

## Objects Classification for Mobile Robots Using Hierarchic Selective Search Method

Cheng Xu<sup>1</sup>, Ting Cao<sup>1</sup>, Zhiqi Li<sup>2</sup> and Xiongren Xiao<sup>1</sup>

<sup>1</sup>*Information Science and Engineering, Hunan University, 410082 Changsha, China*

<sup>2</sup>*Civil Engineering, Hunan University, 410082 Changsha, China*

**Abstract.** Aiming at determining the category of an image captured from mobile robots for intelligent application, classification with the bag-of-words model is proved effectively in near-duplicate/planar images. When it comes to images from mobile robots with complex background, does it still work well? In this paper, based on the merging criterion improvement, a method named hierarchical selective search is proposed hierarchically extracting complementary features to form a combined and environment-adaptable similarity measurement for segmentation resulting a small and high-quality regions set. Simultaneously those regions rather than a whole image are used for classification. As a result, it well improved the classification accuracy and make the bog-of-word model still work well on classification for mobile robots. The experiments on hierarchical selective search show its better performance than selective search on two task datasets for mobile robots. The experiments on classification shows the samples from regions are better than those original whole images. The advantage of less quantity and higher quality object regions from hierarchical selective search is more prominent when it comes to those special tasks for mobile robots with scarce data.

### 1 Introduction

With the rapid development of mobile internet, mobile robots has played increasingly important roles in our daily life. Object classification as important parts for the mobile robots brings us much convenience and high efficiency, such as the traffic monitoring, agricultural products or fruit picking, mobile E-Commerce and so on. Regard a smart phone as a mobile robot according to the definition of robot in [1]. You can imagine, when you're walking on the street, your sight is deeply attracted by a cloth, you just need pick up your smart phone and click the screen, then the similar cloth jump into your eyes.

General methods on objects classification are based on the Bag of Words model (BOW) [2-3] to construct a visual vocabulary. The BOW is a popular form of object representation well applied in the near-duplicate/planar images. However the images captured from mobile robots are easily influenced by the light, viewpoint, complex background and other factors, the same object from different images in various viewpoint or light condition often change great, so how does BOW work well for object classification for mobile robots? In most cases, the construction of a good classifier relies on many samples. When it comes to some specific tasks for mobile robots in which capturing samples becomes hard and the labeling becomes time-consuming, how can an efficient classifier be constructed using those small samples?

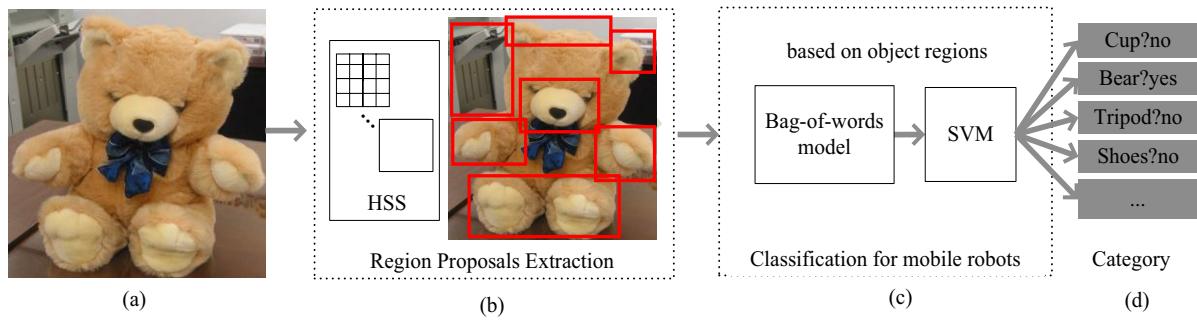
Among mobile robot environment, the key to improve

the performance of object classification is accurately and efficiently to obtain the object in the image. Under the premise of reducing the computational cost, the accuracy and real-time performance of classification should be guaranteed. Since the object can be anywhere in an image, the computation complexity of a traditional exhaustive search method in the whole image is heavily dependent on the size of the image.

Take these into account, in this paper, a method based on Hierarchic Selective Search (HSS) comes up aiming at achieving better classification for the mobile robots. The method uses the small and high-quality object regions set segmented by HSS instead of the whole image to build visual vocabulary and construct classifier. HSS is a newly improved image segmentation method based on region merging. Considering the optimizing of merging criterion, the HSS uses multiple and complementary similarity measurements to get a combined similarity measurement with adjustable weights adapted to the environment of mobile robots during its once running. As a result, HSS method generates a smaller, higher-quality object regions set through an image and reduce the later computation cost for classification. Figure 1 shows the structure of objects classification for mobile robots using HSS.

### 2 Theoretical background

#### 2.1 Graph-based image segmentation model



**Figure 1.** Classification for mobile robots using HSS .

Many scholars use the graph theory to describe the image segmentation problem. For an undirected graph  $G=(V,E)$ , each node  $v_i \in V$  corresponds to a pixel of the image, the edge  $(v_i, v_j) \in E$  connects a pair of adjacent pixels  $v_i$  and  $v_j$ , the weight of edge  $w((v_i, v_j))$  measures dissimilarity (e.g., intensity, color, texture or other attribute) between the neighboring elements  $v_i$  and  $v_j$  with a non-negative value. A segmentation  $S$  is a dividing of  $V$  into different regions. The general methods on evaluating the quality of image segmentation require the pixels in a region to be similar, that is, the weight of two adjacent pixels in the same region is low and the weight of two neighboring pixels in different regions is high.

Define the *internal difference*  $Int(C)$  of one region  $C \subseteq V$  as the largest weight in minimum spanning tree [4] of the region,  $MST(C, E)$ . that is,

$$Int(C) = \max_{e \in MST(C, E)} w(e) \quad (1)$$

Define the *difference* between two regions  $C_1, C_2 \subseteq V$  to be the minimum weight edge connecting the two regions. That is,

$$Dif(C_1, C_2) = \min_{v_i \in C_1, v_j \in C_2, (v_i, v_j) \in E} w((v_i, v_j)) \quad (2)$$

By comparing the minimum of the two regions *internal difference*  $MInt(C_1, C_2)$  and  $Dif(C_1, C_2)$  to check if there exists real boundary  $D(C_1, C_2)$ . ‘true’ denotes there exists boundary between the two regions while ‘false’ denotes not exists. The pairwise comparison predicate is defined as follow:

$$D(C_1, C_2) = \begin{cases} \text{true if } Dif(C_1, C_2) > MInt(C_1, C_2) \\ \text{false otherwise} \end{cases} \quad (3)$$

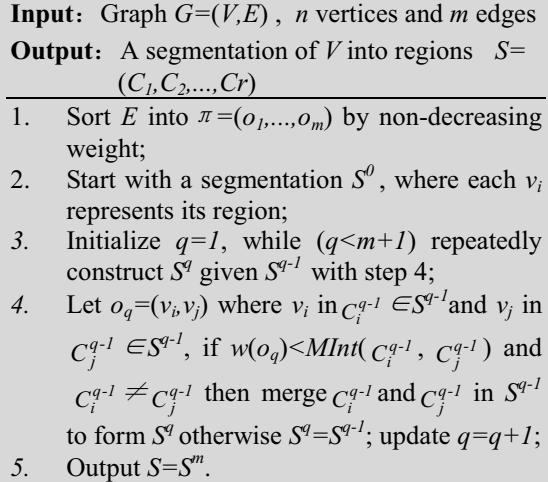
The threshold function  $\tau$  controls the degree to which the difference between two regions must be greater than their *internal differences*.  $k$  is a constant number used to control the scale of region.

$$MInt(C_1, C_2) = \min(Int(C_1) + \tau(C_1), Int(C_2) + \tau(C_2)) \quad (4)$$

$$\tau(C_i) = k / |C_i|$$

The segmentation criteria has a great impact on the complexity of the image segmentation. In order to make it more robust to those outliers, finding an effective segmentation method has proved to be a NP-hard problem [5]. Based on the pairwise comparison predicate shown in (3) and greedy decision, a segmentation method is proposed in [5] which is proved neither too coarse nor

too fine, and its running time is nearly linear in the number of image pixels, so it was often used to produce the initialized segmentation regions. The algorithm is well described in Figure 2.



**Figure 2.** Segmentation algorithm.

## 2.2 Bag of words model

The Bag of words model was initially proposed in [2]. It can be described as four steps: (1) extract visual words from images by local feature descriptors like SURF [6], SIFT[7-8], PCA-SIFT [9], (2) construct the visual vocabulary by the clustering algorithms like k-means [10-11] and the random clustering forests [12], (3) quantify the images by the histogram of the extracted words in the vocabulary, and (4) use the sample images to train and test the classifier. The direct using of BOW model has been proved to be well applied to the classification of the repeated/near-duplicate images. However the irrelevance of space information to vocabulary makes the accuracy vulnerable to the impact of complex background from mobile robots.

As an extension of the BOW, in this paper, the hierachic object regions with spatial information are used to be the inputs of the classification method based on BOW and SVM instead of the whole image. For the less background information interferes the construction of the vocabulary, it is exactly well applied to deal with the various and complex background of images captured from mobile robots.

## 2.3 Classification of object regions

The classifiers appeared in the recent papers on the objects or scene recognition focusing on the K-nearest neighbor algorithm, Support Vector Machine (SVM) [13-14] and so on. The SVM dexterously applies the kernel function to avoid the search of mapping algorithm and the calculation of high-dimensional space. By using different kernel functions and parameters, the SVM get the curved interface to the input space.

In this paper, the LIBSVM [15] is used for the selected object regions classification. For the minimizing of SVM shrinking and caching, the L1-norm is used when dealing with the regions and an optimized sampling strategy is used to remove the redundant regions before the BOW model.

<b>Input:</b>	Color image
<b>Output:</b>	Object location boxes $\mathbf{L}$
1.	Initialize the thresholds $K$ and the similarity measurements set $similarityset$ ;
2.	Concurrently extract object location boxes for each threshold $k_l \in \mathbf{K}$ through step 3-6;
3.	Get start regions $\mathbf{R} = \{r_1, \dots, r_n\}$ with $k_l$ using Figure 2 and initialize similarity set $\mathbf{S} = \emptyset$ , $\mathbf{S}_{new} = \emptyset$ and $S_{struct}(r_i, r_j) = \emptyset$ ;
4.	For each neighboring region pair $(r_i, r_j)$ , calculate the similarity structure set $S_{struct}(r_i, r_j)$ and update $\mathbf{S} = \mathbf{S} \cup S_{struct}(r_i, r_j)$ , each structure contains five similarities $S_{sim}(r_i, r_j)$ with $simi \in similarityset$ and a combined similarity $S_{combined}(r_i, r_j)$ ;
5.	While $\mathbf{S} \neq \emptyset$ do <ol style="list-style-type: none"> <li>Get the structure with the highest combined similarity <math>S_{struct}(r_i, r_j) = \max(\mathbf{S}_{combined})</math>;</li> <li>Merge <math>r_i</math> and <math>r_j</math> to form <math>r_{new} = r_i \cup r_j</math>;</li> <li>Dropout the similarities related to <math>r_i</math>: <math>\mathbf{S} = \mathbf{S} \setminus S_{struct}(r_i, r_*)</math>;</li> <li>Dropout the similarities related to <math>r_j</math>: <math>\mathbf{S} = \mathbf{S} \setminus S_{struct}(r_j, r_*)</math>;</li> <li>Calculate similarity structures <math>\mathbf{S}_{new}</math> between <math>r_{new}</math> and its neighbors;</li> <li>Update <math>\mathbf{S} = \mathbf{S} \cup \mathbf{S}_{new}</math>;</li> <li>Update region set <math>\mathbf{R} = \mathbf{R} \cup r_{new}</math>;</li> </ol>
6.	Extract object region boxes $\mathbf{L}_l$ from all regions in $\mathbf{R}$ ;
7.	Combine all the extracted boxes $\mathbf{L}_l$ from step6 to form and output $\mathbf{L}$ .

Figure 3. Hierarchic selective search algorithm.

### 3 Hierarchic Selective Search

There are many methods to extract the object regions from various images. The popular methods contain Edge Box [16], Selective Search [17-18], Randomized Prim [19], Objectness [20] and so on. They are based on methods of threshold, edge, region, graph or energy functional. Each method has its own special advantages. Considering the computation cost and the various background of mobile robots images, in this paper, the graph-based image segmentation in Figure 2 is used to

initialize the start regions, then the HSS based on region merging is used to form a higher quality, multiple scales and class-independent regions set.

The HSS currently deals with two start regions set for higher quality, the grouping procedure of a start regions set works as follows: firstly calculate the combined similarities between all neighboring regions; then the two most similar regions are merged together, and new similarities are calculated between the resulting region and its neighbors. The process of grouping the most similar regions is repeated until the whole image becomes a single region. The start regions with Figure 2 is well satisfying the requirement of a single object per box. For the regions are hierarchically merged into larger regions, it naturally generates regions at all scales. The detailed description of HSS is present in Figure 3.

#### 3.1 Combined similarity measurement

While the images captured from the vision sensors on the mobile robot are device-dependent and vulnerable to the influence of illumination, occlusion and so on. Using multiple complementary similarity measurements has obvious advantages compared with a single measurement. In Figure 3, the combined similarity measurement is made of color and texture similarity measurements. A structure is used to keep all five single similarity in  $similarityset$  and a combined similarity *combined*.

##### 3.1.1 Color space and texture feature

As for the color models, the RGB, Lab and HSV are used in this paper. The RGB simplifies the architecture and design but relies on the devices heavily. The RG channels of the normalized RGB plus intensity denoted as RGI. The Lab well describes the human visual perception. For its device-independent, it exactly appeals to the various mobile robots and covers the shortage of RGB. The HSV and RGI well describes the hue, saturation, light and shadows for invariance properties.

Texture feature is the intrinsic characteristic of all the surface of the object, it contains important information about the structure of object surface and the relationship with the surrounding environment. Also it reflects visual characteristics of the homogeneous phenomenon in the image without depending on the color and brightness information. In this paper, the fast SIFT-like method [21] is used to extract texture features.

##### 3.1.2 Feature similarity measurement

From the similarity measurements such as Euclidean distance, histogram intersection and so on. In this paper, the distances of histograms on two regions' color and texture are calculated to measure similarities. The formula for computing the components of the combined similarity measurement in Figure 3 is shown below.

$$S_{simi}(r_i, r_j) = \sum_{l=1}^n \min(simil_i^l, simil_j^l) \quad (5)$$

Subscript *simi* refers to the color or texture features in  $similarityset = \{HSV, Lab, RGI, Intensity, Texture\}$ . For the color similarity measurement, a one-dimensional color

histogram is obtained by using 25 bins for each color channel in each region, so the resulting color histogram for a three channel image is  $C_i = \{c_i^1, \dots, c_i^n\}$  where  $n=75$ .

As for the texture similarity measurement, the method is based on fast SIFT-like and takes Gaussian derivatives in eight orientations using  $\sigma=1$  for each color channel. Each color channel extracts a histogram using a bin size of 10, so the texture histogram for a three channel image is  $T_i = \{t_i^1, \dots, t_i^n\}$  where  $n=240$ . Both the color histogram and the texture histogram are normalised using the L1-norm. The color and texture histograms are effectively passed through the hierarchy. The formula is shown in (6) where  $CT_{result}$  refers to the color or texture histogram of the merged region.

$$CT_{result} = \frac{\text{size}(r_i) \times CT_i + \text{size}(r_j) \times CT_j}{\text{size}(r_i) + \text{size}(r_j)} \quad (6)$$

### 3.2 Adjustable weights for various environments

While the images are captured in various scenes and conditions, the emphasis of a similarity measurement is not the same. In this paper, a flexible weight adapted to the environment of the mobile robots is given to the corresponding similarity measurement. For example, when the mobile robot is in an environment with similar color in foreground and background, a higher weight is given to the texture similarity measurement and lower to color similarity measurements. The combined similarity measurement is detailed described in (7) where  $num=5$  is the number of similarity measurements,  $simi_k$  refers to one similarity measurement in  $similarityset$ .

$$S_{combined}(r_i, r_j) = \sum_{k=1}^{num} a_k * S_{simi_k}(r_i, r_j) \quad (7)$$

For the distribution of weights, it depends on the real environment of mobile robot, the paper does not give a standard distribution strategy, mobile robots can check environment to dynamically adjust weights in real scene.

In a word, HSS has changed the merging criterion by using a combined similarity measurement adapting to mobile robots various environment compared to Selective Search (SS) in [17]. While the SS contains multiple color similarity measurements and starting regions for many image cases. But for the color similarity measure in a special starting regions set, they just take a single color space into account during the whole grouping algorithm. HSS uses various color spaces during its once running. It seems that HSS get a intersection among the regions repeatedly running the SS with single color space. So the HSS get a smaller regions set with nearly the same quality compared with SS.

## 4 Experiments Evaluation

The SS performs well in the general object classification and image segmentation on the dataset Pascal VOC 2007 [22] with its fast and multiple strategies. So, in this paper,

the HSS is compared to the SS on both Pascal VOC 2007 and OCS [23]. The OCS is a mobile product images dataset used for retrieving which contains 854 training samples (total 20 categories) and 60 testing samples (total 6 categories). While the Pascal VOC 2007 was an official dataset for image classification and segmentation in the ImageNet challenge [24], many efficient algorithms are evaluated on it, besides the same type of images have various status and most of images contain multiple objects. To some certain extent, these samples can be regarded as the images captured by the mobile robot in a better environment. Some testing images are shown in Figure 4. All experiments are done on the computer with Pentium Dual-Core E6700 3.2GHz, with 4G RAM using the Matlab, C and C++ development languages.



Figure 4. Some testing images on the dataset.

### 4.1 Experimental method

In the paper, the segmentation results are evaluated firstly. The HSS algorithm is tested just with thresholds 50 and 100. The color spaces include HSV, RGI, I (Intensity) and LAB. The weights corresponding to each similarity measurement are set manually according images global properties. For the OCS dataset only provides the sample images, in order to reuse the code, the OCS dataset is organized using the form of Pascal VOC 2007 dataset and firstly get the OCS ground truth annotations, and then calculate overlaps between the resulting region and the true object annotation as the assessment of the quality of segmentation algorithm. In this paper, the mean average best overlap (MABO) and mean number of object regions (MNOR) for all samples are used to comprehensively evaluate HSS, and then compared with SS. To calculate the Average Best Overlap for a specific class  $c$ , firstly the best overlap between each ground truth annotation  $g_i^c \in G^c$  and the object regions  $L$  generated for the corresponding image is calculated.

$$\text{Overlap}(g_i^c, l_j) = \frac{\text{area}(g_i^c) \cap \text{area}(l_j)}{\text{area}(g_i^c) \cup \text{area}(l_j)} \quad (8)$$

$$ABO_c = \frac{1}{|G^c|} \sum_{g_i^c \in G^c} \max \text{Overlap}(g_i^c, l_j) \quad (9)$$

$$MABO = \text{mean} \left( \sum_{c \in C} ABO_c \right) \quad (10)$$

Later the results of object classification for mobile robots are evaluated based on the produced regions. Considering the real-time requirement of mobile robots,

all regions are sampled to get the new training and testing regions set. Then the BOW model represents object features from regions using the SIFT [8] descriptor and k-means method [10], finally the SVM classifier [13] is trained and the accuracy of prediction test regions is calculated. The same procedure is used for regions segmented from SS.

While scholars often use accuracy to evaluate method object classification. Suppose there are  $P$  positive samples and  $N$  negative samples,  $TP$  refers to the positive samples correctly predicted to positive sample,  $TN$  refers to the negative samples correctly predicted to negative sample, the accuracy is described as below.

$$\text{Accuracy} = (TP + TN) / (P + N) \quad (11)$$

## 4.2 Evaluation and analysis

### 4.2.1 Performance of HSS segmentation algorithm

Table 1 and 2 respectively shows the MNOR and MABO for all samples on Pascal VOC 2007 (VOC) dataset and OCS dataset, in which HSS50 and HSS100 separately refer to the condition of using HSS algorithm in threshold  $k_i=50$  and  $k_i=100$ , SS100 refers to the condition of using SS in threshold  $k_i=100$ . The combined similarity is based on a color space and texture, Table 1 and 2 don't list the texture similarity measurement for SS just used a single color similarity measurement. Other tables are the same.

**Table 1.** Comparison under single color space on VOC dataset.

Similarity measure-	MNOR			MABO		
	HSS	HSS1	SS	HSS5	HSS1	SS
ment	<b>50</b>	<b>00</b>	<b>100</b>	<b>0</b>	<b>00</b>	<b>100</b>
HSV	471	288	463	0.691	0.686	0.693
Lab	328	189	318	0.698	0.690	0.690
RGI	370	214	362	0.693	0.684	0.693
Intensity	408	241	399	0.688	0.675	0.670

**Table 2.** Comparison under single color space on OCS dataset.

Similarity measure-	MNOR			MABO		
	HSS	HSS1	SS	HSS5	HSS1	SS
ment	<b>50</b>	<b>00</b>	<b>100</b>	<b>0</b>	<b>00</b>	<b>100</b>
HSV	489	303	429	0.583	0.572	0.580
LAB	277	162	214	0.563	0.551	0.560
RGI	334	186	254	0.572	0.562	0.558
Intensity	378	219	304	0.560	0.552	0.566

Table 3 shows the hierarchical combination strategy result on VOC and OCS dataset. Considering the real object annotations from OCS dataset are based on the boxes rather than the shapes of objects, so the MABO is lower in OCS compared to its value in VOC dataset.

**Table 3.** Comparison under hierarchical combination strategy.

Dataset	Strategy( $k_0=50 k_1=100$ )	MNOR	MABO
Pascal VOC 2007	HSV+LAB HSV+LAB SS	2147	0.799
	LAB+RGI   HSV+I HSS	1976	0.815
	HSV+RGI   I+LAB HSS	1988	0.818
OCS	HSV+LAB HSV+LAB SS	1789	0.611
	LAB+RGI   HSV+I HSS	1206	0.616
	HSV+RGI   I+LAB HSS	1120	0.621

Observing from all table, HSS proposed in this paper

is better than SS in the number of object regions. From the comparison under single color space when  $k_i=100$ , HSS has the MNOR value less than SS nearly 1/3, while the change of MABO value is less than 0.01. Some MABO values from HSS are even better than the MABO from SS. From the comparison under the hierarchical combination strategy, HSS shows its superiority in both MNOR and MABO.

While SS repeatedly runs single color space strategy leading to the object regions redundancy, HSS gets the higher quality regions with less quantity using combined and environment-adjustable similarity measurement. The subsequent classification relies much on the segmented regions, so the quantity and quality of object regions have an important influence on the computation and accuracy of classification. Under the special mobile robots task and Pascal VOC 2007, HSS is relatively better than SS.

### 4.2.2 Performance of classification for mobile robots

Table 4 shows the results of the classification for mobile robots based on a special task dataset OCS. 'HSS-ORS' refers to the regions from HSS; 'SS-ORS' refers to the regions from SS; 'OI' refers to the original images. By changing the samples for classification among the three sources, the accuracy along with the quantity of regions used in training and testing are compared.

**Table 4.** Classification for mobile robots based on OCS.

Samples source	Training samples	Testing samples	Accuracy (%)
SS-ORS	5881	1101	48.97
HSS-ORS	3977	660	51.36
OI	186	60	45.00

Observing the data from above tables, HSS uses less regions than SS. Therefore, during the whole process of classification for mobile robots based on regions, the computational cost of HSS is lower than that of SS. From the accuracy, HSS is also better than SS. Compared with the classifier constructed using original images, the classifier based on object regions has higher accuracy. This is due to the improvement in the quality and quantity of samples based on object regions. The OCS dataset is relatively small and the background of training and testing samples belonging to the same category varies much. Less background interference information and more high-quality object region samples make the accuracy improved but at the expense of time cost. At present, while the computing abilities to mobile robots are quickly upgraded, most of the classification tasks tend to pursue the improvement in accuracy, so the advantages of HSS will be reflected more obviously.

## 5 Conclusion

Image segmentation plays an important role in improving the performance of object classification for mobile robots. With respect to the feature extracted in the whole image, the method based on object regions effectively reduces

the interference of the background and computation cost; accurate object regions can significantly improve the classifier training and testing process and polish up the accuracy of classification in a large extent. As a method of regions capture, HSS used a combined similarity measurement to get a smaller and higher-quality regions set. At the same time, it was applied to the classification for mobile robots based on regions which relieved the background interference in mobile environment. When using the learning methods for object classification, the accuracy is much related to the quantity and quality of samples. In a certain extent, HSS can also solve the problem of data scarcity on some special task for mobile robot cooperating with the method based on regions. The later of this work will be extended to more other datasets for mobile robots tasks.

## Acknowledgments

This work is supported by the National Science and Technology major project (No. 2011ZX05027-005). Thanks to all of the relevant persons.

## References

1. S.B. Niu, Introduction to Robotics: Analysis, Control, Applications. *Prentice Hall/Pearson*, (2004).
2. J. Sivic, A. Zisserman, Video Google: A Text Retrieval Approach to Object Matching in Videos. *IEEE International Conference on Computer Vision*, 1470, (2003).
3. A. Sampath, A. Sivaramakrishnan, K. Narayan, A Study of Household Object Recognition Using SIFT-Based Bag-of-Words Dictionary and SVMs. *Proceedings of the International Conference on Soft Computing Systems*. Springer India, (2016).
4. C.T. Zahn, Graph-theoretic methods for detecting and describing gestalt clusters. *IEEE Transactions on Computing*, **20**:68-86, (1971).
5. P.F. Felzenszwalb, D.P. Huttenlocher, Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, **59**(2):167-181, (2004).
6. B. Herbert, T. Tinne, V. G. Luc, SURF: Speeded Up Robust Features. *Computer Vision & Image Understanding*, **110**(3):404-417, (2006).
7. D. Lowe, Object recognition from local scale-invariant features. *IEEE International Conference on Computer Vision*, **2**:1150-1157, (1999).
8. Q. Zhu, C. Liu, C. Cai, A Novel Robot Visual Homing Method Based on SIFT Features. *Sensors*, **15**(10):26063-26084, (2015).
9. Y. Ke, R. Sukthankar, PCA-SIFT: a more distinctive representation for local image descriptors. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, **2**:506-513, (2004).
10. N. A. M. Isa, S. A. Salamah, U. K. Ngah. Adaptive fuzzy moving K-means clustering algorithm for image segmentation. *IEEE Transactions on Consumer Electronic*, **4**(55): 2145-2153, (2009).
11. C.H. Lin, C.C. Chen, H.L. Lee, Fast K-means algorithm based on a level histogram for image retrieval. *Expert Systems with Applications*, **41**(7): 3276-3283, (2014).
12. G. Fanelli, M. Dantone, J. Gall, Random Forests for Real Time 3D Face Analysis. *International Journal of Computer Vision*, **101**(3): 437-458, (2013).
13. C. Cortes, V. Vapnik, Support-vector networks. *Machine Learning*, **3**(20): 273-297, (1995).
14. B.C. Kuo, H.H. Ho, C.H. Li, A Kernel-Based Feature Selection Method for SVM With RBF Kernel for Hyperspectral Image Classification. *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing*, **7**(7): 317-326, (2014).
15. C.C. Chang, C.J. Lin, LIBSVM: A library for support vector machines. *AcM Transactions on Intelligent Systems & Technology*, **2**(3): 389-396, (2011).
16. C. We, C. X. xin, and Z. Li, Box-counting dimension analysis of edge images detected with classical edge detector. *International Conference on measuring technology and mechatronics automation*, Changsha , China, 756-758, (2010).
17. J. Uijlings, K. Sande, T. Gevers, A. Smeulders, Selective search for object recognition. *International Journal of Computer Vision*, **104**(2): 154-171, (2013).
18. X. Wan , X. Zheng , X. Pang, Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **37**(9): 1904-1916, (2014).
19. S. Manen, M. Guillaumin, L.V. Gool, Prime object proposals with randomized prim's algorithm. *International Conference on Computer Vision*, 2536-2543, (2013).
20. B. Alexe, T. Deselaers, V. Ferrari, Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **34**(11): 2189-2202, (2012).
21. J. Rabin, J. Delon, Y. Gousseau, A contrario matching of SIFT-like descriptors. *Pattern Recognition, IEEE International Conference on*. *Pattern Recognition*, 1-4, (2009).
22. M. Everingham, L.V. Gool, C.K.I. Williams, J. Winn, A. Zisserman, The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. <http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html>, (2007).
23. S.X. Hui, Z. Lin, B. Jonathan, W. Ying, Mobile product image search by automatic query object extraction. *European Conference on Computer Vision*, 114-127, (2012).
24. J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, L. FeiFei, ImageNet Large Scale Visual Recognition Competition 2014. <http://www.imagenet.org/challenges/LSVRC/2014/>, (2014)