Detection and tracking of sea-surface targets in infrared and visual band videos using the bag-of-features technique with scale-invariant feature transform

Tolga Can, A. Onur Karalı, and Tayfun Aytaç*

TÜBİTAK BİLGEM UEKAE/İLTAREN Şehit Mu. Yzb. İlhan Tan Kışlası, 2432. cadde, 2489. sokak, TR-06800, Ümitköy, Ankara, Turkey

*Corresponding author: tayfun.aytac@iltaren.tubitak.gov.tr

Received 3 February 2011; revised 19 September 2011; accepted 30 September 2011; posted 5 October 2011 (Doc. ID 142171); published 18 November 2011

Sea-surface targets are automatically detected and tracked using the bag-of-features (BOF) technique with the scale-invariant feature transform (SIFT) in infrared (IR) and visual (VIS) band videos. Features corresponding to the sea-surface targets and background are first clustered using a training set offline, and these features are then used for online target detection using the BOF technique. The features corresponding to the targets are matched to those in the subsequent frame for target tracking purposes with a set of heuristic rules. Tracking performance is compared with an optical-flow-based method with respect to the ground truth target positions for different real IR and VIS band videos and synthetic IR videos. Scenarios are composed of videos recorded/generated at different times of day, containing single and multiple targets located at different ranges and orientations. The experimental results show that sea-surface targets can be detected and tracked with plausible accuracies by using the BOF technique with the SIFT in both IR and VIS band videos.

OCIS codes: 100.2000, 100.4999, 100.5010, 110.3080, 330.1880, 070.5010.

1. Introduction

Surveillance of the sea surface using infrared (IR) and visual (VIS) band cameras mounted on land or sea platforms to detect, track, and classify symmetric and asymmetric targets gains importance in military and security applications [1–4]. Processing of sea environments in the IR band is an especially challenging research task because sea radiance depends on sky reflections, sun glints, blackbody emissions from wave facets, and atmosphere [5]. Apart from the background clutter, low signal-to-noise ratio, low contrast, sensor noises, and the thermodynamic state of the targets also affect the detection and tracking of the targets in the sea background or at the horizon. Therefore, it is difficult to extract robust features in the IR band. In addition, features may change significantly in the subsequent frames due to illumination changes and target or platform movement, which makes target detection and tracking in the IR band a challenging research subject. Different methods have been proposed for tracking purposes in IR band videos [6–12].

Features obtained using the scale-invariant feature transform (SIFT) [13] are used in target tracking because they have a high differentiation property and are invariant to scale and rotation, robust to noise, and partially invariant to affine transformation and intensity changes. When compared to other interest point detectors, such as those of Moravec [14] and Harris and Stephens [15], SIFT features are more robust to background clutter, noise, and occlusion. Harris corner detection is not invariant to scale, which is important in a sea-surveillance system, where the target may change its orientation

0003-6935/11/336302-11$15.00/0

rapidly or the camera angle changes in the subsequent frames.

In this work, we propose the use of SIFT features together with the bag-of-features (BOF) technique for detection and tracking of sea-surface targets in IR and VIS band videos. Extensive scenario-based comparisons using real IR and VIS band videos and IR synthetic videos are performed, and the performance of the proposed method is compared with a well-known optic flow method, the Kanade–Lucas–Tomasi (KLT) feature tracker [16], using different performance metrics. The scenarios include far located, occluded, and maneuvering targets at different times of day. To the best of our knowledge, no attempt has been made previously to detect and track these types of targets using SIFT features. The major contribution in this work is the investigation of the effects of the SIFT parameters on feature extraction and the usage of these features together with the BOF technique for target detection and tracking.

The paper is organized as follows: in Section 2, a brief review is provided about the SIFT technique and its parameters. In Section 3, the BOF technique is explained. The proposed detection and tracking scheme together with the offline feature selection are explained in detail in Section 4. Experimental results with a comparison using ground truth data with different metrics are provided in Section 5. Concluding remarks are made and directions for future research are provided in Section 6.

2. Scale-Invariant Feature Transform

SIFT features are widely used in applications for target detection [17], tracking [18], classification [19], image matching [20], and constructing mosaic images [21]. The performance of target detection and tracking can be improved by optimizing the SIFT parameters for specific scenarios. For this purpose, over a dataset including sea-surface targets, SIFT parameters, described in the following subsections, are optimized.

A. Contrast Threshold Value

The difference of the Gaussian space is computed to obtain SIFT key points as below [13]:

$$DG(x,y,\sigma) = [G(x,y,n\sigma) - G(x,y,\sigma)] * I(x,y). \quad (1)$$

Here, x and y are the pixel coordinates of the processed image $I(x,y)$, $G(x,y,\sigma)$ is the Gauss filter having variable scale, n is the scale multiplication coefficient, and $DG(x,y,\sigma)$ is the Gauss difference image. In this space, scale-space extremas are detected within the neighborhood of each point in the previous and next scale-space image. In [13], it is stated that the contrast threshold value should be above 0.03 for images having intensity values between 0 and 1. Depending on the value of this threshold, the number of keypoints can be reduced. In our case, we determine this value such that the feature descriptors are representative and the number of features is less for the low computational complexity for sea-surveillance systems.

B. Maximum Harris Corner Coefficient

Keypoints go through a further elimination process by using a cornerness measure. For this purpose, Hessian matrix is computed for each keypoint as

$$H = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}. \quad (2)$$

Here, I_{xx}, I_{xy}, and I_{yy} are the directional derivatives computed in the horizontal, diagonal, and vertical directions, respectively. For the keypoints located on a corner, both eigenvalues of the corresponding Hessian matrix are distinct and take high values. Equation (3) meets these constraints and is used to decide whether a keypoint is located on a corner or not:

$$\frac{(\alpha+\beta)^2}{\alpha\beta} \le \frac{(r+1)^2}{r}. \quad (3)$$

In the equation, α and β are the eigenvalues of the Hessian matrix. The corner coefficient r is a parameter to be considered in feature extraction.

C. Orientation Assignment and Descriptor Definition

An orientation is computed and assigned for each keypoint to achieve rotation invariance. A gradient histogram is constructed, and the local peak within the 80% of the highest peak is assigned as the orientation of the keypoint. If more than one local peaks exist, then a keypoint is defined for each orientation.

A SIFT descriptor is constructed using the gradient magnitudes and orientations around the keypoint detected before. The scale of the keypoint is used to create a Gaussian window and to weight the gradient magnitudes around the keypoint with this window. Gradient orientations are rotated according to the predefined keypoint orientation. Finally, gradient histograms are constructed for the four by four regions around the keypoint and each histogram is inserted into a row vector in order to construct the SIFT descriptors.

D. Feature Matching

SIFT descriptors are highly distinctive, and the Euclidian distance measure is used to match these descriptors. First, the distances between all descriptors are found. Then, for each descriptor d, first the two closest descriptors, c_1 and c_2, are selected. If the ratio of the distance between d and c_1 to the distance between d and c_2 is greater than a predefined threshold, descriptors d and c_1 are considered as a match. After describing the parameters for the SIFT feature extraction, we provide the BOF technique, used for target detection, in the next section.

3. Bag-of-Features Technique

In order to retrieve related documents based on query words, documents should be indexed by using representative words (keywords). In addition, documents should be classified according to those keywords to reduce the search time. In the case of a text search, document classification or indexing is much more easy compared to a visual search. For example, we have two documents as follows:

- $D_1 = \{\text{BOF is used in computer vision}\}$
- $D_2 = \{\text{We have used SIFT features}\}$.

In these two documents, there are 10 distinct words as {BOF, is, used, in, computer, vision, we, have, SIFT, features}. Documents can be represented by using the number of occurrences of words as:

- $D_1 = \{1, 1, 1, 1, 1, 1, 0, 0, 0, 0\}$
- $D_2 = \{0, 0, 1, 0, 0, 0, 1, 1, 1, 1\}$.

However, in the case of an image search, there are no words for indexing images. The BOF technique is used to identify images as a combination of representative words as it is in the text search. It is an adaptation of the "term frequency inverse document frequency" method from information retrieval to computer vision and is widely used for visual object/image categorization and retrieval applications [22–24]. The main motivation of using BOF features lies in its intra- and interclass discrimination power.

Each image is considered as a combination of visual terms (*visterm*) obtained by clustering features extracted from the training set. The flow diagram of the BOF is given in Fig. 1. Identification of *visterms* and representing each image by *visterms* are considered as offline learning steps. In the classification phase, features extracted from the test frame are used to create a *visterm* histogram. This histogram is compared with that obtained from the training dataset to find the actual category of the test frame, as seen in the last step in Fig. 1.

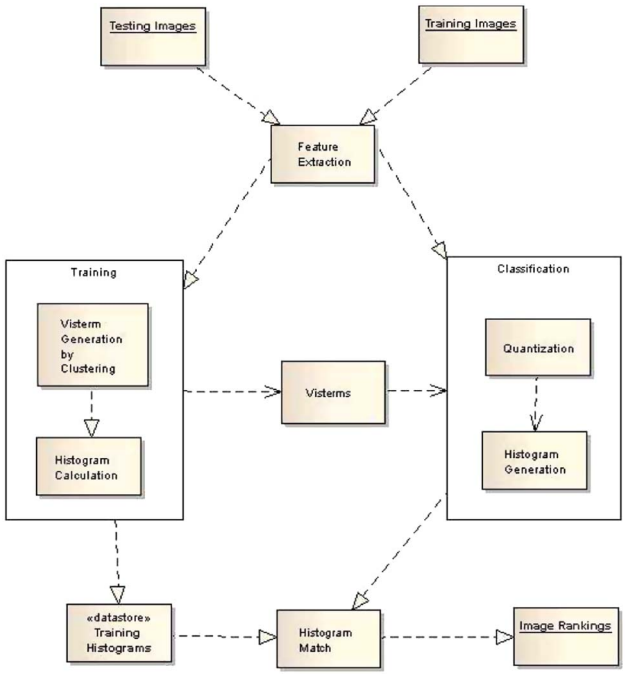

Fig. 1. (Color online) Flow diagram of the BOF procedure.

Any type of feature can be included to the BOF technique because it is independent of feature definition. However, the features that could tolerate changes in rotation, illumination, and scale should be used for successful target detection. Therefore, the SIFT features that satisfy the needs of robust tracking scenarios are used in this work.

Because our study is focused on sea surveillance, i.e., images containing almost always sea-surface, sky, and target, categorizing frames is not meaningful. However, subimages or features extracted from the images can be classified. In this work, the BOF technique is modified to classify SIFT descriptors as SIFT descriptors corresponding to the target or background instead of classifying images. As a result of classification, regions containing target SIFT descriptors are considered as the target region.

4. Target Detection and Tracking with SIFT Features

The proposed target detection and tracking method is composed of training and test phases. In the training phase, SIFT descriptor characteristics (*visterm*s) corresponding to the target and background are determined by clustering SIFT features extracted from the training set. Based on the SIFT descriptor characteristics learned on the training phase, targets are detected and tracked using SIFT descriptors on the test phase. In the following sections, we will describe the training and test phases in detail.

A. Training Phase: Classifying SIFT Features

The training phase is the learning phase of the BOF technique. This phase can be summarized as the clustering of SIFT descriptors extracted from the target and background regions. The steps, which are horizon detection, target/background determination, and *visterm* generation, are explained in the following sections.

1. *Horizon Detection*

The intensity difference between the sea and sky in the IR images can be easily seen in Fig. 2. Intensity values at the sky are almost always greater than

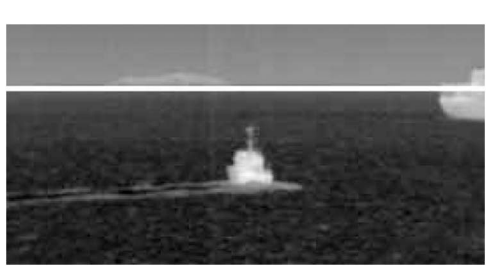

Fig. 2. Detected horizon is marked with a solid white line.

those in the sea. The horizon detection used in this work depends on this fact. With the detection of the horizon, the sky background is discarded and the sea background is taken into account. For the VIS band case, there is no need to detect the horizon because no SIFT features are extracted above the horizon by properly adjusting the SIFT parameters.

The horizon is detected by computing the intensity difference between the consecutive rows for each pixel as given in Eq. (4):

$$D_x(y) = I(x,y) - I(x+1,y), \tag{4}$$

where $D_x(y)$ is the intensity difference vector between rows x and $x+1$ (assuming top row of the frame is the first row). The number of the positive differences is checked for each row (D_x). If at any row the number of positive differences is equal or greater than 80% of the width of the frame and this value is the global maximum, this row is defined as a horizon. An example of the detected sky line is given in Fig. 2. This method may fail in the case when the camera rotates or there exists a target longer than 20% of the sky line length on the sky line. In our case, it is assumed that the camera does not rotate and only moves in the azimuth and elevation angles. As this process is only used in the training phase, it has no effect on the performance of target detection and tracking.

2. *Target/Background Determination*

In the target/background determination, a semiautomated approach is adopted, where the operator first specifies the target location by drawing a rectangle around the target. Then, the SIFT descriptors located in this rectangle are assumed as SIFT descriptors corresponding to the target and the rest as background SIFT descriptors. The effects of the SIFT descriptors extracted from the sea regions in the selected target region can be omitted because the dominant part of the SIFT descriptors are due to the target.

3. *Visterm Generation*

Target and background SIFT descriptors have different characteristics because they have different gradient histograms. The BOF technique makes use of clustering methods to discriminate the characteristics of background and target descriptors. We used the k-means algorithm [25–28] to cluster the target and background SIFT descriptors by using different k values for target and background. Resulting cluster centroids are considered as *visterm*s, which are the characteristics of the SIFT descriptors for target and background regions.

B. Test Phase: Target Detection and Tracking

We use the fact that there cannot be significant changes in the target direction at the consecutive frames in sea-surveillance videos because the speed of the above-sea platforms is limited when compared to other platforms. Therefore, the SIFT descriptors extracted from two consecutive frames should be similar both on and around the target locations. The main steps of the detection and tracking, which are descriptor classification, descriptor matching, and region-of-interest (ROI) selection, are described in detail in the following sections.

1. *Descriptor Classification*

SIFT descriptors should be classified as target or background descriptors in order to match the target descriptors in the consecutive frames. Descriptor classification is performed by using the cluster centroids obtained in the training phase. First, SIFT descriptors are extracted from the current frame. For each descriptor (desc_j), the Euclidean distances to each target centroid (cent_k^t) and background centroid (cent_k^b) are calculated by using Eq. (5):

$$\mathrm{dist}_{jk}^{t,b} = \sum_{n=1}^{p} [\mathrm{desc}_j(n) - \mathrm{cent}_k^{t,b}(n)]^2, \tag{5}$$

where dist_{jk}^t (dist_{jk}^b) is the distance between the jth descriptor and the kth target (background) cluster centroid, $\mathrm{desc}_j(n)$ is the nth element of the jth descriptor, $\mathrm{cent}_k(n)$ is the nth element of the kth cluster centroid, and p is the size of the SIFT descriptor vector (128 in our case).

The distance between the SIFT descriptors and the cluster centroids is normalized as

$$\mathrm{dist}_{jk_{\mathrm{norm}}}^{t,b} = \frac{\mathrm{dist}_{jk}^{t,b}}{\max[\mathrm{dist}_{jk}^{t,b}]}, \tag{6}$$

where $\mathrm{dist}_{jk_{\mathrm{norm}}}^t$ ($\mathrm{dist}_{jk_{\mathrm{norm}}}^b$) is the normalized distance value between the SIFT descriptor j and the target (background) centroid k and $\max[\mathrm{dist}_{jk}^t]$ ($\max[\mathrm{dist}_{jk}^b]$) is the maximum distance value between the SIFT descriptor and the target (background) centroids. The minimum normalized distance is calculated for both target and background centroids. Then, the SIFT descriptor is classified as

$$\text{decision} = \begin{cases} \text{target}, & \min[\mathrm{dist}_{jk_{\mathrm{norm}}}^t] \le \min[\mathrm{dist}_{jk_{\mathrm{norm}}}^b], \\ \text{background}, & \text{otherwise}. \end{cases}$$

where $\min[\mathrm{dist}_{jk_{\mathrm{norm}}}^t]$ ($\min[\mathrm{dist}_{jk_{\mathrm{norm}}}^b]$) is the smallest normalized distance value between the SIFT descriptor and the target (background) centroids. A sample classification result of the SIFT descriptors is given in Fig. 3, in which circles indicate the SIFT descriptors classified as the target, and plus signs indicate the SIFT descriptors classified as the background.

Until a target region is detected, all target SIFT descriptors extracted from the previous frame are

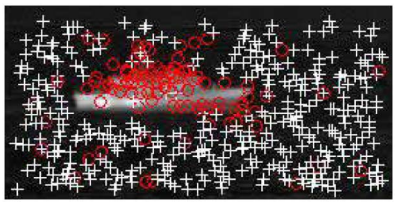

Fig. 3. (Color online) Classification results: circles and plus signs indicate the target and background SIFT descriptor locations, respectively.

used to match with the target SIFT descriptors in the current frame. However, after detecting a target region, only the target descriptors located in this target region are used for descriptor matching. This procedure reduces the running time of the algorithm and provides more accurate results.

2. *Descriptor Matching*

In this work, the method proposed by Lowe [13] is used to match the SIFT descriptors in the consecutive frames. However, this matching procedure may give false matches, as seen in Fig. 4. Additionally, the classification of the SIFT descriptors as target or background may result in false alarms, as seen in Fig. 3. The main cause of those false alarms is that neither classification nor matching steps make use of the important information about SIFT descriptors, i.e., their locations. The position information can be used to eliminate false alarms resulting from the classification or the matching steps. In this step, descriptor matches corresponding to the targets are pruned by appropriate rules.

Pruning of matched features: Information between two consecutive frames can be used to prune false matches. While pruning the matched SIFT descriptors in the consecutive frames, we use the fact that the descriptors should match in a consistent way such that the descriptors extracted from the front part of the target in the ith frame should match with those extracted from the front part of the target in the $(i+1)$th frame. This consistency can be checked by the tangent slope and the length constraints to prune false positive matches in the consecutive frames.

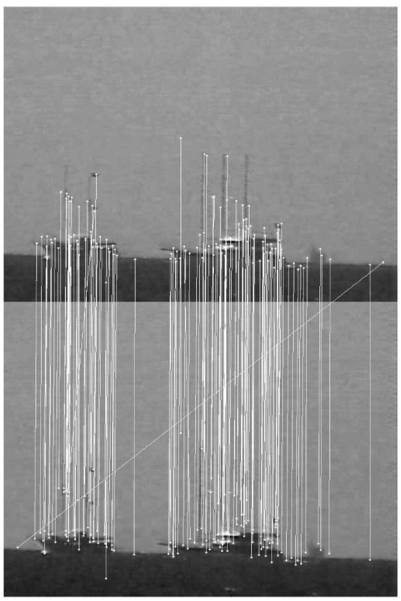

Fig. 4. SIFT match sample between two consecutive frames.

When two matched points in the consecutive frames are connected by a line, the slope of the line can be easily calculated. The slopes calculated from the matched descriptors should be almost equal in all matched pairs. In this work, the slope is considered as the difference between the y coordinates of the matched points, as can be seen in Eq. (7):

$$\Delta(m) = m_y^i - m_y^{i+1}, \tag{7}$$

where m_y^i and m_y^{i+1} are the y coordinates of the matched SIFT point m in the ith and $(i+1)$th frames, respectively.

In order to detect false matched SIFT descriptors, the mean and standard deviation of the slopes are calculated among the matched descriptors in the consecutive frames, and then the matched SIFT descriptors are eliminated according to their slopes using the relation below:

$$\text{decision}(m)\begin{cases}\text{true}, & \mu_s - \sigma_s < \Delta(m) < \mu_s + \sigma_s \\ \text{false}, & \text{otherwise}\end{cases}.$$

Here, m is the matched SIFT descriptor pair in the consecutive frames and μ_s and σ_s are the mean and the standard deviation of all slopes.

In this paper, our aim is to match the SIFT descriptors coming from the same parts of the target in the consecutive frames. For example, SIFT descriptors extracted from the front part of the target at frame i should match with those of the target at frame $i+1$. In order to achieve this, a line is obtained by connecting two matched points in the consecutive frames and the length of this line is calculated. The length of the matched descriptors should be almost equal in all matched pairs. The length is calculated by using Eq. (8):

$$L(m) = m_x^i - m_x^{i+1} + w. \tag{8}$$

Here, m_x^i (m_x^{i+1}) is the x coordinate of the matched SIFT point m in ith [$(i+1)$th] frame and w is the width of the frame. The width w is a fixed value for each frame sequence, so this parameter can be omitted in the calculations for the sake of simplicity.

Matches in the consecutive frames should have similar lengths because they are expected to match SIFT descriptors extracted from the same parts of the targets. The matched points are eliminated based on their lengths:

$$\text{decision}(m)\begin{cases}\text{true}, & \mu_l - \sigma_l < L(m) < \mu_l + \sigma_l \\ \text{false}, & \text{otherwise}\end{cases},$$

where $L(m)$ is the length of the matched SIFT descriptors in the consecutive frames and μ_l and σ_l are the mean and standard deviation of length values of all matched points, respectively.

3. *Region-of-Interest Selection*

After matching the SIFT descriptors in the consecutive frames and pruning the false matches, the ROI including the target region needed to be determined. The ROI is considered as the minimum rectangle that encloses all matched SIFT descriptors.

A separate ROI is calculated for each frame. However, this ROI is verified and updated if needed by using the ROI of the previous frame. Verification of the new ROI is performed based on Eq. (9):

$$\text{ROI\_Validity} = \begin{cases}\text{invalid} & w_{\text{diff}}/w_{\max} > T \quad \text{or} \quad h_{\text{diff}}/h_{\max} > T \\ \text{valid} & \text{otherwise}\end{cases}, \tag{9}$$

where w_{diff} (h_{diff}) is the absolute difference between the width (height) of the ROIs of the consecutive frames and $w_{\max}$ ($h_{\max}$) is the maximum of widths (heights) of the ROIs for consecutive frames. The threshold T is taken as 0.20 in our trials.

If Eq. (9) gives *invalid* result for the ROI of the current frame, then the ROI is updated by using Eq. (10):

$$\text{ROI} = \frac{\text{ROI}_{\text{prev}} + \text{ROI}_{\text{cur}}}{2}, \tag{10}$$

where ROI_{prev} is the ROI calculated for the previous frame and ROI_{cur} is the ROI that is marked as invalid from Eq. (9). These verification and update methods help the proposed method to control the growth of the ROI in a reasonable manner by eliminating the false ROI detection results.

In the detection step, ROI is calculated by finding the largest candidate region after performing a dilation operation. However, the ROI is calculated based on all matched SIFT descriptors in the tracking steps.

5. Experimental Studies

In the implementation of the proposed approach, SIFT features are extracted with the algorithm described in [29]. Four scale levels are created for each octave of the Gaussian scale-space to extract SIFT features from both IR and VIS band videos. Contrast threshold values are set to 0.005 and 0.01 for the IR and VIS band videos, respectively. The corner threshold r is set to 15 and 10 for the IR and VIS band videos, respectively. The KLT feature tracker is used to compare the performance of the proposed method. In the literature, the method is used to extract interest points and to match them in the subsequent frames [16,30,31]. The KLT feature tracker is a sparse optical flow method based on three assumptions: constant brightness, small movements in time, and coherent motion at neighboring elements. A sparse iterative version of the KLT feature tracker in pyramids, described in [32], is implemented in this study. Initial target detection is done manually. A 10 by 10 search window and five-level pyramid is used for tracking purposes. In the following sections, we will describe the performance metrics and the dataset used for performance comparisons.

A. Evaluation Criteria

We evaluate the proposed approach on several videos by using four metrics based on different morphological similarity. These four performance metrics can be defined as follows:

- **Metric 1** (M_1): M_1 is the Euclidean distance between the center of the ground truth of the target region and the center of the detected target region ($|JK|$ in Fig. 5).
- **Metric 2** (M_2): M_2 is the city block distance between the center of the ground truth data and the center of the detected target area. This metric can be visualized in Fig. 5 by the summation of $|OJ|$ and $|OK|$.
- **Metric 3** (M_3): The ratio between the undetected target area and the total target area (false negative rate) is used as a third metric. This metric gives what percentage of the target is missed. The area enclosed by ABMHLD divided by the area enclosed by ABCD provides an illustration of this metric in Fig. 5.
- **Metric 4** (M_4): This metric gives the true positive rate, which is calculated as the ratio between the correctly detected target area and the whole detected target area. In Fig. 5, this metric is illustrated as the ratio between the area enclosed by HMLC divided by the area enclosed by HGFE.

For each video, the four metrics are calculated by averaging each metric along the frame sequences.

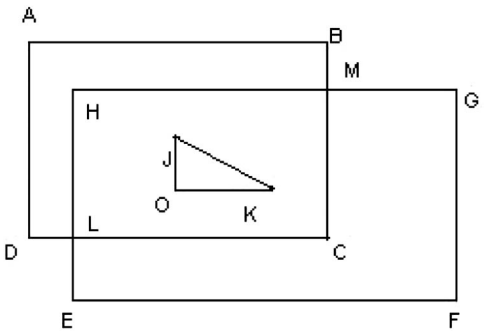

Fig. 5. The rectangle represented by ABCD is the target area, and EFGH is the detected area. J represents the center of the ABCD rectangle, and K represents the center of the EFGH rectangle. O corresponds to the origin of the coordinate system.

We expect low values for the first three metrics (M_1, M_2, and M_3) and high values (closer to 1.0) for the last metric (M_4). The main motivation to use different metrics is to evaluate the proposed method from different aspects to obtain robust results. The metrics defined above are complementary of each other. For example, if only M_3 or M_4 is used, then whenever the ROI is miscalculated and the whole image is selected as the ROI, M_4 will favor a false detection. On the other hand, if only some part of the target is selected as the ROI, then M_3 will favor a false detection. Whenever the ROI is found to be the same as the ground truth data, M_3 will be zero and M_4 will be one, which indicates a perfect detection. Boundary values for M_3 and M_4 are defined for the decision of the true target detection. Hence, the number of true and false detections can be obtained and used to determine the overall detection performance of the method. The maximum value that M_3 can take is selected as 0.5 in this work, and values obtained below this threshold are assumed to indicate true target detection. Similarly, the minimum value that M_3 can take is also selected as 0.5, which indicates that the target should cover at least 50% of the detected target region. True target detection is assumed if both of these constraints are satisfied.

B. Datasets

Mainly, we have three types of datasets: real and synthetic IR videos and VIS band videos. The frames of each video are labeled to identify by using a rectangle target region. One sample ground truth frame can be seen in Fig. 6. In the same figure, the detected ROI is also illustrated. The experimental results for each dataset are given in the following subsections. For each dataset, different videos are selected for training and testing sets.

1. Results for Real IR Band Videos

In this dataset, the proposed technique is trained and tested on different sea-surveillance videos. Three videos are used for training and eight videos are used for testing. The sample videos examined in this work are obtained at field trials using a long-wave IR camera working in $8-12\,\mu$m range. The captured image has dimensions of [136,272] and is located on the ground to observe the scene. Different scenarios include images containing single and multiple targets and targets located at different ranges and orientations, and recorded at different times of day.

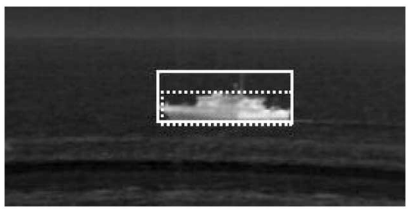

Fig. 6. Ground truth and the detected target areas for a sample frame. The ground truth is marked with the solid rectangle. The detected region is marked with the dashed lines.

The content of the training videos can be summarized as follows:

- **Real_Video_Train_1**: A target is moving in the scene, and after some time, two targets are appearing in the scene.
- **Real_Video_Train_2**: A target is approaching from a different distance and orientation to the camera.
- **Real_Video_Train_3**: A target is coming too close to the camera while moving in the scene.

The content of test videos can be summarized as follows:

- **Real_Video_Test_1**: A target is coming toward the camera with an angle of approximately 45°.
- **Real_Video_Test_2**: A target is moving in a circular route.
- **Real_Video_Test_3**: A target is moving at almost the same distance to the camera position.
- **Real_Video_Test_4**: A target is moving at almost the same distances according to the camera position.
- **Real_Video_Test_5**: A target is moving at almost the same distance to the camera position.
- **Real_Video_Test_6**: A target is moving toward the camera, and then it changes its direction and moves away from the camera.
- **Real_Video_Test_7**: A target is moving away from the camera by changing its direction.
- **Real_Video_Test_8**: A target is moving at almost the same distance to the camera position.

In the training phase, more than 7000 and 16,000 SIFT descriptors are extracted from real IR videos for the target and background regions, respectively. Best performance is achieved by using 100 clusters for target descriptors and 300 clusters for background descriptors.

Results for real IR videos by using *visterm*s obtained from real IR videos are given in Table 1. In the ground truth data, the targets are specified by a rectangle including the mast of the sea-surface target. However, masts are not so much visible in IR videos because of their sizes compared to the target. Because the masts are not detectable by our approach (no interest points can be detected on or around masts), the detected target area does not always contain masts. Therefore, an increase is observed in our false-negative rate (M_3). The algorithm is tested on several real IR videos, and the average performance achieved for M_4 is 79.62%. As defined in the previous section, target detection rates are found using the thresholds defined for M_3 and M_4. An average detection rate of 75% for the proposed technique over each video of the real IR video dataset is achieved.

Table 1. Real IR Video Results by Using Centroids Extracted from the Real Videos[a]

Video Reference	M_1	M_2	M_3	M_4
Real_Video_Train_1	6.37	7.85	0.32	0.80
	85.12	105.90	1.00	0.00
Real_Video_Train_2	13.92	17.11	0.35	0.72
	20.02	24.15	0.37	0.54
Real_Video_Test_3	14.80	18.37	0.45	0.78
	20.95	25.80	0.41	0.61
Real_Video_Test_1	12.41	14.36	0.29	0.87
	19.07	21.63	0.29	0.77
Real_Video_Test_2	14.41	17.14	0.53	0.81
	24.62	30.06	0.46	0.62
Real_Video_Test_3	18.69	20.67	0.38	0.77
	34.07	38.07	0.67	0.62
Real_Video_Test_4	15.25	17.73	0.36	0.79
	14.65	15.87	0.47	0.91
Real_Video_Test_5	18.63	23.08	0.46	0.73
	22.79	28.26	0.45	0.49
Real_Video_Test_6	15.48	17.71	0.47	0.75
	23.45	24.87	0.46	0.60
Real_Video_Test_7	18.44	22.65	0.70	0.88
	20.71	24.74	0.45	0.76
Real_Video_Test_8	25.53	28.97	0.33	0.77
	28.61	33.69	0.36	0.52

[a]The top rows represent the proposed method results, and the bottom rows represent the KLT results.

Sample tracking results are given in Fig. 7. In this figure, the ground truth region is specified by solid lines, and the detected region is specified by the dashed lines. As shown, although the size of target changes in time because of the target motion, the target can be detected correctly.

The performance of the proposed technique is compared with that of KLT feature tracker. Tracking results for real IR videos are given in the bottom rows of Table 1. The proposed method provides better tracking performance, and the ROI detected by the proposed technique is more similar to ground truth

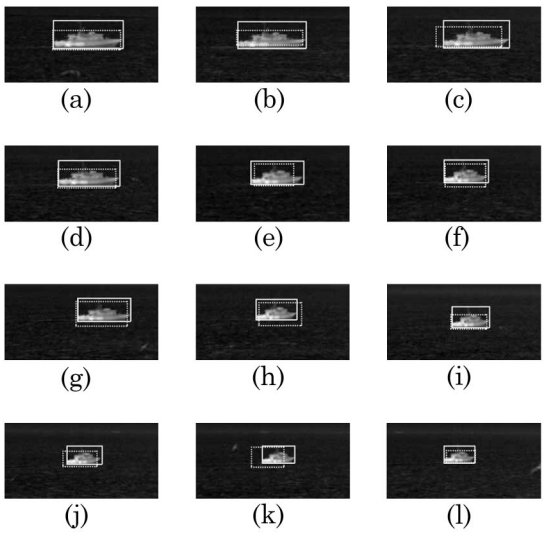

Fig. 7. Tracking results from a real IR video.

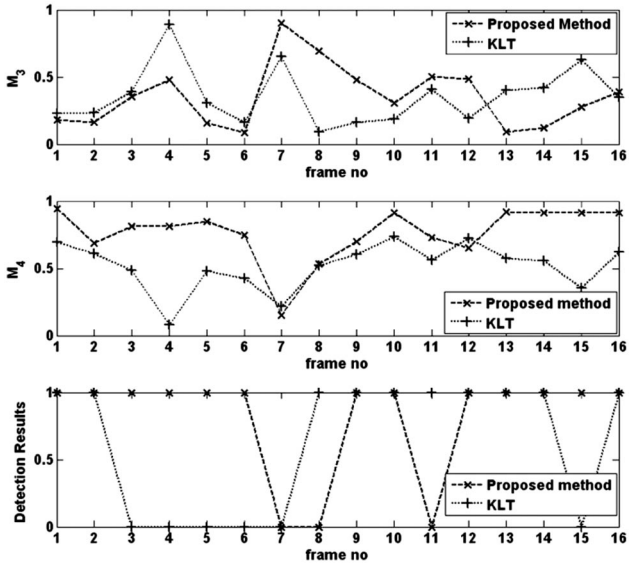

Fig. 8. M_3, M_4, and detection results for a sample real IR video sequence.

data compared to that obtained using the KLT feature tracker. Also, the KLT tracker is not robust to illumination changes. Detection performance of the KLT tracker over the same dataset is 55% worse than that achieved by the proposed method (75%). For a sample real IR video sequence, we provide M_3, M_4, and the detection performances in Fig. 8.

The tracking results for training and testing sets by using *visterm*s obtained from IR synthetic videos are presented in Table 2. As expected, in some videos, synthetic *visterm*s perform worse compared to *visterm*s obtained from real videos (Real_Video_Train_2). In fact, this result is expected to occur because the synthetic and real IR videos have different histograms.

2. *Results for Synthetic IR Band Videos*

The method proposed is also tested using synthetic IR videos. The IR scene is generated by texture mapping for sea and sky backgrounds. Sea and sky backgrounds are validated via histogram comparisons with real IR data [33]. Then a 3D noise component,

Table 2. Real IR Video Results by Using Centroids Extracted from the Synthetic IR Videos

Video Reference	M_1	M_2	M_3	M_4
Real_Video_Train_1	7.78	9.42	0.23	0.67
Real_Video_Train_2	56.00	68.80	0.78	0.25
Real_Video_Test_3	17.16	21.32	0.43	0.72
Real_Video_Test_1	14.76	17.14	0.35	0.87
Real_Video_Test_2	16.22	18.90	0.53	0.84
Real_Video_Test_3	16.52	18.39	0.41	0.84
Real_Video_Test_4	14.15	17.50	0.42	0.82
Real_Video_Test_5	18.64	23.44	0.48	0.74
Real_Video_Test_6	14.90	17.83	0.51	0.78
Real_Video_Test_7	23.19	27.27	0.65	0.82
Real_Video_Test_8	23.78	28.84	0.58	0.77

Table 3. Synthetic IR Video Results by Using Centroid Extracted from the Synthetic Videos[a]

Video Reference	M_1	M_2	M_3	M_4
Syn_Video_Train_2	5.26	6.68	0.33	0.94
	17.53	19.70	0.22	0.54
Syn_Video_Train_3	4.81	6.07	0.28	0.91
	8.13	10.17	0.06	0.48
Syn_Video_Test_1	4.68	5.48	0.47	0.85
	10.75	12.07	0.18	0.66
Syn_Video_Test_2	5.80	7.52	0.46	0.91
	13.55	16.48	0.21	0.46
Syn_Video_Test_3	5.60	7.22	0.45	0.92
	12.96	15.81	0.32	0.45

[a]The top rows represent the proposed method results, and the bottom rows represent the KLT results.

which is independent in spatial and temporal space, is added to simulate sensor effects [34]. The standard deviation of the noise components is selected to be 7 for the 8 bit dynamic range IR images. The field of view of the imaging system is selected to be 6°, and the distance to the target platform varies between 2500 to 5000 m. The testing and training videos are created by using different camera positions and target orientations. In the experiments, three videos are used for training and three videos for testing. Synthetic IR video frames have a resolution of 256 × 256.

More than 26,000 background SIFT descriptors and more than 10,000 target SIFT descriptors are extracted from the synthetic training videos. The best performance is achieved by using 200 clusters for target descriptors and 500 clusters for background descriptors.

The tracking results for both training and testing are given in Table 3 by using the centroids obtained from the synthetic IR videos. The average performance achieved for M_4 is 89.33%. The KLT feature tracker performs worse on IR synthetic videos compared to real IR videos and the proposed method. The KLT feature tracker cannot locate consistent points as does the SIFT. An average detection rate of 55% for the proposed technique over each video of the synthetic IR video dataset is achieved. Performance of the KLT tracker over the same dataset is obtained to be 51%. Sample tracking result obtained using *visterm*s extracted from the synthetic training set is given in Fig. 9.

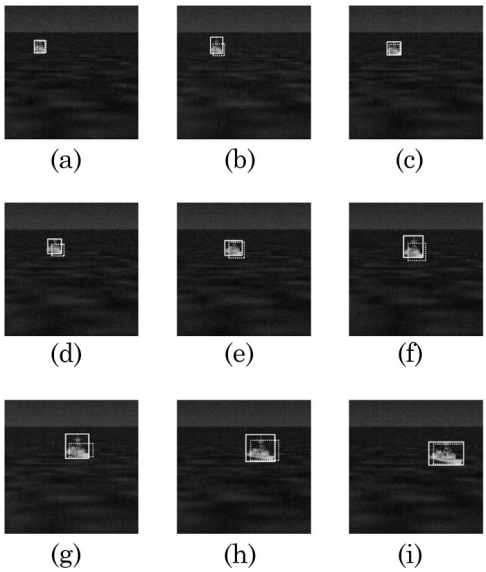

Fig. 9. Tracking results from synthetic IR video.

Table 4. Synthetic IR Video Results by Using Centroids Extracted from Real Videos

Video Reference	M_1	M_2	M_3	M_4
Syn_Video_Train_2	5.37	6.67	0.42	0.95
Syn_Video_Train_3	8.38	10.17	0.51	0.92
Syn_Video_Test_1	10.03	12.55	0.72	0.90
Syn_Video_Test_2	6.32	8.05	0.52	0.93
Syn_Video_Test_3	6.09	7.88	0.51	0.92

The tracking results for synthetic training and testing videos are given in Table 4 for *visterm*s extracted from the real IR videos, respectively.

3. *Results for VIS Band Videos*

The proposed technique is tested on three different VIS band videos. The frame size for this dataset is 640 × 480. Because SIFT descriptors are extracted from the gray-scale images, a process is performed on red-green-blue to gray-scale conversion. Then, 20,000 SIFT descriptors from target regions and 160,000 SIFT descriptors from background regions are extracted. Training samples are collected from different sea-surface targets in different sea-surface scenarios. In order to observe the effect of the number of clusters in the clustering target and background SIFT descriptors, the algorithm is run several times using a different number of clusters.

The best results are obtained by using 500 centroids for target SIFT descriptors and 5000 centroids for background SIFT descriptors. The centroids obtained from the synthetic and real IR videos are also used in the experiments. But targets cannot be detected in VIS band tests by using these centroids, as expected. The results for VIS band videos are given in Table 5. The proposed method is tested on several VIS band videos, and the average performance achieved for M_4 is 94.66%. Because the KLT feature

Table 5. Visual Band Video Results[a]

Video Reference	M_1	M_2	M_3	M_4
Visual_Band_Video_1	10.19	13.00	0.59	0.97
	7.24	8.96	0.07	0.80
Visual_Band_Video_2	2.09	2.62	0.22	0.96
	2.20	2.69	0.00	0.74
Visual_Band_Video_3	13.49	17.00	0.18	0.91
	101.17	132.10	0.54	0.30

[a]The top rows represent the proposed method results, while the bottom rows represent the KLT results.

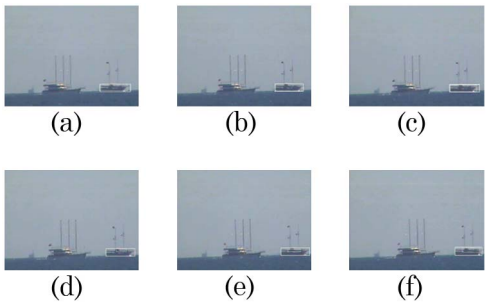

Fig. 10. (Color online) Tracking results from VIS band videos.

tracker is very sensitive to illumination changes, satisfactory results could not be obtained for Visual_Band_Video_3. The visualization of sample tracking is given in Fig. 10.

6. Discussion and Conclusion

We propose a method to automatically detect and track sea-surface targets on IR and VIS band videos based on the BOF technique with SIFT features. Our method has mainly two phases: training and testing. In the training phase, the BOF technique is used to identify the characteristics of SIFT descriptors extracted from the target and background. Then, target detection and tracking is performed in the test phase by using the learned model of the SIFT descriptors and an extended matching scheme. In the method, the BOF technique and SIFT descriptors are used together for robust target detection and tracking against illumination, scale, and rotational changes. By the use of heuristic rules such as tangent slope and the length, false alarms are eliminated in the matching of the target features in the subsequent frames.

As a future study, the proposed method will be extended to track multiple targets. Motion estimation techniques such as Kalman filtering will be planned to assist the proposed heuristic rules for target gate determination. The fusion of SIFT features extracted in IR and VIS band images may improve both detection and tracking accuracy, which is a another possible extension of this study.

The authors would like to thank Dr. S. Gökhun Tanyer, Dr. Cemil B. Erol, and Dr. Alper Yıldırım for their support in this study; Serdar Çakır for helpful discussions; and Onur Bekmen for generating synthetic IR videos.

References

1. M. Diani, A. Baldacci, and G. Corsini, "Novel background removal algorithm for Navy infrared search and track systems," Opt. Eng. **40**, 1729–1734 (2001).
2. Y. Xiong, J.-X. Peng, M.-Y. Ding, and D.-H. Xue, "An extended track-before-detect algorithm for infrared target detection," IEEE Trans. Aerospace Electron. Syst. **33**, 1087–1092 (1997).
3. M. de Visser, P. B. W. Schwering, J. F. de Groot, and E. A. Hendriks, "Passive ranging using an infrared search and track sensor," Opt. Eng. **45**, 1–14 (2006).
4. S. Çakır, T. Aytaç, A. Yıldırım, and Ö. Nezih Gerek, "Classifier-based offline feature selection and evaluation for visual tracking of sea-surface and aerial targets," Opt. Eng. **50**, 107205 (2011).
5. C. R. Zeisse, C. P. McGrath, K. M. Littfin, and H. G. Hughes, "Infrared radiance of the wind-ruffled sea," J. Opt. Soc. Am. A **16**, 1439–1452 (1999).
6. A. Bal and M. S. Alam, "Dynamic target tracking with fringe-adjusted joint transform correlation and template matching," Appl. Opt. **43**, 4874–4881 (2004).
7. F. A. Sadjadi, "Infrared target detection with probability density functions of wavelet transform subbands," Appl. Opt. **43**, 315–323 (2004).
8. A. Bal and M. S. Alam, "Automatic target tracking in FLIR image sequences using intensity variation function and template modeling," IEEE Trans. Instrum. Meas. **54**, 1846–1852 (2005).
9. H.-W. Chen, S. Sutha, and T. Olson, "Target detection and recognition improvements by use of spatiotemporal fusion," Appl. Opt. **43**, 403–415 (2004).
10. J. F. Khan, M. S. Alam, and S. M. A. Bhuiyan, "Automatic target detection in forward-looking infrared imagery via probabilistic neural networks," Appl. Opt. **48**, 464–476 (2009).
11. Z. Zalevsky, D. Mendlovic, E. Rivlin, and S. Rotman, "Contrasted statistical processing algorithm for obtaining improved target detection performances in infrared cluttered environment," Opt. Eng. **39**, 2609–2617 (2000).
12. J. S. Shaik and K. M. Iftekharuddin, "Detection and tracking of rotated and scaled targets by use of Hilbert-wavelet transform," Appl. Opt. **42**, 4718–4735 (2003).
13. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," Int. J. Comput. Vis. **60**, 91–110 (2004).
14. H. P. Moravec, "Visual mapping by a robot rover," in *Proceedings of the 6th International Joint Conference on Artificial Intelligence* (Morgan Kaufmann, 1979), Vol. 1, pp. 598–600.
15. C. Harris and M. Stephens, "A combined corner and edge detection," in *Proceedings of The Fourth Alvey Vision Conference* (The British Machine Vision Association and Society for Pattern Recognition, 1988), Vol. 15, pp. 147–151.
16. C. Tomasi and T. Kanade, "Detection and tracking of point features," Technical report (Carnegie Mellon Univ., 1991).
17. C. Park, K. Baea, and J.-H. Jung, "Object recognition in infrared image sequences using scale invariant feature transform," Proc. SPIE, **6968**, 69681P (2008).
18. H. Lee, P. G. Heo, J.-Y. Suk, B.-Y. Yeou, and H. Park, "Scale-invariant object tracking method using strong corners in the scale domain," Opt. Eng. **48**, 017204 (2009).
19. P. B. W. Schwering, H. A. Lensen, S. P. van den Broek, R. J. M. den Hollander, W. van der Mark, H. Bouma, and R. A. W. Kemp, "Application of heterogeneous multiple camera system with panoramic capabilities in a harbor environment," Proc. SPIE, **7481**, 74810C (2009).
20. J.-Z. Liu, X.-C. Yu, L.-Q. Gong, and W.-S. Yu, "Automatic matching of infrared image sequences based on rotation invariant," in *Proceedings of the IEEE International Conference on Environmental Science and Information Technology* (IEEE, 2009), pp. 365–368.
21. Y. Wang, A. Camargo, R. Fevig, F. Martel, and R. R. Schultz, "Image mosaicking from uncooled thermal IR video captured by a small UAV," in *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation* (IEEE, 2008), pp. 161–164.
22. G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proceedings*

of the European Conference on Computer Vision Workshop on Statistical Learning in Computer Vision (Springer, 2004), pp. 59–74.
23. T. Kinnunen, J. K. Kamarainen, L. Lensu, and H. Kalviainen, "Bag-of-features codebook generation by self-organisation," in *Proceedings of the 7th International Workshop on Advances in Self-Organizing Maps* (Springer-Verlag, 2009), pp. 124–132.
24. J. C. van Gemert, J. M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders, "Kernel codebooks for scene categorization," in *Lecture Notes in Computer Science: European Conference on Computer Vision* (Springer, 2008), pp. 696–709.
25. E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classification," in *Proceedings of the European Conference on Computer Vision* (Springer, 2006), pp. 490–503.
26. T. Leung and J. Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons," Int. J. Comput. Vis. **43**, 29–44 (2001).
27. M. Weber, M. Welling, and P. Perona, "Unsupervised learning of models for recognition," in *Proceedings of the European Conference on Computer Vision* (Springer, 2000), pp. 18–32.
28. J. Winn, A. Criminisi, and T. Minka, "Object categorization by learned universal visual dictionary," in *Proceedings of the Tenth IEEE International Conference on Computer Vision* (IEEE, 2005), Vol. 2, pp. 1800–1807.
29. A. Vedaldi, www.vlfeat.org/vedaldi/code/sift.html, SIFT (2010).
30. B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th International Joint Conference on Artificial Intelligence* (Morgan Kaufmann, 1981), Vol. 2, pp. 674–679.
31. J. Shi and C. Tomasi, "Good features to track," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 1994), pp. 593–600.
32. J.-Y. Bouguet, "Pyramidal implementation of the Lucas–Kanade feature tracker," Technical report, Intel Corp., Microprocessor Research Labs, 1999.
33. M. I. Smith, M. Bernhardt, C. R. Angell, D. Hickman, P. Whitehead, and D. Patel, "Validation and acceptance of synthetic infrared imagery," Proc. SPIE, **5408**, 9–21 (2004).
34. J. D. Agostino and C. Webb, "Three-dimensional analysis framework and measurement methodology for imaging system noise," Proc. SPIE, **1488**, 110–121 (1991).