Abstract—Autonomous round-the-clock observation of wide critical maritime areas can be a powerful support for border protection agencies to avoid criminal acts like illegal immigration, piracy or drug trafficking. These criminal acts are often accomplished by using small boats to decrease the probability of being uncovered. In this paper, we present an image exploitation approach to detect and classify maritime objects in infrared image sequences recorded from an autonomous platform. We focus on high robustness and generality with respect to variations of boat appearance, image quality, and environmental condition. A fusion of three different detection algorithms is performed to create reliable alarm hypotheses. In the following, a set of well-investigated features is extracted from the alarm hypotheses and evaluated using a two-stage-classification with support vector machines (SVMs) in order to distinguish between three object classes: clutter, irrelevant objects and suspicious boats. On the given image data we achieve a rate of 97 % correct classifications.

I. INTRODUCTION

Criminal activities at sea such as illegal immigration, piracy or trafficking of drugs, weapons and illicit substances have become an uprising issue in the recent years. Often, small and midsize boats, which are difficult to detect, are used for such activities. Until now, border agencies observe and protect the critical maritime areas by ships, planes or helicopters, which is very expensive and full coverage is difficult to obtain.

To improve this situation, the European research project AMASS (Autonomous MArine Surveillance System) investigates to use a network of unmanned surveillance platforms located in a considerable distance from shore. The platforms are equipped with different sensors, the optical sensors being uncooled thermal infrared cameras. In order to exploit the data delivered by the camera, detection and tracking algorithms which are able to work with a moving sensor under a variety of weather and visibility conditions are required.

In our work, we focus exactly on this topic to investigate and implement approaches for a robust detection of maritime objects, reliable generation of alarms with low rate of false positives and a final classification of these alarms with the aim of distinguishing between irrelevant objects like big ships, buoys or animals, and small, suspicious boats. Thus, a human operator is strongly supported not only due to the autonomous round-the-clock monitoring, but also due to a prior selection of alarms, which are really worth to be transmitted.

Related work for detection approaches includes for example template-based cross-correlation along the prior detected horizon in infrared images [1], edge-detection and -grouping above the prior detected horizon in visual color images [2] and anomaly detection using pixel-wise Mahalanobis distance and mean shift in visual and infrared images [3].

As classification is the main topic of this paper, we go more into detail for the related work in this field of research. In [4], small boats are to be classified in visual and infrared image data. Therefore, a precise segmentation is performed using adaptive progressive thresholding (APT) on the visual and graph-cuts on the infrared images. The object’s shape is the crucial classification information which is extracted by applying a principal component analysis (PCA) directly on the image pixels. Correct classification rates of 84 % for visual and 77 % for infrared data are achieved.

In [5], an on-shore visual camera is used for classification of suspicious boats and ships in a harbor environment. Again precise segmentation and object shape are the main issues.

After a background subtraction and hysteresis thresholding, the MPEG-7 region-based shape descriptor [6] is extracted and classified with k-Nearest-Neighbor (k-NN). About 90 % of the given data is correctly classified.

The next approaches only consider big ships to distinguish between different ship types like tanker, cargo ships, carrier or cruiser. For all classes to be found, three-dimensional ship models are available e.g. for training or to create silhouette templates. In [7], two experimental systems are developed for infrared images. The first is working with edge-histograms and the second with invariant moments (Hu moments) on the silhouette and a neural network. Classification rates of 80 % respectively 70 % are obtained. In [8], the usage of a more precise segmentation approach, Hu moments and a support vector machine (SVM) for classification perform better than in [7] on infrared data. Finally, in [9] the PCA is discovered to produce better classification results using silhouette information than Hu moments in combination with neural networks.

All presented approaches use the shape of the detected objects for classification. This is only possible, if the object is...
fully visible and close enough to the image sensor. We avoid contour and edge information like object silhouette in order to be more independent from the object’s scale, distance, texture, direction and appearance as well as the image quality and the environmental conditions.

This paper provides the following organization: After a short system overview in section II, we describe the detection and alarm generation process in section III. Main part is the classification module including feature extraction and two-stage-classification with support vector machines in section IV. Some evaluation results gained by conducting experiments on current test data are presented in section V. Finally, a conclusion and an outlook to future works are given in section VI.

II. SYSTEM OVERVIEW

The system overview of the whole image exploitation module can be seen in Fig. 1. The algorithms are structured by a layer model. At first, the camera orientation is estimated with an inertial measurement unit (IMU). In the next step, the initial location of the horizon computed from camera orientation is refined based on image analysis. Supported by the just located horizon, the detection takes place in order to find conspicuous areas in the image. The top layer contains the alarm generation as well as the classification of these alarms. For an explicit description of the system itself and the process of system integration, refer to [10].

We focus on the two upper layers in this paper. Hence, Fig. 2 offers a closer look to them. Right after the horizon detection, the three detection approaches to be presented more detailed in the next section are applied. The detection results are passed to the top layer and used as input for the fusion and alarm hypothesis generation unit. Detection results for a short time interval (e.g. 50 images) to be analyzed are collected as a set of bounding boxes in a global scene coordinate frame which is aligned with the estimated horizon line and the initial north direction. The transformation from image coordinates to scene coordinates results in a compensation of estimated camera motion. The scene data structure which is used to store the collected bounding boxes is thus able to provide a motion-stabilized interface to the detection results.

III. DETECTION AND ALARM GENERATION

Exploitation of infrared image sequences is used to detect maritime objects. As seen in Fig. 1, the image exploitation algorithms are structured by a layer model. The boat detection layer uses results from the underlying horizon detection layer to set up search areas for the implemented detection algorithms. Different detection algorithms based on complementary image cues are used in the detection layer. The first algorithm is a track-before detect algorithm [13] using spatio-temporal integrated blob strength, the second one exploits stable image regions [14], and the third algorithm is based on tracking salient image points [15]. The motivation for using a combination of different detection algorithms is the need for robustness with respect to variations of boat appearance, image quality, and environmental conditions.

Detection results for a short time interval (e.g. 50 images) to be analyzed are collected as a set of bounding boxes in a global scene coordinate frame which is aligned with the estimated horizon line and the initial north direction. The transformation from image coordinates to scene coordinates results in a compensation of estimated camera motion. The scene data structure which is used to store the collected bounding boxes is thus able to provide a motion-stabilized interface to the detection results.
In the next software layer, spatio-temporal fusion of the detection results is performed on the scene data structure using a voting procedure. Voting results in a set of bounding box clusters which extend both in space and time. To generate alarm hypotheses and to provide input for the classification algorithm, a representative image has to be determined for each detected cluster. From the images used to build the scene data structure, we select for each spatio-temporal cluster the image which provides the maximum support for this cluster. Support is defined by the number of detections covered by a cluster. From the supporting detections a final bounding box is computed which, together with the selected image, represents an alarm hypothesis that has to be validated by the classification component. In Fig. 3 an example for the detection process is visualized.

![Detected horizon line and bounding boxes resulting from different detection algorithms. Shown is the automatically selected image with the maximum number of supporting detections. Different colors (red, green, and yellow) are used for different detection algorithms.](image)

![Final bounding box (red) computed from the detections in Fig. 3(a). Image and bounding box form a detection hypothesis used as input for classification and alarm generation.](image)

**IV. CLASSIFICATION**

This section is divided in two subsections: The extensive research on feature extraction and the two-stage-classification with SVMs.

**A. Feature extraction**

The feature extraction consists of a *pre-processing step*, where a background subtraction is performed, the *feature calculation*, where a high-dimensional feature vector is created containing information of several feature classes, and finally the *feature evaluation*, where the features are normalized, analyzed and finally reduced in their dimensionality for better separability and faster computation time.

1) *Pre-processing*: As some of the features to be calculated need an object segmentation first, a background subtraction based on a single intensity-threshold is conducted. Four different approaches for an easy and fast background subtraction have been investigated:

- **Bounding box difference image**: The given bounding box is mirrored at the vertical image center axis to create a new bounding box which is assumed to contain only background. The new bounding box image is subtracted from the given one and just the object area itself is left. Its advantage is the high possibility to adapt to each background appearance, but in heavy sea with strong waves, the background is too irregular and some wave areas survive the subtraction as well.

- **Bounding box histogram difference**: Like in the first approach a new bounding box is generated by mirroring the given one. For both bounding box areas the histogram is calculated. The new bounding box histogram is subtracted bin-wise from the given one and the first bin of the difference histogram containing enough elements is taken as gray-value threshold. This way, the background appearance is not a problem anymore, but specific effects like an unequal distribution of sea and sky in the both bounding-boxes can affect the background subtraction negatively.

- **Image histogram difference**: The histograms of the bounding box and the whole image are created. After the whole image histogram has been downscaled bin-wise by using one experimentally determined scale-factor, again the histogram difference is calculated and the first bin of the histogram difference containing enough elements is taken as threshold. This approach is more robust against unequal distributions of sea and sky.

- **Row-wise image histogram difference**: Row-wise application of the same approach as described for image histogram difference right before. This can be performed either in parallel to the horizontal image coordinate axis or to the horizon. Hence, we receive one gray-value threshold per row, which is more robust against background irregularities. On the other hand we need a very precise horizon estimation in order to avoid severe
segmentation mistakes in some rows, where sea and sky merge.
During the evaluation with some experiments, it has been shown, that the image histogram difference performs best. Some examples are presented in Fig. 4.

2) Feature calculation: Without any special consideration or expectation towards the given data, a huge, but very generalized set of features was implemented. With the adjective generalized, we want to emphasize, that no prior knowledge about the target class “suspicious boats” like contour, edge constellation, expected texture, scaling, distance, size, direction, etc. was used. This way we want to avoid specialization and to pay attention to heavy variations of boat appearance, image quality, and environmental conditions. The resulting feature vector had a dimensionality of 342 before the evaluation. To keep track of this complexity, the features were divided in several feature classes. The vector is created simply by concatenating all calculated features. An overview of the chosen feature classes is shown in Fig. 5 and a description of each class is given in the following:

- **Invariant moments**: Hu moments [16][17] are calculated on the bounding box image both with and without background subtraction.
- **Co-occurrence matrices**: According to [18], the gray-tone spatial-dependence matrices (co-occurrence matrices) for the bounding box image are calculated and features extracted like variance, contrast, entropy, sum variance, sum difference.
- **Texture analysis**: Mainly the same features like for the co-occurrence matrices are extracted, but directly on the gray-value bounding box image.
- **Kernel analysis**: After background subtraction, the object blob (kernel) is compared to its surrounding area (frame). Some area next to the object blob is not considered in order to avoid merging effects. Feature extraction consists of calculating relations between the kernel and the frame like means difference, means ratio, variances difference, variances ratio.
- **Row analysis**: For each single row, data like gray-value mean, variance and standard deviation are computed. Features are created e.g. by comparing this data row-wise or by grouping it together to calculate ratios between the upper and lower half of the bounding box for example. This way, the vertical spatial information of the features is kept alive.
- **Blob analysis**: After background subtraction it is assumed that only the object blob is left. This object blob is analyzed by calculating its mean, variance, centroid, central moments.
- **Gradient analysis**: Several filters like Sobel or Normalized Gradients of Gaussians [19] have been tested to create histograms of oriented gradients (HOGs) as well as absolute and oriented gradient images of maximum gradients using an absolute threshold, and all gradients. Various statistical histogram-features are extracted and a texture analysis directly on the gradient images is conducted.
- **Local Binary Pattern analysis**: We follow the investigations of T. Ojala [20] to create circular and “uniform” Local Binary Pattern (LBPs). Feature extraction contains various statistical features of LBP-histograms as well as a texture analysis directly on the LBP-image.

Row analysis, invariant moments, gradient analysis and LBP analysis are performed on both with and without background subtraction. The resulting feature vector containing 342 features is now to be evaluated.

3) Feature evaluation: The feature evaluation aims to find the features with best separability and to discard features with weak separability in order to reduce the feature vector’s dimensionality. This way, the calculation becomes faster, as not all features have to be extracted, and the upcoming classification becomes faster as well, as the input-vector is of lower dimensionality.

At first, the features need to be normalized. Thus, their influence towards the classification task is equalized and the domination of few features with high value range is avoided. Normalization is performed by using standard deviation of each feature. Only the positive samples are used for normalization in order to create a model of the target object’s expected value range.

For evaluation we use a linear discriminant analysis (LDA) to figure out the features with highest variance for the given la-
beled training data. Features with maximum variance between the two classes have best separability. A greedy algorithm chooses the best feature combination with respect to maximize the overall separability.

As we use a two-stage-classification where the first stage separates clutter from objects and the second stage distinguishes between suspicious boats – the target class – or irrelevant objects, the features need to be evaluated separately for being capable for the first or the second stage.

The feature vector for the first stage contains 11 features coming from co-occurrence matrices, texture analysis, kernel analysis, row analysis and gradient analysis. With three as the highest amount of features originating from the same feature class, good orthogonality and low covariance of the used features is guaranteed. Powerful features are standard-deviation as in the clutter examples no objects and thus no strong intensity changes appear, as well as the disparity of kernel and frame mainly due to the same reason.

For object classification in the second stage, we use a feature vector consisting of 7 features gathered from co-occurrence matrices, texture analysis and gradient analysis. Dominant feature class is the gradient analysis, so local intensity changes coded in HOGs and absolute gradient texture offer the highest separability for the decision between a suspicious boat or an irrelevant object on the given data.

### B. Two-stage-classification

As already mentioned, the classification is divided in two stages: Separating objects from clutter in the first and suspicious boats from irrelevant objects in the second stage. In each stage, a SVM is trained for its specific feature set. The SVMs are taken from the OpenCV-library [21], which offers good training and prediction methods based on the libSVM [22]. Fig. 6 shows the whole classification process where initially the big set of features is divided in two feature vectors and used as input for the two SVMs. The first stage is mainly a cross-check of the given alarm as we assume clutter to be filtered by the alarm generation process. Hence, the training data contains only positive samples. Negative samples are created randomly and by mirroring the given positive sample bounding box by the vertical image center axis, if possible. This way, we generated a set of 1877 training samples with 621 positive (object) and 1256 negative (clutter) samples. A training process is applied using cross-validation to iteratively optimize the SVM-parameters.

In the second stage only samples are considered which have been classified as objects in the first stage. For the training, the object data set consisting of 621 elements, which were created by the alarm generation, was manually labeled in 235 suspicious boats and 386 irrelevant objects. The same training process using cross-validation as in the first stage is applied.

### V. Experimental results

The training- and test-data consists of 19 scenes of variable duration with one or more different objects per scene. The images are coming from a thermal infrared sensor and have a resolution of 576×472 pixels. With the alarm generation module, we successfully extracted 749 objects in total from the given data. The upcoming experiments just focus on the classification module.

In order to generate negative training samples for the first stage of classification – the separation of clutter and objects – we created additional samples symmetrically or randomly as described in section IV-B. The whole data set contains 2205 samples with 749 positives (objects) and 1456 negatives (clutter). We divided this set in a training set of 1877 samples with 621 positives and 1256 negatives, and a test set of 328 samples with 128 positives and 200 negatives. The training data was used to train the SVM-parameters and the test data contains just unseen objects for a meaningful evaluation. After the training process, we achieved a correct classification rate of 99.1 % on the test data with only 3 false positives.

For the second stage, the set of 749 objects has been divided in training data containing 621 samples with 235 positives (suspicious boats) and 386 negatives (irrelevant objects), and test data consisting of 128 samples with 54 positives and 74 negatives. The correct classification rate on the test data after the training step was 97.7 %.

The overall correct classification rate of both stages can now be calculated by multiplying the rates of the two single states to get a result of 96.78 % correct classifications on the given data for the three classes: clutter, irrelevant objects and suspicious boats. All results are clearly represented in Table I and some example classifications are given in Fig. 7. The whole

<table>
<thead>
<tr>
<th>classifier</th>
<th>SVM 1 (clutter)</th>
<th>SVM 2 (objects)</th>
<th>2-stage-SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>correct rate</td>
<td>99.10 %</td>
<td>97.66 %</td>
<td>96.78 %</td>
</tr>
<tr>
<td>false positives</td>
<td>0.90 %</td>
<td>0.00 %</td>
<td>0.54 %</td>
</tr>
<tr>
<td>false negatives</td>
<td>0.00 %</td>
<td>2.34 %</td>
<td>2.68 %</td>
</tr>
</tbody>
</table>

feature extraction and classification process needs a processing time of about 140 ms per alarm on a quad core processor with 2.66 GHz and 3 GB of RAM. This is no problem, as the current application only needs a real-time detection for the alarm generation but no real-time classification.
VI. CONCLUSION AND FUTURE WORKS

In this work, we presented an image exploitation approach to detect, verify and classify maritime objects in order to identify small boats, which are often used for criminal activities. Three different detection methods have been fused for high robustness and checked for spatio-temporal stability of the detection results to generate reliable alarms for maritime objects. These alarms are used to extract additional features for a two-stage-classification with SVMs for separating clutter and objects in the first as well as irrelevant objects and suspicious boats in the second stage. We achieved a very promising correct classification rate of about 97% on the currently available data.

Future works will include recording and creating more training and test data including ground-truth of the object’s distance, velocity and direction. This way, we will be able to estimate the distance range for reliable alarms and classifications.

Another topic is the usage of temporal information for the classification process. With a tightly coupled object tracking and classification, temporal and reliable physical features can be extracted and evaluated for their applicability towards the presented classification task.

ACKNOWLEDGMENT

This work has been done in cooperation with Carl Zeiss Optronics GmbH and was supported with funds from the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement No. SP1-Cooperation-218290.
REFERENCES


Classification of small boats in infrared images for maritime surveillance

DOI:
10.1109/WSSC.2010.5730289 (http://dx.doi.org/10.1109/WSSC.2010.5730289)

© 2010 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Details:
Institute of Electrical and Electronics Engineers -IEEE-:
Waterside Security. 2nd International Conference WSS 2010 : 3-5 November 2010, Marina di Carrara, Italy
New York, NY: IEEE, 2010
ISBN: 978-1-4244-8894-0
7 pp.