Real-time Detection and Recognition of Landmarks for Mobile Robot in Indoor Scenes

Yanbing Xue[1, 2], Hua Zhang[1, 2], Yucui Ju[1, 2] and Jin Wang[3]

[1] Key Laboratory of Computer Vision and System, Tianjin University of Technology, Tianjin 300191, China

2 Tianjin Key Laboratory of Intelligence Computing and Novel Software Technology, Tianjin University of Technology, Tianjin 300191, China

3 Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science &Technology, Nanjing, 210044, China

Abstract

This paper proposes a robust real-time artificial landmarks detection and recognition system for indoor mobile robot. Landmarks detection and recognition for indoor robots faces two major difficulties, one is the illumination changes and the other is processing speed. In this paper, first, histograms of oriented gradient (HOG) features are extracted to resolve the problem of illumination changes. Second, AdaBoost based algorithm is used in detection phase to increase the processing speed. Finally, RBF-SVM classifier is used for recognition. Experimental results show a high detection and recognition accuracy and the processing speed is about 10 frames per second.

Keywords: *mobile robot; landmarks detection and recognition; AdaBoost; HOG features; SVM classifier*

1. Introduction

Indoor mobile robots are gradually entering our lives as services and office assistants. Landmark detection and recognition is an important task in mobile robot environment perception [1] which is also the key step to ensure the robot can be positioned correctly. Landmark detection and recognition system usually includes two stages: detection and recognition [2].

The proposed system searches the regions of interest of the image in the detection stage. Due to the color and shape are the main characteristics of the landmark, most of the algorithms use color threshold segmentation and Hough transform [3-6] to detect the landmarks. However, the method of using color threshold segmentation will be influenced by the changes of illumination and the method based on the shape analysis (Hough transform) will bring an error positioning due to the similarity of the landmarks. The recognition stage, the system evaluates regions found in the detection stage and identifies the landmarks. Template matching is widely used in landmarks recognition for its simplicity [7-9], which on other hand has the disadvantage of low accuracy. These algorithms can obtain better results in a certain extent, but most of them have the disadvantages of error rate is high and the detection speed is slow.

This paper presents a fusion of AdaBoost [10] and SVM [11-12] algorithm for landmark recognition. First of all, filtering out the most likely the candidate collection of images through AdaBoost, and then, using RBF-SVM to recognize on the collection of candidates. The proposed method greatly reduces the number of sub-images that the RBF-SVM need to recognition, and ensures the accuracy of recognition to some extent. In order to demonstrate the performance of the proposed system, we make another experiment using liner SVM instead of AdaBoost to detect and using RBF-SVM to recognize.

Figure 1 shows the overall proposed system architecture for landmark detection and recognition. The output from the detection system will be the input to recognition system. The robot uses the detection system to detect its interested landmark in the current environment and determine the position, the size and the posture of the landmark. Then the precise positioning of the landmark is sent to the recognition system to identify and to determine the meaning of the signs. Both of the detection system and the recognition system use HOG features. We import the positive landmark sample patches and negative non-landmark patches to HOG feature extractor to form the feature vectors for each patch.

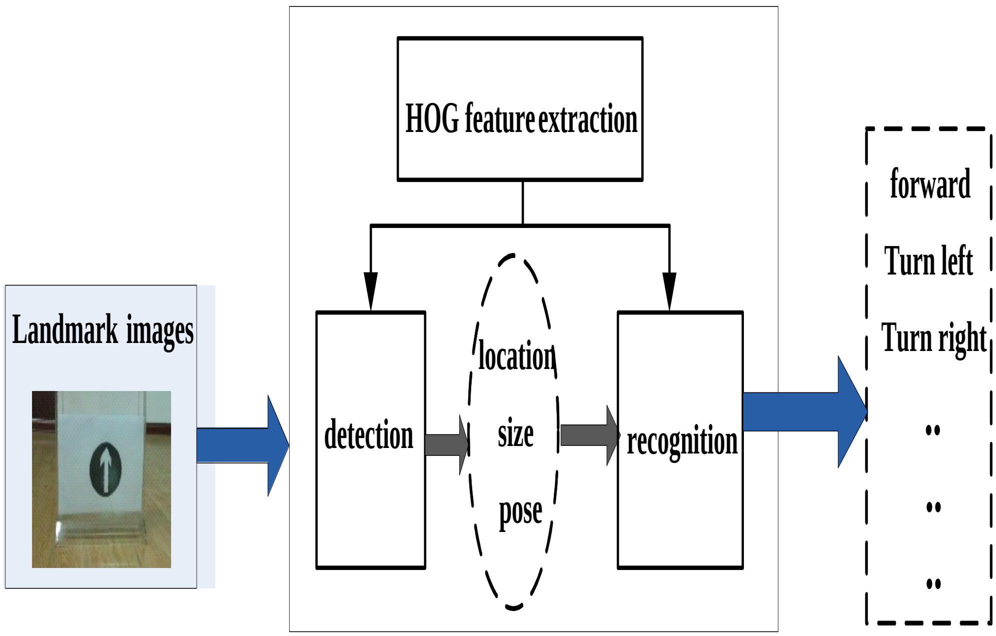

Figure 1. The Diagram of Landmark Detection and Recognition System

The paper is organized as follows: Section 2 focuses on the HOG feature extraction. Section 3 describes AdaBoost learning algorithm based on the HOG features for landmarks detection. Section 4 shows the landmark recognition based on SVM. Section 5 presents the experimental results of proposed method. Section 6 gives our conclusions and future work.

2. Feature extraction

HOG feature is a feature descriptor which is used for object detection proposed by Dalal [13] in 2005. The advantage of HOG feature is that it is based on the distribution histogram of oriented gradient. Therefore, it can describe contour feature of landmarks and is not sensitive to illumination and a small amount of offset at the same time.

In this paper, in order to make the dimension of feature vector suitable for machine learning, all the landmarks are gray images and normalized to the size of 32 by32 pixels. Gradient is calculated using a simple center symmetric operator [-1, 0, 1], the size of the image block is 16 by16 pixels and the initial orientation angle is unsigned 0 ° to 180 °which is divided into nine angles. We use the L1-norm to normalize the gradient of the block.

The extraction steps of landmark HOG feature are as follows:

1). Gradient Computation. The horizontal G_x and vertical gradient G_y obtained by convolving the gradient operator [-1, 0, 1] and [1, 0,-1] with the landmark image.

$$G_x(x,y) = I(x+1,y) - I(x-1,y)$$
$$G_y(x,y) = I(x,y+1) - I(x,y-1) \quad (1)$$

The gradient direction and gradient magnitude of each pixel is calculated as equation 2.

$$m(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2}$$
$$\theta(x,y) = \arctan(G_y(x,y) / G_x(x,y)) \quad (2)$$

2). The division of sub-blocks and histogram acquisition. As shown in Figure 2(a), the landmark is divided into nine image blocks (BLOCK), each block is divided into four units (CELL) and the size of each cell is 8 by 8 pixels. The gradient direction of each pixel is 0 ° ~ 180 °and if the gradient magnitude of a pixel is greater than 180 °, its final direction value is $\theta - 180^\circ$. Then the final gradient orientation is divided into nine bins and in order to get the corresponding 9-dimensional feature vector of each cell, we adopt the method of weighted projection in the histogram using gradient orientation for each pixel within the cell (see Figure 2).

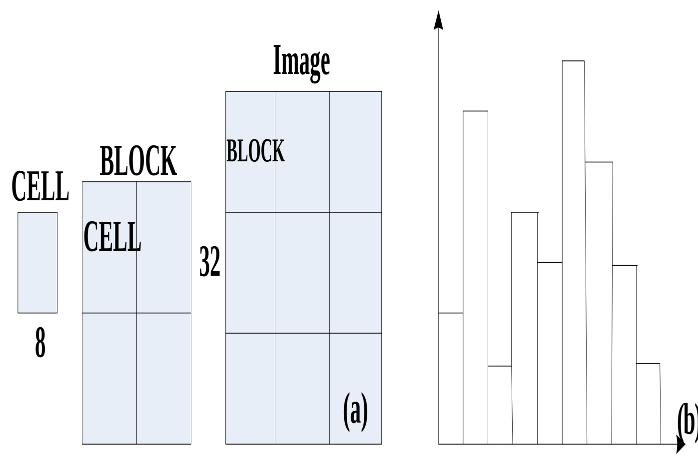

Figure 2. Division of the Image and the Histogram of Oriented Gradient

3). Normalization. In order to make the feature vector space having robustness to local illumination changes, shadows, and the edge changes, we will need to normalize the gradient intensity. In this paper, L1-norm is used for normalizing as followed:

$$v \leftarrow v / (\|v\| + e) \quad (3)$$

4). Generation of HOG feature vectors. Hog Feature extraction process as shown in Figure 3, high dimensional vectors of 9× 9 × 4=324 data are generated by the above steps. The first “9” means that there are nine bins in each cell, the second “9” means that there are nine blocks ,"4" means that there are four cells in each block and then the whole HOG feature vectors are generated.

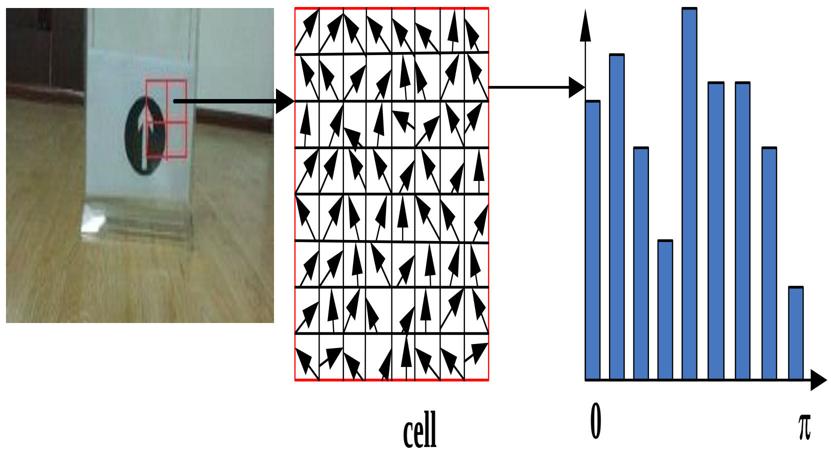

Figure 3. The Extraction Process of HOG Features

3. Landmark Detection based on AdaBoost

3.1. Using Integral Image to Calculate the HOG Feature for High Speed

As we know that different positions and sizes of feature will produce different features. For a 32× 32 window will have thousands of features, and then the calculation of feature values will spend a lot of time. So in order to improve calculation speed of feature values, we use the method of integral image [14] to calculate. The integral image simplifies the computational complexity and improves the detection efficiency.

The integral image at location x, y contains the sum of the pixels above and to the left of x, y. Then the value of the integral image at location (x, y) is:

$$ii(x,y) = \sum_{x' \le x} \sum_{y' \le y} i(x', y') \tag{4}$$

Where $i(x', y')$ is the pixel value at location (x', y') in the image, ii(x, y) is the integral image at location (x, y).

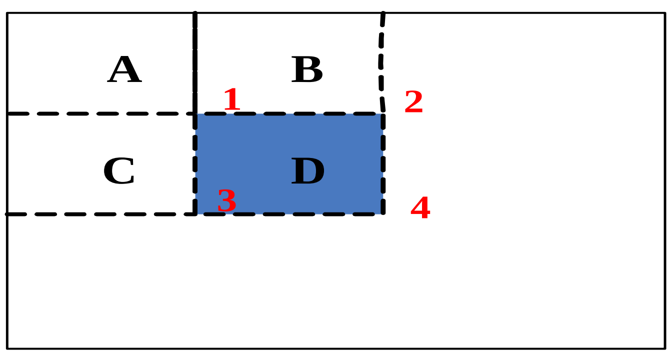

Figure 4. Integral Image

As shown in Figure 4, the value of the integral image at location 1 is the sum of the pixels in rectangle A. The value at location 2 is A+B, at location 3 is A+C, and at location 4 is A+B+C+D. The sum within D can be computed as 4+1-(2+3).

Inspired by HAAR feature, we exploit a fast way of calculating the HOG feature. First, we discretize each pixel's orientation (including its magnitude) into 9 histogram bins. We compute and store an integral image for each bin of the HOG and use them to compute efficiently the HOG for any rectangular image region. This requires 4 × 9 image access operations.

3.2. Constructing Weak Classifier of HOG Adapt to AdaBoost

The weak classifier is the smallest part in the landmark detection. Combination of weak classifiers which are selected according to certain rules, you can construct different strong classifiers and cascade classifier. The structure of weak classifiers are simpler, the calculation times are less so that the detection speed can be improved. Therefore, the construction of weak classifiers is the focus of the AdaBoost algorithm. As we know that not all the features can adapt to the AdaBoost algorithm, so the following describes the structure of weak classifier of HOG-AdaBoost proposed by this paper.

HOG feature is a 36D feature vector and therefore we cannot consider a HOG feature as a weak classifier. The HOG feature of each cell contains important information on how to separate landmarks from other objects, therefore, the set of weak classifiers are created for each cell in this paper. As the HOG is a histogram with bins indicating local gradient distribution, we set a threshold T and then compare the value of one bin with T, if the value is larger than T, we consider it as landmarks otherwise not. The histogram has nine bins in this paper, and then we have nine weak classifiers corresponding to each bin. The weak classifier is defined as follows:

$$h_{k,i}(x) = \begin{cases} 1 & if \;\; p_i Hist_{k,i}(x) < p_i \theta_i \\ 0 & otherwise \end{cases} \tag{5}$$

In the above equation, x indicates the input image (the set of training samples). For the cell k, the feature $p_i Hist_{k,i}(x)$ presents the value on ith bin of the histogram in that cell. θ_i is the decision threshold corresponds to the ith feature. $p_i = \pm 1$ determines the direction of the inequality.

3.3. Strong Classifier

Since there are many kinds of HOG features with different locations and sizes which will bring a huge amount of calculation if we calculate all of the features in the detection process. It is very important on how to select a small number of the best features of weak classifier to form the final classifier which can not only reduce the amount of calculation of features, but also can achieve accurate classification. Training AdaBoost strong classifier is to achieve this purpose. The training process of AdaBoost classifier as shown in Table 1:

Table 1. The Training Process of the Strong Classifier

Step1: Given weak learning algorithm and training set $\{(x_1,y_1),(x_2,y_2),...,(x_n,y_n)\}$, where x_i indicates the feature vector of samples, and y_i signifies the category of the sample, 0 for negative samples, 1for positive samples.

Step2：Given initialize weights $w_{1,i}=\frac{1}{2m},\frac{1}{2l}$ for y=0, 1 respectively, where m and l are the number of negatives and positives respectively and m+l=n.

Step3：For t=1, 2, …T (T is the number of iterations)

1. Normalize the weights of samples:

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$$

2. For each feature j, train a classifier h_j. For each weak classifier, calculate the error rate under the current:

$$\varepsilon_j = \sum_i w_i \left| h_j(x_i) - y_i \right|$$

3. Choose the classifier which has the lowest error

$$\varepsilon_t = \min \sum_i w_i \left| h_j(x_i) - y_i \right|$$

4. Update the weight according to the best classifier:

$$w_{t+1,i} = w_{t,i}\beta_t^{1-e_i}$$

$$\beta_t = \frac{\varepsilon_t}{1-\varepsilon_t}$$

Where e_i =0 if sample x_i is classified correctly, 1 otherwise. This will increase the weight of the samples which are classified error and reduce the weight of the samples which are classified correct, therefore, the samples are classified error will be valued on the weak classifier selection during the next iteration.

Step4：The final strong classifier is:

$$h(x)=\begin{cases} 1 & \sum_{t=1}^{T}\alpha_t h_t(x) \ge \frac{1}{2}\sum_{t=1}^{T}\alpha_t \\ 0 & otherwise \end{cases} \quad 其中\alpha_t = \log\frac{1}{\beta_t}$$

4. The Landmark Recognition based on RBFSVM

SVM is a machine learning algorithm which can classify data into several groups. It is based on the concept of decision planes where the training data is mapping to a higher dimensional space and separated by a plane defining the two or more classes of data. The formulation of SVM deals with structural risk minimization (SRM). SRM minimizes an upper bound on the Vapnik Chervonenkis dimension, and it clearly differs from empirical risk minimization, which minimizes the error on the training data.

Landmark recognition is implemented by SVM with RBF kernels. For the training process of SVM, we used the library LIBSVMS [15]. In many cases, the data cannot be separated by a linear function. A solution is to map the input data into a different space $\Phi(x)$. Due to the fact that the training data are used through a dot product, if there was a "kernel function," so that we satisfy $K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle$, we can avoid computing $\Phi(x)$ explicitly and use the kernel function $K(x_i, x_j)$.

In this paper, we have used a RBF kernel as follows:

$$K(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}} \tag{6}$$

and the decision function for a new input vector is

$$f(x) = \mathrm{sgn}(\sum_{i=1}^{N_s} \alpha_i y_i K(s_i, x) + b) \tag{7}$$

Where N_s is the number of support vectors, and S_i are the support vectors.

The basic processes of landmark recognition are shown in Figure 5 and from it we can see that the landmark recognition system mainly consists of two parts: feature extraction and SVM classifier training. HOG features are extracted and it has been introduced above, then we will introduce the proposed establishment method of SVM classifier.

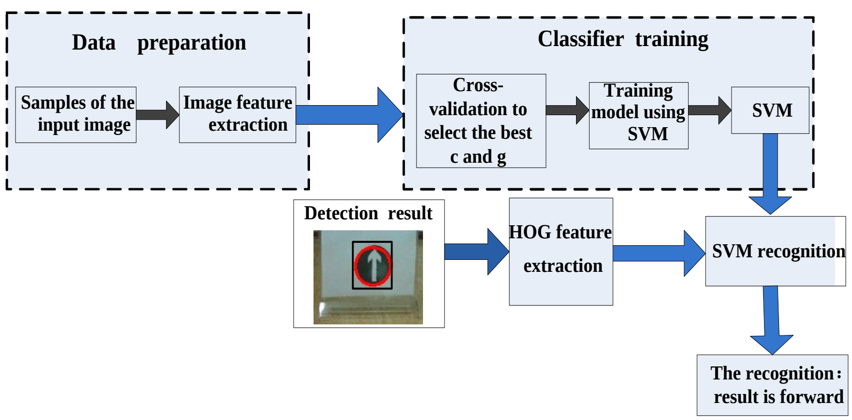

Figure 5. Landmark Recognition Process

4.1. Establishment of SVM Classifier

In this paper, the one-against-one classification method is used for multi-class support vector machine classifier of a landmark recognition system. First, scaling the initial feature vector to the range [0, 1]. Second, training the best classification parameters. The paper uses the RBF kernel of SVM to build classifier. There are two parameters while using RBF kernels: C and r, different C and r will produce different accuracy and the goal is to identify good (C, r) so that the classifier can accurately predict unknown data. The original methods usually use the default values or manually to find the best C and r which is time-consuming and not accurate enough. The paper uses "web searching" to find the optimal C and r. 75% of the total samples are used as the training samples for cross validation which are divided into 6 groups. Each group takes turns as test samples, the rest as the training samples. Figure 6 shows the parameter values obtained from the optimization and the classification accuracy of training set. The best parameters can be seen from the figure: C=8, Gamma=0.0078125, the correct rate is 100%. After getting the best parameters, we can obtain the optimal classifier model by training on the training samples. Finally, the optimum classification model is used to recognize the remaining 25% test samples.

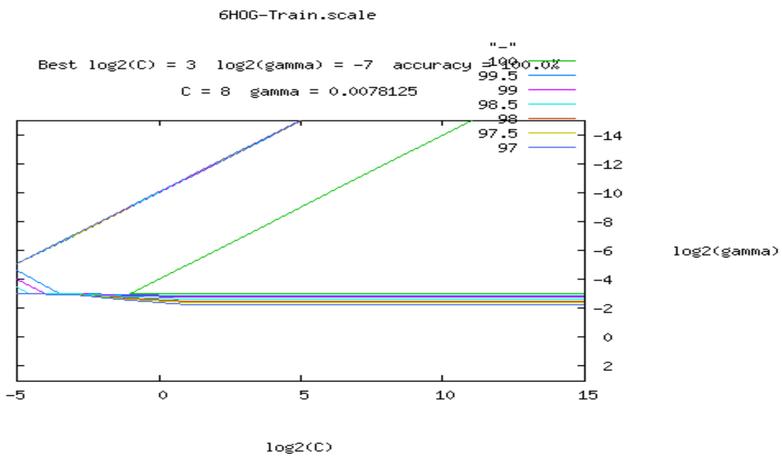

Figure 6. The Optimization of Landmark Classifier

5. Experimental Results and Analysis

In this paper, mobile robot in the indoor scene is used to validate our algorithm (Figure 7).

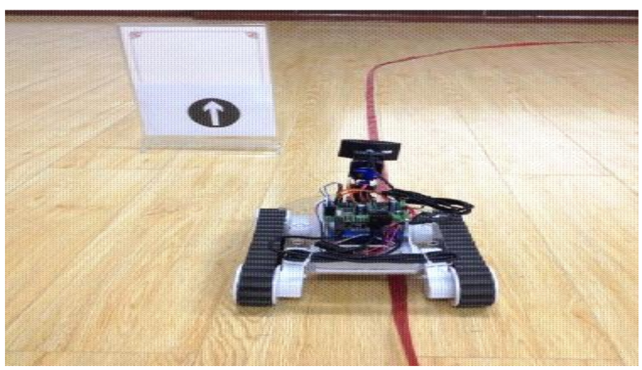

Figure 7. The Mobile Robot

In the detection phase, the landmark samples are acquired by a camera mounted on a vehicle at the height of 0.50 m (above the ground). We extract the HOG features of the prepared samples and use the AdaBoost and SVM algorithm to train respectively. When the AdaBoost and liner SVM classifier are obtained, we compare the performance on the testing samples.

In the recognition phrase, the RBFSVM, which are obtained in the training process, are used to provide the most effective representation of the landmarks. We show the recognition results of different landmarks which are detected by the AdaBoost and liner SVM classifier.

Finally, we test the performance of our overall system.

5.1. Experimental Results of Landmark Detection

Samples of this paper are divided into positive samples and negative samples.1) positive samples: We make five kinds of landmarks including forward, turn left, turn right, rotate and stop. For every kind of landmarks, we use the mobile robot to take several shots respectively from different directions and different distances which is shown in Figure 7.

As shown in Figure 8(a), we performed the cut operation on preserved landmark pictures. In this paper, we first take four points roughly, and then make the bounding rectangle of a circle landmark through the fine adjustment. Finally, the pictures which have been taken out of images are normalized to the size of 32 by 32 pixels and the final positive samples are made successfully.2) negative samples: Negative samples consist of two parts, one part is the arbitrary regions excluding landmarks which are cropped from the captured photos, and another part is some background images which are downloaded from the Internet. We try to insure that the negative samples have diversity and negative samples are normalized to 64 by 128 pixels (32 by 32 pixels for liner SVM).

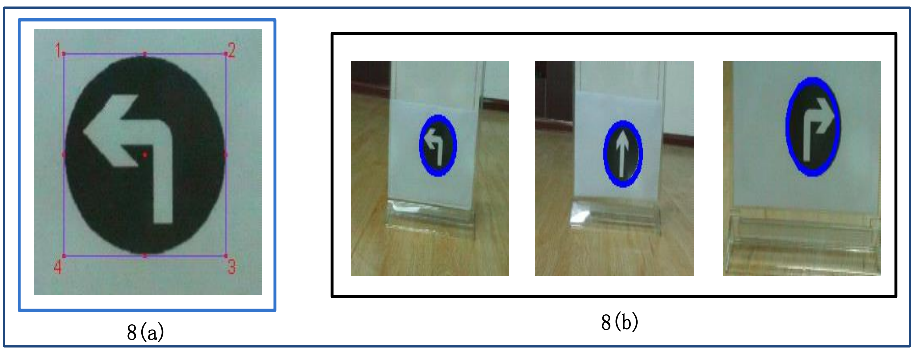

Figure 8. The Cropping Ways of Positive Samples and the Part of Detection Results

Figure 8(b) shows the part of the detection results. In the training stage, we use 1762 positive samples and 5000 negative samples, the results of the landmark detection are shown in Table 2.

Table 2. The Results of Landmark Detection

type	Liner SVM + HOG					AdaBoost + HOG				
	Test samples	True positives		False positives		Test samples	True positives		False positives	
		#	%	#	%		#	%	#	%
forward	123	119	96.7	2	1.6	123	122	99.2	1	0.8
left	125	121	96.8	3	3.2	125	123	98.4	3	2.4
right	133	131	98.4	4	3.0	133	132	99.2	3	2.3
rotation	128	123	96.1	5	3.9	128	124	96.9	6	4.8
stop	138	134	97.1	3	2.2	138	137	99.3	3	2.2

As can be seen from the Table 2, the AdaBoost classifier has higher true positives than the liner SVM classifier and they have nearly the same number of false positives. The experiment shows that the proposed detection method has a good performance.

5.2. Experimental Results based on SVM Landmark Recognition

We divide the samples into six groups in the experiment. As shown in Figure 9, the five groups of samples are normalized to the size of 32 by 32 pixels and the HOG features are extracted from the five groups of samples which are used for training SVM.

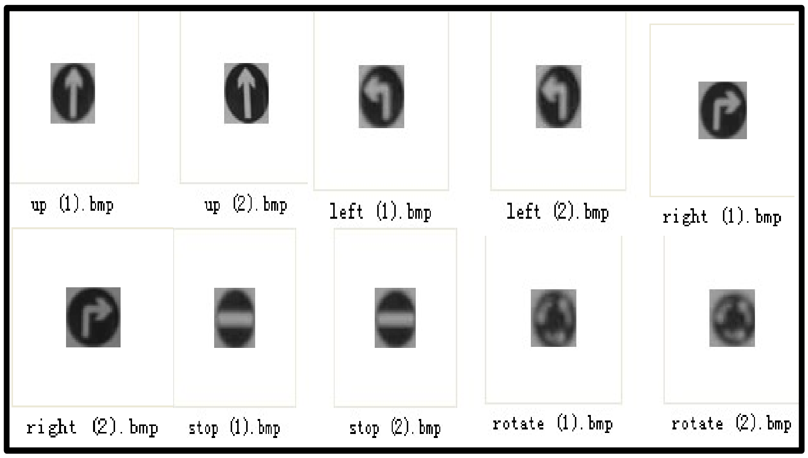

Figure 9.The Training Samples of SVM

As shown in Figure 10, the test experiments are divided into five groups which are as followed from left to right: the first group is “forward”, the second group is “turn left”, the third group is “turn right”, the fourth is “rotation”, and the fifth group is “stop”.

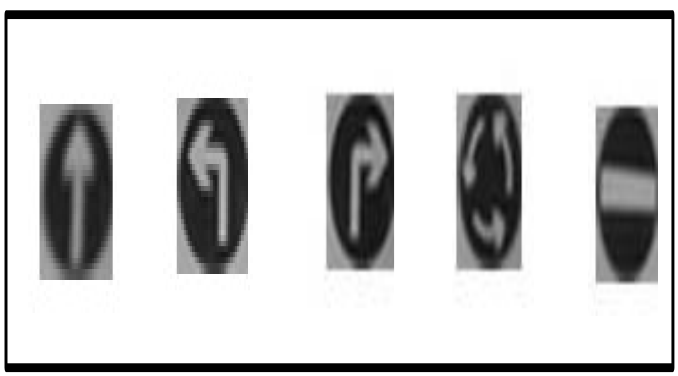

Figure 10.The Test Samples of SVM

Signs used in our experiment are taken by mobile robot in indoor environment (see Figure 7). We received a total number of 1794 samples, in which the 1446 samples are used for training samples and the rest of 648 samples are used for test samples.

Recognition results of each group are shown in Table 3, the “false recognition” is that the landmark is detected in the detection phrase but not recognize in the recognition phrase.

Table 3. The Recognition Results of the Five Kinds of Signs

type	Liner SVM + RBF- SVM					AdaBoost + RBF -SVM				
	Test samples	Correct recognition		False recognition		Test samples	Correct recognition		False recognition	
		#	%	#	%		#	%	#	%
forward	123	117	95.1	6	4.9	123	118	95.9	5	4.1
left	125	120	96.0	5	4.0	125	122	97.6	3	2.4
right	133	130	97.7	3	2.3	133	130	97.7	3	2.3
rotation	124	120	96.8	4	3.2	128	126	98.4	2	1.6
stop	133	128	96.2	5	3.8	138	134	97.1	4	2.8

The experimental results show that the recognition results of AdaBoost and RBF-SVM are better than liner SVM and RBF-SVM. We find that the liner SVM cannot location correctly sometimes, so it has slower correct recognitions than AdaBoost.

5.3. The Performance of our Overall System

The performance of overall system is shown in Table 4 and 5.The “false recognition” in the overall system is the sum of the number of the landmarks which are not detected in the detection phrase and the number of the landmarks which are detected in the detection phrase but not recognize in the recognition phrase.

Table 4. The Performance of Overall System

stype	Liner SVM + RBF -SVM					AdaBoost + RBF- SVM				
	Test samples	Correct recognition		False recognition		Test samples	Correct recognition		False recognition	
		#	%	#	%		#	%	#	%
forward	123	113	91.9	10	8.1	123	117	95.1	6	4.9
left	125	116	92.8	9	7.2	125	120	96.0	5	4.0
right	133	128	96.2	5	3.8	133	129	97.0	4	3.0
rotation	124	115	92.7	9	6.3	128	122	95.3	6	4.7
stop	133	124	93.2	9	6.8	138	133	96.4	5	3.6

Table 4 shows that the proposed system has higher correct recognitions and lower false recognitions than the other one. Table 5 shows that the average recognition time of the AdaBoost and RBF-SVM is 99.84ms while that of the liner SVM and RBF-SVM is

201.39ms. Experimental results show that the overall proposed system not only has high correct recognition, but also achieve real-time.

Table 5. The Recognition Time of the Overall System

The average recognition time (frame/ms)	Liner SVM + RBF- SVM	AdaBoost + RBF- SVM
	201.39	99.84

6. Conclusions

The paper presents a real-time artificial landmark detection and recognition system for mobile robot. The detection method is based on HOG feature and AdaBoost learning algorithm. In the recognition phrase, we use HOG feature combined with RBF-SVM classifier. An experimental comparative study on the two algorithm verified that the proposed algorithm is not only able to meet real-time requirements, but also to obtain a very high recognition rate by applying the whole system to the mobile robot.

Acknowledgements

This research has been supported by funding from National Natural Science Foundation of China (61001174, 61202168, and 61201234), Key project in Science and technology pillar program of Tianjin (10ZCKFGX00400), Natural Science Foundation of Tianjin (12JCYBJC10200), and Tianjin Education Committee science and technology development Foundation (No. 20120802). It was also supported by the Industrial Strategic Technology Development Program (10041740) funded by the Ministry of Knowledge Economy (MKE) Korea, and by the Natural Science Foundation of Jiangsu Province (No. BK2012461).

References

[1] S. H. Piao, W. Zhao, Q. B. Zhong and Y. Q. Liu, “Research of Vision-Based Global Localization for Mobile Robot”, Journal of Computational Information Systems, vol. 7,**(2011)**, pp. 144-151.
[2] Y. Nguwi and A. Kouzani, “A study on automatic recognition of road signs”, Cybernetics and Intelligent Systems”, **(2006)**, pp. 1-6.
[3] M. A. Garcia-Garrido, M. A. Sotelo and E. Martm-Gorostiza, “Fast traffic sign detection and recognition under changing lighting conditions”, Intelligent Transportation Systems Conference, **(2006)**, pp. 811-816.
[4] Y. B. Damavandi and K. Mohammadi, “Speed limit traffic sign detection and recognition”, Cybernetics and Intelligent Systems, vol. 2, **(2004)**, pp. 811-816.
[5] H. Huang, C. Chen, Y. Jia and S. Tang, “Automatic Detection and Recognition of Circular Road Sign”, Mechtronic and Embedded Systems and Application, **(2008)**, pp. 626-630.
[6] A. De La Escalera, L. E. Moreno, M. A. Salichs and J. M. Armingol, “Road traffic sign detection and classification”, IEEE Transactions on Industrial Electronics, vol. 44,**(1997)**, pp. 848-859.
[7] Y. B Lauziere, D. Gingras and F. P. Ferrie, “A model-based road sign identification system”, Computer Vision and Pictograph Recognition, vol. 1, **(2001)**, pp. 1163-1170.
[8] R. Malik, J. Khurshid and S. Ahmad, “Road sign detection and recognition using color segmentation, shape analysis and template matching”, International Conference on Machine Learning and Cybernetics, **(2007)**, vol. 6, pp. 3556-3560.
[9] V. Andrey and K. H. Jo, “Automatic detection and recognition of traffic signs using geometric structure analysis”, SICE-ICASE International Joint Conference, **(2006)**, pp. 1451-1456.
[10] Z. Yang, X. Hua and Y. Shao, “Empirical Analysis of AdaBoost Twin Bounded Support Vector Machine”, Journal of Information and computational Science, vol. 9,**(2012)**, pp. 5085-5092.
[11] C. Cortes and V. Vapnik, “Support-vector networks”, Machine Learning, vol. 20, **(2005)**, pp. 273-297.
[12] V. Vapnik, Editor, “The Nature of Statistical Learning Theory”, Springer-Verlag, NewYork, **(1995)**.

[13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", Computer Visio and Pattern Recognition, vol. 1, **(2005)**, pp. 886-893.
[14] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", Computer Visio and Pattern Recognition, vol. 1, **(2001)**, pp. 511-518.
[15] R.-E. Fan, P.-H. Chen and C.-J. Lin, "Working set selection using second order information for training SVM", Journal of Machine Learning Research, vol. 6, **(2005)**, pp. 1889-1918.

Authors

Yanbing Xue received the B.S. degree from Shandong Normal University ,China, in 2002, the M.S. degree from Tianjin University of Technology, China, in 2005. He is an assistant researcher in the School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin, China. His research interest includes computer vision and human - computer interaction.

Hsua Zhang received the B.S. degree, the M.S. degree and doctor degree from Tianjin University in 1983, 1988 and 2008 respectively. She is a professor in the School of Computer and Communication Engineering, Tianjin University of Technology, Tianjin, China. Her research interests include multimedia analysis and virtual reality.

Yucui Ju received the B.S. degree in Electronic and Information Engineering from Shandong University of Technology, China, in 2011. She is pursuing her master degree in the school of Computer and Communication Engineering, Tianjin University of Technology. Her research interest includes computer vision and Pattern Recognition.

Jin Wang received the B.S. and M.S. degree from Nanjing University of Posts and Telecommunications, China in 2002 and 2005, and Ph.D. degree from Kyung Hee University Korea in 2010. During 2010-2011, he was a post-doctor in Kyung Hee University Korea. Now, he is a professor in School of Computer and Software, Nanjing University of Information Science and Technology. His research interests mainly include routing algorithm/protocol design, performance evaluation and optimization for wireless ad hoc and sensor networks. He is a member of the IEEE and ACM.