

A hierarchical oil tank detector with deep surrounding features for high resolution optical satellite imagery

Lu Zhang, Zhenwei Shi*, *Member IEEE* and Jun Wu

Abstract

Automatic oil tank detection plays a very important role for remote sensing image processing. To accomplish the task, a hierarchical oil tank detector with deep surrounding features is proposed in this paper. The surrounding features extracted by the deep learning model aim at making the oil tanks more easily to recognize, since the appearance of oil tanks is a circle and this information is not enough to separate targets from the complex background. The proposed method is divided into three modules: candidate selection, feature extraction and classification. Firstly, a modified Ellipse and Line Segment Detector based on gradient orientation is used to select candidates in the image. Afterwards, the feature combing local and surrounding information together is extracted to represent the target. Histogram of Oriented Gradients which can reliably capture the shape information is extracted to characterize the local patch. For the surrounding area, the Convolutional Neural Network trained in ImageNet Large Scale Visual Recognition Challenge 2012 contest is applied as a blackbox feature extractor to extract rich surrounding feature. Then the linear Support Vector Machine is utilized as the classifier to give the final output. Experimental results indicate that the proposed method is robust under different complex background and has high detection rate with low false alarm.

Index Terms

Oil tank detection, Surrounding information, Deep learning, Convolutional Neural Network, Ellipse and Line Segment Detector.

The work was supported by the National Natural Science Foundation of China under the Grants 61273245 and 91120301, the Beijing Natural Science Foundation under the Grant 4152031, the funding project of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University under the Grant VR-2014-ZZ-02, the Fundamental Research Funds for the Central Universities under the Grant YWF-14-YHXY-028 and the Grant YWF-15-YHXY-003, and the Open Research Fund of The State Key Laboratory of Space-Ground Integrated Information Technology under grant NO. 2014_CXJJ-YG_08. (Corresponding author: Zhenwei Shi)

Lu Zhang (e-mail: lu_zhang0928@163.com) and Zhenwei Shi (Corresponding Author, e-mail: shizhenwei@buaa.edu.cn) are with State Key Laboratory of Virtual Reality Technology and Systems, School of Astronautics, Beihang University, Beijing 100191, China and with Beijing Key Laboratory of Digital Media, Beihang University, Beijing 100191, China and also with Image Processing Center, School of Astronautics, Beihang University, Beijing 100191, China.

Jun Wu (wujunok@foxmail.com) is with Space Star Technology Co., Ltd. Beijing 100086, China and with State Key Laboratory of Space-Ground Integrated Information Technology Beijing 100086, China.

I. INTRODUCTION

With the success of remote sensing technology, more high-resolution data is now available, facilitating a wide range of applications such as city surveying, disaster monitoring, military reconnaissance. In these applications, automatic object detection plays a very important role and has received increasing research interests. Some relevant works have been done for the task [1] [2] [3]. The feature extraction and learning methods [4] [5] [6] also provides some helps and inspirations for this topic.

In the optical satellite images, oil tank storing such valuable products which are necessary for transportation and industrial production is also one of the important targets [7]. Some early attempts about the oil tank detection have been proposed. Chen *et al.* [8] proposed a hierarchical model for oil depot detection, including image segmentation, circular oil tank detection and localization. Li *et al.* [9] applied a gradient fuzzy Hough transform for oil tank detection to avoid computational complexity and false diffusion peaks, then a post-processing method was used to remove false positives. Han *et al.* [10] raised a developed Hough transform to select oil tanks, and a graph search strategy was presented to cluster the selected areas. Soon after, Han *et al.* [11] proposed a two step method based on a saliency model and graph search. Yao *et al.* [12] presented a method based on salient region and geometric features. Cai *et al.* [13] proposed an algorithm by visual saliency and Hough Transform. Zhu *et al.* [14] put forward a coarse-to-fine framework for oil tank detection, and the framework was composed of two operations: oil tank selection based on the probabilistic latent semantic analysis model and oil tank detection with Hough transform, template matching. Kushwaha *et al.* [15] proposed a hierarchical model to detect circular shaped bright oil tanks in satellite images. But there is a problem existing in the aforementioned methods that the focused targets are selected from the well-contrast areas where the targets are generally brighter than the background. This may not be the case for practical applications when the oil tanks are in low-contrast. In 2014 Ok *et al.* [16] proposed a method based on the tank's shadow information regardless of the contrast constraints, but this have a low efficiency when the shadows are not visible or complete. Then in 2015 Ok *et al.* [7] [17] raised an approach considering the symmetric nature of circular oil depots. However, the above methods still focused on the local area with simple shape information and this could not be enough to discriminate oil tanks from the complex background (See the first row in Figure 1), and there might not be further information that could be used in terms of the local patch.

To further investigate the existing problem, we proposed a novel algorithm taking the surrounding information into consideration. Similar ideas have been used in other satellite object detection tasks as prior knowledge, for instance, detecting cars in the asphalted area [18] [19], detecting aircrafts in the airport [20] [21], detecting ships at sea [22] [23]. For oil tanks, the area around the target is apt to include shadows, pipelines and some other oil tanks, and this information can help us distinguish targets from the background. In Figure 1, it's hard to distinguish the objects in the first row, but if the surroundings in the second row are taken into account, the objects are more easily to recognize. In this case, we combine local and surrounding information together to represent the targets' area.

The proposed algorithm can be divided into three operations: candidate selection, feature extraction and classi-

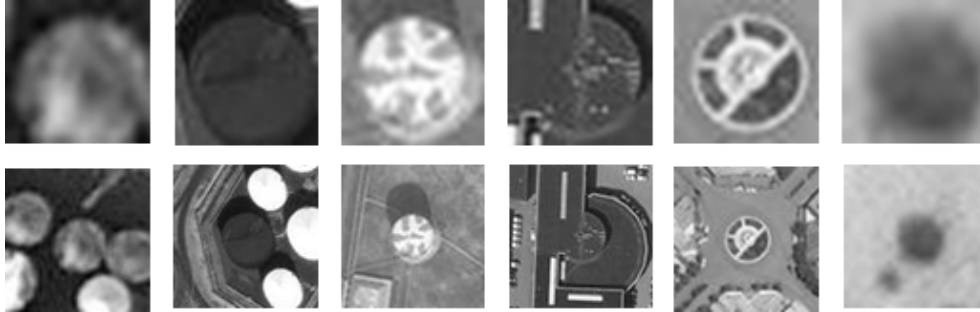


Fig. 1. Objects and their surroundings. The first, the second and the third column are oil tanks and their surroundings; the forth (pool), the fifth (crossroad) and the sixth (bush) column are negative samples with their surroundings.

fication.

The candidate selection herein is used to improve the efficiency of the system. Traditional manners using sliding windows on satellite images were computationally inefficient and suffered from multi-scale trouble. Instead, the problem is solved by using the “recognition using regions” paradigm which has been announced success for object detection task [24] [25]. In this paradigm, the candidate regions are firstly selected and resized to the fixed size. Then the resized regions are input to extract features and classify to generate the final output. A circle detection method is utilized to select candidates in the image because of the circle shape of oil tanks. Automated circle extraction has been an open research area. A classical method is standard circular Hough transform (SCHT) first proposed by Duda and Hart [26]. It makes use of the edge detection result of the image and converses the result to the parameter space whose dimension is predefined as three (radius and center coordinates) via an accumulation process. The peaks in the parameter space correspond to the circles in the original image. A number of methods were also proposed to decrease the computation complexity and improve the efficiency of the method [27] [28] [29]. However, the Hough based methods may fail when the edge of oil tanks are not clear enough. In this case, we apply a modified Ellipse and Line Segment Detector (ELSD) [30] in the paper for the candidate selection. The ELSD based on gradient orientation shows robustness for oil tanks under low contrast with the background, and the modified ESLD is much more accurate than the traditional manner.

Features combining local and surrounding information are then extracted on the regions announced by the candidate selection method. Histogram of Oriented Gradients (HOG) [31] which shows good performance to capture shape information is used to represent the local area. But what features should be extracted from the surroundings? Adding a bad representation of the surrounding area may lead to a even worse result. Local Binary Pattern (LBP) [32], HOG [2], Gabor [33] are the most popular handcrafted features for object detection task [3]. HOG can capture the shape feature; LBP and Gabor can extract the texture of the image. But their performances are not that satisfying as it is really hard to give accurate surrounding descriptions even for humans, and it is also tedious to design another new feature. Deep learning, which has become a hot-spot since 2006 [34] [35] and made remarkable achievements in many domains, can learn the surrounding features automatically. Convolutional Neural Network

(CNN) is one of the typical deep learning models and shows good performance in image processing field [25] [36] [37]. So it is appropriate and promising to apply the CNN to extract surrounding features. However, training a large CNN is a very challenging task as the labeled oil data currently available is very scarce. In this case we utilize the Krizhevsky's CNN model [36] trained in ImageNet Large Scale Visual Recognition Challenge 2012 (ILSVRC2012) contest as a blackbox feature extractor and apply a classifier to give the final result. Then we just need to train the classifier which does not demand too much training samples. Such algorithm can also be considered as a transfer learning model from ImageNet to the oil tank detection task, and the transfer ability of Krizhevsky's CNN model has already been verified in RCNN [25], DeCAF [37]. In practice the CNN feature shows better performance than the classical handcrafted features. A feature visualization method t-SNE (unsupervised learning feature visualization algorithm proposed by Maaten and Hinton, 2008 [38]) is also applied in the experimental part to evaluate CNN feature's performance and the function of the surrounding information.

Support Vector Machine (SVM) proposed by Cortes and Vapnik is an important classification algorithm of statistical pattern recognition [39]. It can select the optimal hyperplane with high confidence in terms of training data and its corresponding labels. In recent years, SVM has been widely used in deep learning transfer models as the final classifier [25] [40] [37]. DeCAF [37] used deep features with linear SVM, which shew better performance compared with Logistic Regression in Caltech-101 and office dataset. RCNN [25] applied 4096-dimensional CNN feature and linear SVM for PASCAL VOC detection challenge and achieved state-of-art results. DLSVM [40] compared SVM with softmax for the deep features in MINIST and CIFAR-10 dataset, and SVM is better behaved. The SVM's performance may owe to the idea of the largest margin, which generates an optimal hyperplane separating the positive and negative samples as much as possible, and this could improve the generalization ability between different datasets. Therefore, switching from traditional softmax or logistic manners to the SVM seems to be credible and appears to be helpful to the classification task in the transfer model. In this case, we utilize the linear SVM in our algorithm to give the final detection result.

The overview of the whole oil tank detection system is shown in Figure 2. In the figure, the original image is input to extract the candidates firstly. Afterwards, HOG features are extracted on the local area, CNN features are extracted on the surrounding area, and then the linear SVM classifier gives the final output. In the final detection result, the red circle indicates the missing target, the green circles refer to the correct detections, and the blue circles would be also shown in the figure if there are some false positives .

The main contributions of our approach are summarized in two aspects below:

- 1) We propose a novel strategy combining local and surrounding information for oil tank detection, which can be helpful to distinguish targets from the complex background.
- 2) The Krizhevsky's CNN model is utilized as a blackbox feature extractor for the surrounding area of satellite images in this paper for the first time and shows better performance than classical handcrafted features.

The remaining paper is organized as follows: Section II details the candidate selection process; Section III depicts the feature extraction process; Section IV presents the experimental results; Section V provides the conclusions and future works of this paper. Section VI gives the acknowledgement.

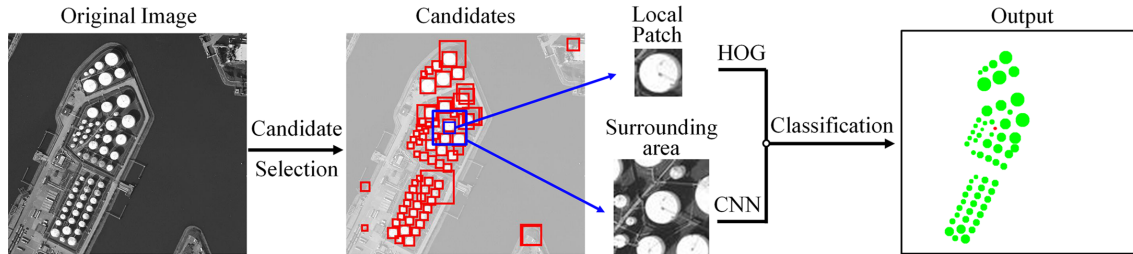


Fig. 2. Outline of our algorithm, which can be divided into three modules: candidate selection, feature extraction and classification.

II. CANDIDATE SELECTION

Traditional detection approach using sliding window suffered from large computation and multi-scale problem. For remote sensing process, the image is often very large and the oil tanks are in different scales. In this case the sliding window manner may be infeasible for oil tank detection in high-resolution images. Thus, a candidate selection method is utilized to improve the efficiency of the system. Since the circle shape of the oil tanks, it is rational to think of a circle detection method for the candidate selection. Hough transform [9] [10] [26] is one of the most popular choices, but it may fail to handle the real images when the edges of oil tanks are not clear enough. Then the ELSD method which relies on the orientation of gradients shows robustness for oil tanks under low contrast with the background. However there is still one problem that the ELSD has a lot of unrealistic selections in the background making the computation a little more inefficient. To improve the property of the method, we modify the method and give another validation step after the ELSD process. The detail of the modified ELSD is stated below.

A. ELSD

ELSD, a parameterless line segment and elliptical arc detector [30], is a two-stage process in our algorithm: curve candidate selection and validation.

In the candidate selection step, the process is similar to the region growing. Region growing groups pixels together if they share same gradient orientation up to a given rule. For the curve candidate selection, the process chains neighbor rectangle regions under some constraints. The rectangle regions can be accessed by the region growing. The constraints presented here are that the candidate regions should be convex and roughly smooth. Additionally, a conic fitting technique is used to compute the parameters that fit the selected regions.

The validation step, which is based on the probabilistic contrario approach introduced by Desolneux *et al.* [41], makes a comparison between the candidate with a completely stochastic area. The number of aligned pixels in the candidate region is firstly computed in this step. For a curve candidate c , a pixel p is said aligned up to a precision σ if

$$\text{Angle}(\nabla x(p), \text{dir}_{\perp}(\tan_c(p))) \leq \sigma\pi \quad (1)$$

where $Angle(\mathbf{a}, \mathbf{b})$ indicates the absolute angle value between \mathbf{a} and \mathbf{b} , $\nabla x(p)$ is the gradient of the image x at p and $dir_{\perp}(tan_c(p))$ indicates the direction orthogonal to the tangent line to the curve c in p . The parameter σ is set to $1/8$, which is proved to be satisfactory in practice [30] [42].

After that, $k_x(c_i)$ is got as the number of aligned pixels in curve candidate c_i in the original image x . Assume the number of aligned pixels in stochastic image X is $k_X(c_i)$. Making a comparison, the candidate region in the original image can be said a false positive with a probability $P(k_X(c_i) \geq k_x(c_i))$. In practice, a binomial law can be used to compute the probability, as the pixels in X are independent variables with σ probably to be aligned in stochastic image.

$$\begin{aligned} P(k_X(c_i) \geq k_x(c_i)) &= B(l(c_i), k_x(c_i), \sigma) \\ &= \sum_{i=k_x(c_i)}^{l(c_i)} C_l^i \sigma^i (1-\sigma)^{l-i} \end{aligned} \quad (2)$$

$l(c_i)$ is the total number of pixels in the candidate area c_i .

Additionally, the size of the image and 6 free parameters representing the circular arcs are used to estimate the candidates number in the image. Suppose the image is $m \times n$, then the number of candidates in the image can be estimated as $(mn)^3$. The final circular arc validation standard is described as follows:

$$NFA_{circle} = (mn)^3 B(l, k, \sigma) \leq \varepsilon \quad (3)$$

ε can be considered as false detection under our tolerance in the image. It is set to a small number 1, and the same value has been used in [30] [42].

So far, the process of the traditional ELSD is finished, and we can get the selected curves as well as their fitting parameters.

B. Modified ELSD

The traditional ELSD based on the region growing manner can only detect continuous circular arcs and is sensitive to outliers in the circle. In this case, only part of the circles and a lot of circular arcs in the background can be detected at the same time. But these selections in the background do not stand for circles and can be further removed if we just focus on the circle detection. Therefore, we propose another validation step after the ELSD process to improve the computation efficiency of the traditional manner. Similar ideas have been used in [16] [17] to evaluate the roundness of the selected circles.

In this paper, the fitting parameters (center coordinates and radius) announced of the valid curves are used to create a complete circular ring firstly. The pixels on the ring should satisfy the following constraint:

$$|\sqrt{(Cir_r - Cen_r)^2 + (Cir_c - Cen_c)^2} - R| \leq \eta \quad \eta \geq 0 \quad (4)$$

where Cir_r and Cir_c respectively indicate the row and column index of the pixels on the ring, Cen_r and Cen_c respectively indicate the row and column index of the circle's center, R is the circle's radius. Cen_r , Cen_c and R are obtained by using the traditional ELSD. η is the threshold that controls the thickness of the ring. In practice,

a very large η could include some irrelevant pixels that do not have useful gradient information, a very small η for example 0 could result in a discontinuous ring. The value of the parameter is experimentally set to be 1 in this paper, the experiment part will give detailed explanations.

Then the number of the aligned pixels is computed on the ring not just on the circular arcs. The aligned principle is the same as ELSD. The aligned ratio of the circle is computed as below:

$$R_{circle} = \frac{k_x(circle)}{l_{circle}} \quad (5)$$

where l_{circle} indicates the total pixels of the ring, $k_x(circle)$ is the number of aligned pixels. A threshold is set to segment the results to get the final validated selections. After this step, the circles with continuous or discontinuous circular arcs can be reserved, and in the meantime some false positives are removed. Algorithm 1 gives the main steps of the modified ELSD. In this algorithm, the traditional ELSD stops at step 14, and step 15 to step 18 indicate the final circle validation in the modified ELSD. Some of the comparison results between the traditional ELSD and

Algorithm 1 Modified ELSD

- 1: **Input:** Gray-scale image x , **parameters:** Threshold r .
 - 2: $grad \leftarrow$ compute the gradients of input image x ;
 - 3: **for** pixel p_i in x **do**
 - 4: $RL \leftarrow$ line region grows using seed point p_i and gradient information $grad$;
 - 5: $RC \leftarrow$ initialize the curve region using line region RL ;
 - 6: **for** endpoints in RC **do**
 - 7: $RL \leftarrow$ line region grows using endpoint in RC and gradient information $grad$;
 - 8: $line \leftarrow$ estimate the rectangle parameters of the line region RL ;
 - 9: $RC \leftarrow$ curve region grows using the previous curve region RC and line parameter $line$;
 - 10: **end for**
 - 11: $curve \leftarrow$ estimate the curve parameters of the curve region RC ;
 - 12: $CvN \leftarrow$ compute the number of aligned pixels in the region RC using the parameter $curve$;
 - 13: $NFA_{curve} \leftarrow$ compute curve decision principle using aligned pixel number CvN and parameter $curve$;
 - 14: $Cv \leftarrow$ validate the RC using NFA_{curve} ;
 - 15: $circle \leftarrow$ get the center and radius parameters by the parameter $curve$ in the validated curve Cv ;
 - 16: $CN \leftarrow$ compute the number of aligned pixels on the circular ring using the parameter $circle$;
 - 17: $Cr \leftarrow$ compute the ratio of the circle using aligned pixel number CN and the parameter $circle$;
 - 18: $C \leftarrow$ validate the circle using the ratio Cr and threshold r ;
 - 19: **end for**
 - 20: **Output:** list of valid circles.
-

the modified ELSD are shown in Figure 3. In the figure, the first column refers to the original images, the circular arcs in the second column indicate the results announced by the traditional ELSD and the rectangles in the third

column reflect the results of the modified ELSD. We can see that the number of rectangles is much less than the circular arcs, and the modified ELSD will not miss any targets.

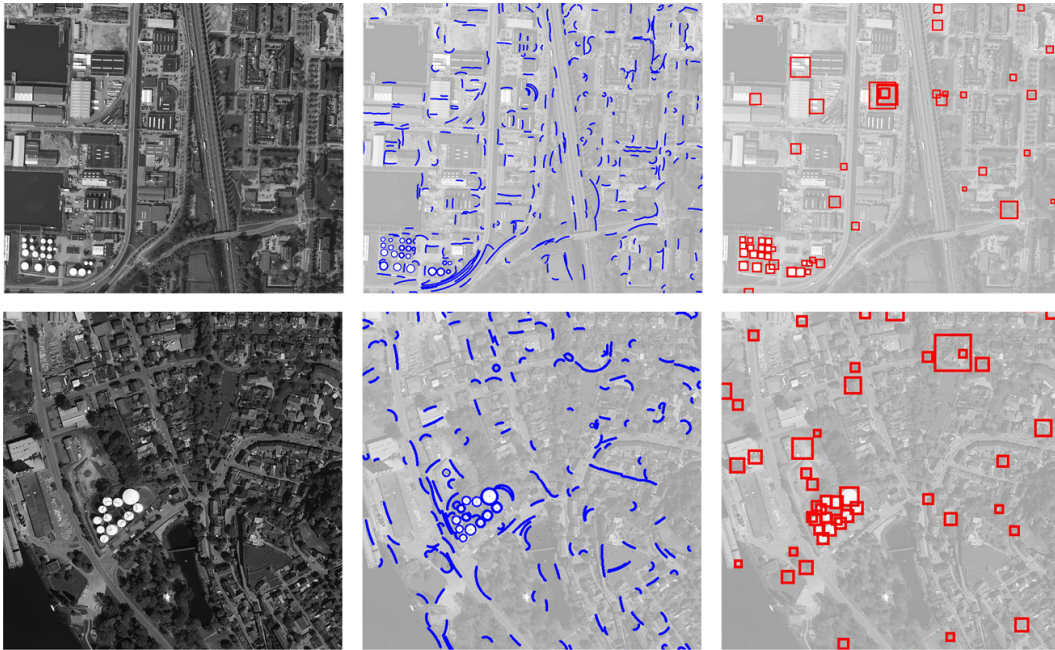


Fig. 3. Comparison between the modified ELSD and the traditional ELSD. The first column refers to the original images. The circular arcs in the second column indicate the selected results announced by the traditional ELSD. The rectangles in the third column reflect the results of the modified ELSD.

III. FEATURE EXTRACTION

Features combining local and surrounding information are extracted in this step on the regions announced by the candidate selection method. The surrounding area of the oil tank which is apt to include shadows, pipelines and some other oil tanks can make the target more easily to recognize. A transfer learning CNN model trained in ILSVRC2012 contest is applied for the oil tank detection task to extract rich surrounding feature, and we call it as Surrounding-CNN feature. As for the local patch, it can reflect the characteristics of the oil tank. In practice, there are some negative samples that contain more than one oil tank or are close to the target, and their surrounding areas also include some oil tanks. Some examples are shown in Figure 4. In this case, just the surrounding feature is probably confused and makes a wrong decision. Therefore the local feature is combined helping to improve the system's property. The HOG is used to extract the local feature, and we call it as Local-HOG feature. The final combined feature is named as LHOG-SCNN (Local-HOG plus Surrounding-CNN) feature in this paper. The details of the LHOG-SCNN feature process are stated below.

The size of the local patch is selected as 1.2 times of the circle's diameter. The size of the surrounding area is used as 3 times of the circle's diameter. The experiment will give the explanation about the two parameters (local



Fig. 4. Some negative samples whose surrounding areas include some oil tanks. These samples announced by the modified ELSD are cut from different images. The first row indicates the local patches, the second row refers to their corresponding surroundings. The positions of the local patches are shown with rectangles in their corresponding surrounding areas.

patch size and surrounding area size). The diameters of the circles are got from the candidate selection result. The two kinds of features are chained together to represent the candidates at the end of this step.

A. Local-HOG Feature

HOG, which shows good performance to reflect the shape information [2] [31], is used to extract the local feature. It is based on the idea that the distribution of gradient directions can characterize the shape or the appearance of the objects rather well [31].

In the HOG process, the gradient image is firstly computed. After that, the histogram of gradient directions is counted in each cell of the gradient image. Then the histograms are connected and normalized over a larger block which contains several cells. At last the normalized histograms of blocks sliding over the whole image are combined to generate the final HOG descriptor.

For example, in terms of a 50×50 patch from the original image, the cell size is 10×10 , and the size of a block containing four cells is 20×20 . The gradient direction values are quantified into nine direction bins. Then for each cell, a 9-dimension feature of the histogram is got from the gradient patch. Afterwards, the four cells' histograms are connected and normalized by the energy of the block, and output the $4 \times 9 = 36$ -dimension feature. The block is set to slide through the gradient patch with the step of 10 pixels. At last the $16 \times 36 = 576$ -dimension HOG feature is got to express the 50×50 patch. Figure 5 shows the main process of HOG.

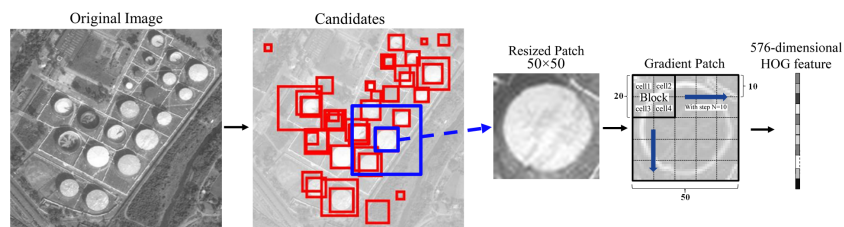


Fig. 5. Process of HOG feature extraction.

In our algorithm, the patch size, cell size and block size are just the same as the example stated before. The

576-dimension HOG feature is extracted from the local candidate patch announced by the modified ELSD.

B. Surrounding-CNN Feature

Convolutional Neural Network (CNN) is a multi-layer structure. With the rise of deep learning [34] [35], it has become one of the research hot spots and shows good performance in image processing. The structure of the CNN model is shown in Figure 6.

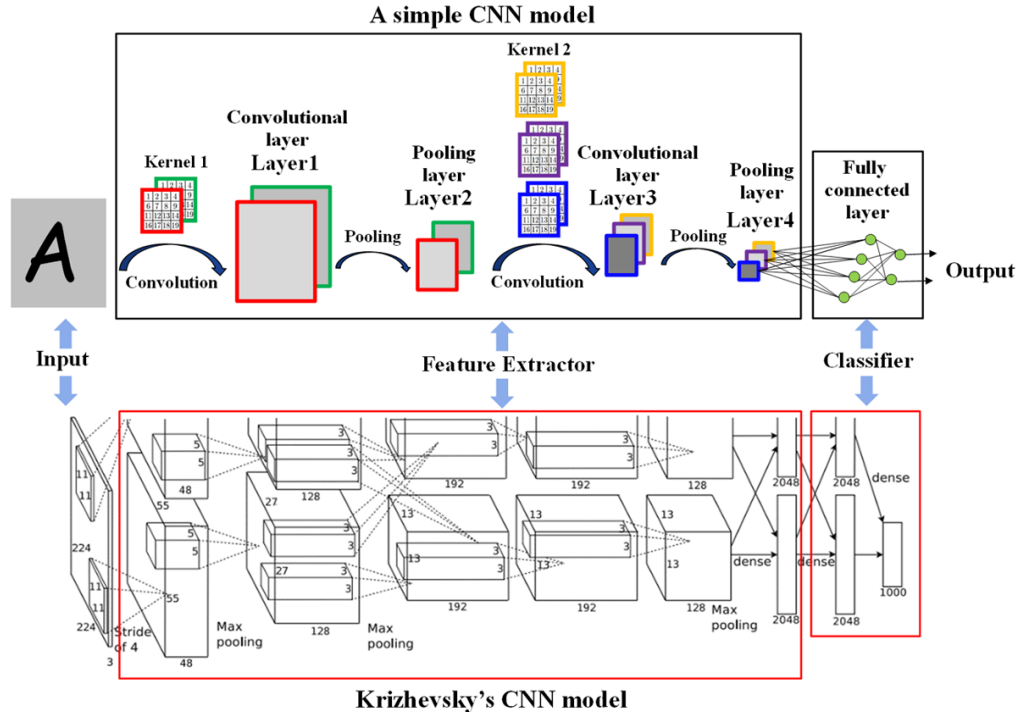


Fig. 6. CNN models' structure, the upper is a simple CNN model and the lower is Krizhevsky's CNN model in 2012 [36].

In Figure 6, the upper is a simple CNN model and the lower is Krizhevsky's CNN model in 2012 [36]. Both of them are composed of two stage process: feature extraction and classification. The treatment of the each phase (set of convolutions with optional pooling) within the feature extraction stage represents the information extraction from raw pixels, to low-level features, to mid-level features, up to concept-level features that are fed into the classifier (fully connected layers) to give the final output.

At a convolution layer, the feature maps of the previous layer are convolved with learnable kernels, and the generated results plus a bias parameter are input to the activation function to form the output feature map. Each output map may correspond to convolutions with multiple input feature maps. For example, considering the convolution process between Layer2 and Layer3 in the simple CNN model in Figure 6, the kernels with the same color in Kernel2 correspond to the two feature maps in Layer2 and generate one output feature map with the same color in

Layer3. Its mathematical form is stated below:

$$x_j^l = f \left(\sum_{i \in M_j} x_i^{l-1} * k_{ij}^l + b_j^l \right) \quad (6)$$

Where x_j^l represents the j_{th} feature map in layer l , x_i^{l-1} represents the i_{th} feature map in the previous layer, k_{ij}^l is the learnable kernel, b_j^l is the bias parameter, $f(\cdot)$ refers to the activation function and M_j indicates a selection of input feature maps. Usually, a convolutional layer is interspersed with a pooling layer to reduce algorithm complexity and to gradually build up further rotation and translation invariance, but this is not absolute. At a pooling layer, the number of output feature maps is the same as the input. It generates a downsampled version of the input feature map. The process is simply described as follows:

$$x_j^l = pooling(x_j^{l-1}) \quad (7)$$

Where x_j^l represents j_{th} feature map in layer l , x_j^{l-1} represents the j_{th} feature map in layer $l - 1$, ‘pooling’ corresponds to the downsample process. Considering the pooling process between Layer1 and Layer2 in the simple CNN model in Figure 6, the number of feature maps between the two layers are the same and every feature map in Layer2 generates one feature map with the same color in Layer3 using the pooling process. Some of the common pooling manners include max-pooling, average-pooling, etc.

In practice, training a deep CNN is a very challenging task since the labeled oil data is very scarce. Therefore an existing CNN model trained by Krizhevsky in the ILSVRC2012 contest is applied to extract surrounding feature in our algorithm. The structure of the model can be seen in Figure 6. The network requires a colored image input with fixed size (224×224), and generates 1000 dimensional output. Its structure includes five convolutional layers, some of which are followed by max-pooling layers. The activation function of the convolutional layer is the non-saturating nonlinearity function $f(x) = max(0, x)$. Three fully connected layers are connected at last generating the final result. Moreover, some tricks such as local response normalization, overlapping pooling and drop out have been used in the network to improve its property. The network is trained through the training set of 1.2 million samples. The training process may take five to six days on two NVIDIA GTX 580 3GB GPUs. The model outperformed all other methods in the recognition challenge at that time.

This paper is the first to show the dramatic performance of Krizhevsky’s CNNs model for the oil tank detection task. It can be considered as a transfer learning structure form ILSVRC-2012 to oil tanks. The transfer ability of this model has already been verified in DeCAF [37] and RCNN [25].

DeCAF analyzed and visualized the deep convolutional features for scene recognition, object detection and domain adaptation tasks. For scene recognition, the features trained in ILSVRC2012 were generalized to SUN-397, and the features showed very good clustering performance. In the object detection task, the trained CNN model outperformed the previous methods in the Caltech-101 database. For domain adaptation, the deep convolutional features were used for office data from different camera devices and illustrated robust ability to resolution changes.

RCNN applied the CNN model to the object detection task on the PASCAL VOC dataset, yielding state-of-art detection performance.

In our algorithm, the CNN model is used as a blackbox feature extractor. Specially, the surrounding area of the valid circle is firstly resized to the compatible CNN size (224×224). After that, the gray scale image is copied to 3 channels generating the colored image that meets the model's input requirement. The 4096-dimensional output of the last feature extract layer is used as the extracted feature (DeCAF and RCNN have used the model in the same way).

The main process of this step is shown in Figure 7.

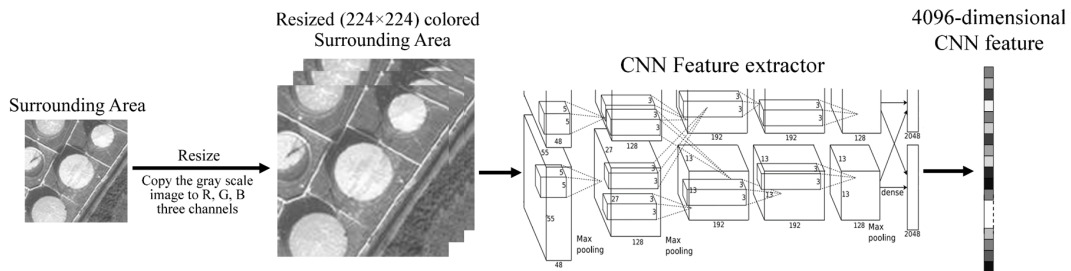


Fig. 7. Process of the Surrounding-CNN feature extraction.

IV. EXPERIMENTS

The influence of some parameters and the performance of the proposed method are evaluated in this section.

A. Dataset

The process of the proposed algorithm is all based on the data downloaded from Google-Earth. Totally, we get 54 large colored images with unified resolution of 1m. 42 images are used for parameter selection and classifier training, and the rest are used for the test. We compute the mean value of images' three channels (RGB) to get their gray-scale model and just use the gray-scale images in our algorithm. The purpose of this is to reduce the requirement for the data resource and to make this method a general application for different image models.

The training samples used for the SVM classifier are got from the 42 training images according to the result of the modified ELSD. We get 11383 samples with 4264 positive samples and 7119 negative ones. However, the surrounding area of these samples do not include all situations, since the arrangement of oil tanks in the surrounding area is directional. It can be seen in Figure 8, the samples are got from different training images. Therefore, we enlarge the data by means of rotation to improve the system's generalization ability. In detail, we rotate the samples every 45° . This means that we can get another 7 different new samples from the original one. Finally we get 91064 training samples with 34112 positive ones and 56952 negative ones. Details of the SVM training samples are shown in Table I.

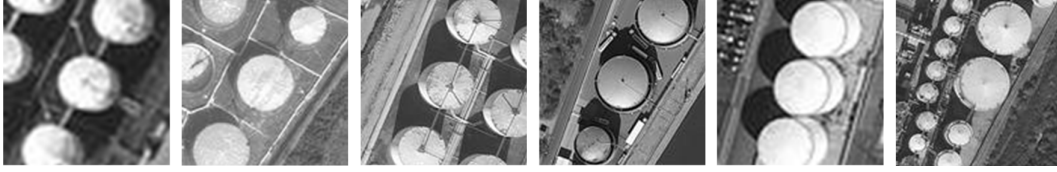


Fig. 8. The arrangement of oil tanks in the surrounding area is somewhat directional (the samples are got from different training images).

TABLE I
DETAILS OF THE TRAINING DATA FOR SVM CLASSIFIER

| | Training image number | Positive samples | Negative samples | Totally |
|------------------|-----------------------|------------------|------------------|---------|
| Before expanding | 42 | 4264 | 7119 | 11383 |
| After expanding | 42 | 34112 | 56952 | 91064 |

The test dataset includes 12 large images with different sizes (the largest is 3712×3008 pixels, the smallest is 1763×1356 pixels). In these images the diameter of the smallest oil tank is about 10 pixels, and the diameter of largest one is about 50. Through observations, the oil tanks whose diameters are smaller than 10 pixels do not have clear circle shape. In this case, we just focus on the targets whose diameters are larger than 10. Actually this is also a very hard detection task. The whole training and test process is shown in Figure 9.

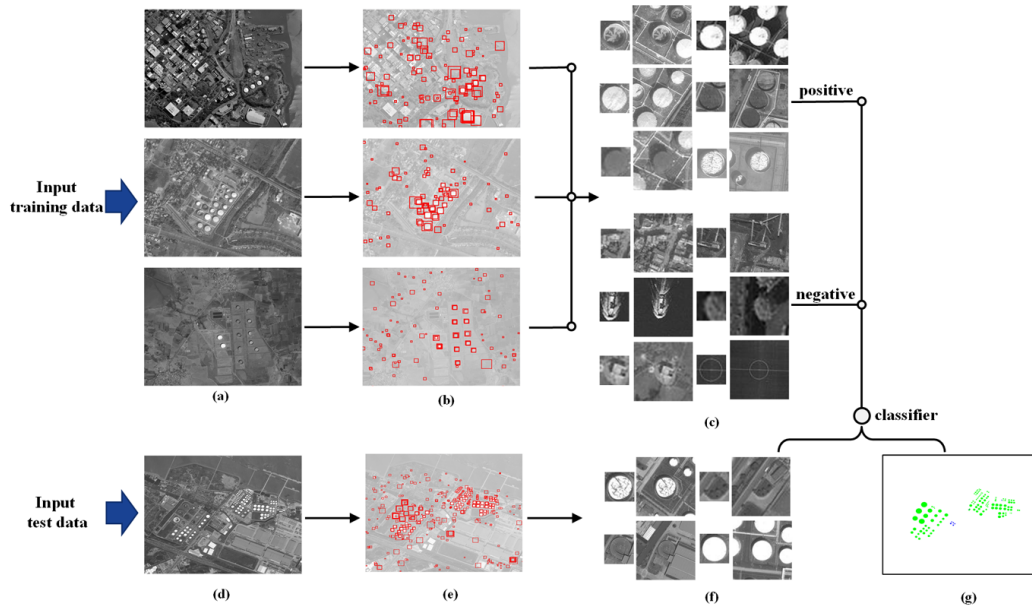


Fig. 9. Training and test process. (a) indicates the training images that can be used for the parameter selection of the modified ELSD. (b) refers to the candidates of the training images. (c) reflects training samples used for the SVM classifier. (d) indicates the test images. (e) refers to the candidates of the test images. (f) reflects the candidate samples of the test images. (g) indicates the final test results.

B. Parameter Selection for the Modified ELSD

The traditional ELSD is a parameterless ellipse detector, and the parameter here refers to the ratio threshold R_{circle} and the circle threshold η in final circle validation step of the modified ELSD. R_{circle} and η can be used to measure the roundness of the selections. In this experiment, the precision-recall graph [43] is applied to evaluate the performance. Precision and recall are defined as Figure 10. TP indicates the positive samples that are selected

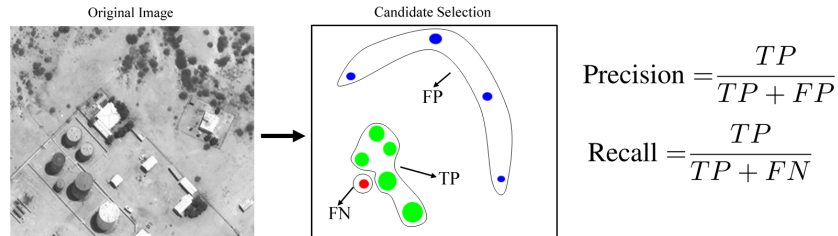


Fig. 10. The definition of precision and recall.

as positive correctly, namely the correct detections. FN indicates the positive samples which are not selected as positive, namely the missing targets. FP refers to the negatives that are considered to be the positive ones, namely the false positives. In Figure 10, the red circle is the missing target FN , the blue circles are the false positives FP , the green circles are the true positives TP . In this case, $Precision = 5/(5 + 4)$, $Recall = 5/(5 + 1)$. The recall corresponds to the ratio of missing targets to the total targets, while the precision corresponds to the ratio of false positives to the total detections. For a candidate selection method, a high recall rate should be guaranteed firstly and on this basis a high precision can help to improve the computational efficiency of the system.

The controlling variables method is utilized in this part to select the two parameters since both of the parameters have influence on the candidate selection result. Firstly for the ratio threshold R_{circle} , we set a fixed $\eta = 1$ and compute the results of different R_{circle} (0, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55) on the 42 training images with 3401 valid oil tanks whose diameters are larger than 10 pixels. The precision-recall graph is shown in Figure 11.

From Figure 11, it can be seen that when ratio threshold switches from 0.4 to 0.45, there is a big change of the system's recall rate. Additionally, when the threshold increases from 0 to 0.4, the recall only descends 0.6175% while the precision grows from 7.77% to 26.12%. In this case, we choose 0.4 as the final validation threshold, and it shows good performance to improve the system's property. Moreover, another issue can be also found in Figure 11 that when the ratio threshold is set 0, there are still some missing targets. These targets are missed by the traditional ELSD process. Through observations, most of them are very small oil tanks, the others are the oil tanks in very low contrast. Both of them do not have clear shape information, and it's really hard to recognize them even for humans. Figure 12 shows some of the missing targets.

Secondly for the parameter η , we set a fixed $R_{circle} = 0.4$ which shows good performance in the previous experiment. Then we choose different circle thresholds (0, 0.5, 1, 1.5, 2, 4) and compute precisions and recalls on the 42 training images. The result is shown in Table II. From the table, one can find that, it is impossible to choose

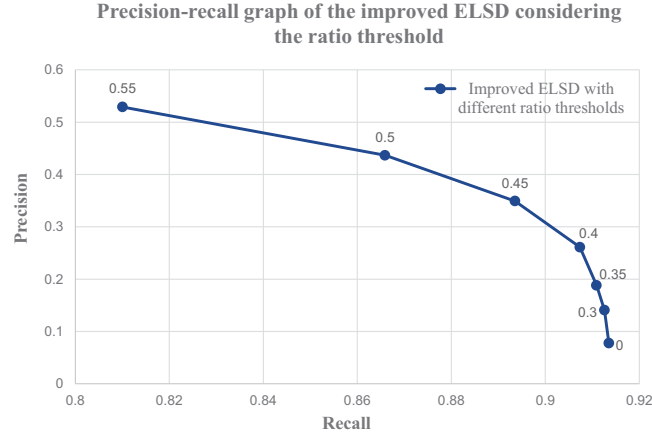


Fig. 11. Precision-recall graph of the modified ELSD considering the ratio threshold.

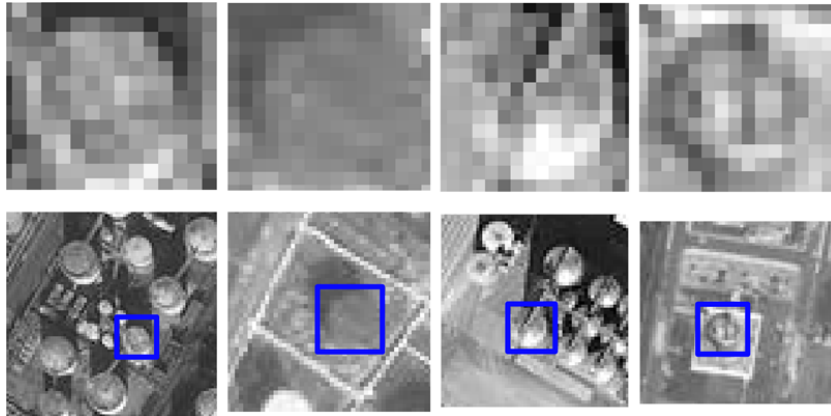


Fig. 12. Some of the missing targets of the modified ELSD when the ratio threshold is set 0. The rectangles in the second row indicate the positions of the missing circles.

a η with the best recall and precision simultaneously. However, a relative balanced results could be obtained if we set $\eta = 1$.

TABLE II
SELECTION RESULT WITH DIFFERENT CIRCLE THRESHOLDS η

| Circle threshold η | 0 | 0.5 | 1 | 1.5 | 2 | 4 |
|-------------------------|--------|--------|--------|--------|--------|--------|
| Recall | 0.8808 | 0.9082 | 0.9074 | 0.9032 | 0.8999 | 0.88 |
| Precision | 0.1036 | 0.1795 | 0.2062 | 0.2180 | 0.2439 | 0.2780 |

C. Parameter Selection for the Penalty Factor of SVM

The classifier used in our algorithm is a linear SVM which is a simple manner but has shown good performance for the CNN transfer learning model. Its mathematical form is described as follows:

$$\begin{aligned} \min_{w,b,\xi} \quad & \frac{1}{2}w^T w + C \sum_{n=1}^N \xi_n \\ \text{s.t.} \quad & y_n(w^T x_n + b) \geq 1 - \xi_n, \quad n = 1, \dots, N \\ & \xi_n \geq 0, \quad n = 1, \dots, N \end{aligned} \quad (8)$$

(x_n, y_n) , $n = 1, \dots, N$, $x_n \in \mathbb{R}^D$, $y_n \in \{-1, +1\}$ indicate the training data and its corresponding labels, $w \in \mathbb{R}^D$, $b \in \mathbb{R}$ are learning parameters, ξ_n is slack variable penalizing data point that violates margin demand. The penalization coefficient C is the only parameter that needs to be determined in this part.

The cross validation process is applied for the parameter selection. Firstly, the training data is equally divided into 4 groups. Every time, three groups are used for the SVM training and the left one is used for the test. Then the SVM with a determined penalty factor can have 4 test accuracies. We can get their mean value to measure the performance of the penalty factor.

The LHOG (Local HOG feature) is used in this experiment to choose the penalization coefficient. The same parameter is used for other SVM classifiers considering the surrounding features. In this case, the result can be more persuasive to evaluate the performance of the surrounding information. We compute the accuracies of different penalty factors (0.01, 0.1, 1, 10) in this experiment. The size of the local patch is set as 1.2 times of the circle's diameter fixedly. The experiment result is shown in Table III. It can be seen that when the penalty is set to be 1, the average test accuracy is higher than the results of other values. Therefore, we select 1 as the value of the penalization coefficient in this paper.

TABLE III
CROSS VALIDATION RESULT WITH DIFFERENT PENALTY FACTORS

| Penalty factor | 0.01 | 0.1 | 1 | 10 |
|-----------------------|--------|--------|---------------|--------|
| Average test accuracy | 0.9271 | 0.9291 | 0.9293 | 0.9290 |

D. Parameter Selection for the feature extraction

The experiments for setting the local patch size and the surrounding area size are discussed in this section. The cross validation result of SVM is utilized to measure different features' performances. The penalty factor of the linear SVM is set to be 1 and the division of the data set is similar as the experiment for the SVM parameter selection. Intuitively, these two parameters are relevant to the size of oil tanks. Both of them can be represented as $x \times R$. x refers to the variable in this experiment and R indicates the circle's diameter announced by the candidate selection result.

For the local patch size, we set different x values (1, 1.1, 1.2, 1.3) and extract HOG features on the local patches with different sizes. We compute the average test accuracies of these HOG features. The result is shown in Table IV. When x is set to be 1.2, the average test accuracy is higher than the results of other values. Therefore, we

TABLE IV
CROSS VALIDATION RESULT OF LOCAL HOG FEATURES WITH DIFFERENT LOCAL PATCH SIZE

| x | 1 | 1.1 | 1.2 | 1.3 |
|-----------------------|--------|--------|---------------|--------|
| Average test accuracy | 0.9265 | 0.9284 | 0.9293 | 0.9247 |

select the local patch size as 1.2 times of the circle's diameter.

For the surrounding area size, we set different x values (2, 3, 4, 5) and extract CNN features on the surrounding areas with different sizes. These CNN features are then combined with the local HOG feature to represent the targets. The size of the local patch is set as 1.2 times of the circle's diameter. We compute the average test accuracies of different combined features. The result is shown in Table V. When x is set to be 3, the average test accuracy is

TABLE V
CROSS VALIDATION RESULT OF COMBINED FEATURES WITH DIFFERENT SURROUNDING AREA SIZE

| x | 2 | 3 | 4 | 5 |
|-----------------------|--------|---------------|--------|--------|
| Average test accuracy | 0.9744 | 0.9754 | 0.9753 | 0.9680 |

higher than the results of other values. In this case we select the size of the surrounding area as 3 times of the circle's diameter.

E. Comparison of Modified ELSD, ELSD and Hough Transform

In this part, the modified ELSD is compared with the traditional ELSD and the Hough transform. At first, the parameters of the three methods are set to get relative good results. The parameters of the modified ELSD are set according to the previous experiment, and there are not any parameters that need to be set in the traditional ELSD. In terms of the Hough transform, it makes use of image's edge detection result and converses the result to the parameter space whose dimension is predefined as three (center coordinate and radius) via an accumulation process. The peaks segmented by the accumulator threshold in the parameter space correspond to the circles in the original image. The function in opencv library is used to get the Hough Transform result in this experiment. Different parameters are set according to the image's standard deviation as it's really hard to set unified parameters for all input images. The result of different methods on the 42 training images along with their average processing time is presented in Table VI. The unit of the processing time is second. It can be seen that the two ELSD methods are faster and more accurate than the Hough transform. Moreover it's really hard to get a relatively optimal Hough

TABLE VI
RESULTS OF DIFFERENT CANDIDATE SELECTION METHODS

| Method | Recall | Precision | Average Processing Time |
|------------------|---------------|---------------|-------------------------|
| Hough Transform | 0.6521 | 0.003507 | 75.56 |
| Traditional ELSD | 0.9136 | 0.07771 | 31.98 |
| Modified ELSD | 0.9074 | 0.2612 | 33.64 |

transform model with a lot of parameters that need to be selected. The modified ELSD has a much higher precision rate than the traditional manners. It means that much less candidates will be input for the following processes (feature extraction and classification). Therefore, the modified ELSD improves the efficiency of the whole detection system.

However, the precision of the modified ELSD is still not satisfying if we consider it as the final detection result. The reason for the low precision can be summarized in two aspects. One is that there are actually some circles in the background such as the bushes. The other one is that the oil tanks are not always critical circles especially for small oil tanks and the oil tanks in low contrast. If we want to detect these ones, we have to loosen our constraints. In this case, some objects with circular arcs or similar to circles in the background can be detected at the same time. This can also reflect that just the circle shape is far from enough for the oil tank detection task. But, the modified ELSD just plays as a candidate selection method in our algorithm. Its performance is already enough for the detection task.

F. Visualization and Comparison of Different Features

A feature visualization method t-SNE (unsupervised learning feature visualization algorithm proposed by Maaten and Hinton, 2008) is applied in this section to find a 2-dimensional embedding of the high-dimensional feature space.

Firstly, the LHOG-SCNN feature is compared with Local-HOG feature to evaluate the function of the surrounding information. The visualization of surrounding-CNN feature alone is also represented in this part. The experiment is tested on the training samples with 34112 positive ones and 56952 negative ones. The result is shown in Figure 13. We can see that the LHOG-SCNN feature combining local and surrounding information together is more separable than the Local-HOG feature. Moreover, in the LHOG-SCNN feature visualization, the positive and negative features are apt to cluster together depending on their labels, and this indicates that the targets' features combining the surrounding information are more consistency than those just focusing on the local patch. As for the visualization of SCNN feature, the samples are also very separable (but shows a little worse cluster ability than the LHOG-SCNN feature). This can demonstrate that the surrounding information is the driving force that can make oil tanks more easily to recognize.

Secondly, the t-SNE algorithm is used to visualize the performances of different features in the surrounding area.

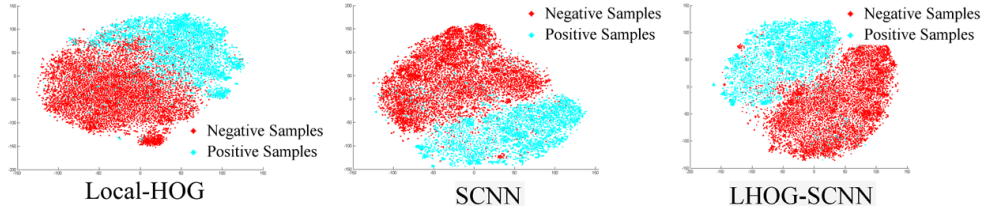


Fig. 13. t-SNE visualization of LHOg-SCNN feature, SCNN feature and Local-HOG feature.

LBP, Gabor and HOG which are widely used for object detection task, are compared with the CNN feature. The result is also obtained on the training dataset.

For LBP feature, each pixel is compared with its 8 neighbors and encoded with the 8 binary values in the uniformed manner to get the Local Binary Pattern image. After this, the histograms in different blocks divided from the image are computed and chained to get the final LBP feature. Specially, the surrounding area is firstly resized to the 225×225 which is almost the same as the capable CNN size (224×224), and then the resized area is divided into 5×5 blocks with block size 45×45 . We count and normalize the histogram of the uniformed LBP in each block, and connect all block histograms to form the final 1475-dimensional LBP feature of the area. The t-SNE visualization result is demonstrated in the upper right corner in Fig. 14.

For Gabor feature, the 48 feature maps after Gabor filter are firstly got. Gabor filter could be written as:

$$g(x, y; \lambda, \theta, \sigma) = \exp\left(-\frac{x'^2 + y'^2}{2\sigma^2}\right) \exp\left(i\left(2\pi\frac{x'}{\lambda}\right)\right)$$

where

$$\begin{aligned} x' &= x \cos \theta + y \sin \theta \\ y' &= -x \sin \theta + y \cos \theta \end{aligned}$$

λ indicates the wavelength of the sinusoidal function, θ refers to the orientation, σ is the standard deviation of the Gaussian envelope, x and y indicate the coordinates of the pixels in the filter. In this paper, 0.8, 1, 1.2 are set as wavelengths λ , 0, 45, 90, 135 are set as flip angles θ and 1, 2, 3, 4 are set as standard deviations σ . Therefore, we get 48 feature maps after the Gabor filter. Afterwards, each feature map is resized to 225×225 , and the resized feature map is divided into 5×5 blocks (45×45) similar to LBP. We compute the mean and variance values of each block, and normalize them of all blocks in each feature map. Then we chained the 50-dimensional feature in each feature map to get a 2400-dimensional (48 feature maps \times 50-dimensional feature of each feature map) feature of the surrounding area. The t-SNE visualization result is shown in the lower left corner in Fig. 14.

For HOG feature, the process is almost the same as the local patch feature extraction. We just need to replace the local patch with surrounding area. The size of the resized surrounding area is 225×225 , the block size is 90×90 , the cell size is 45×45 , and the step size is 45. The t-SNE result is revealed in upper left corner in Fig. 14.

From Fig. 14, it can be seen that the CNN feature outperforms all other features and shows good cluster ability in the surrounding area.

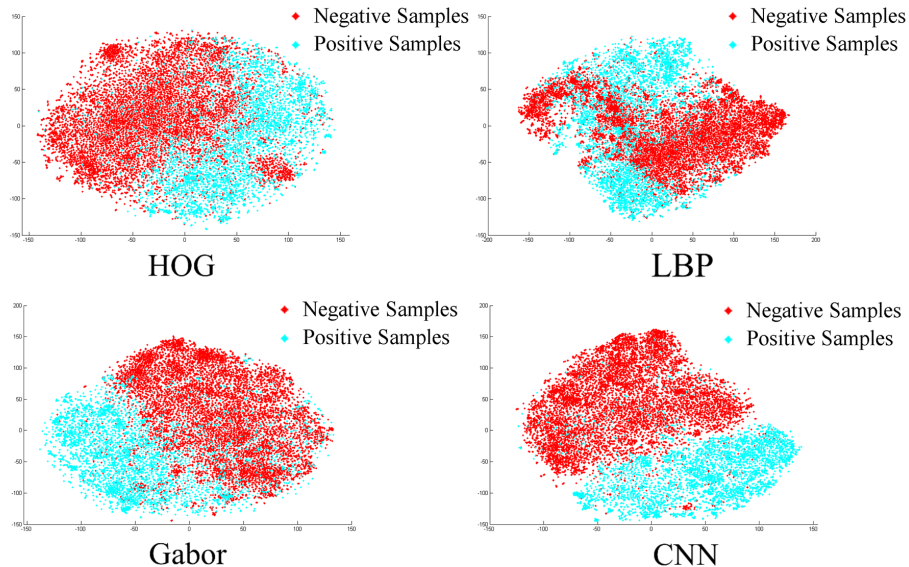


Fig. 14. t-SNE visualization of different features in the surrounding area.

Moreover, we test and record the training accuracy of different features through the training samples. The result is shown in Table VII. The liblinear library in MATLAB form provided by Chih-Jen Lin is applied to accomplish the training task. The training time of different features is also shown in the table. The unit of the training time is second. It illustrates that different features except for LHOG-SCNN and SCNN are not capable to accomplish the difficult training task let alone the test data.

TABLE VII
TRAINING ACCURACY OF DIFFERENT FEATURES ON THE TRAINING DATA

| Local Feature | Surrounding Feature | Training Accuracy | Training Time |
|---------------|---------------------|-------------------|---------------|
| HOG | None | 0.9467 | 22.14 |
| HOG | HOG | 0.9619 | 68.54 |
| HOG | LBP | 0.9778 | 45.06 |
| HOG | Gabor | 0.9782 | 56.26 |
| None | CNN | 1 | 50.03 |
| HOG | CNN | 1 | 49.87 |

The precision-recall graph is also drawn to show the performance of different features on the test dataset. We use the output of SVM ($w^T x_n + b$) as the score of the sample to a positive detection and set different thresholds to compute different precisions and recalls. The result is shown in Figure 15, and the LHOG-SCNN feature is the best performed.

From the performances of LHOG and SCNN feature in Table VII and Figure 15 we can also conclude that the

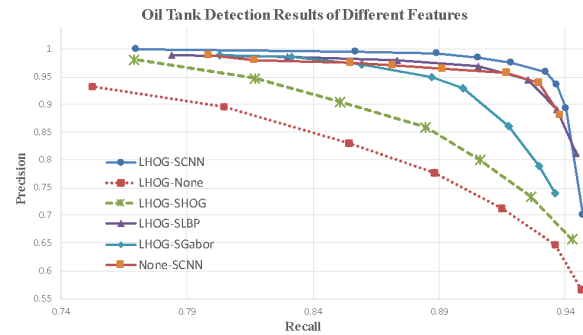


Fig. 15. Precision-recall graph of different features, the points indicate the results using different segment thresholds for SVM outputs.

SCNN feature considering the surrounding information could be the driving performance data set and the LHOG is just additive to the SCNN feature.

G. Final Detection Result

The processing time of modified ELSD plus LHOG-SCNN feature classification is compared with Ok proposed method in [7]. The detail of the test time is shown in Table VIII, the unit of the processing time is second. For the method in [7], the implementation is performed in MATLAB and on a computer with Intel i7 processor with 2.40 GHz and 16 GB RAM. The result is provided by the author. In terms of the method proposed in this paper, all our application is on an computer with Intel i7 processor with 4.0 GHz, 16 GB RAM and a GPU of GeForce GTX 760. The candidate selection process is performed with C++ code, the SCNN feature extraction is accelerated by GPU and the others are in MATLAB form. It can be seen from Table VIII, the processing time of our method is acceptable.

The result of the method in [7] is also compared with the results of different features in this part. Some pieces of the result are shown in Figure 16. In the figure, the red circles indicate the missing targets, the blue circles reflect the false positives, the green circles are the true positives. The precisions and recalls of different methods along with their processing time in the test dataset are shown in Table IX. The unit of the processing time is second. The recall after using the modified ELSD is 0.9497 and the precision is 0.2182. The implementations of HOG, LBP and Gabor are all in the MATLAB form. It can be seen that the LHOG-SCNN achieves better performance than the method in [7] and the other features. The results of LHOG-SCNN in the large size images are shown in Figure 17.

V. CONCLUSIONS AND FUTURE WORKS

A hierarchical oil tank detector for optical satellite images is presented in this paper. It is built on the applications of surrounding information and deep learning. The surrounding area which is apt to include shadows pipelines and some other oil tanks makes the oil tanks more separable from the complex background. The deep learning which has

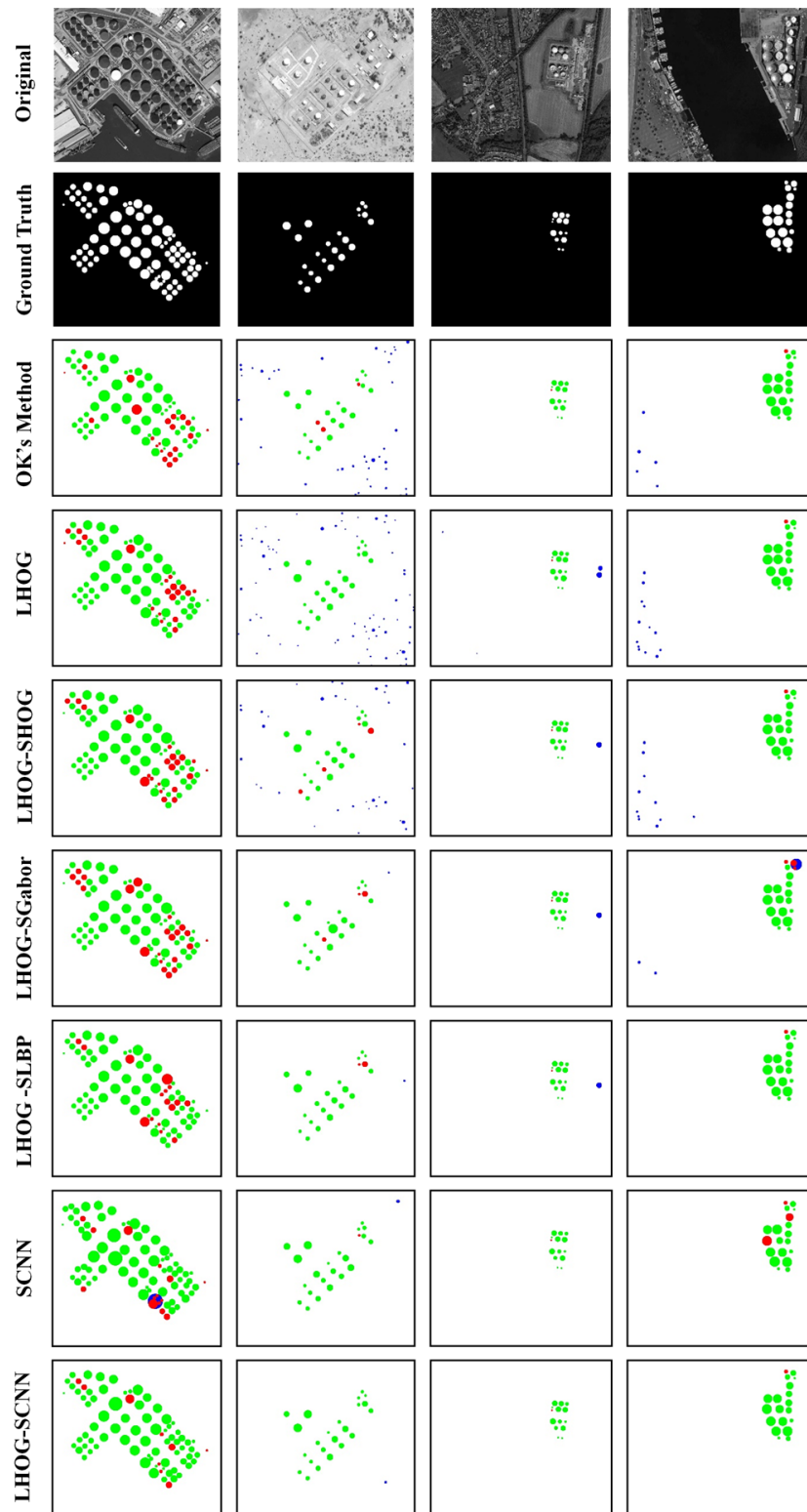


Fig. 16. Detection result of different features and OK's proposed method in [7]. The longitude of the image in the first column is -95.105483 and the latitude is 29.751176. The longitude of the image in the second column is 44.994777 and the latitude is 10.427440. The longitude of the image in the third column is -1.672366 and the latitude is 52.571756. The longitude of the image in the forth column is -118.250652 and the latitude is 33.840113.



Fig. 17. Detection results with LHO-SCNN feature in the large size images. The size of the image in the first row is 2781×2445 , the longitude is -118.238100 and the latitude is 33.801487 . The size of the image in the second row is 2158×2290 , the longitude is -95.128800 and the latitude is 29.744275 . The size of the image in the third row is 3712×3008 , the longitude is -1.672366 and the latitude is 52.571756 .

TABLE VIII
PROCESSING TIME

| ImageID | Image Size | Prop. Method | | | | | Method in [7] |
|--------------|------------|---------------------|--------------|--------------------|------|--------------|---------------|
| | | Candidate Selection | | Feature Extraction | | Total Time | |
| | | ELSD | Post Process | LHOG | SCNN | | |
| 1 | 3712×3008 | 113.31 | 10.21 | 1.97 | 1.36 | 126.84 | 118.6 |
| 2 | 2114×1858 | 31.45 | 3.90 | 2.98 | 2.05 | 40.38 | 71.1 |
| 3 | 1792×1536 | 25.33 | 1.29 | 2.90 | 5.00 | 34.50 | 23.7 |
| 4 | 1813×2000 | 35.91 | 3.73 | 2.04 | 2.36 | 44.03 | 44.4 |
| 5 | 2500×1379 | 41.85 | 2.93 | 1.23 | 2.03 | 48.04 | 56.2 |
| 6 | 2158×2290 | 57.2 | 3.90 | 1.80 | 1.95 | 64.85 | 114.3 |
| 7 | 1858×2016 | 30.97 | 3.88 | 2.02 | 2.08 | 38.95 | 55.2 |
| 8 | 1892×1618 | 28.17 | 2.42 | 1.50 | 1.81 | 33.90 | 39.3 |
| 9 | 1765×1356 | 13.88 | 2.38 | 1.05 | 1.08 | 18.39 | 27.2 |
| 10 | 2781×2445 | 66.92 | 8.58 | 3.57 | 4.70 | 83.77 | 194.5 |
| 11 | 1958×2023 | 46.54 | 2.61 | 0.99 | 0.99 | 51.12 | 34.7 |
| 12 | 2712×2652 | 50.33 | 3.31 | 1.74 | 1.75 | 57.12 | 59.1 |
| Average Time | | | | | | 53.49 | 69.86 |

TABLE IX
DETECTION RESULT OF DIFFERENT METHODS

| | Method in [7] | LHOG | LHOG-SHOG | LHOG-SGabor | LHOG-SLBP | SCNN | LHOG-SCNN |
|-------------------------|---------------|--------------|-----------|-------------|-----------|--------|---------------|
| Recall | 0.8620 | 0.8880 | 0.8845 | 0.8872 | 0.9054 | 0.8715 | 0.9184 |
| Precision | 0.9019 | 0.7762 | 0.8592 | 0.9490 | 0.9684 | 0.97 | 0.9751 |
| Average Processing Time | 69.86 | 51.23 | 62.64 | 410.85 | 77.47 | 51.51 | 53.49 |

a lot of achievements in many fields help us find a better surrounding representation than the traditional handcrafted features. The deep learning algorithm applied in this paper is an existing CNN model trained by Krizhevsky in the ILSVRC2012 contest because of the lack of labeled oil data. Additionally, a modified ELSD method is proposed to select candidates to improve the system's efficiency. Experimental results demonstrate that the proposed method is robust under different complex background and has high detection rate with low false alarms. Moreover, we have found that:

- 1) The gradient direction is more robust than the gradient amplitude for satellite images especially for the samples in low contrast with the background. This could be one of the reasons that the ELSD based method shows better performance than the traditional Hough transform manner.
- 2) The surrounding feature is the driving performance feature set and the local feature is simply additive to the surroundings. After all, the surrounding area consists of more information than the local patch. However, the local information is still necessary especially for the negative samples whose surrounding areas contain some oil tanks.

- 3) Not all surrounding features combining the local information can show better performance than the feature just focusing on the local patch. On the one hand such features remove some false positives, but on the other their detection accuracy could be even lower if they can not reliably grasp the surrounding information. This is hard to see from the statistics, but we can find the negative impact of the surrounding information in some specific cases. In the second column in Figure 16, the features (HOG, LBP and Gabor) combining surrounding information can remove some false positives but they miss some targets in the mean time. The complex surrounding information needs a higher requirement for the feature.
- 4) The idea using the surrounding information can be applied to other target detection task and could have a better performance than the manner just focusing on the local patch.

For further works, there are three aspects of the paper that need to be more deeply studied. Firstly, the method is sensitive to different resolutions. A higher resolution could bring more false positives and a lower resolution could result in more missing targets. Therefore it will require another training process with different resolution samples in such cases. Secondly, the targets near the image borders may not include enough surrounding area and may give a wrong decision result because of the poor representation of the surrounding features. For this problem, we will train the classifier with the new samples whose surrounding areas contain black borders. Thirdly, the CNN model used as a blackbox feature extractor in our algorithm is not targeted to the oil data. The property of the system can be further improved if we can fine tuning the transfer model with our own data set.

VI. ACKNOWLEDGEMENT

The authors sincerely thank the Associate Editor and the four anonymous reviewers for their very useful comments and suggestions which greatly improve the quality of this paper. The authors would like to thank A.O. Ok for sharing their process result and data set for the method in [7] [17].

REFERENCES

- [1] J. Leitloff, S. Hinz, and U. Stilla, "Vehicle detection in very high resolution satellite images of city areas," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 48, no. 7, pp. 2795–2806, 2010.
- [2] Z. Shi, X. Yu, Z. Jiang, and B. Li, "Ship detection in high-resolution optical imagery based on anomaly detector and local shape feature," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 52, no. 8, pp. 4511–4523, 2014.
- [3] X. Chen, S. Xiang, C. Liu, and C. Pan, "Aircraft detection by deep belief nets," in *Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on*, pp. 54–58, IEEE, 2013.
- [4] X. Huang, L. Zhang, and P. Li, "Classification and extraction of spatial features in urban areas using high-resolution multispectral imagery," *Geoscience and Remote Sensing Letters, IEEE*, vol. 4, no. 2, pp. 260–264, 2007.
- [5] X. Huang and L. Zhang, "An svm ensemble approach combining spectral, structural, and semantic features for the classification of high-resolution remotely sensed imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 51, no. 1, pp. 257–272, 2013.
- [6] X. Huang, Q. Lu, and L. Zhang, "A multi-index learning approach for classification of high-resolution remotely sensed images over urban areas," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 90, pp. 36–48, 2014.
- [7] A. O. Ok and E. Baseski, "Circular oil tank detection from panchromatic satellite images: A new automated approach," *Geoscience and Remote Sensing Letters, IEEE*, vol. 12, no. 6, pp. 1347–1351, 2015.
- [8] A. Chen and J. Li, "Automatic recognition method for quasi-circular oil depots in satellite remote sensing images," *Opto-Electronic Engineering*, vol. 33, no. 9, pp. 96–100, 2006.

- [9] B. Li, D. Yin, X. Yuan, and G. Li, "Oilcan recognition method based on improved hough transform," *Opto-Electronic Engineering*, vol. 35, no. 3, pp. 30–44, 2008.
- [10] X. Han, Y. Fu, and G. Li, "Oil depots recognition based on improved hough transform and graph search," *Journal of Electronics & Information Technology*, vol. 33, no. 1, pp. 66–72, 2011.
- [11] X. Han and Y. Fu, "Circular array targets detection from remote sensing images based on saliency detection," *Optical Engineering*, vol. 51, no. 2, pp. 026201–1, 2012.
- [12] Y. Yao, Z. Jiang, and H. Zhang, "Oil tank detection based on salient region and geometric features," in *SPIE/COS Photonics Asia*, pp. 92731G–92731G, International Society for Optics and Photonics, 2014.
- [13] X. Cai, H. Sui, R. Lv, and Z. Song, "Automatic circular oil tank detection in high-resolution optical image based on visual saliency and hough transform," in *Electronics, Computer and Applications, 2014 IEEE Workshop on*, pp. 408–411, IEEE, 2014.
- [14] C. Zhu, B. Liu, Y. Zhou, Q. Yu, X. Liu, and W. Yu, "Framework design and implementation for oil tank detection in optical satellite imagery," in *Geoscience and Remote Sensing Symposium (IGARSS), 2012 IEEE International*, pp. 6016–6019, IEEE, 2012.
- [15] N. K. Kushwaha, D. Chaudhuri, and M. P. Singh, "Automatic bright circular type oil tank detection using remote sensing images," *Defence Science Journal*, vol. 63, no. 3, pp. 298–304, 2013.
- [16] A. O. Ok, "A new approach for the extraction of aboveground circular structures from near-nadir vhr satellite imagery," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 52, no. 6, pp. 3125–3140, 2014.
- [17] A. O. Ok and E. Baseski, "Automated detection of oil depots from high resolution images: a new perspective," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 1, pp. 149–156, 2015.
- [18] T. Moranduzzo and F. Melgani, "Detecting cars in uav images with a catalog-based approach," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 52, no. 10, pp. 6356–6367, 2014.
- [19] X. Jin and C. H. Davis, "Vector-guided vehicle detection from high-resolution satellite imagery," in *Geoscience and Remote Sensing Symposium, 2004. IGARSS'04. Proceedings. 2004 IEEE International*, vol. 2, pp. 1095–1098, IEEE, 2004.
- [20] Z. An, Z. Shi, X. Teng, X. Yu, and W. Tang, "An automated airplane detection system for large panchromatic image with high spatial resolution," *Optik-International Journal for Light and Electron Optics*, vol. 125, no. 12, pp. 2768–2775, 2014.
- [21] Z. Li and L. Itti, "Saliency and gist features for target detection in satellite images," *Image Processing, IEEE Transactions on*, vol. 20, no. 7, pp. 2017–2029, 2011.
- [22] W. Wu, J. Luo, C. Qiao, and Z. Shen, "Ship recognition from high resolution remote sensing imagery aided by spatial relationship," in *Spatial Data Mining and Geographical Knowledge Services (ICSDM), 2011 IEEE International Conference on*, pp. 567–569, IEEE, 2011.
- [23] C. Corbane, F. Marre, and M. Petit, "Using spot-5 hrg data in panchromatic mode for operational detection of small ships in tropical area," *Sensors*, vol. 8, no. 5, pp. 2959–2973, 2008.
- [24] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.
- [25] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 580–587, IEEE, 2014.
- [26] R. O. Duda and P. E. Hart, "Use of the hough transformation to detect lines and curves in pictures," *Communications of the ACM*, vol. 15, no. 1, pp. 11–15, 1972.
- [27] N. Kiryati, Y. Eldar, and A. M. Bruckstein, "A probabilistic hough transform," *Pattern recognition*, vol. 24, no. 4, pp. 303–316, 1991.
- [28] C. F. Olson, "Constrained hough transforms for curve detection," *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 329–345, 1999.
- [29] C. Hollitt, "A convolution approach to the circle hough transform for arbitrary radius," *Machine vision and applications*, vol. 24, no. 4, pp. 683–694, 2013.
- [30] V. Pătrăucean, P. Gurdjos, and R. G. Von Gioi, "A parameterless line segment and elliptical arc detector with enhanced ellipse fitting," in *Computer Vision–ECCV 2012*, pp. 572–585, Springer, 2012.
- [31] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886–893, IEEE, 2005.
- [32] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.

- [33] J. G. Daugman, "Complete discrete 2-d gabor transforms by neural networks for image analysis and compression," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 36, no. 7, pp. 1169–1179, 1988.
- [34] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
- [35] G. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
- [36] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.
- [37] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 647–655, 2014.
- [38] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.
- [39] V. Vapnik, S. E. Golowich, and A. Smola, "Support vector method for function approximation, regression estimation, and signal processing," *Advances in neural information processing systems*, pp. 281–287, 1997.
- [40] Y. Tang, "Deep learning using linear support vector machines," *ICML 2013 Workshop on Representation Learning*, 2013.
- [41] A. Desolneux, L. Moisan, and J. M. Morel, *From Gestalt theory to image analysis: a probabilistic approach*, vol. 34. Springer Science & Business Media, 2007.
- [42] R. von Gioi, J. Jakubowicz, J. Morel, and G. Randall, "Lsd: A fast line segment detector with a false detection control," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 4, pp. 722–732, 2010.
- [43] D. Powers, "Evaluation: From precision, recall and f-factor to roc, informedness, markedness & correlation (tech. rep.)," *Adelaide, Australia*, 2007.