Journal of Software Engineering and Applications, **2013, 6, 511-518**
http://dx.doi.org/10.4236/jsea.2013.69061 Published Online September 2013 (http://www.scirp.org/journal/jsea)

Object Detection Using SURF and Superpixels*

Miriam Lopez-de-la-Calleja[1], Takayuki Nagai[2], Muhammad Attamimi[2], Mariko Nakano-Miyatake[1], Hector Perez-Meana[1]

[1]ESIME Culhuacan, Instituto Politecnico Nacional, Mexico City, Mexico; [2]University of Electro-Communications, Tokyo, Japan.
Email: hmperezm@ipn.mx

Received July 19th, 2013; revised August 20th, 2013; accepted August 28th, 2013

ABSTRACT

This paper proposes a novel object detection method in which a set of local features inside the superpixels are extracted from the image under analysis acquired by a 3D visual sensor. To increase the segmentation accuracy, the proposed method firstly performs the segmentation of the image, under analysis, using the Simple Linear Iterative Clustering (SLIC) superpixels method. Next the key points inside each superpixel are estimated using the Speed-Up Robust Feature (SURF). These key points are then used to carry out the matching task for every detected keypoints of a scene inside the estimated superpixels. In addition, a probability map is introduced to describe the accuracy of the object detection results. Experimental results show that the proposed approach provides fairly good object detection and confirms the superior performance of proposed scene compared with other recently proposed methods such as the scheme proposed by Mae *et al.*

Keywords: Object Detection; SURF; SLIC Superpixels; Keypoints Detection; Local Features; Voting

1. Introduction

In the area of intelligent systems, the autonomous mobile robots are expected to have the ability to recognize their surrounding environment in real time. Object detection, which is a task for searching and localizing a target in a particular scene, can be considered as prime feature for autonomy. This fact has stimulated the research in this field and as a result several algorithms have been proposed during the last several years. Lai *et al.* [1], Ozuysal, *et al.* [2], Harzallah, *et al.* [3], Dalal and Triggs [4] proposed to use the standard sliding window approach in which the system evaluates a score function for all positions and scales in an image; and sets limits to the scores to obtain bounding boxes for each instance. Each detector window has a fixed size and search across 20 scales on an image in a pyramidal form. For efficiency a linear score function is considered. The performance of the classifier heavily depends on the data and also the features used for the object detection [1]. Another popular approach is to extract local interest points from the image and then to classify each of the regions around these points, rather than looking at all possible sub windows [5-7]. A weakness shared by all of the above approaches is that they can fail when local image information is insufficient, that is, if the target is very small or highly occluded. To reduce these problems, Mae *et al.* [8] included a local feature matching algorithm using local geometric consistency for object detection. When it is online, the system uses SIFT for scene feature extraction and compares them with those of the reference image. This research is suited for objects that have texture, and performs better when the objects have flat surface or when they are observed from the same view angle. The advantage of this approach is the simplicity of the implementation and portability for various robot control systems, minimal knowledge for the target pattern and fairly good performance. The main disadvantages are that it is limited to the patterns with texture and the flat surface assumption for pose estimation. As a result, the matching could worsen if the object has non-planar surface and if it is observed from a different view-point.

Other proposals are based on the appearance [9-15], which are offline methods based on a collection of small patches. These approaches provide good detection rates although its computational complexity is large, requiring in general long processing time to generate the model of

*Object detection for robotic vision.

each object [15].

Thus the current progress in object detection still requires further research to achieve efficiency close to 100% in real time. To contribute to the improvement of some issues in object detection, this paper proposes an object detection algorithm based on super pixel together with the Speed-Up Robust Feature (SURF) as a feature extraction method to perform the matching task. Evaluation results show that despite a cluttered background and occlusion, the proposed algorithm is able to detect the specific object among several other similar looking ones. This property makes the proposed algorithm suitable for using on robotic platforms which may operate in natural sceneries.

The rest of this paper is organized as follows: Section 2 provides a detailed description of proposed method for object detection. Section 3 provides the experimental results and discussion together with a comparison of the performance of the proposed method with other recently proposed algorithms. Finally, the main conclusions are presented in Section 4.

2. Proposed Method

The proposed method is based on the use of SLIC super pixel [6] and SURF [7], together with a voting process and the probability map, which is introduced in this work in order to improve the accuracy of object detection. **Figure 1** shows the block diagram of the proposed method. Here given an input image color and the Time of Flight (TOF) data, is segmented using SLIC super pixel [16,17]. Next several key points are extracted and labeled as a feature, for matching, using SURF [18]. Then the extracted key points of the input image are compared against the learned key points stored in a database. Next, a voting, which is similar to a histogram of Ids, is calculated for the input key points and then the final Id is determined as the greatest number in this step. Then we use a probability map, which is generated for each Id, to increase the accuracy of the position estimated of a given object in the scene. Finally using these Ids, the desired object is detected in the scene. Next sub-sections provide a detailed description of each stage of proposed system.

2.1. Database

A 3D visual sensor [16] which consists of a TOF and two CCD cameras is used to capture color and 3D information to construct a database. To obtain the visual information, a small handheld observation table with an XBee wireless controller is installed on a robot that enables the observation of the object from various viewpoints. Here, 10 objects are used and 40 different views of each object are captured. Considering the computational cost, the SURF algorithm [17] is used to collect a set of 128 dimensional descriptors from each captured image which is stored in the database.

2.2. Input Image

The proposed object detection method uses visual sensor, shown in **Figure 2**, which acquires color information in real time by calibrating the TOF (Time of Flight) and two CCD cameras [16].

2.3. Segmentation Process

Superpixels has been applied in several computer vision applications such as depth estimation [17], image segmentation [18,19] and object localization [20], etc. Because in most of these applications superpixels have performed fairly well, several approaches to calculate such superpixels have been proposed in the last decade [21-26]. Among them a suitable approach is the so called Simple Linear Iterative Clustering (SLIC) [27] because it is faster to compute, achieve high segmentation quality and provides accurate segmentations.

Assume that an N-pixels image is divided in K non-overlapped sub-blocks of size $S \times S$ pixels, where $S = (N/K)^{1/2}$, whose center is given by (x_i, y_i). To avoid the superpixel center being located on an edge or a noisy pixel, it is estimated as the point with the smallest gradient in a window of 3 × 3 pixels around the center of sub-block

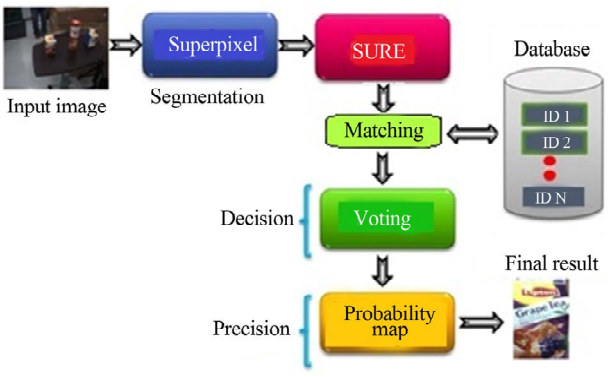


Figure 1. Proposed method.

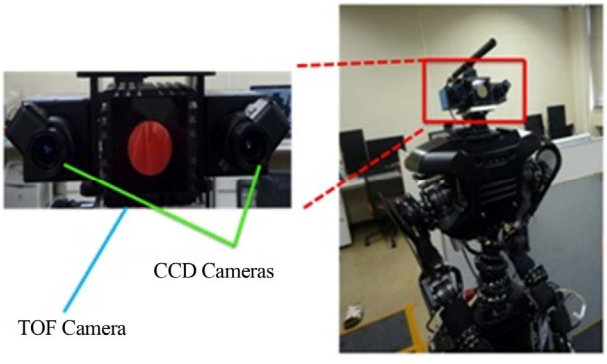


Figure 2. Visual sensor used for object detection.

under analysis [27]. After the center of *i-th* superpixel is obtained, the center of the *i-th* cluster center is determined as follows:

$$C_i = \left[L_i, a_i, b_i, \hat{x}_i, \hat{y}_i\right],\ I = 1, 2,..,K,\ L_i \qquad (1)$$

where $\left(\hat{x}_i, \hat{y}_i\right)$ is the center of *i-th*-cluster, L_i is its lightness, a_i its redness-greenness, b_i its yellowness-blueness. Once the *K* initial centers are determined, each pixel in a neighborhood of $2S \times 2S$ pixels is associated with the superpixel, in such neighborhood, whose distance, *D*, be minimum, where

$$D = \sqrt{\left(\mathrm{d}c\right)^2 + \left(\frac{\mathrm{d}s}{S}\right)^2 m^2}\ , \qquad (2)$$

$$dc = \sqrt{\left(L_j - L_i\right)^2 + \left(a_j - a_i\right)^2 + \left(b_j - b_i\right)^2}\ , \qquad (3)$$

$$ds = \sqrt{\left(x_j - \hat{x}_i\right)^2 + \left(y_j - \hat{y}_i\right)^2}\ , \qquad (4)$$

$$S = \sqrt{N/K}\ , \qquad (5)$$

and $1 \le m \le 40$ is a constant that controls the importance of color similarity and spatial distance. Thus when m is large the spatial distance is more relevant and the resulting superpixels are more compact, while when *m* is small the color becomes more important and the superpixels becomes more irregular in size and shape, approaching more closely to the image boundaries. Finally, after all pixels have been associated the closest superpixel, a new center C_i, $I = 1, 2,\ldots,K$, is estimated by averaging all pixels belonging to the *i-th* superpixel.

Proposed algorithm assumes $K = 200$ and to control the compactness of a superpixel we select $m = 10$ which provides a good balance between the color similarity and spatial proximity.

2.4. Feature Extraction

The Speeded-Up Robust Features (SURF) [28], which is a scale and rotation-invariant detector and descriptor, is used for feature extractions. The main task of SURF is finding point correspondences between two images of the same object. The structure of the SURF algorithm is divided in three steps: 1) Interest key point detection; 2) estimation of a feature vector, called descriptor; 3) matching between images. The interest point detection is employed to find relevant points in one image or object, in order to allocate valuable information that is going to be computed by a local descriptor, by means of the Hessian-Laplace matrix detectors. The Estimation of a feature vector describes the relevant regions within the interest point neighborhood. It has to be distinctive and at the same time robust to noise, detection, geometric and photometric displacement deformations. Finally in most situations the scenery has many key points that must be identified with labels, which can be achieved using locality-sensitive hashing (LSH) [29,30]. It is an indexing scheme for performing approximate search in high dimensional environments by enumerating all nearest neighbors and choosing the nearest point. To filter the matching results, the Euclidean distance between the matched descriptor and the most similar one is firstly calculated. The descriptors with low distances comparing a predetermined threshold are used. Finally, the best *k* results that satisfy the threshold will be used for voting step described in the next step.

2.5. Voting

In the voting process, the key points estimated by the feature extraction stage that are inside a given boundary are considered in the voting process. A voting technique is applied to estimate the object inside the superpixel and then to obtain the *Id* to capture the object. Let *N* be the number of objects in the database; *V*(*i*,*j*) is the number of matched key points of the object *j* inside the *i-th* superpixel and *V*(*i*,*N* + 1), is the number of unmatched key points inside the *i-th* superpixel. Then, the resulting parameter $Id_{\max}\left(1 \le id \le N\right)$ is determined by the maximum vote number. This *Id* is used to segment the superpixels that results of the whole segmentation of a given input image.

2.6. Probability Map

The probability map is used to determine the probability of the *Id* in each segmented part of the image. The process of estimating the probability map consists of finding the occurrence rate of each *Id* in the image, such that a particular *Id* can be selected according to the occurrence rate to that *Id*. The probability of object *j* at pixel (*a*, *b*) inside the *i-th* superpixel is then given by

$$P_m\left(a,b\middle| j\right) = \frac{V\left(i,j\right)}{\sum_{k=1}^{N+1} V\left(i,k\right)} \qquad (6)$$

where $P_m\left(a,b,\middle| j\right)$ is defined as the probability map that represents the accuracy of object detection at pixel (*a*, *b*), *V*(*i*, *j*) is the vote number of matched keypoints of object *j* inside the *i-th* superpixel. The Probability map is estimated for all *Id*'s in order to determine the probability of the detected objects. This method, as well as the voting helps to increase the accuracy of the object detection algorithm.

3. Experimental Results

This section presents the results of the experimental evaluation of proposed system. The experiment is carry out in a room using 10 different objects as shown in the

Figure 3(a). The user shows each object in various angles to the robot to build the database, as shown in **Figure 3(b)**, which acquires features vectors of 40 consecutive frames for each object that are generated by the robot in the learning phase.

3.1. Detection Performance

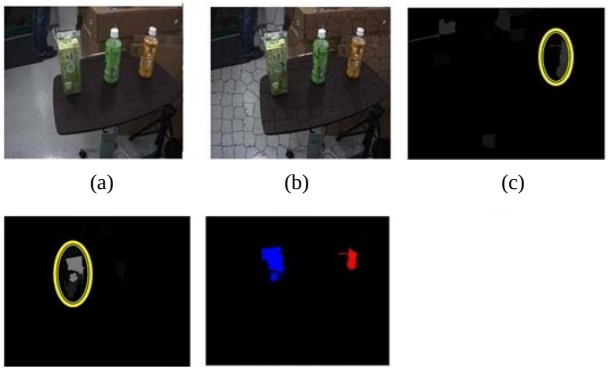
To evaluate de object detection capability of proposed system, 10 different experimental setups are constructed as shown in **Tables 1** and **2**. Some of the experimental processes used for obtaining the object detection performance of proposed scheme are shown in **Figures 4-7**.

Figure 4(a) shows the setup 1, where we have 3 objects, two of them belonging to the database, which means that the robot will only have to detect two of them. **Figure 4(b)** shows the superpixel estimation used for segmenting the desired region; **Figures 4(c)** and **(d)** give the desired probability map for orange juice bottle and green tea carton box. Finally, **Figure 4(e)** shows the detected region both objects, as it was expected.

Figure 5(a) shows the setup 2 in which we have several objects, 3 of them belonging to the database, which means that the robot has to detect the 3 objects belonging to the

(a) (b)

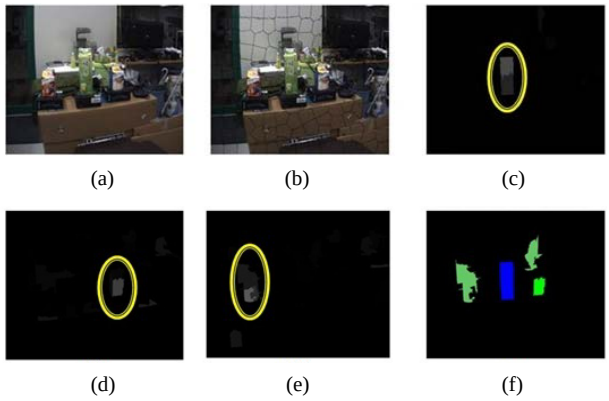
Figure 3. Learning phase, (a) Objects used for training; (b) A people showing the objects to the robot from different angles.



(a) (b) (c)

(d) (e)

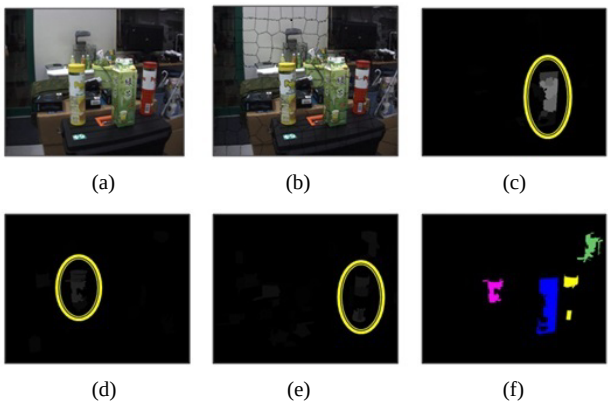
Figure 4. Experimental setup 1: (a) Input image; (b) Superpixels evaluation result; (c) Probability map of orange tea bottle; (d) Probability map of green tea carton box; (e) Detected objects.



(a) (b) (c)

(d) (e) (f)

Figure 5. Experimental setup 2: (a) Input Image; (b) Superpixels evaluation; (c) Probability map of green tea carton box; (d) Probability map of milk tea carton box; (e) Probability map of potatoes red carton box; (f) Detected object.



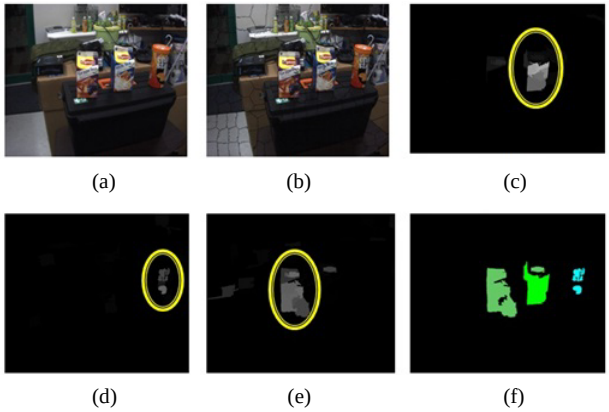
(a) (b) (c)

(d) (e) (f)

Figure 6. Experimental setup 8: (a) Input image; (b) Superpixel evaluation results; (c) Probability map of milk tea cardbox; (d) Probability map of yellow potatoes package; (e) Probability map of red potatoes package; (f) Detected objects.

Table 1. Experimental setups used for system evaluation.

Setup	Object 1	Object 2	Object 3
1	Green tea carton box	Orange tea bottle	
2	Grape tea carton box	Potatoes red box	Milk tea carton box
3	Green tea carton box	Potatoes yellow box	Pringles
4	Chipstar	Seafood noodles plastic box	Pringles
5	Pringles	Seafood noodles plastic box	Chipstar
6	Grape tea carton box	Green tea carton box	Orange tea bottle
7	Potatoes yellow box	Green tea carton box	Potatoes red box
8	Potatoes yellow box	Milk tea carton box	Potatoes red box
9	Grape tea carton box	Orange tea bottle	Chipstar
10	Orange tea bottle	Green tea carton box	



(a) (b) (c)

(d) (e) (f)

Figure 7. Experimental setup 4: (a) input image; (b) Superpixel evaluation results; (c) Probability map green tea cardbox; (d) Probability map of orange bottle tea; (e) Probability map of chipstar (f) Detected objects.

Table 2. Evaluation results obtained for the ten different setups used for system evaluation.

Setup number	Number of objects	Detected objects	Detection rate
1	2	2	100%
2	3	3	100%
3	3	2	66%
4	3	1	33%
5	3	1	33%
6	3	3	100%
7	3	1	33%
8	3	3	100%
9	3	3	100%
10	2	2	100%

database. **Figure 5(b)** shows the superpixels used for segmenting the desired regions, **Figures 5(c)-(e)** shows the desired probability map of the green tea, milk tea carton boxes and the yellows box containing potatoes; finally, **Figure 5(f)** shows that, as we expected, the proposed algorithm was able to correctly detect the three objects.

Figure 6(a) shows the setup 8 which have several objects, 3 of them belonging to the database, which means that the robot has to detect the 3 objects belonging to the database. **Figure 6(b)** shows the superpixels used for segmenting the desired region, **Figures 6(c)-(e)** show the desired probability map of milk tea carton box, as well as yellow and red potatoes packages; finally **Figure 6(f)** shows that proposed system is able to correctly detect the objects belonging to the database.

Figure 7(a) shows the setup 9 which also have several objects, 3 of them belonging to the database which means that the robot has to detect the 3 object belonging to the database. **Figure 7(b)** shows the superpixels used for segmenting the desired region, **Figures 7(c)-(e)** show the desired probability of green tea carton box, orange tea and chipstar, respectively. Finally **Figure 7(f)** shows that the proposed system is able to correctly detect the three objects in the database.

3.2. Evaluation Criterion

In the literature there are different evaluation criterions to assess the local descriptors. Among the most remarkable works it is worth to mention those that operate within the ROC space [31], and those that employs the Recall vs. 1-Precision space [32,33].

In this work the descriptors evaluation was carried out by using the work reported by Mikolajczyk and Shmid [22,23], which employs the recall versus 1-precision criterion. This evaluation criterion is based on the number of correct and false matches obtained for an image pair. The test is based on the number of correct matches and false descriptors obtained from a pair of images. The real positive, *Tp* (true positive) and false positives, *Fp* (false positive) denote the correct and false correlation that were detected by the system. The False negative *Fn* (false negative) and true negatives, *Tn* (true negative) represents the correct and false correlation that were not detected by the system. The descriptor evaluation employs the following parameters:

Precision (*P*): is the fraction of detected region where are the objects belonging to the database.

$$P = \frac{Tp}{Fp + Tp} \tag{7}$$

Recall (*R*): is defined as the fraction of object region that is detected.

$$R = \frac{Tp}{Fn + Tp} \tag{8}$$

F–Measure (*F*): is the harmonic-mean of *P* and *R*.

$$F = \frac{2PR}{P + R} \tag{9}$$

Precision and recall are the basic measurements employed in the evaluation of searching strategies. **Figure 8** shows the matching aspects between *Fn*, *Tp* and *Fn*. *Fn* represents the relevant items that have not been detected. On the other hand, the items that have been detected, but are not relevant, are placed on the right (*Fp*). Recall is obtained from the matching between *Fn* and *Tp*; which determines the fraction of the detected area. The precision is determined by the matching between *Tp* and *Fn*, which determines the fraction of the region where the objects that belong to the same data base are detected.

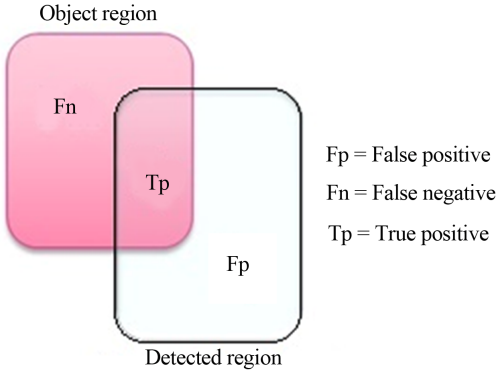


Figure 8. Illustration of meaning of false positive (*Fp*), false negative (*Fn*) and true positive (*Tp*).

In this work the evaluation criteria was applied for the descriptors SURF employing the data set from visual sensor. Note that recall and 1-precision are independent terms. Recall is computed with respect to the number of corresponding regions and 1-precision with respect to the total number of matches.

Table 3 shows the results obtained for the recall (%), precision (%) and F-measure, using the evaluation criteria proposed by Mikolajczyk and Schmid [33], when the proposed algorithm is applied to the experimental setups described in **Table 1**. From these results it follows that the proposed system performs fairly well in most situations, although it has difficulties when it is required to detect plastic bottles. **Table 4** shows a summary of the performance of proposed algorithm using the evaluation criteria proposed by Mikolajczyk and Schmid [33]. The average results obtained for recall is 47.5%, for precision is 79% and for F-measure is 57%, even it appears to be a low detection rate, the result from 10 sceneries show that the proposed system is able to detect 21 from 28 objects, it means that 76.5% objects are correctly detected. The plastic bottle gives the worst F-measure and from **Table 4**, its turns out that the transparent bottles are responsible for the low recall, precision and F-measure rates. **Figure 9** illustrate the difficulties found when it is required to detect plastic bottles.

3.3. Comparison with the Y. Mae *et al.* [8]

The proposed method differs from the method proposed by Mae *et al.* [8] in three main respects: 1) We use for feature extraction the SURF [28] algorithm, while the Mae *et al.* employed the Scale-invariant feature transform (SIFT) [34-36] for this task; 2) To find the best match for each feature we use the LSH [29,30], meanwhile Hough transform [35] was used by [8]; and 3) We use 10 different small objects as carton bottle, plastic bottle, and circular objects at a distance of 1.5 meters, while in the experiments of [8], they used six small static objects. **Figure 10** shows the results obtained using the

Table 3. Detection of each object in terms of the parameters recall, precision and measure.

Object	Recall (%)	Precision (%)	Measure
Orange tea bottle	7.920	5.405	0.0642
Milk carton box	32.88	85.17	0.5470
Green tea carton box	69.50	97.69	0.8043
Red potatoes carton box	47.85	84.93	0.6114
Yellow potatoes carton box	23.21	94.53	0.3703
Chipstar	38.48	95.58	0.5421
Pringles	67.11	83.95	0.6269
Plastic seafood noodle	22.29	31.06	0.1676
Grape tea carton box	80.66	64.16	0.7147
Green tea bottle	37.32	55.42	0.4460

Table 4. Global evaluation in terms of precision, recall and measure.

Characteristic	Recall (%)	Precision (%)	Measure
Plastic bottle	22.62	30.41	0.0642
Cardboard	80.65	87.55	0.8313
Circular object	50.98	96.55	0.7459

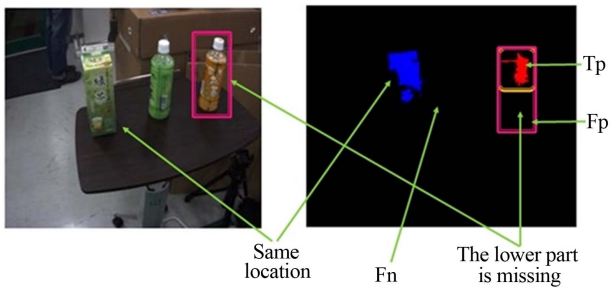


Figure 9. Evaluation of descriptors.

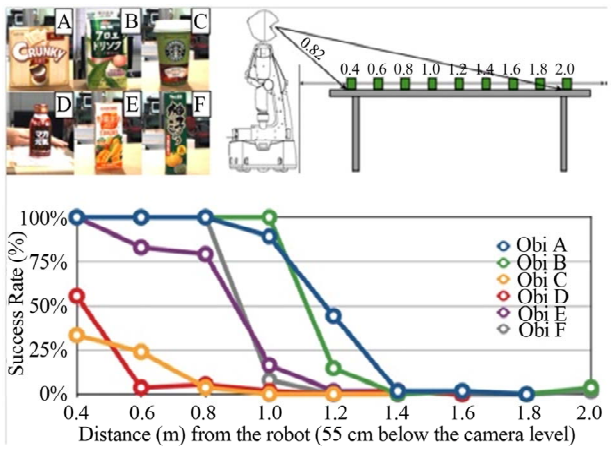


Figure 10. The result of detection for static objects proposed by Mae *et al.* [8].

Mae *et al.* method [8]. As we can see, the object detection can be reliably obtained when they are in close range, but the success rate drops dramatically when it is relatively far. Objects C (carton cup) and D (canned) are small and have round surfaces, and their input images show quite large perspective deformation from the reference images that are taken from a perpendicular viewpoint. As a result, their success rates are dramatically lower than other objects with flat surface. The Object E and F (juice carton box) at a distance of 1m is below about 25%. On the other hand the proposed method with a distance of 1.5 meters between the objects and the robot provides better results even with the plastic bottle that provides the worst results. Thus using the proposed method the object detection can be improved using the method when the distance between the object and robot is larger than 1.3 m.

Using the criteria proposed by Mikolajczyk and Schmid [33] in terms of Precision–Recall we can see that the proposed method provides a fairly good performance with objects located at a distance of 1.5 meters, providing a significant improvement even with small, non flat and circular objects.

4. Conclusion

This paper proposes a novel object detection method using local features inside the superpixel. The proposed algorithm shows that object detection could be improved using SURF features and SLIC superpixel. Our approach can be used in an online robot system for a search task in real environment. Experimental results illustrated the capabilities of the algorithm to detect the object, which is used as the average criteria of evaluation for recall and precision as well as the detection rate. Evaluation results show that the proposed algorithm performs fairly well in the majority of the sceneries, although its performance degrades when it is required to detect transparent plastic objects. Our method exceeds the state of the art on Mae *et al*. [8] for object detection, obtaining better results for objects that are small and have round surfaces, especially when the distance between the object and robot is larger than 1.2 m. In the future work, we propose to increase the database of objects and improve our object detection system using 3D information.

5. Acknowledgements

The authors thank to the National Polytechnic Institute, The University of Electro-communications and the National Council of Science and Technology (CONACYT) for their support during the realization of this research.

REFERENCES

[1] K. Lai, L. Bo, X. Ren and D. Fox, (2011) "A Large-Scale Hierarchical Multi-View RGB-D Object Dataset," *Proceedings of International Conference on Robotics and Automation*, Shanghai, 2011, pp. 1817-1827.

[2] M. Özuysal, V. Lepetit and P. Fua, "Pose Estimation for Category Specific Multiview Object Localization," *Proceedings of International Conference on Computer Vision and Pattern Recognition*, Miami, 20-25 June 2009, pp. 775-785.

[3] H. Harzallah, F. Jurie and C. Schmid, "Combining Efficient Object Localization and Image Classification," *International Conference on Computer Vision* (*ICCV*), Kyoto, 29 September -2 October 2009, pp. 237-244.

[4] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," *Proceedings of International Conference on Computer Vision and Pattern Recognition* (*CVPR*), San Diego, 25 June 2005, pp. 886-893.

[5] G. Bouchard and B. Triggs, "A Hierarchical Part-Based Model for Visual Object Categorization," *Proceedings of International Conference on Computer Vision and Pattern Recognition* (*CVPR*), San Diego, 20-25 June 2005, pp. 710-715.

[6] F. Lafarge, X. Descombe, J. Zerubia and P. Desillingy, "Structural Approach for building Reconstruction from a Single DSM," *IEEE Trans on Pattern Analysis and Machine Intelligence*, Vol. 32, No. 1, 2010, pp. 135-147. doi:10.1109/TPAMI.2008.281

[7] F. Lafarge, X Descombe., J. Zerubia and P. Desillingy, "Structural Approach for Building Reconstruction from a Single DSM," *IEEE Trans on Pattern Analysis and Machine Intelligence*, Vol. 32, No. 1, 2010, pp. 135-147. doi:10.1109/TPAMI.2008.281

[8] Y. Mae, J. Choi, H. Takahashi, K. Ohara, T. Takubo and T. Arai, "Interoperable Vision Component for Object Detection and 3D Pose Estimation for Modularized Robot Control," *Mechatronics*, Vol. 21, No. 6, 2011, pp. 983-992. doi:10.1016/j.mechatronics.2011.03.008

[9] B. Leibe, A. Leonardis and B. Schiele, "Combined Object Categorization and Segmentation with an Implicit Shape Model," *Workshop on Statistical Learning in Computer Vision*, Prague, May 2004, pp. 1-16.

[10] J. Gall and V. Lempitsky, "Class-Specific Hough Forests for Object Detection," *Proceedings of International Conference on Computer Vision and Pattern Recognition* (*CVPR*), Miami, 20-25 June 2009, pp. 1022-1029.

[11] S. Lazebnik, C. Schmid and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories," *Proceedings of International Conference on Computer Vision and Pattern Recognition* (CVPR), New York, 2006, pp. 2169-2178.

[12] J. Shotton, M. Winn, C. Rother and A. Criminisi, "Textonboost: Joint Appearance, Shape and Context Modeling for Multi-Class Object Recognition and Segmentation," *Lecture Notes in Computer Science*, Vol. 3951, 2006, pp. 1-15.

[13] P. Viola and M. Jones, "Rapid Object Detection Using a Boosted Cascade of Simple Features," *Proceedings of International Conference on Computer Vision and Pattern* Recognition (*CVPR*), Vol. 1, 2001, pp. 511-518.

[14] A. Andreas-Opelt and A. Zisserman, "A Boundary-

Fragment-Model for Object Detection," *Lecture Notes in Computer Science*, Vol. 3952, 2006, pp. 575-578. doi:10.1007/11744047_44

[15] J. Ponce, S. Lazebnik, F. Rothganger and C. Schmid, "Toward True 3d Object Recognition," *Proceedings of International Conference on Computer Vision and Pattern Recognition* (*CVPR*), Washington, 2004, pp. 4034-4041.

[16] M. Attami, A. Mizutami, T. Nakamura, T. Nagai, K. Funakoshi and M. Nakano, "Real-Time 3D Visual Sensor for Robust Object Recognition," *Proceedings of International Conference on Intelligent Robots and Systems*, Taipei, 18-22 October 2010, pp. 4560-4565.

[17] D. Hoiem, A. Efros and M. Hebert, "Automatic Photo Pop-Up," *Proceedings of International Conference on Computer Graphics and Interactive Techniques*, (*SIGGRAPH*), Los Angeles, July 2005, pp. 1-8.

[18] Y. Li, J. Sun, C. Tang and H. Shum, "Lazy Snapping," *Proceedings International Conference on Computer Graphics and Interactive Techniques* (*SIGGRAPH*), Los Angeles, 2004, pp. 303-308.

[19] X. He, R. Zemel and D. Ray, "Learning and Incorporating Top-Down Cues in Image Segmentation," *Lecture Notes in Computer Science*, Vol. 3951, 2006, pp. 338-351. doi:10.1007/11744023_27

[20] B. Fulkerson, A. Vedaldi and S. Soatto, "Class Segmentation and Object Localization with Superpixel Neighborhoods," *Proceedings of International Conference on Computer Vision*, (*ICCV*), Nara, 29 September-2 October 2009, pp. 670-677.

[21] X. Ren and J. Malik, "Learning a Classification Model for Segmentation," *Proceedings of International Conference on Computer Vision* (*ICCV*), Nice, 13-16 October 2003, pp. 10-17. doi:10.1109/ICCV.2003.1238308

[22] G. Mori, "Guiding Model Search Using Segmentation," *Proceeding of International Conference on Computer Vision*, (*ICCV*), Las Vegas, 17-21 October 2005, pp. 1417-1423.

[23] P. Felzenszwalb and D. Huttenlocher, "Efficient Graph-Based Image Segmentation," *International Journal of Computer Vision*, Vol. 9, No. 2, 2004, pp. 167-181.

[24] A. Vedaldi and S. Soatto, "Quick Shift and Kernel Methods for Mode Seeking," *European Conference on Computer Vision*, Marseille, 2008, pp. 705-718.

[25] A. Levinshtein, A. Stere, K. Kutulakos, D. Fleet, S. Dickinson and K. Siddiqi, "Turbopixels: Fast Superpixels Using Geometric Flows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 31, No. 12, 2009, pp. 2290-2297. doi:10.1109/TPAMI.2009.96

[26] A. Moore, S. Prince, J. Warrell, U. Mohammed and G. Jones, "Superpixel Lattices," *Proceedings of International Conference on Computer Vision and Pattern Recognition* (*CVPR*), Anchorage, 23-28 June 2008, pp. 1-8.

[27] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua and S. Süsstrunk, "SLIC Superpixels," School of Computer and Communications Sciences, EPFL Technical Report 149300, 2010.

[28] B. Herbert, E. Andreas, T. Tinne and V. Luc, "Speed-Up Robust Features (SURF)," *Computer Vision and Image Understanding*, Vol. 110, No. 3, 2008, pp. 346-359. doi:10.1016/j.cviu.2007.09.014

[29] L. Qin, W. Josephson, Z. Wang and C. Kai-Li, "Multi-Probe LSH: Efficient Indexing for High-Dimensional Similarity Search," *Proceedings of Very Large Database Conference*, Vienna, 23-28 September 2007, pp. 950-961.

[30] S. Har-Peled, P. Indyk and R. Motwani, "Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality," *Theory of Computing*, Vol. 8, No. 1, 2012, pp. 321-350.

[31] O. Miksik and K. Mikolajczyk, "Evaluation of Local Detectors and Descriptors for Fast Feature Matching," *Proceedings of International Conference on Pattern Recognition*, Tsukuba, 2012, pp. 2681-2684.

[32] Y. Ke and R. Sukthankar, "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors," *Workshop on Generic Object Recognition and Categorization*, Washington DC, 27 June-2 July 2004, pp. 506-513.

[33] K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 27, No. 10, 2005, pp. 1615-1630. doi:10.1109/TPAMI.2005.188

[34] http://www.robots.ox.ac.uk/~vgg/research/affine/

[35] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, Vol. 20, No. 1, 2004, pp. 91-110. doi:10.1023/B:VISI.0000029664.99615.94

[36] M. Lopez-de-la-Calleja, T. Nagai and H. Perez-Meana, "Superpixel-Based Object Detection Using Local Feature Matching," *Proceedings of the* 29*th Conference of the Robotics Society of Japan*, Toyosu, 2011, pp. 11-17.

