Hindawi
Complexity
Volume 2021, Article ID 8885821, 11 pages
https://doi.org/10.1155/2021/8885821

Research Article

Feature Tracking for Target Identification in Acoustic Image Sequences

Jue Gao, Ya Gu, and Peiyi Zhu

School of Electrical Engineering and Automation, Changshu Institute of Technology, Changshu 215500, Jiangsu, China

Correspondence should be addressed to Ya Gu; guya927819@163.com

Received 11 August 2020; Revised 21 September 2020; Accepted 1 March 2021; Published 16 March 2021

Academic Editor: Jing Na

This paper proposes underwater target identification with local features and a feature tracking algorithm for acoustic image sequences. Feature detectors and descriptors are key to feature tracking. Their performance in underwater scene is evaluated by the change of multitarget parameters. A comprehensive quantitative investigation into the performance of feature tracking is thereby presented. Experimental results confirm that the proposed algorithm can accurately track potential targets and determine whether the potential targets are static targets, dynamic targets, or false alarms according to the tracking trajectories and statistical data.

1. Introduction

Underwater target identification has a wide range of applications in biology, geophysics, oceanography, and military [1–3]. Through underwater acoustic imaging technology, the operation is carried out in two steps [4, 5]: (a) the sonar system is used to obtain images of underwater scenes in the region of interest (ROI) and (b) acoustic images are processed and analysed to obtain potential targets. However, the disadvantages of acoustic images such as high speckle noise, low resolution, and intensity alterations pose serious challenges to identify the target in acoustic image sequences.

According to the research of bionics [6], the biological vision system divides an object into several subsystems and realizes the identification through the synthesis of local information. In acoustic image sequences, local features are different from the image patterns of the nearest neighbour [7, 8]. Analysing the local characteristics of ROI can not only obtain the target related information but also provide the clues to identify the potential target. Usually, the local feature involves detector and descriptor.

The feature detection finds significant image regions, which makes it robust to all possible image transformations. The leading algorithms [9–11] are divided into three categories: corner detection, spot detection, and region detection. The corner can be defined as the point with high curvature and generally captured by Harris detector and Features from Accelerated Segment Test (FAST) detector. Instead of corners, spot detection concerns the local extremum of the response of some filters. Both the Laplace of Gaussian (LOG) and the determinant of Hessian matrix (DOH) indicate the local structural information of image. Furthermore, difference of Gaussian (DoG) and FAST Hessian detector are the improved version of the former. Region detection divides the image into several regions according to some similar properties of pixels. Typical methods include the Maximally Stable Extremal Regions (MSER) detection.

Subsequently, a descriptor representing the local neighbourhood around must be created. The most direct descriptor is the image block around the point. However, it has no invariance. The Scale Invariant Feature Transform (SIFT) [12] is probably the most famous local descriptor, which stimulates several subsequent works. As an alternative to SIFT, Speeded Up Robust Features (SURF) [13] is adopted to accelerate the calculation speed. Moreover, binary descriptors [14, 15], namely, Binary Robust Independent Elementary Features (BRIEF), Oriented fast and Rotated BRIEF (ORB), Binary Robust Invariant Scalable Keypoint

(BRISK), and FAST REtina Keypoint (FREAK), use simple pixel comparison to produce binary strings of usually shorter length.

Feature detection and description are not isolated tasks but are the foundation of feature tracking in image sequence. It can solve the tasks including visual odometer, Simultaneous Localization and Mapping (SLAM), Augmented Reality (AR), and so on [16–18]. However, few comparative evaluations regarding the local features are performed for target representation in acoustic image. In particular, we are not aware of any work devoted to using feature tracking to identify targets. The main contributions of this paper fall into the following two categories:

(1) An extensive evaluation of detector-descriptor-based target representation is carried out. The performances are investigated under the influence of SNR, target position, and size, and the optimal of local features is selected for underwater task.

(2) A feature tracking algorithm for target identification in acoustic image sequences is proposed. According to tracking trajectories and statistics, it can be determined whether the potential target is static target, dynamic target, or false alarm.

This paper is organized as follows. The analysis of feature detection and the description are described in Sections 2 and 3, respectively, in which we make a short review of each detector and descriptor and evaluate their performances by simulation experiments. In Section 4, we propose the feature tracking algorithm, tune the experimental parameters and implementation details, and discuss the results. The conclusions are drawn in Section 5.

2. Feature Detectors

The feature detectors are used to associate potential targets in acoustic image. The following sections review the detectors including Harris corner, FAST corner, DoG spot, Hessian spot, and MSER and investigate the application in acoustic image sequences.

2.1. Harris Corner Detector. Given an image I, the autocorrelation matrix $M(x, y)$ is calculated at each pixel (x, y):

$$M(x,y) = \sum_{u,v} \omega(u,v) \begin{bmatrix} [I_x(x,y)]^2 & I_x(x,y) \cdot I_y(x,y) \\ I_x(x,y) \cdot I_y(x,y) & \left[I_y(x,y)\right]^2 \end{bmatrix}, \tag{1}$$

where I_x and I_y denote the derivatives of image I and $\omega(u, v)$ is the window patch at position (x, y). Let λ_1 and λ_2 be eigenvalues of M, and the local shape of the neighbourhood can be classified as either uniform (both small), edge region (one small, one large), or corner region (both large). In order to avoid tedious computation of eigenvalues, corner points can be distinguished by corner response value and given by

$$c(x,y) = \det(M(x,y)) - \alpha \cdot [\operatorname{trace}(M(x,y))]^2, \tag{2}$$

where α is chosen accordingly and takes values around 0.04. Taking an example of an actual acoustic image containing two ROIs in the underwater scene, the result of Harris corner detection is shown in Figure 1.

2.2. FAST Corner Detector. FAST employs a circle of 16 pixels around the pixel of interest (x, y) numbered from 1 to 16 clockwise. Let $I(x, y)$ be the intensity of the pixel (x, y) and set a threshold value thr. If a set of N contiguous pixels in the circle are all brighter or all darker than (x, y), then (x, y) is labelled as a candidate FAST corner. Empirically, N is chosen to be 12. The following score is computed for each candidate point:

$$c(x,y) = \max\left\{ \sum_{(x_i,y_i)\in S+} \left|I(x,y) - I(x_i,y_i)\right| - \text{thr}, \sum_{(x_j,y_j)\in S-} \left|I(x,y) - I\left(x_j,y_j\right)\right| - \text{thr} \right\}, \tag{3}$$

where $S+$ is the subset of pixels brighter than (x, y) and $S-$ is the subset of pixels darker than (x, y). It can be seen from the FAST corner detection results shown in Figure 2 that detected points are also distributed on the edge of the ROI area. Only one feature is detected in ROI1, while three features are detected in ROI2. By comparison, FAST detector determines the feature according to the intensity as well as the position.

2.3. DoG Spot Detector. Detecting local extrema using DoG is a part of SIFT algorithm, and the descriptor part is described in Section 3.1. The scale space representation at different scales is obtained by convolution of the image and the Gaussian kernel:

$$\text{DoG}(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y), \tag{4}$$

where σ is the scale space factor $G(x, y, \sigma)$ is the Gaussian kernel function and is expressed as

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2} e^{-((x^2+y^2)/(2\sigma^2))}. \tag{5}$$

The pixel (x, y) is identified by comparing the value of a pixel with its 8 neighbours in the same scale and the 18 pixels in the two neighbouring scales. If the pixel value of (x, y) corresponds to a maximum in this neighbourhood, then it is

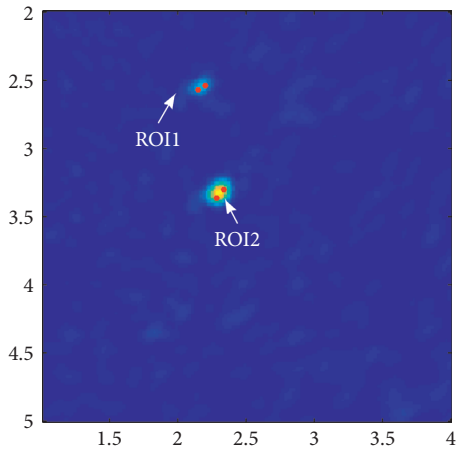

Figure 1: Harris corner detection result.

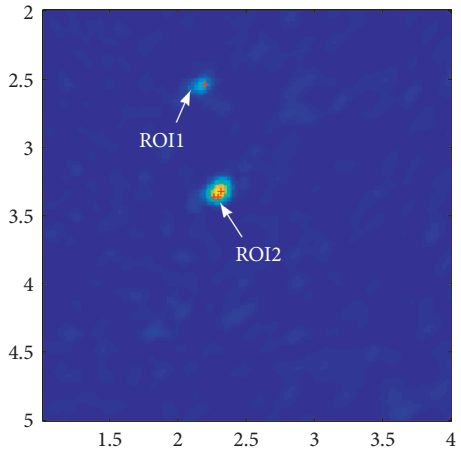

Figure 2: FAST corner detection result.

labelled as a feature. The DoG spot detection result is shown in Figure 3. It is clear that one feature is detected in ROI1, and three features are detected around ROI2, of which one is located in the center and two are distributed at the edge. Similar to FAST, more DoG features are extracted from the ROI area.

2.4. Hessian Spot Detector. Similar to DoG, Hessian is also the detection part of SURF algorithm. The Hessian matrix with scale at point (x, y) is defined as

$$H(x, y, \sigma) = \begin{bmatrix} \frac{\partial^2}{\partial x^2} G(\sigma) * I(x, y) & \frac{\partial}{\partial x}\frac{\partial}{\partial y} G(\sigma) * I(x, y) \\ \frac{\partial}{\partial x}\frac{\partial}{\partial y} G(\sigma) * I(x, y) & \frac{\partial^2}{\partial y^2} G(\sigma) * I(x, y) \end{bmatrix}. \tag{6}$$

Transforming the convolution operation into the box filtering operation, the score value of the candidate points can be calculated by

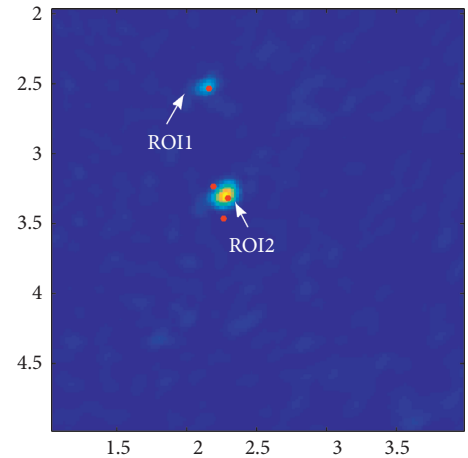

Figure 3: DoG spot detection result.

$$c(x, y, \sigma) = D_{xx}(\sigma) \cdot D_{yy}(\sigma) - \left(0.9 D_{xy}(\sigma)\right)^2, \tag{7}$$

where D_{xx}, D_{xy}, and D_{yy} are the convolution results of the filters and the approximate coefficient of $[H(x, y, \sigma)]$ is 0.9. The Hessian spot detection result is shown in Figure 4. Two features are detected distributing in the center of each ROI. Compared with the previous algorithms, Hessian algorithm ignores the edge point.

2.5. MSER Detector. Mark R as the boundary of the connected region, use ∂R to denote the pixels on the border, and use int (R) to denote the pixels in the area contained by the border. The determination of the extreme value area can be achieved by

$$\begin{cases} I|_{\text{int}(R)} < I|_{\partial R}, \\ I|_{\text{int}(R)} > I|_{\partial R}. \end{cases} \tag{8}$$

The stability of the area can be defined as

$$\psi(R) = \frac{A(R)}{(\mathrm{d}/\mathrm{d}t) A(R)}, \tag{9}$$

where $A(R)$ is the area surrounded by the boundary R. The results are shown in Figure 5. Three MSER are found in ROI1, and five MSER are found in ROI2. It exhibits that stable extreme regions are formed around the ROI.

2.6. Performance Evaluation of Detectors. An underwater scene simulation model is established to evaluate the performance of the detector. The parameters are set as follows: coverage sector is 140°, number of beams is 256, image size is 201×201, and resolution grid is $0.01 \times 0.01\ \mathrm{m}^2$. Construct a square target with SNR being 15 dB, the length being 0.15 m, and the center position being (−0.07 m, 0.87 m). The background follows the Weibull distribution, with scale parameter $k = 6.67$ and the shape parameter $\lambda = 0.45$. The simulation is shown in Figure 6, in which the target is surrounded by a dashed frame. The accuracy performance can be defined as

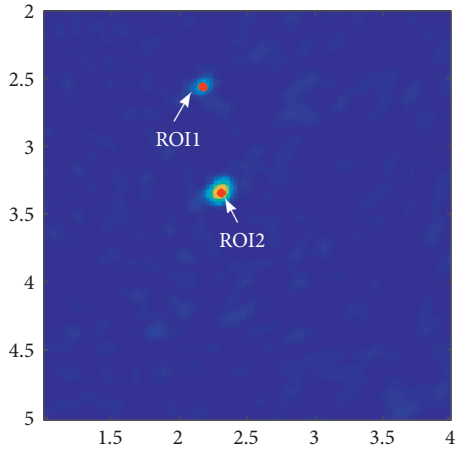

Figure 4: Hessian spot detection result.

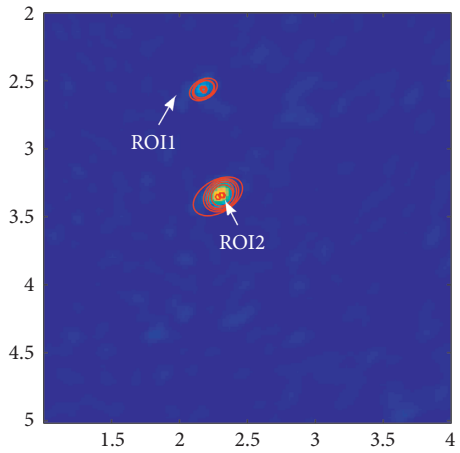

Figure 5: MSER detection in ROIs of an acoustic image.

$$P = \frac{N_{\text{obj}}}{N_{\text{all}}} \times 100\%, \tag{10}$$

where N_{all} is the total number of features and N_{obj} is the number of features falling into the man-made target.

The influence factors including size, SNR, and position on detection performance are investigated based on Figure 6. The simulation length of target is between 0.03m~0.3m, SNR is between −10dB~40dB, and center position is placed randomly. Subsequently, use the detectors mentioned above with the variation of each parameter and repeat the set of experiments 200 times. The corresponding relationships between the average detection accuracy and each parameter are shown in Figure 7. As shown in Figure 7(a), when the target size is larger than 0.06 m, Hessian and MSER fluctuate around 50% and 40%,

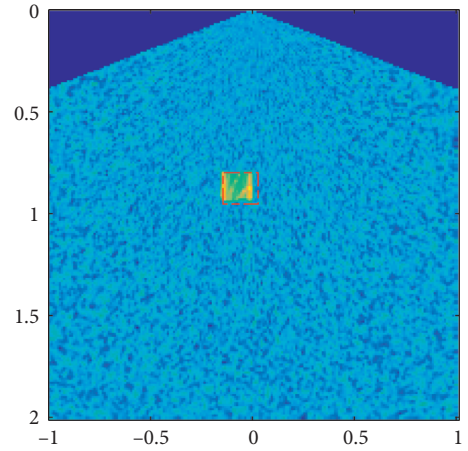

Figure 6: Simulated image of underwater scenes.

FAST and Harris stabilize at 18% and 5%, and DoG slowly rises to 4%. In Figure 7(b), while SNR is greater than 10 dB, the detection rates are gradually increased, and the rates drop to below 5% while SNR is less than 5 dB. As the effect of distance shown in Figure 7(c), the detection rate of Hessian, FAST, and MSER decreases significantly, while Harris and DoG are always at a low level.

Table 1 shows the statistical characteristics, where C is the mean square deviation of the accuracy rate and $\overline{N_{\text{all}}}$ and $\overline{P}$ represent the total features and the accuracy rate in average per experiment. It can be seen from Table 1 and Figure 7 that both Harris and DoG obtain a higher $\overline{N_{\text{all}}}$ and $\overline{P}$, while the lower C makes the curve change smoothly, MSER and Hessian behave the exact opposite to the above two detectors, and FAST is moderate in all aspects. In addition, one can learn that the detection performance is most affected by the change of the SNR, the position change is the least, and the size is in the middle.

3. Feature Descriptors

In order to measure the similarity in acoustic image sequence, the descriptors have to be implemented subsequently. Four state-of-the-art feature descriptors including SIFT, SURF, BRISK, and FREAK are used for research.

3.1. SIFT Descriptor. SIFT descriptor is computed using the gradient magnitude and orientations in a 16×16 window around the feature. These are stacked in 8-bin histograms formed in 4×4 subregions and weighted by a Gaussian window, yielding a descriptor vector of length 128. The gradient magnitude $m(x, y)$ and orientation $\theta(x, y)$ of a point (x, y) in the Gaussian image are calculated by

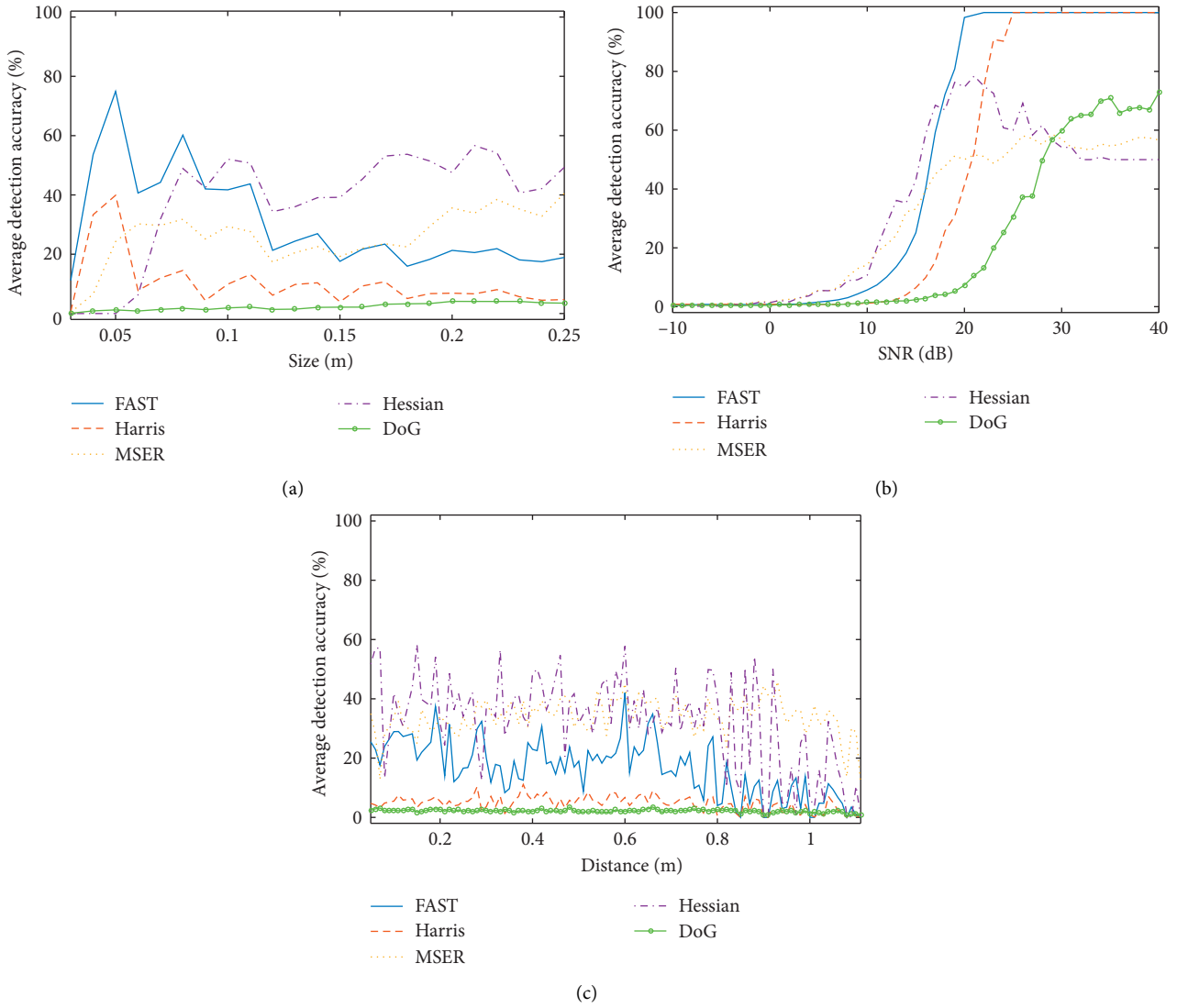

Figure 7: The average detection accuracy versus three parameters. (a) Variation of target size. (b) Variation of SNR. (c) Variation of target position.

Table 1: Statistical characteristics of detection performances.

Detector	Size			SNR			Position		
	$\overline{N_{\text{all}}}$	$\overline{P}$ (%)	σ_p	$\overline{N_{\text{all}}}$	$\overline{P}$ (%)	σ_p	$\overline{N_{\text{all}}}$	$\overline{P}$ (%)	σ_p
FAST	19.7	28.0	0.20	134.0	48.1	0.47	17.5	18.0	0.07
Harris	107.4	9.3	0.12	212.6	40.5	0.46	114.9	5.2	0.02
MSER	13.1	30.0	0.14	36.8	30.1	0.24	12.3	34.9	0.06
Hessian	5.7	38.6	0.18	17.4	33.8	0.28	4.3	34.2	0.11
DoG	286.6	2.7	0.01	233.1	20.8	0.28	286.8	2.2	0.01

$$m(x,y) = \sqrt{[L(x+1,y) - L(x-1,y)]^2 + [L(x,y+1) - L(x,y-1)]^2},$$
$$\theta(x,y) = \tan^{-1}\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}. \tag{11}$$

3.2. SURF Descriptor. SURF descriptor designs a $\pi/3$ fan-shaped sliding window, which rotates with a step length of 0.2 radians, and accumulates the sum of Haar wavelet response values $\mathrm{d}x$ and $\mathrm{d}y$. A square region is split up into 4×4 subregions and the following feature vector is

$$V = \left[\sum \mathrm{d}x, \sum |\mathrm{d}x|, \sum \mathrm{d}y, \sum |\mathrm{d}y|\right]. \tag{12}$$

Each subregion has a descriptor vector containing 4 entries yielding a 64-element SURF descriptor. The corresponding $m(\omega)$ and $\theta(\omega)$ are given by

$$\begin{aligned} m(\omega) &= \sum_{\omega} \mathrm{d}x + \sum_{\omega} \mathrm{d}y, \\ \theta(\omega) &= \arctan\left[\frac{\sum_{\omega} \mathrm{d}x}{\sum_{\omega} \mathrm{d}y}\right]. \end{aligned} \tag{13}$$

3.3. BRISK Descriptor. BRISK descriptor is composed as bit-string of length 512 by concatenating the results of simple brightness comparison tests. It applies the sampling pattern rotated by $\alpha = \arctan 2(g_y, g_x)$ around the points. The bit-vector descriptor d_k is assembled by performing all the short-distance intensity comparisons of point pairs $(p_i^{\alpha}, p_j^{\alpha}) \in S$, such that each bit $b$can be expressed by

$$b = \begin{cases} 1, & I\left(p_j^{\alpha}, \sigma_j\right) > I\left(p_i^{\alpha}, \sigma_i\right), \\ 0, & \text{otherwise}, \end{cases} \quad \forall\left(p_j^{\alpha}, p_i^{\alpha}\right) \in S. \tag{14}$$

3.4. FREAK Descriptor. Similar to BRISK descriptor, FREAK descriptor also employs a hand-crafted sampling pattern. A binary descriptor F is constructed by thresholding the difference between pairs of receptive fields with their corresponding Gaussian kernel:

$$F = \sum_{0 \le a \le N} 2^a T(P_a), \tag{15}$$

where P_a is a pair of receptive fields, N is the desired size of the descriptor, and $T(P_a)$ is judged by

$$T(P_a) = \begin{cases} 1, & I\left(P_a^{r_1}\right) > I\left(P_a^{r_2}\right), \\ 0, & \text{otherwise}, \end{cases} \tag{16}$$

where $I(P_a^{r_1})$ is the smoothed intensity of the first receptive field of the pair P_a.

3.5. Descriptor Matching. It is necessary to select an appropriate similarity measurement method for tracking application. SIFT and SURF adopted matching methods based on nearest-neighbour ratio [7]. In contrast, binary descriptors such as BRISK and FREAK sample Hamming distances to measure the similarity [15]. Since only bitwise XOR operation and counting operation are required, the computational complexity is greatly reduced compared with the Euclidean distance. Figure 8 shows the matching result using SURF descriptors in two consecutive acoustic frames. Although the background contains a lot of noise and the target intensity along with position varies dramatically, the targets in the last frame are found in the next frame precisely by SURF descriptors.

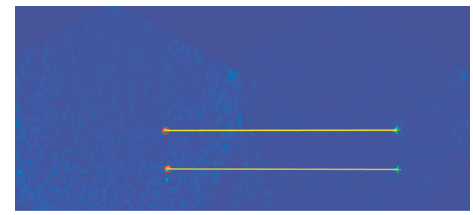

Figure 8: Matching in sequential acoustic images using SURF descriptors.

3.6. Performance Evaluation of Descriptors. The same model in Section 2.6 is used to compare the matching performance, and the acoustic image shown in Figure 6 is considered as a reference image. Set the reference target length as 0.15 m, SNR as 15 dB, and the center position as $(-0.07\ \mathrm{m},\ 0.87\ \mathrm{m})$. The evaluation of matching performance usually uses the nearest neighbour matching accuracy, which is defined as

$$P = \frac{N_{\mathrm{cor}}}{N_{\mathrm{cor}} + N_{\mathrm{fal}}} \times 100\%, \tag{17}$$

where N_{cor} is the matching number and N_{fal} is the mismatching number.

The influences of three target parameter on matching performances are shown in Figure 9. As observed from Figure 9(a), SURF maintains the accuracy rate above 80% except for individual breakpoints, and other descriptors have relatively lower matching accuracy. In Figure 9(b), both SIFT and SURF have a higher matching rate within a greater SNR variation range, and BRISK and FREAK obtain a lower matching probability when SNR alters widely. Specially, SIFT descriptor shows strong matching ability when the SNR is above 20 dB. Figure 9(c) exhibits that BRISK and FREAK only have a greater matching probability with a slight change of the distance, while SIFT and SURF still maintain a higher matching probability with a large fluctuation of distance. As the distance increased further, SURF has better matching performance than SIFT.

Table 2 lists the corresponding statistical characteristics. $\overline{P}$ is the matching probability of feature pairs in average per experiment, and σ_p is the mean square error of $\overline{P}$. It is observed that SURF is more robust with the variation of target location and size, SIFT has the best robustness when SNR changes, and binary descriptors have lower matching performance in all respects. Therefore, the most significant factor for matching is SNR.

4. Target Identification Using Feature Tracking

4.1. Feature Tracking Algorithm. Inspired by the Track Before Detect (TBD) strategy [19, 20], a new idea involving feature tracking has been innovated for underwater target

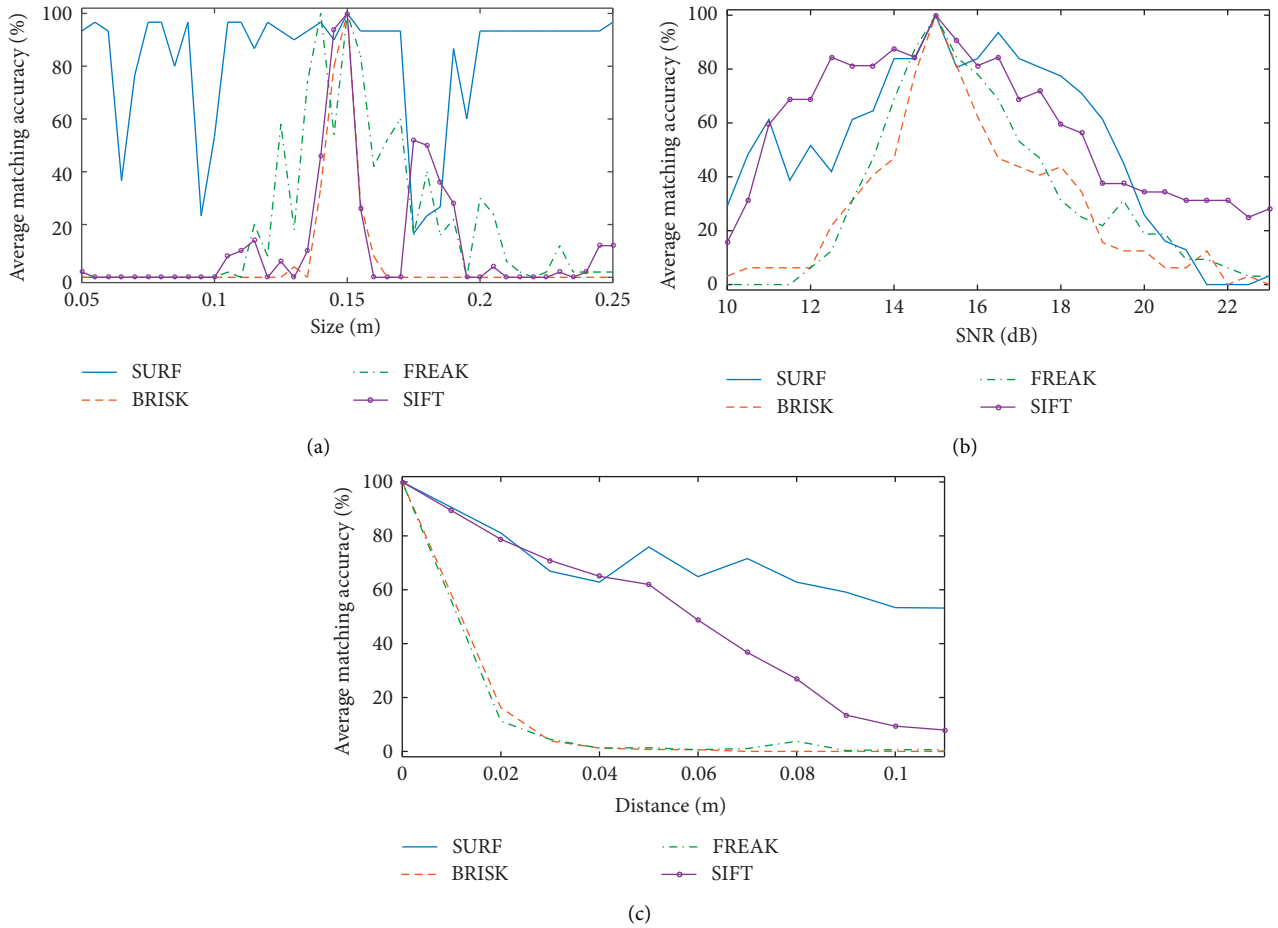

Figure 9: The average matching accuracy versus three parameters. (a) Variation of target size. (b) Variation of SNR. (c) Variation of target position.

Table 2: Statistical characteristics of matching performances with variable parameters.

Descriptors	Size		SNR		Position	
	$\overline{P}$ (%)	σ_P	$\overline{P}$ (%)	σ_P	$\overline{P}$ (%)	σ_P
SURF	50.2	0.15	50.2	0.31	63.0	0.11
BRISK	6.0	0.20	28.5	0.28	8.3	0.26
FREAK	19.4	0.29	31.9	0.31	8.8	0.25
SIFT	12.7	0.24	58.0	0.25	44.2	0.28

identification. It does not judge whether there is a target in a single frame of image but tracks multiple targets at the same time and then discriminates the potential targets according to the motion trajectory. In the underwater application, the features are used to characterize the potential targets, and the matching of descriptors measures the relevance of potential targets between frames. The flowchart of the proposed algorithm shown in Figure 10 comprises five main stages:

(1) Input the first frame I_1, obtain the feature set D_1, and save it as a template M.

(2) Read the subsequent frame I_i, acquire the feature set D_i, and match the extracted featureF_i from M.

(3) Make the matching feature F_i as potential targets and update the corresponding feature in F_i from M.

(4) Remove the mismatching feature F_i of consecutive k frames from M (considering that acoustic imaging is susceptible to environmental interference resulting in insufficient stability, k is rounded to 10% of the total number of frames).

(5) After traversing the entire image sequence, determine whether the remaining feature F_i represents the real target and then obtain feature trajectory.

According to the previous research, five combinations of Hessian + SURF, DoG + SIFT, MSER + SURF, FAST + BRISK, and FAST + FREAK are selected for feature tracking research, and an acoustic image sequence with 15 frames is simulated. In the initial frame, set the target length as 0.15 m, SNR as 15 dB, and the center position at (0 m, 1 m). The mobile distance of the target center x in each subsequent frame is the random number within −0.01 m ~ 0.01 m and that of y is the random number within −0.02 m ~ 0 m. The range of target SNR is 14 dB ~ 17 dB, and the range of target size is 0.14 m ~ 0.16 m. Four test scenarios are designed as shown in Table 3: Test I is only the change of target center position, Test II is the change of target center position and

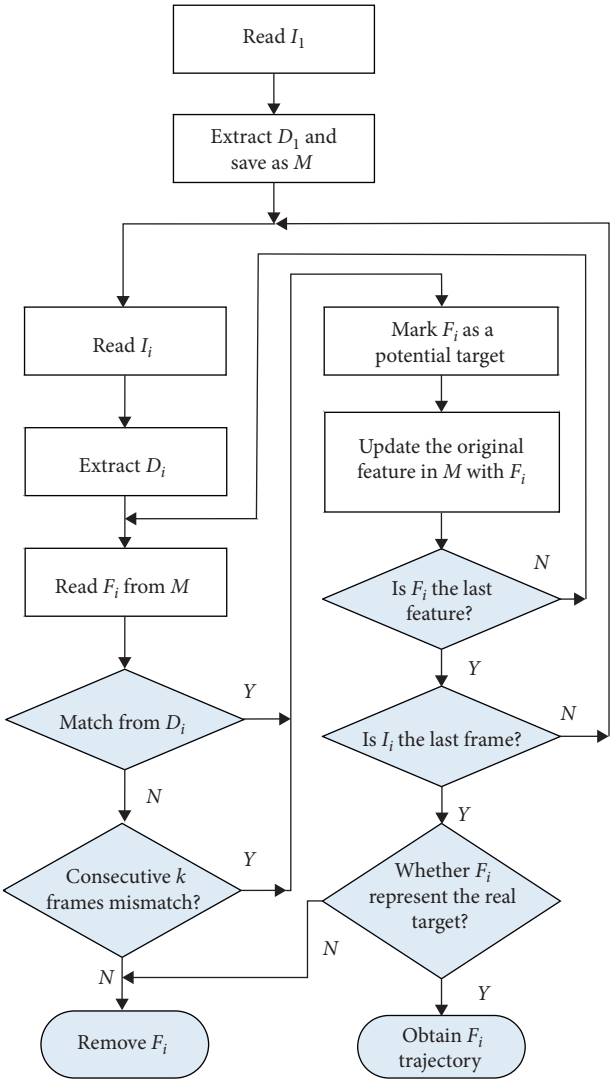

Figure 10: The feature tracking procedure.

SNR, Test III is the change of target center position and size, and Test IV is the change of all three parameters.

The matching of local features associated with the target in two consecutive frames is regarded as successful tracking, and the measure of tracking performance is given by

$$P = \frac{F_{\text{cor}}}{F_{\text{all}}} \times 100\%, \tag{18}$$

where F_{cor} is the number of frames for successful target tracking and F_{all} is the total number of frames. The test is repeated 200 times under four scenarios, and the performance comparisons of feature tracking are shown in Figure 11, where $\overline{P}$ is the average tracking rate. It is clear that Hessian + SURF reaches the highest $\overline{P}$, DoG + SIFT and MSER + SURF achieve the close $\overline{P}$, and local features using binary descriptors behave worse, particularly, FAST + FREAK get the lowest $\overline{P}$. In addition, $\overline{P}$ is highest when the target position alters alone, and it decreases when the target SNR or size also fluctuates.

Table 3: The parameters of the four experimental scenes.

Test scenario	Center position		SNR (dB)	Size (m)
	x (m)	y (m)		
I	−0.01–0.01	0.98–1	15	0.15
II	−0.01–0.01	0.98–1	14–17	0.15
III	−0.01–0.01	0.98–1	15	0.14–0.16
IV	−0.01–0.01	0.98–1	14–17	0.14–0.16

4.2. Experimental Results and Analysis. A typical dataset collected during a trial is used for verifying the effectiveness of the proposed method. The acoustic image sequence contains 38 acoustic images with a size of 776×646 and a resolution grid of $0.02 \times 0.02\,\text{m}^2$. It presents a $16 \times 13\,\text{m}^2$ water scene parallel to the water surface, in which a circular steel tank with a diameter of 0.35 m is used as a static target and a small ball with a diameter of 0.2 m is used as a dynamic target.

From the initial frame shown in Figure 12(a), it is observed that a static target is centered at (1.1 m, 12.8 m), a dynamic one is located at (6 m, 12.4 m), and the red dotted box represents the mobile area of the dynamic target. The motion trajectory is divided into two sections. The first trajectory as shown in Figure 12(b) is from frame 1 to frame 20. The dynamic target approaches the static target horizontally from right to left, and the total mobile distance of the target is 2.70 m. The second trajectory as shown in Figure 12(c) is from frame 21 to frame 38. The dynamic target approaches the static target horizontally reversely, and the total mobile distance of the target is 2.68 m. Statistically, the target mobile distance between adjacent frames is from 0.05 m to 0.27 m, with an average of 0.15 m and a mean square error of 0.06.

The tracking process of features is shown in Figure 13. In the initial frame, 131 Hessian + SURF, 1220 DoG + SIFT, and 35 MSER + SURF features are acquired, respectively. The remaining 7, 6, and 5 features are obtained in the end frame. In contrast, 3 FAST + BRISK and 3 FAST + FREAK features are obtained in the initial frame, all FAST + FREAK features are lost in the 10th frame, and only one FAST + BRISK feature remains in the end. By comparing the feature coordinate with the motion trajectory, only Hessian + SURF successfully tracked the dynamic target with 30 frames corresponding to the dynamic target. DoG + SIFT and MSER + SURF lost the dynamic target at the 18th and the 10th frame, respectively, and missed the dynamic target.

The tracking statistics are shown in Table 4. The total offset of the Hessian + SURF for tracking success is close to the actual mobile distance of the target, while the movement offset of DoG + SIFT and MSER + SURF for tracking failure is close to the actual movement distance in first section and quite different in second section. In addition, Hessian + SURF, DoG + SIFT, MSER + SURF, and FAST + BRISK have 37, 25, 29, and 29 frames corresponding to static targets, respectively. Analysing the offset and coordinates of remaining features, the features located around the static target related to the cable of the circular steel tank and the features without clear correspondence are judged as false alarms.

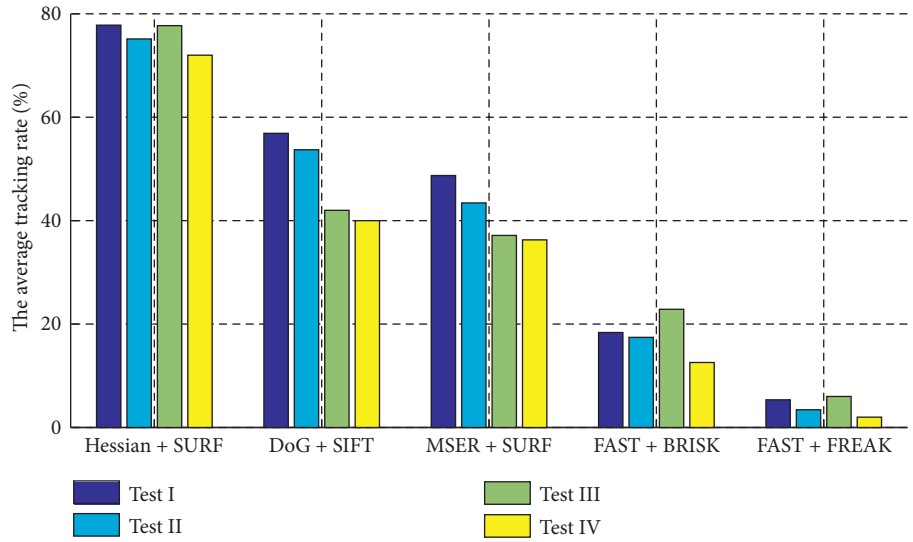

Figure 11: Performance comparisons of feature tracking.

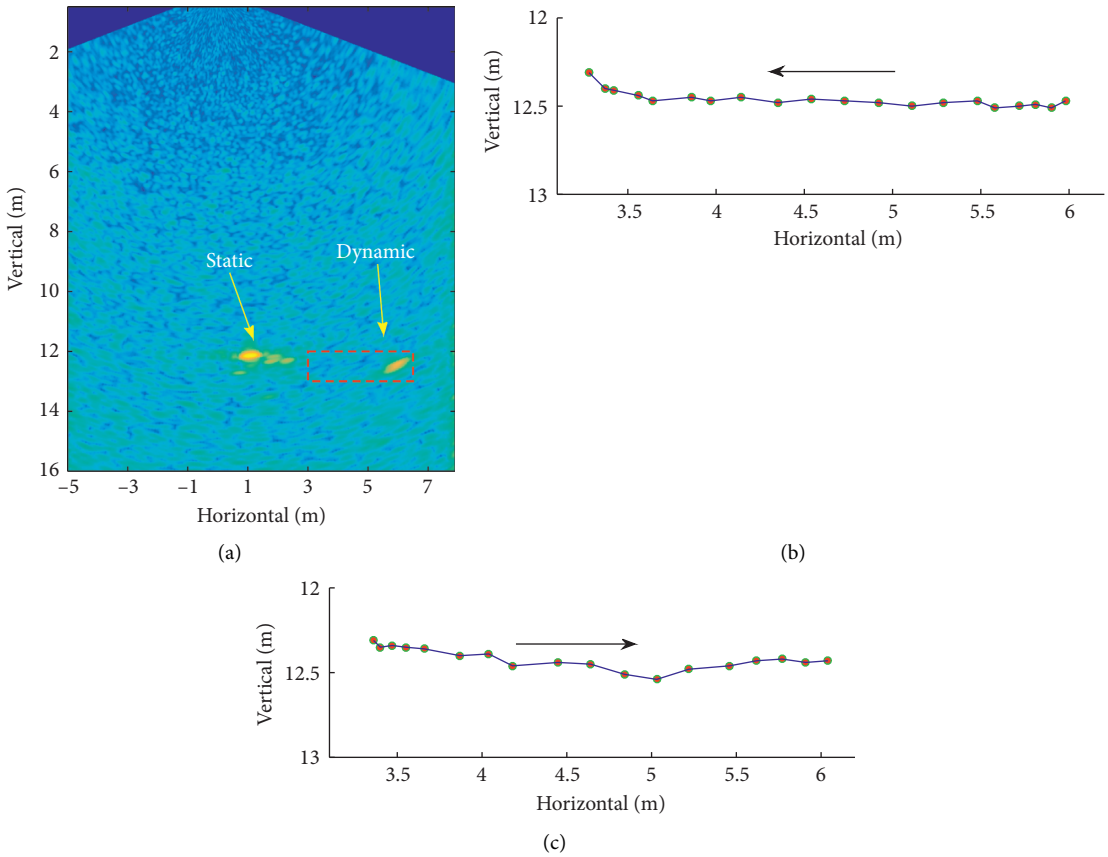

Figure 12: Feature tracking on experimental data. (a) The initial frame. (b) The first trajectory. (c) The second trajectory.

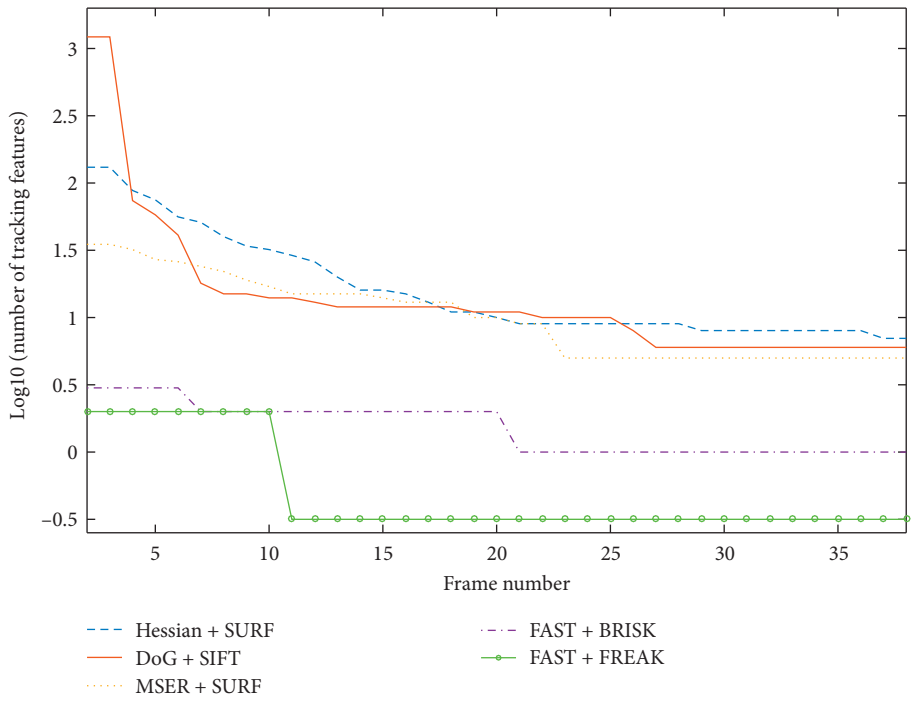

Figure 13: Feature tracking process.

Table 4: Statistical information of feature tracking.

Local features	Number of valid frames	Tracking rate (%)	Section I		Section II		Target type
			Average offset (m)	Total offset (m)	Average offset (m)	Total offset (m)	
Hessian + SURF	30	78.95	0.27	2.62	0.28	2.70	Dynamic
	37	97.37	0.01	0.02	0.01	0.02	Static
	0	0.00	0.45	0.58	0.44	0.57	False
	0	0.00	0.02	0.03	0.02	0.03	Cable
	0	0.00	0.36	1.45	0.35	0.57	False
	0	0.00	0.09	0.05	0.09	0.03	Cable
	0	0.00	0.40	1.74	0.42	0.08	False
DoG + SIFT	16	42.11	0.18	2.35	0.19	1.47	False
	25	65.79	0.02	0.02	0.02	0.01	Static
	0	0.00	0.07	0.06	0.07	0.12	Cable
	0	0.00	0.01	0.01	0.02	0.02	Cable
	0	0.00	0.01	0.01	0.01	0.02	Cable
	0	0.00	0.02	0.01	0.02	0.02	Cable
MSER + SURF	8	21.05	0.19	2.01	0.20	1.07	False
	29	76.32	0.01	0.02	0.01	0.01	Static
	0	0.00	0.01	0.02	0.01	0.02	Cable
	0	0.00	0.14	0.72	0.07	0.57	False
	0	0.00	0.08	0.58	0.09	1.51	False
FAST + BRISK	29	76.32	0.02	0.04	0.01	0.01	Static

5. Conclusion

We have introduced local features for identifying underwater targets, investigated the feature detectors and descriptors for target representation, and proposed a novel feature tracking algorithm. Hessian + SURF has achieved robust tracking results for dynamic targets. By comparison, DoG + SIFT acquires more features and can better track multiple static targets. The remaining combination have relatively poor tracking results. In the case that several frames fail to match during the tracking process, the feature is still passed on unless the rejection condition is triggered, and the potential target will not be lost. The algorithm used in this paper can be applied to linear and uncertain nonlinear systems [21–25].

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (no. 61903050), the Natural Science Foundation of Jiangsu Province (no. BK20181033), and the Natural Science Fundamental Research Project of Colleges and Universities in Jiangsu province (no. 18KJB120001).

References

[1] A. Abu and R. Diamant, "Unsupervised local spatial mixture segmentation of underwater objects in sonar images," *IEEE Journal of Oceanic Engineering*, vol. 44, no. 4, pp. 1179–1197, 2019.

[2] S. Cui, Y. Wang, S. Wang, R. Wang, W. Wang, and M. Tan, "Real-time perception and positioning for creature picking of an underwater vehicle," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 3783–3792, 2020.

[3] K. Colbo, T. Ross, C. Brown, and T. Weber, "A review of oceanographic applications of water column data from multibeam echosounders," *Estuarine, Coastal and Shelf Science*, vol. 145, no. 5, pp. 41–56, 2014.

[4] W. Kong, J. Yu, Y. Cheng, W. Cong, and H. Xue, "Automatic detection technology of sonar image target based on the three-dimensional imaging," *Journal of Sensors*, vol. 2017, Article ID 8231314, 8 pages, 2017.

[5] A. Trucco, M. Garofalo, S. Repetto, and G. Vernazza, "Processing and analysis of underwater acoustic images generated by mechanically scanned sonar systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 7, pp. 2061–2071, 2009.

[6] C. Barngrover, A. Althoff, P. Deguzman, and R. Kastner, "A brain-computer interface (BCI) for the detection of mine-like objects in sidescan sonar imagery," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 1, pp. 123–138, 2016.

[7] W. Zhou, L. Yu, Y. Zhou, W. Qiu, M.-W. Wu, and T. Luo, "Local and global feature learning for blind quality evaluation of screen content and natural scene images," *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2086–2095, 2018.

[8] S. Gauglitz, T. Höllerer, and M. Turk, "Evaluation of interest point detectors and feature descriptors for visual tracking," *International Journal of Computer Vision*, vol. 94, no. 3, pp. 1646–1655, 2011.

[9] A. Mustafa, H. Kim, and A. Hilton, "MSFD: multi-scale segmentation-based feature detection for wide-baseline scene reconstruction," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1118–1132, 2019.

[10] H. X. Wen, Z. Sheng, X. Y. Lu, F. Wu, and W. Zhang, "Moving object detection in aerial infrared images with registration accuracy prediction and feature points selection," *Infrared Physics & Technology*, vol. 92, no. 8, pp. 318–326, 2018.

[11] F. Bowen, J. Hu, and E. Y. Du, "A multistage approach for image registration," *IEEE Transactions on Cybernetics*, vol. 46, no. 9, pp. 2119–2131, 2016.

[12] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[13] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[14] H. Yang, C. Huang, F. Wang, K. Song, and Z. Yin, "Robust semantic template matching using a superpixel region binary descriptor," *IEEE Transactions on Image Processing*, vol. 28, no. 6, pp. 3061–3074, 2019.

[15] B. Fan, Q. Kong, T. Trzcinski, Z. Wang, C. Pan, and P. Fua, "Receptive fields selection for binary feature description," *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society*, vol. 23, no. 6, pp. 2583–2595, 2014.

[16] S. Yao, G. H. Wang, and Z. Li, "Correlation filter learning toward peak strength for visual tracking," *IEEE Transactions on Cybernetics*, vol. 48, no. 4, pp. 1290–1303, 2018.

[17] W. M. Arnold, M. C. Dung, and R. Cucchiara, "Visual tracking: an experimental survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1442–1468, 2014.

[18] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: a review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, 2011.

[19] S. P. Ebenezer and A. Papandreou-Suppappola, "Generalized recursive track-before-detect with proposal partitioning for tracking varying number of multiple targets in low SNR," *IEEE Transactions on Signal Processing*, vol. 64, no. 11, pp. 2819–2834, 2016.

[20] T. Northardt and S. C. Nardone, "Track-before-detect bearings-only localization performance in complex passive sonar scenarios: a case study," *IEEE Journal of Oceanic Engineering*, vol. 44, no. 2, pp. 482–491, 2019.

[21] Y. Q. Wang, F. Chen, G. M. Zhuang, and G. Yang, "Dynamic event-based mixed H∞ and dissipative asynchronous control for markov jump singularly perturbed systems," *Applied Mathematics and Computation*, vol. 386, no. 1, Article ID 125443, 2020.

[22] Y. Wang, F. Chen, and G. Zhuang, "Dynamic event-based reliable dissipative asynchronous control for stochastic Markov jump systems with general conditional probabilities," *Nonlinear Dynamics*, vol. 101, no. 1, pp. 465–485, 2020.

[23] Y. Q. Wang, G. M. Zhuang, X. Chen, Z. Wang, and F. Chen, "Dynamic event-based finite-time mixed H∞ and passive asynchronous filtering for T-s fuzzy singular markov jump systems with general transition rates," *Nonlinear Analysis: Hybrid Systems*, vol. 36, Article ID 100874, 2020.

[24] J. Na, Y. Xing, and R. Costa-Castelló, "Adaptive estimation of time-varying parameters with application to roto-magnet plant," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 2, pp. 731–741, 2021.

[25] Y. Xing, J. Na, and R. Costa-Castelló, "Real-time adaptive parameter estimation for a polymer electrolyte membrane fuel cell," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 11, pp. 6048–6057, 2019.