PAPER Coarse-to-Fine Evolutionary Method for Fast Horizon Detection in Maritime Images

Uuganbayar GANBOLD^{†a)}, Junya SATO^{††b)}, Nonmembers, and Takuya AKASHI^{†c)}, Member

SUMMARY Horizon detection is useful in maritime image processing for various purposes, such as estimation of camera orientation, registration of consecutive frames, and restriction of the object search region. Existing horizon detection methods are based on edge extraction. For accuracy, they use multiple images, which are filtered with different filter sizes. However, this increases the processing time. In addition, these methods are not robust to blurting. Therefore, we developed a horizon detection method without extracting the candidates from the edge information by formulating the horizon detection problem as a global optimization problem. A horizon line in an image plane was represented by two parameters, which were optimized by an evolutionary algorithm (genetic algorithm). Thus, the local and global features of a horizon were concurrently utilized in the optimization process, which was accelerated by applying a coarse-to-fine strategy. As a result, we could detect the horizon line on high-resolution maritime images in about 50ms. The performance of the proposed method was tested on 49 videos of the Singapore marine dataset and the Buoy dataset, which contain over 16000 frames under different scenarios. Experimental results show that the proposed method can achieve higher accuracy than state-ofthe-art methods.

key words: horizon detection, genetic algorithm, coarse-to-fine approach, global and local features

1. Introduction

Autonomous surface vehicles have been developed for applications such as environmental protection and coastal guard [1]. These vehicles usually utilize radar, light detection and ranging, inertial systems, and GPS for navigation and obstacle detection [2]. With the rapid development of computer vision and the increase in camera resolution, information from cameras is used to solve various problems, such as object detection, recognition, and tracking. Videos captured by optical systems are valuable for autonomous surface vehicles to perceive surrounding information for obstacle detection, remote control, and estimation of the spatial orientation. Video processing in maritime scenarios is quite challenging because of random camera shaking caused by waves and the processing time of high-resolution images.

To address the above challenges, a horizon on maritime images is used for the following purposes: estimation of the

[†]The authors are with Graduate School of Engineering, Iwate University, Morioka-shi, 020–8551 Japan.

a) E-mail: uuganbayar@scv.cis.iwate-u.ac.jp

spatial orientation of a camera/ship and image registration, which aligns consecutive frames into a one-coordinate system for object detection and object tracking [3]–[5]. In addition, the horizon is used to determine the region of interest to reduce the processing time and false detection [5], [6].

Therefore, the accurate detection of the horizon line is critically important for maritime image processing as an initial step. However, the detection of the horizon line faces several issues caused by complex maritime environments, such as waves, ocean color, light changing, and partial occlusions by maritime objects. Another challenge is that the pixels of the horizon line features are fewer than those of the entire image [7]. Thus, an accurate extraction of the horizon line features is required.

In the last two decades, several approaches have been proposed for detecting the horizon line in maritime environments. In the maritime scenario, the horizon line is generally represented as a straight line because the sea surface can be assumed to be flat and vanishes into a line on the image plane. Related works can be divided into local feature-based methods [8], [9], global feature-based methods [10], [11] and hybrid methods [5], [12]–[14]. The local feature-based methods extract a line segment of the horizon from local features such as edge information using line detection techniques such as Hough, and Radon. Global feature-based methods optimize the horizon line parameters using the horizon features on the entire image. Although hybrid methods, which utilize both global and local features, can achieve higher accuracy, the estimation of these features for all horizon candidates requires considerable computation time.

In this work, we formulate the horizon line detection as an optimization problem and propose a new method called coarse-to-fine evolutionary method, abbreviated as CFEM. As suggested by the name, we adopted the genetic algorithm (GA) to optimize the parameters of the horizon line. The GA is an evolutionary algorithm that can optimize the parameters using a criterion, which concurrently assumes global and local features. In addition, a coarse-to-fine approach was adopted to accelerate the processing. First, the GA optimizes the parameters of the horizon line on down-sampled image of input image by using an optimization criterion that utilizes both global and local features. Then, the coarsely estimated parameters from the previous step are fine-tuned on a higher resolution image within a narrow range of line parameters with the GA using local feature estimation.

Our main contributions are:

Manuscript received March 23, 2021.

Manuscript revised July 14, 2021.

Manuscript publicized September 8, 2021.

^{††}The author is with Faculty of Engineering, Gifu University, Gifu-shi, 501–1193 Japan.

b) E-mail: jsato@gifu-u.ac.jp

c) E-mail: akashi@iwate-u.ac.jp (Corresponding author) DOI: 10.1587/transinf.2021EDP7064

- A heuristic approach, which is independent from edgeextraction unlike existing methods;
- A combination of the GA and coarse-to-fine approach for fast processing;
- Integration of local and global features in the optimization criteria for high performance;
- A quick estimation of the local and global features for fast processing.

In Sect. 2, we discuss related works on horizon detection and the GA. Section 3 outlines the proposed method for horizon detection. The experimental results and details of the parameters are discussed in Sect. 4. Finally, we conclude with an overview of this study in Sect. 5.

2. Related Work

For most horizon detection methods in the maritime scene, the horizon is considered a straight line. Related works can be classified as local and global feature-based methods. The local feature-based methods [8], [9] identify a prominent line as a horizon line using line segment extraction methods from an edge information of the input image. Hough transform [8], Radon transform [12], and line segmentation algorithms are mostly used to extract line features from an edge information. Although the horizon line can be detected in real-time [8], [9], it cannot be established by a prominent line owing to oceanic color differences and noise caused by waves and blurring of an input image. Another limitation of this approach is that it is difficult to distinguish the horizon line from the extracted lines [7].

Several methods that improve the local feature-based method have been introduced. Fefilatyev et al. [5] introduced a candidate-first approach. First, a few candidate lines are selected by the Hough transform based on an edge map. Then, a global feature of the horizon line is used to find an optimal solution from candidates, and calculates the difference of the color distributions in two regions divided by the candidate line. A similar method was proposed by Lipschutz et al. [15], in which a color histogram was used to model the color-space distribution of two regions to reduce the processing time. Prasad et al. [12], [16] used multi-scale edge extraction approaches to extract edges from multiple images filtered with different filter sizes and accurately extract the edge information of the horizon line. In MSCM-LiFe [16], Canny edge detection and Hough transform are used to select the first modal candidates on multiple images filtered with different filter sizes and the maximum intensity variation is calculated to select the second modal candidates. Then, to select the final solution from the candidates, the goodness of the two modals and the geometric proximity of the pair of modals are measured. In MusCoWERT [12], a weighted edge map is computed for each image filtered with a multi-scale filter and candidates are selected by the Radon transform from the weighted edge map. Then a voting system is used for the final solution from all the candidates. Jeong at el. [17] combined multi-scale edge detection and convolutional neural network for reliable edge extraction. Then, they used linear curve fitting along with median filtering to find an optimal horizon line. In experimental results of [17], the above multi-scale approaches achieved the highest accuracy. But, they required expensive computation for real-time processing in high-resolution images for detecting the optimal horizon line.

In addition, global feature-based methods [8], [10], [11], [15] have also been proposed. The global feature is used as an optimization criterion to optimize the horizon line parameters. The horizon line can be represented by two parameters, orientation and position. Ettinger et al. [10], [11] considered that the horizon line divides an image into two different regions, namely sky and sea, thus the difference of the two regions was used as an optimization criteria. To find the optimal parameters of the horizon line, they calculated the statistical distance metrics of distributions in the two regions for all combinations of the horizon line parameters. These methods are not dependent on edge information and they can detect the horizon line on blurred and noisy images. They achieved real-time processing on a low-resolution image using a coarse-to-fine approach. However, the result of [12] shows that this method requires tens of seconds to detect the horizon line on the high-resolution image because it requires calculation of the statistical distribution of the two regions for all candidates and uses exhaustive search to optimize the parameters.

Recently, sky-sea region extraction methods have been proposed to reduce the processing time by restricting the search region [13], [14]. Liang et al. [14] extracted the skysea region using probabilities that were distributed on vertically divided regions by weighted textures. Then, candidate lines were extracted from the sky-sea region using an edge detector and Hough transform. Finally, a voting method was applied to obtain the final solution. The extraction of the sea-sky regions reduced the processing time and false detection. However, only part of the horizon line is obtained when there are occlusions near the horizon and a large angle gradient along the horizontal axis. Jeong et al. [13] also vertically divided an image into several regions and extracted the sky-sea region using the difference between the color distributions of the consecutive regions. The difference between the two regions was calculated by the Bhattacharyya distance, and a region with the largest distance was selected as the sky-sea region. Then, multi-scale edge detection was applied to the sky-sea region and merged into one edge map. Finally, the Hough and least-squares methods were sequentially used to find the horizon line.

Except for Ettinger's methods [10], [11], the above methods extract the candidates of horizon using local features as edge information and use consecutive filtering on several stages, which are based on features of the horizon. One limitation of this approach is that filtered candidates in the previous stage cannot be considered in the next stage, even though these candidates have survival candidates in the filtering of the next stages. Thus, multi-scale approaches have been proposed to extract sufficient candidates in an early stage. However, these methods require along processing time. In addition, Jeong et al. [13] stated that the methods that depend on edge information cannot detect the horizon line when the input image is blurred or the boundary between the sky and sea region is gradually changed.

Therefore, in our previous study [18], we proposed a novel method that optimizes the parameters of the horizon line. For the fast and accurate detection of the horizon, we considered several improvements. First, we used global optimization algorithms to solve the horizon detection problem, and applied GA for efficient optimization to reduce the processing time. Second, instead of considering all pixels of an image for each combination of the horizon line parameters, such as in [10], we defined the local features of the horizon line using a vanishing line characteristic. The result of [18] shows that utilization of the local features of the horizon for optimization criteria can reduce the processing time and can increase an accuracy. However, this method has limitation in certain scenarios such as for mostly occluded horizon and drastic changes in the color of the sea.

Our study extends the previous method [18] for improving accuracy by adding the factor of global feature to the optimization criterion. Consequently, the accuracy of the proposed method was improved in frames, whose sea color drastically changes. One advantage of optimizationbased approaches is that several optimization criterion can be used as fitness functions. In our case, local and global features were utilized concurrently in the fitness functions.

3. Proposed Method

3.1 Overview

In this section, we introduce a horizon detection method called coarse-to-fine evolutionary method. A diagram of the proposed method is presented in Fig. 1. The proposed method consists of three steps. First, an input image is down-sampled and a probability map of the horizon is created during the pre-processing stage. Before downsampling, the input image is filtered by a Gaussian filter. The probability map of the horizon is used for optimization criterion as a factor of the global feature of the horizon in the next step. Subsequently, coarse-to-fine optimization is performed for detect the horizon line. Coarse-to-fine approaches are widely used in computer vision to improve the efficiency [19], [20]. In the coarse-optimization stage, the parameters of the horizon are roughly optimized on the gray-scale image of the down-sampled image. The global and local features are concurrently utilized in the optimization criterion. Finally, the fine-optimization of the parameters is performed at high-resolution to improve the accuracy. The fine-optimization stage is performed in a narrow region close to the parameters that are roughly optimized by coarse optimization.

For quick optimization of the horizon line parameters, GA is used, which provides optimization utilizing fewer combinations of parameters compared to exhaustive search. The GA is broadly applied to efficiently solve combinatorial optimization problems in computer vision such as template matching and object detection [21]–[23].

For all optimization methods, an optimization criterion significantly affects the processing time and accuracy. We introduced the fast estimation of local and global features for the optimization criteria. In the next subsection, for horizon detection, a GA and utilization of local and global features are presented.

3.2 Optimization by Genetic Algorithm

The horizon is projected onto a single line in an image plane. Therefore, the problem of the horizon detection can be regarded as a global optimization problem. GA is a popular evolutionary algorithm for global optimization and has been applied to various combinatorial optimization problems in computer vision [21], [23]. Thus, we used the GA to optimize the parameters of horizon line in both coarse and fine optimization. A simple GA requires the generation of an initial population of individuals for every frame of a sequence. Each individual within the population represents a possible solution, a so-called candidate. For every iteration, the individuals of the population are evaluated by a fitness function, and then updated by genetic operations such as selection, crossover, and mutation.

The generation of an initial population for each frame of a sequence is time-consuming. Therefore, we used evolutionary video processing (EVP) [24]. It generates the initial population once at the initial frame of a sequence and inherits a population of the last generation into an initial generation of the next frame. Akashi et al. [24] stated that evolutionary video processing can improve optimization ac-



Fig. 1 Diagram of the proposed method.

curacy and reduce processing time. In addition, we used an elite saving strategy to improve the efficiency of GA, which is the process of preserving previous high-performance solutions from the current generation to the next. To solve the combinatorial problem with GA, a representation of a solution to the problem as chromosomes and a formulation of the fitness function must be determined, which are essential for optimization accuracy and speed, and will be explained in the following sections.

3.2.1 Representation of Parameters in Chromosomes

The chromosome of an individual is often represented by bit strings because of it is faster than the real coded GA in the processing of crossover and mutation operations [21]. The chromosome contains a set of parameters, that are necessary to solve a problem. In our case, the horizon can be a straight line, and the ground truth of the horizon line is given by a straight line on the datasets [4], [25]. Hence, a horizon can be represented by two parameters of the straight line: vertical position *Y* and orientation angle θ . An adjustment values for the two parameters decoded as a chromosome of individuals. These were the orientation ρ and the height adjustments of the horizon line λ . *Y* and θ were calculated as follows,

$$Y = (Y_0 + \lambda),\tag{1}$$

$$\theta = (\theta_0 + \rho), \tag{2}$$

where Y_0 is the initial vertical position, and θ_0 is the initial orientation angle. In the coarse-step optimization, the initial value of the vertical position $Y_0 = H/2$ was located in the center of an image, and the initial value of orientation $\theta_0 = 0$ was set parallel to the horizontal edge of the image. In the fine-step optimization, the initial parameters of the horizon line were the elite-candidate-line parameters of the coarse-optimization stage as follows,

$$Y_0 = Y_e \times S,\tag{3}$$

$$\theta_0 = \theta_e,\tag{4}$$

where Y_e and θ_e are the elite-candidate-line parameters of the coarse-optimization stage, and S is a scale used to down-sample an input image into a low-resolution image.

3.2.2 Designing of the Fitness Function

In a GA, a fitness function is used to guide the simulation toward an optimal solution, and it evaluates the goodness of each individual. Therefore, designing the fitness function is very important for quick convergence on an appropriate solution, and it has a significant impact on computational time. The fitness function should precisely evaluate how to fit a given solution and should be fast to compute. Existing optimization-based [11], [15] methods for horizon detection use an optimization criterion, which calculates the color distribution across all pixels of an image. Thus, these methods require a significant amount of time to achieve an accurate detection of the horizon line. In our previous work [18], we determined the local feature of the horizon line, which was used in the fitness function. As a result, the processing speed and accuracy of horizon detection was improved. However, the method in [18] failed in certain scenarios, such as changes in the color of the sea and the mostly occluded horizon line. To improve the accuracy in the above scenarios, we assumed a global feature factor in the fitness function. As mentioned before, the global feature estimation for each candidate is time-consuming because it covers a wide area of the input image. Thus, we created a probability map of the horizon line in the pre-processing stage and used it as a global feature factor in the fitness function. In the coarse step, the fitness function *F* was designed with the global feature factor *G* and local feature factor *L* as follows,

$$F^{c}(Y,\theta) = G(Y) \times L(Y,\theta).$$
(5)

In the fine-tuning step, the global feature factor effects were weaker than those of the local feature factor for the optimization because the position of the horizon line was roughly determined in the coarse-optimization. Therefore, we assumed only the local feature factor in the fine-tuning step into the fitness function as follows,

$$F^{f}(Y,\theta) = L(Y,\theta).$$
(6)

3.3 Global Feature

In the pre-processing stage, an input image is filtered by the Gaussian filter and downsampled. The downsampled image was used for global feature estimation and coarse stage optimization. Global feature estimation covers all pixels of image, but it is computationally expensive. Thus, we create a probability map of the horizon, which indicates the probability of the horizon at each row of an image. The map was used as a global feature factor in the fitness function. A textural feature and a color feature were used to extract the region that contains the horizon [13], [14]. Although both features are significant in extraction process for input images without blur, the textural feature is not applicable, where the color feature performs better. We used a color feature to determine the probability of existence of the horizon, similar to [13]. The creation steps of the probability map of the horizon are shown in Fig. 2. First, the image were divided into nine regions (I = 9). The height of these regions h was a fifth of the image height H, and 50 percent of the regions overlapped with neighboring regions as shown in Fig. 2 (b). The region with larger change in color distribution compared with neighboring regions has a higher probability of containing the horizon. A color histogram was calculated for each region to evaluate the color distribution due to processing speed considerations, and N = 64 bins were used for each color. To compare the histograms of two regions, the Hellinger distance was calculated as follows,



Fig.2 Creation steps of the probability map of horizon. (a) Downsampled image, (b) divided regions, (c) existence probability of horizon for each region, (d) probability map of horizon.



Fig. 3 (a) Description of the horizon line in image plane, where bold line *h* indicates horizon line, the line *X* is line which intersects with *h* on the image plane *I* passes through the camera center *C* and is parallel to the sea surface plane *O*, the p_j and p_i pixels of image, the s_j and s_i are regions of sea surface those are projected into p_j and p_i respectively. (b) Local features of the horizon line in the sample frame and can be estimated in a narrow range close to the horizon.

$$D(R_n, R_m) = \sqrt{1 - \frac{\sum_{j=0}^{N-1} \sqrt{H_n(j) \times H_m(j)}}{\sqrt{\sum_{j=0}^{N-1} H_n(j) \times \sum_{j=0}^{N-1} H_m(j)}}, \quad (7)$$

where H_n and H_m are the histograms of neighboring regions R_n and R_m , respectively. H(j) indicates the *j*-th bin of the histogram. Then, as shown in Fig. 2 (c), the existence probability of horizon was calculated for each region as follows,

$$P(i) = \begin{cases} D(R_i, R_{i+1}) & \text{if } i = 0\\ D(R_i, R_{i-1}) & \text{if } i = I - 1\\ \frac{D(R_i, R_{i-1}) + D(R_i, R_{i+1})}{2} & \text{otherwise} \end{cases}$$
(8)

where I is the number of the divided regions and nine in this research. Subsequently, a probability map of horizon was created for each row of the image as shown in Fig. 2 (d). The probability of horizon at *Y*-th row of the image is calculated as follows,

$$G(Y) = \begin{cases} P(0) & \text{if } Y < h/2\\ P(I-1) & \text{if } Y > (H-h/2) \\ \frac{P(\lfloor \frac{Y}{(h/2)} \rfloor - 1) + P(\lfloor \frac{Y}{(h/2)} \rfloor)}{2} & \text{otherwise} \end{cases}$$
(9)

3.4 Local Features

For the fast evaluation of a candidate line, we defined the local features of the horizon line using a vanishing line characteristic. As shown in Fig. 3 (a), we propose three essential characteristics. First, the horizon line is a straight line. Second, an appearance of the above horizon line is different from that of the side below the horizon line. Third, regions close to the horizon line tend to be texture-less. The reason is that the wide area of the sea surface is projected to a few pixels near the horizontal. Thus, the regions, which are close to the horizon line, have texture-less. As shown in Fig. 3 (b), these characteristics can be estimated in the narrow regions close to the horizon line as local features. Reducing the area of the region, which evaluates the candidate, can reduce the processing time of the optimization process. The same local feature estimation is used in both the coarse and fine-step optimizations. In addition, the local features are estimated on a gray-scale image using the following equation:

$$L(Y,\theta) = \frac{1}{N_c \times W_{\max}} \times \sum_{j=1}^{Z} \sum_{i=0}^{K} C(Y,\theta, j, x_i),$$
 (10)

where Z is a parameter to control the evaluation range of the local feature, K is the number of samples according to the image width (*W*) and it is K = W/d. *d* is the sampling step and x_i follows $x_i = i \times d$. *C* is a function to estimate the local feature of the horizon line and it consists of the following three functions:

$$C(Y, \theta, j, x_i) = AB(Y, \theta, j, x_i) + A(Y, \theta, j, x_i) + B(Y, \theta, j, x_i),$$
(11)

where AB is a function that estimates the difference in appearance above and below the horizon area. A and B are functions that estimate the texturelessness of the above and below-side regions of the horizon line, respectively. To estimate the local features at given Y, θ , j, and x_i , four pixels (S1, S2, M1, and M2) were assumed, as shown in Fig. 4. The S1 and M1 points are symmetric with respect to the candidate line and symmetric to point O1. The S2 and M2 points are symmetric with respect to the candidate line and symmetric to point O2. O1 and O2 are points on the candidate line at a given x_i and $x_i - d/2$, respectively. When the four pixels are in the image plane, the function C is calculated using Eq. (11) otherwise C is 0. N_c is the total number of combinations of Y, θ , j, and x_i , when the four pixels are in the image plane. A W_{max} is the total value of the weights of the features used in functions AB, A, and B. Function AB, A, and B are as follows:

$$AB(Y, \theta, j, x_i) = \begin{cases} w_1 & \text{if } |I_{S1} - I_{M1}| > T\\ w_2 & \text{otherwise} \end{cases},$$
(12)

$$A(Y, \theta, j, x_i) = \begin{cases} w_3 & \text{if } |I_{S1} - I_{S2}| < T\\ 0 & \text{otherwise} \end{cases}$$
, (13)

$$B(Y, \theta, j, x_i) = \begin{cases} w_4 & \text{if } |I_{M1} - I_{M2}| < T \\ 0 & \text{otherwise} \end{cases}$$
(14)

Here, *AB* counts symmetric points with respect to the candidate line with a different color. Functions *A* and *B* count the points that have similar neighboring points along with the candidate line. I_{S1} , I_{S2} , I_{M1} , and I_{M2} are the pixel values at *S* 1, *S* 2, *M*1 and *M*2, respectively. As shown in Fig. 4, the distance from the points *S* 1, *M*1, *S* 2, and *M*2 into the candidate line is *j*. A threshold value of *T* is used to evaluate



Fig. 4 Local feature estimation. Four pixels are used for local feature estimation at given *Y*, θ , *j*, and *x_i*. These are *S*1, *S*2, *M*1, and *M*2. The red line describes a candidate line given by an orientation angle θ and a vertical position *Y*.

whether the points are similar or different. w_1 , w_2 , w_3 , and w_4 are the weights of the features.

4. Experimental Results

4.1 Dataset and Evaluation Criteria

We verified the performance of the proposed method using the Singapore Maritime Dataset (SMD) and Buoy-Dataset (BD), which are publicly available. The SMD consists of onboard and onshore videos. The onboard videos were captured by a camera mounted on a moving board, and the onshore videos are captured by a static camera installed onshore. The videos contain complex maritime scenes that have strong noise caused by wakes and waves, and color changes in the sea. The resolution of the SMD videos was 1920×1080 pixels. The BD consists of videos captured by a camera mounted on a floating buoy with resolution of 800×600 pixels. A challenge for the onboard videos of SMD and BD is the large variation in the orientation and position of the horizon line between adjacent frames. The details of SMD and BD are given in Table 1. The ground truth of the horizon line is given by the vertical position Yand the orientation angle θ .

In previous studies [2], [7], [12]–[14], the performance is commonly evaluated by mean absolute error (MAE) and percentile error at 25th, 50th, and 95th. The 95th percentile error indicates the detection result on complex scenes and it is used to imply that the how detection method is robust and consistent over datasets with great diversity [12], [13]. To compare our results, our study used the same percentile errors. Moreover, MAE was used for analyzing parameter analysis.

4.2 Parameter Setting

The proposed method needs to adjust the values of the parameters. Because they affect the performance, it is necessary to investigate the optimal values of parameters. However, the number of parameters is very large. Hence, we focused on population size, generation size, and threshold of the local features (T) because these have a significant influence on performance. The experimental results are described in Sect. 4.3. The other parameters were empirically

Table 1Details of Datasets [12].

Dataset	Buoy	Singapore maritime		
		Onboard	Onshore	
No. of videos	10	11	28	
No. of frames	996	2772	12604	
Min(Y-mean(Y))	-281.68	-436.30	-13.54	
Max(Y-mean(Y))	307.82	467.86	9.95	
Standard deviation of Y	107.98	145.10	1.52	
$Min(\theta-mean(\theta))$	-15.72	-26.34	-9.99	
$Max(\theta-mean(\theta))$	20.72	12.99	0.51	
Standard deviation of θ	4.40	1.11	0.04	

fixed. The details are described as follows. The population and generation sizes of the GA are 20 and 20 in both coarse and fine optimization stages. The crossover and mutation probabilities are 0.6 and 0.07, respectively, which affect the convergence speed and diversity of the population. The orientation adjustment parameter ρ and height adjustment parameter λ are decoded from the chromosome. In the coarse-optimization stage, the adjustment ranges of each parameters are described as follows,

- Orientation adjustment ρ : $[-\pi/4, \pi/4]$,
- Height adjustment λ : [-3H/5, 3H/5].

To detect the horizon line that is particularly out of an image, we set the range of height adjustment by a value that larger than the height of an image. In coarse optimization, the parameters were roughly optimized. Hence, in the fine-step optimization, the orientation and height adjustment ranges exhibited a reduction of six times and two times, respectively, with the ranges of coarse optimization. The chromosome length has 16 bits because each parameter is represented by eight bits in both optimization stages. We downsampled the input image for global feature estimation and coarse optimization. The scale to downsample was S = 1/4for the BD and S = 1/8 for the SMD. For quick local features estimation, the range parameter to evaluate the local features was Z = 6 and the sampling step was d = 4. In addition, the threshold value was T = 20, and the weights of the local feature were $w_1 = 3$, $w_2 = -2$, $w_3 = 1$, and $w_4 = 2$.

4.3 Effectiveness of EVP and Parameter Analysis

As described in Sect. 3.2, in our coarse-to-fine evolutionary method, the EVP is used for the optimization of HL parameters. In this section, to confirm the effectiveness of EVP in this study, its performance was compared with the simple GA (see Sect. 3.2) and exhaustive search (ES) on the SMD. The same fitness function was used for the EVP and the simple GA, also the same function was used as an evaluation function in the ES. In the experiment of this paper, the same size of candidates was used in both coarse and fine optimization stages. In the EVP and simple GA, the population and generation sizes were the same, 5, 10, 20, 30, 40, and 50. In the ES, we uniformly sampled the search space of two parameters (*Y* and θ) for HL and the sampling resolutions were similarly 5, 10, 20, 30, 40, and 50.

Figure 5 shows the MAE of vertical position and average processing time per frame on various numbers of candidates. The MAE of EVP was smaller than the simple GA and ES in all the candidate sizes. Moreover, the processing times of all methods were almost the same. Therefore, the EVP is effective for HL detection on the video. Although the number of candidates on each stage increased from 20×20 , the accuracy of the EVP increased a little. Considering fast detection of HL, 20 and 20 for the population and generation sizes were optimal combination for the proposed method. Hence, these values were used for comparison experiments with related work.



Fig. 5 Accuracy and processing times (EVP, simple GA, and ES), which were performed by fixing the threshold value of the local features (T = 20) and changing the number of candidates on SMD.



Fig.6 Accuracy of the proposed method, which was performed on two datasets, SMD and BD by fixing the generation and population sizes (set as 20) and changing the threshold value of the local features.

To find the optimal threshold value of the local features (T), various values were set to the proposed method to compare. The results are shown in Fig. 6. The lowest MAEs on BD and SMD were T = 10 and T = 20, respectively. However, while comparing our method with related works, we used the same threshold value T = 20 on all datasets like Ganbold et al. [18].

4.4 Comparison Results and Consideration

The performance of the proposed method was compared with the state-of-the-art HL detection methods, which were compared in [13]. For fair comparison, the same datasets and same evaluation criteria are used in this study. Jeong et al. [13] compared their proposed method on the SMD and BD with state-of-art methods including MusCoW-ERT [12], MSCM-Life [16], the method of Fefilatyev et al. (FGSL) [25], the method of Lipschutz et al. (LHSL) [15] and two methods on [8] those are the Hough method (Hough) and the intensity variation analysis based method (IntV). We briefly summarize the above methods in the related work section. As far as we know, there is no method

	Vertical position error (pixels)			Orientation angle error (degrees)			
	25th percentile	50th percentile	95th percentile	25th percentile	50th percentile	95th percentile	
Jeong ROI [13]	0.51	1.23	3.99	0.05	0.12	0.39	
MusCoWERT [12]	0.54	1.49	8.17	0.06	0.25	0.88	
MSCM-Life [16]	1.16	2.84	505.78	0.17	0.38	5.50	
LHSL [15]	13.78	25.65	507.92	0.88	1.37	6.52	
FGSL [13]	5.28	10.85	581.44	0.67	1.00	3.88	
IntV [8]	13.36	24.89	498.17	0.87	1.35	6.12	
Hough [8]	2.27	221.67	520.34	0.25	1.00	4.57	
Ganbold [18]	0.58	1.26	4.74	0.06	0.14	0.55	
Ours (CFEM)	0.51	1.09	3.83	0.05	0.11	0.44	

 Table 2
 Comparision of horizon detection on onboard videos from the SMD

 Table 3
 Comparision of horizon detection on onshore videos from the SMD

	Vertical position error (pixels)		Orientation angle error (degrees)				
	25th percentile	50th percentile	95th percentile	25th percentile	50th percentile	95th percentile	
Jeong ROI [13]	0.99	2.09	12.87	0.04	0.10	0.67	
MusCoWERT [12]	1.14	2.63	11.41	0.14	0.21	1.07	
MSCM-Life [16]	1.63	3.88	81.59	0.11	0.18	1.14	
LHSL [15]	14.96	27.92	109.00	0.75	1.03	3.86	
FGSL [13]	5.88	11.53	64.70	0.75	1.00	2.87	
IntV [8]	2.08	5.82	39.89	0.14	0.52	5.37	
Hough [8]	3.12	165.02	460.24	0.14	0.36	3.80	
Ganbold [18]	0.83	1.94	32.72	0.03	0.09	0.44	
Ours (CFEM)	0.77	1.95	11.00	0.03	0.07	0.74	

Table 4	- C	Comparision	of	horizon	detection	on	videos	from	the	Buoy
---------	-----	-------------	----	---------	-----------	----	--------	------	-----	------

	Vertical position error (pixels)			Orientation angle error (degrees)			
	25th percentile	50th percentile	95th percentile	25th percentile	50th percentile	95th percentile	
Jeong ROI [13]	0.53	1.07	2.98	0.07	0.15	0.45	
MSCM-Life [16]	1.54	2.97	11.56	0.33	0.57	11.56	
LHSL [15]	0.66	1.50	3.76	0.17	0.33	0.67	
FGSL [13]	0.60	1.35	3.84	0.18	0.36	0.79	
IntV [8]	0.84	1.91	55.06	0.14	0.32	13.24	
Hough [8]	0.77	1.76	4.46	0.18	0.37	0.89	
Ganbold [18]	0.44	0.94	2.74	0.08	0.17	0.59	
Ours (CFEM)	0.37	0.80	2.29	0.06	0.14	0.44	

to use previous frame information.

The statistics of the errors in parameters Y and θ on SMD are listed in Tables 2 and 3. The statistics of the errors in parameters Y and θ on BD are listed in Table 4. The experimental results show that the proposed method performs better on the SMD and BD datasets. In particular, the median positional error and median orientation error of the proposed method were relatively smaller than those of all the compared methods in all datasets. In addition, the proposed method can detect the horizon line when the input image is blurred, as shown in the bottom image of Fig. 8 (d). In the 95th percentile, the orientation angle error of [13] was smaller than that of the proposed method on the SMD, but the position error of the proposed method was smaller than that of the other methods. As shown in Fig.7, the utilization of global features in optimization criteria had the effect of reducing false positives caused by changes in sea color.



Fig.7 Comparison of the detection results. (a) Result of [18]. (b) Result of proposed method.



Fig.8 Sample frames of horizon detection results using the proposed method. (a) the detection results the onboard dataset of SMD. (b) Detection results on the onshore dataset of SMD. (c) Detection results on BD. (d) Detection results on complex scenarios, with occlusion of the horizon and blurred images. The red and blue dashed lines indicate the detection result of the horizon line and the coarse-step optimization result, respectively, and the green line indicates the ground truth.

 Table 5
 Average processing time per frame in seconds

	Onboard	Onshore	Buoy
Jeong ROI [13]	0.07	0.07	0.02
MusCoWERT [12]	9.2	9.5	5.8
MSCM-Life [16]	6.73	6.83	2.26
LHSL [15]	13.75	13.76	2.30
FGSL [13]	36.58	36.63	8.61
IntV [8]	0.30	0.30	0.01
Hough [8]	0.11	0.10	0.01
Ganbold [18]	0.04	0.04	0.02
Ours (CFEM)	0.05	0.05	0.02

Although the vertical position error decreased, orientation angle error increased when the horizon line was mostly occluded by objects, as shown in the bottom row images in Fig. 7.

Average processing times per image by our and comparative methods are provided in Table 5. The results of the Jeong ROI [13], MSCM-Life [16], FGSL [25], LHSL [15], Hough [8], IntV [8] were taken from [13]. They were implemented using Phyton and executed on an Intel E5-1680 CPU. The MusCoWERT was taken from [12] and it was implemented using MATLAB 2015b, and the result was obtained on an Intel i7-3770 CPU. The proposed method and method of Ganbold et al. [18] were implemented using C/C++ and were executed on an Intel i7-3770 CPU. The MusCoWERT, MSCM-Life, FGSL, and LHSL were required several seconds per image. The methods of IntV and Hough were relatively fast, but they were low detection accuracy. The methods of Jeong ROI [13], Ganbold et al. [18], and the proposed method processed the image within one-tenth seconds and they were reliable detection accuracy.

The experimental results demonstrate that the proposed method can detect the HL at high speed with high accuracy, is effective for HL detection on high resolution video data. Example frames of the horizon line detection results for SMD and BD are shown in Fig. 8 which indicates that the proposed method accurately detected the horizon line in various maritime scenes. The detection results for complex scenarios are shown in Fig. 8 (d). The proposed method failed for data in which the horizon line was mostly occluded by objects.

5. Conclusion

In this study, we proposed a novel fast horizon line detection method that optimizes the horizon parameters by using a GA. We also adopted a coarse-to-fine approach to meet realtime processing requirements. In addition, we introduced a fast estimation of global and local feature estimations for quick optimization. Previous methods extracted the candidates of horizon using the edge information and use consecutive filtering to find the final solution. A limitation of these methods is that if the candidates cannot be extracted from edge information in the previous stage, they are not considered in the next stage, even though these candidates are survival candidates in the filtering of the next stages. Unlike these methods, our method is a heuristic optimization-based method and local and global features are concurrently utilized to evaluate each candidate. The proposed method does not extract edge information from multi-scale images, and even for blurred input images, it can detect the horizon line.

The proposed method was tested on the SMD and BD, which are publicly available datasets that contain complex maritime scenes. In addition, we compared the performance of the proposed method with that state-of-the-art methods, which used the same datasets. The experimental results indicated that the proposed method could detect the horizon line more accurately than the compared methods. In particular, the median positional error and median orientation error of the proposed method were relatively smaller than those of all the compared methods in all datasets. The processing speed of our method was approximately 20 f ps for highresolution images. However, the proposed method failed in scenarios in which the horizon line was mostly occluded by objects as shown at the top of Fig. 8 (d). The mostly occluded case is out of range of the proposed method. In future work, we plan to explore other optimization criteria and features for robust detection of HL in complex scenarios such as the horizon is mostly occluded and the coastal line is visible.

References

- L. Wang, Q. Wu, J. Liu, S. Li, and R.R. Negenborn, "State-of-the-art research on motion control of maritime autonomous surface ships," Journal of Marine Science and Engineering, vol.7, no.12, p.438, 2019.
- [2] A. Stateczny, W. Kazimierski, P. Burdziakowski, W. Motyl, and M. Wisniewska, "Shore construction detection by automotive radar for the needs of autonomous surface vehicle navigation," ISPRS International Journal of Geo-Information, vol.8, no.2, p.80, 2019.
- [3] Y. Wang, R. Tan, G. Xing, J. Wang, X. Tan, X. Liu, and X. Chang, "Aquatic debris monitoring using smartphone-based robotic sensors," IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks, pp.13–24, IEEE, 2014.
- [4] D.K. Prasad, D. Rajan, L. Rachmawati, E. Rajabally, and C. Quek, "Video processing from electro-optical sensors for object detection and tracking in a maritime environment: a survey," IEEE Transactions on Intelligent Transportation Systems, vol.18, no.8, pp.1993–2016, 2017.
- [5] S. Fefilatyev, D. Goldgof, M. Shreve, and C. Lembke, "Detection and tracking of ships in open sea with rapidly moving buoy-mounted camera system," Ocean Engineering, vol.54, pp.1–12, 2012.
- [6] H. Bouma, D.J.J. de Lange, S.P. van den Broek, R.A. Kemp, and P.B. Schwering, "Automatic detection of small surface targets with electro-optical sensors in a harbor environment," Electro-Optical Remote Sensing, Photonic Technologies, and Applications II, p.711402, International Society for Optics and Photonics, 2008.
- [7] Y. Sun and L. Fu, "Coarse-fine-stitched: A robust maritime horizon line detection method for unmanned surface vehicle applications," Sensors, vol.18, no.9, p.2825, 2018.
- [8] E. Gershikov, T. Libe, and S. Kosolapov, "Horizon line detection in marine images: which method to choose?," International Journal on Advances in Intelligent Systems, vol.6, no.1, pp.79–88, 2013.
- [9] G.-Q. Bao, S.-S. Xiong, and Z.-Y. Zhou, "Vision-based horizon extraction for micro air vehicle flight control," IEEE Transactions on Instrumentation and Measurement, vol.54, no.3, pp.1067–1072, 2005.
- [10] S.M. Ettinger, M.C. Nechyba, P.G. Ifju, and M. Waszak, "Towards flight autonomy: Vision-based horizon detection for micro air vehicles," Florida Conference on Recent Advances in Robotics, 2002.
- [11] S.M. Ettinger, M.C. Nechyba, P.G. Ifju, and M. Waszak, "Vision-

guided flight stability and control for micro air vehicles," Advanced Robotics, vol.17, no.7, pp.617–640, 2003.

- [12] D.K. Prasad, D. Rajan, L. Rachmawati, E. Rajabally, and C. Quek, "Muscowert: multi-scale consistence of weighted edge radon transform for horizon detection in maritime images," JOSA A, vol.33, no.12, pp.2491–2500, 2016.
- [13] C.Y. Jeong, H.S. Yang, and K. Moon, "Fast horizon detection in maritime images using region-of-interest," International Journal of Distributed Sensor Networks, vol.14, no.7, p.1550147718790753, 2018.
- [14] D. Liang and Y. Liang, "Horizon detection from electro-optical sensors under maritime environment," IEEE Transactions on Instrumentation and Measurement, vol.69, no.1, pp.45–53, 2019.
- [15] I. Lipschutz, E. Gershikov, and B. Milgrom, "New methods for horizon line detection in infrared and visible sea images," Int. J. Comput. Eng. Res, vol.3, no.3, pp.1197–1215, 2013.
- [16] D.K. Prasad, D. Rajan, C.K. Prasath, L. Rachmawati, E. Rajabally, and C. Quek, "Mscm-life: multi-scale cross modal linear feature for horizon detection in maritime images," 2016 IEEE region 10 conference (TENCON), pp.1366–1370, IEEE, 2016.
- [17] C. Jeong, H.S. Yang, and K. Moon, "A novel approach for detecting the horizon using a convolutional neural network and multi-scale edge detection," Multidimensional Systems and Signal Processing, vol.30, no.3, pp.1187–1204, 2019.
- [18] U. Ganbold and T. Akashi, "The real-time reliable detection of the horizon line on high-resolution maritime images for unmanned surface-vehicle," 2020 International Conference on Cyberworlds (CW), pp.204–210, IEEE, 2020.
- [19] M. Atiquzzaman, "Coarse-to-fine search technique to detect circles in images," The International Journal of Advanced Manufacturing Technology, vol.15, no.2, pp.96–102, 1999.
- [20] C. Spence, J. Pearson, and J. Bergen, "Coarse-to-fine image search using neural networks," Advances in Neural Information Processing Systems, vol.7, pp.981–988, 1994.
- [21] J. Sato and T. Akashi, "High-speed multiview face localization and tracking with a minimum bounding box using genetic algorithm," IEEJ Transactions on Electrical and Electronic Engineering, vol.12, no.5, pp.736–743, 2017.
- [22] U. Ganbold, T. Akashi, C. Zhang, H. Tomita, and H. Kubota, "Horizontal line detection using genetic algorithm," International Workshop on Advanced Image Technology (IWAIT) 2019, p.110490S, International Society for Optics and Photonics, 2019.
- [23] T. Nakane, N. Bold, H. Sun, X. Lu, T. Akashi, and C. Zhang, "Application of evolutionary and swarm optimization in computer vision: a literature survey," IPSJ Transactions on Computer Vision and Applications, vol.12, no.1, pp.1–34, 2020.
- [24] T. Akashi, Y. Wakasa, K. Tanaka, S. Karungaru, and M. Fukumi, "Using genetic algorithm for eye detection and tracking in video sequence," Journal of Systemics, Cybernetics and Informatics, vol.5, no.2, pp.72–78, 2007.
- [25] S. Fefilatyev, M. Shreve, and D. Goldgof, "Detection of the vanishing line of the ocean surface from pairs of scale-invariant keypoints," International Conference on Image Analysis and Processing, pp.161–169, Springer, 2013.



Uuganbayar Ganbold received her B.E. from National University of Mongolia in 2007 and M.E. degree in Engineering from Iwate University, Japan, in 2019. He is now Ph.D. candidate at Design and Media Technology, Graduate School of Engineering, Iwate University. His research interests include feature matching and optimization in computer vision problems. He is a student member of the IEEE.



Junya Sato received B.E., M.E., and Ph.D. degrees in Engineering from Iwate University, Japan, in 2012, 2014, and 2017, respectively. Since 2017, he has been an Assistant Professor with the Department of Mechanical Engineering, Faculty of Engineering, Gifu University, Japan. His research interests include applications of evolutionary computations to computer vision. Dr. Sato is a member of IEEE, IEEJ, and the Japan Society for Precision Engineering.



Takuya Akashireceived a PhD in theSystem Design Engineering from the Universityof Tokushima in 2006.Since April 2009, hehas been with the Department of Electrical En-gineering, Electronics and Computer Science,Iwate University.In 2015, he is a visiting asso-ciate with the California Institute of Technology.He is currently an associate professor with theIwate University.His research interests includeevolutionary algorithms, image processing andhuman sensing.He is a member of the IEEE,

RISP, IEICE, and IEEJ.