This translates to 0.317 s per frame image. However, similar to

edge-based methods, the horizon information extracted via SSM may not be precise enough in complex scenes.

In a comprehensive comparison, the proposed method is a coastline extraction algorithm based on minimal manual interaction, which definitely used less computational resources than the method based on SSM. Our method first performs an upfront calculation of the complexity of the scene, in addition to extracting the initial ROI using manual annotation and subsequent matching. These three steps are time-consuming but greatly reduce the region to be searched in the specific edge extraction phase, which can save some time. Moreover, for conventional EDTSM-like-method algorithms, which take the least amount of time, our method is able to obtain more accurate edge information, which is of the utmost importance. Therefore, for a USV with a long navigation time, the method in this paper can help obtain relatively accurate information about the horizon line in real time.

#### 6. Conclusions

We proposed a horizon line detection method based on minimal manual interaction, which specifically includes evaluating the complexity of the navigation scene using the ASM parameters of GLCM. For highly complex scenes, we use manual interactions to dynamically extract the ROI in the initialization phase of the method, and then use key feature points to match the ROI in the next image frame as a way to continuously exclude the interference of the complex background environment. We then use Zernike moments to extract the edge features of the current ROI, and finally use the method of least squares to linearly fit all the lower edge feature points to the horizon line. To evaluate the effectiveness of our method, comparative experiments were designed for navigation scenarios of different complexities, also quantitative comparisons were conducted between this paper and the conventional EDTSMs and SSMs. Our experiments show that this method has the potential to be applied to the autonomous navigation and control of USVs.

Our method partially solves the problem of horizon extraction in complex environments but has some shortcomings. At present, the anti-interference and stability of our approach are poor in some extreme or unexpected situations. For example, during the long-term navigation of a USV, the front camera is frequently disturbed by water splashes, resulting in unclear images and failure of the dynamic region-matching algorithm to converge, leading to the failure of the horizon line extraction. In addition, as the endurance mileage of USV becomes longer, errors in the horizon line extracted by the linear fitting algorithm at the early stage may accumulate, requiring manual correction during the journey. The above two aspects are the main shortcomings of our method. In the future, we will focus on improving the anti-interference and stability of our approach. Specifically, we will concentrate on improving the hardware part of the image acquisition device to address the problem of susceptibility to water splash disturbance, and explore new discrete point-fitting methods to improve the long-term stability of the algorithm.

Author Contributions: B.S.: Conceptualization, data curation, formal analysis, investigation, methodology, funding acquisition, validation, visualization, writing—original draft, and writing—review and editing. C.W.: Conceptualization, resources, software, and writing—review and editing. Y.D.: Conceptualization, resources, software, funding acquisition, and writing—review and editing. J.G.: Measurement, investigation, methodology, and software. Z.Z.: Investigation, methodology, and software. Y.L.: Conceptualization, formal analysis, funding acquisition, methodology, project administration, resources, software, supervision, and writing—review and editing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the Natural Science Foundation of China (No. 52201363), Natural Science Foundation Hubei Province (No. 2020CFB306), Hubei Provincial Education Department Scientific Research Program Project (Nos. Q20222202, Q20212204), Ideological and Political Department Project of Hubei Province (No. 21Q210), and Hubei University of Economics Research and Cultivation Key Project (No. XJZD202106).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

**Data Availability Statement:** The datasets generated and analysed during this study are available from the corresponding author upon request.

Acknowledgments: We thank the Hubei University of Economics and Hubei Minzu University for their support.

**Conflicts of Interest:** The authors declare no conflicts of interest. The funders had no role in the design of this study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

# References

- 1. Zhang, Y.; Li, Q.; Zang, F. Ship detection for visual maritime surveillance from non-stationary platforms. *Ocean. Eng.* **2017**, 141, 53–63. [CrossRef]
- Liang, D.; Liang, Y. Horizon detection from electro-optical sensors under maritime environment. *IEEE Trans. Instrum. Meas.* 2019, 69, 45–53. [CrossRef]
- Kahveci, N.E.; Ioannou, P.A. Adaptive steering control for uncertain ship dynamics and stability analysis ScienceDirect. *Automatica* 2013, 49, 685–697. [CrossRef]
- 4. Jin, X.; Zhang, G.; Ma, Y.; Peng, X.; Shi, B. Cooperative multi-task traversing with complex marine environment for multiple unmanned surface vehicles inspired by membrane computing. *Ocean. Eng.* **2022**, *266*, 112586. [CrossRef]
- Zhan, W.; Xiao, C.; Yuan, H.; Wen, Y. Effective Waterline detection for unmanned surface vehicles in inland water. In Proceedings of the 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA), Montreal, QC, Canada, 28 November–1 December 2017.
- 6. Shi, B.; Su, Y.; Lian, C.; Xiong, C.; Long, Y.; Gong, C. Obstacle type recognition in visual images via dilated convolutional neural network for unmanned surface vehicles. *J. Navig.* **2022**, *75*, 1–18. [CrossRef]
- 7. Praczyk, T. A quick algorithm for horizon line detection in marine images. J. Mar. Sci. Technol. 2018, 2, 164–177. [CrossRef]
- 8. Long, Y.; Liu, S.; Qiu, D.; Li, C.; Guo, X.; Shi, B.; AbouOmar, M.S. Local Path Planning with Multiple Constraints for USV Based on Improved Bacterial Foraging Optimization Algorithm. *J. Mar. Sci. Eng.* **2023**, *11*, 489. [CrossRef]
- 9. Steccanella, L.; Bloisi, D.; Blum, J.; Farinelli, A. Deep learning waterline detection for low-cost autonomous boats. *Adv. Intell. Syst. Comput.* **2019**, *867*, 613–625.
- 10. Wiehle, S.; Lehner, S. Automated waterline detection in the Wadden Sea using high-resolution Terra SAR-X images. *J. Sens.* 2015, 2015, 450857. [CrossRef]
- 11. Jeong, C.; Yang, H.S.; Moon, K. A novel approach for detecting the horizon using a convolutional neural network and multi-scale edge detection. *Multidimens. Syst. Signal Process.* **2019**, *30*, 1187–1204. [CrossRef]
- 12. Wei, Y.; He, Y. Shadow verification based waterline detection for unmanned surface vehicles deployed in complicated natural environment. *Int. J. Adv. Robot. Syst.* 2018, 15, 172988141881873. [CrossRef]
- Sarah, W.; Ruth, M.; Eileen, K.; Simon, S. A review of recent innovation in psychosocial interventions for reducing violence and aggression in adults using a horizon scanning approach. *Aggress. Violent Behav.* 2022, 62, 101685.
- 14. Antonio, V.; José Ignacio, V.; Paula, G. Measurement of radar horizon in a real marine environment and its influence on the reduction of interferences. *Measurement* **2018**, *122*, 186–191.
- 15. Shi, B.; Su, Y.; Wang, C.; Wan, L.; Luo, Y. Study on intelligent collision avoidance and recovery path planning system for the waterjet-propelled unmanned surface vehicle. *Ocean. Eng.* **2019**, *182*, 489-498. [CrossRef]
- 16. Shi, B.; Su, Y.; Zhang, D.; Wang, C.; Mahmoud, S.A. Research on trajectory reconstruction method using automatic identification system data for unmanned surface vessel. *IEEE Access* 2019, 7, 170374–170384. [CrossRef]
- 17. Shi, B.; Guo, J.; Wang, C.; Su, Y.; Di Y.; Mahmoud, S.A. Research on the visual image-based complexity perception method of autonomous navigation scenes for unmanned surface vehicles. *Sci. Rep.* **2022**, *12*, 10370. [CrossRef]
- Prasad, D.K.; Rajan, D.; Rachmawati, L. Video processing from electro-optical sensors for object detection and tracking in a maritime environment: A survey. *IEEE Trans. Intell. Transp. Syst.* 2017, 18, 1993–2016. [CrossRef]
- Bovcon, B.; Mandeljc, R.; Pers J. Stereo obstacle detection for unmanned surface vehicles by IMU-assisted semantic segmentation. *Robot. Auton. Syst.* 2018, 104, 1–13. [CrossRef]
- Kong, X.; Liu, L.; Qian, Y.; Cui, M. Automatic detection of sea-sky horizon line and small targets in maritime infrared imagery. *Infrared Phys. Technol.* 2016, 76, 185–199. [CrossRef]
- Sungho, K. Sea-Based infrared scene interpretation by background type classification and coastal region detection for small target detection. Sensors 2015, 15, 24487–24513.
- 22. Fefilatyev, S.; Goldgof, D.; Shreve, M. Detection and tracking of ships in open sea with rapidly moving buoy-mounted camera system. *Ocean Eng.* **2012**, 54, 1–12. [CrossRef]
- Prasad, D.K.; Deepu, R.; Lily, R. MuSCoWERT: Multi-scale consistence of weighted edge Radon transform for horizon detection in maritime images. J. Opt. Soc. Am. A Opt. Image Sci. Vis. 2016, 33, 2491–2500. [CrossRef] [PubMed]

- 24. Wang, B.; Su, Y.; Wan, L. A sea-sky line detection method for unmanned surface vehicles based on gradient saliency. *Sensors* 2016, *16*, 543–561. [CrossRef]
- 25. Nicola, C.; Enrico, P.; Emanuele, M.; Alberto, P. Receding horizon task and motion planning in changing environments. *Robot. Auton. Syst.* **2021**, *145*, 103863.
- Shen, Y.; Krusienski, D.; Li, J. A hierarchical horizon detection algorithm. *IEEE Geosci. Remote Sens. Lett.* 2013, 10, 111–114. [CrossRef]
- 27. Liu, C.; Zhang, Y.; Tan, K. Sensor fusion method for horizon detection from an aircraft in low visibility conditions. *IEEE Trans. Instrum. Meas.* **2014**, *63*, 620–627. [CrossRef]
- Dai, Z.; Yi, J.; Zhang, H.; Wang, D.; ; Huang, X.; Ma, C. CODNet: A center and orientation detection network for power line following navigation. *IEEE Geosci. Remote Sens. Lett.* 2022, 19, 8014805. [CrossRef]
- Vandaele, R.; Sarah, L.; Varun, O. Automated water segmentation and river level detection on camera images using transfer learning. In *Pattern* Recognition: 42nd DAGM German Conference, DAGM GCPR 2020, Tübingen, Germany, 28 September–1 October 2020; Springer: Berlin/Heidelberg, Germany, 2021; Volume 12544, pp. 232–245.
- 30. Erceg, S.; Erceg, B.; Polach, F.; Ehlers, S. A simulation approach for local ice loads on ship structures in level ice. *Mar. Struct.* 2022, *81*, 103117. [CrossRef]
- 31. Timotheatos, S.; Piperakis, S.; Trahanias, P. Visual horizon line detection for uav navigation. Int. J. Mech. Control 2019, 20, 35–51.
- 32. Zhan, W.; Xiao, C.; Wen, Y. Autonomous visual perception for unmanned surface vehicle navigation in an unknown environment. *Sensors* 2019, *19*, 2216–2228. [CrossRef]
- Touqeer, A.; George, B.; Monica, N.; Ara, N.; Terry, F. Horizon line detection using supervised learning and edge cues. Comput. Vis. Image Underst. 2020, 191, 102879.
- 34. Haralick, R.M.; Shanmugam, K.; Dinstein, I. Textural features for image classification. *IEEE Trans. Syst. Man Cybern.* 1973, 6, 610–621. [CrossRef]
- Gao, S.; Zhao, M.; Zou, Y.; Zhang, L. Improved algorithm about subpixel edge detection of image based on Zernike orthogonal moments. *Acta Autom. Sin.* 2008, 34, 1163–1168. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.