A Tale of Two Object Recognition Methods for Mobile Robots

Arnau Ramisa[1], Shrihari Vasudevan[2], Davide Scaramuzza[2], Ramón López de Mántaras[1], and Roland Siegwart[2]

[1] Artificial Intelligence Research Institute (IIIA-CSIC), Campus UAB, 08193 Bellaterra, Spain
[2] Autonomous Systems Lab, ETH Zurich, Switzerland

Abstract. Object recognition is a key feature for building robots capable of moving and performing tasks in human environments. However, current object recognition research largely ignores the problems that the mobile robots context introduces. This work addresses the problem of applying these techniques to mobile robotics in a typical household scenario. We select two state-of-the-art object recognition methods, which are suitable to be adapted to mobile robots, and we evaluate them on a challenging dataset of typical household objects that caters to these requirements. The different advantages and drawbacks found for each method are highlighted, and some ideas for extending them are proposed. Evaluation is done comparing the number of detected objects and false positives for both approaches.

1 Introduction

Robots like the Sony Aibo or the Robox are ready to enter our homes, but they lack a lightweight object perception method that allows them to interact with the environment. In order to make robots useful assistants for our everyday life, the ability to learn and recognize objects is of essential importance. For example, in [1] the authors investigate underlying representations of spatial cognition for autonomous robots. Although not specifically addressed in that work, object perception is an essential component that the authors reported to be the most limiting factor. Object recognition in real scenes is one of the most challenging problems in computer vision, as it is necessary to deal with difficulties such as viewpoint changes, occlusions, illumination variations, background clutter or sensor noise. Furthermore, in a mobile robotics scenario a new challenge is added to the list: computational complexity. In a dynamic world, information about the objects in the scene can become obsolete even before it is ready to be used if the recognition algorithm is not fast enough. All these complications make object recognition in real scenes a hard problem, that will demand significant effort in the years to come.

Numerous methods for object recognition have been developed over the last decades, but few of them actually scale to the demands posed by a mobile robotics scenario. Furthermore, most of them concentrate on specific cases, like

A. Gasteratos, M. Vincze, and J.K. Tsotsos (Eds.): ICVS 2008, LNCS 5008, pp. 353–362, 2008.

faces or pedestrians. This paper moves towards object recognition for mobile robots comparing two very popular object recognition techniques suitable to be used in this context. The work itself is aimed at the bigger goal of developing a robust yet lightweight object perception system that can actually be used by mobile robots and meet their hard constraints. Two recent and successful general object recognition approaches include: the constellation method proposed by Lowe together with its interest point detector and descriptor SIFT [2] and a *bag of features* approach, the one developed by Nistér and Stewénius [3]. The authors of both approaches have specifically addressed the issue of computational complexity and claim that proper implementations of their algorithms can recognise a significant number of objects in real time. An object training dataset with different types of object was acquired. Three different categories of objects occurring in typical household environments are considered: textured, untextured and with repetitive textures. Each object has approximately 20 training images and every category consists of three different objects. To evaluate the methods, a test dataset with the same objects was acquired. The test dataset includes occlusions, illumination changes, blur and other typical nuisances that will be encountered while navigating with a mobile robot. Both datasets (training and testing) are available for download[1].

The rest of this work is organized as follows: in Section 2 the methods evaluated are outlined. In order to perform our tests, some modifications had to be done to the bag of features method, they are explained in Section 3. In Section 4 the image dataset and the experiments performed are detailed. Finally, in Section 5 we conclude the article with our findings.

2 Methods

To make this work more self-contained, the two evaluated methods are briefly reviewed in this section. For further information on the object recognition methods used, the reader is referred to [2] for the Lowe method and [3] for the Nistér and Stewénius bag of features approach.

2.1 Lowe Constellation Method

Lowe's object recognition approach is a single view object detection and recognition system with some interesting characteristics for mobile robots, most significant of which is the ability to detect and recognize objects at the same time in an unsegmented image. Another interesting features is the Best-Bin-First algorithm used for approximate fast matching, which reduces the search time by two orders of magnitude for a database of 100,000 keypoints for a 5% loss in the number of correct matches. The first stage of the approach consists on matching individually the SIFT descriptors of the features detected in a test image to the ones stored in the object database using the Euclidean distance. False matches are rejected if the distance of the first nearest neighbor is not distinctive enough when

[1] `http://www.asl.ethz.ch/research/asl/cogniron`

Fig. 1. Matching stage in the Lowe object recognition method

compared with that of the second. In Figure 1, the matching features between a test and model images can be seen. The presence of some outliers can also be observed. Once a set of matches is found, the generalized Hough transform is used to cluster each match of every database image depending on its particular transformation (translation, rotation and scale change). Although imprecise, this step generates a number of initial coherent hypotheses and removes a notable portion of the outliers that could potentially confuse more precise but also more sensitive methods. All clusters with at least three matches for a particular image are accepted, and fed to the next stage: the Iterative Reweighed Least Squares is used to improve the estimation of the affine transformation between the model and the test images.

2.2 Bag of Features Method

The bag of features (or bag of words) approach to object classification comes from the text categorization domain, where the occurrence of certain words in documents is recorded and used to train classifiers that can later recognize the subject of new texts. This technique has been adapted to visual object classification substituting the words with local descriptors such as SIFT [4]. In order to make the local descriptors robust to changes in point of view and scale, local feature detectors are often used to select the image patches that will be used [5], although some authors point that using bigger numbers of randomly selected patches gives better results than a limited number of regions defined around local features [6]. A histogram of descriptor occurrences is built to characterize an image. In order to limit the size of the histogram, a code-book or vocabulary computed applying a clustering method to the training descriptors is used. This code-book should be general enough to distinguish between different descriptor

types but specific enough to be insensitive to small variations in the local patch. Next a multi-class classifier is trained with the histograms of local descriptor counts. In the approach used in this work, the problem of recognizing a large number of objects in an efficient way is addressed. A hierarchical vocabulary tree is used, as it allows to code a larger number of visual features and simultaneously reduce the look-up time to logarithmic in the number of leaves. The vocabulary tree is built using hierarchical k-means clustering, where the parameter k defines the branch factor of the tree instead of the final number of clusters. The signature of an image is a histogram with a length equal to the number of nodes of the tree. For each node i, a histogram bin is computed in the following way:

$$q_i = n_i \omega_i, \tag{1}$$

where n_i is the number of descriptor vectors of the image that have a path through node i of the vocabulary tree, and ω_i is the weight assigned to this node. To improve retrieval performance a measure based in entropy is used for the weights:

$$\omega_i = ln(\frac{N}{N_i}), \tag{2}$$

where N is the number of images in the database, and N_i is the number of images in the database with at least one descriptor vector path through node i. To compare a new query image with a database image, the following score function is used:

$$s(q,d) = \| \frac{q}{\|q\|} - \frac{d}{\|d\|} \| \tag{3}$$

where q and d are the signatures of the query and database image. The normalization can be in any desired norm, but L1-norm was found to perform better. The class of the object in the query image is determined as the dominant in the k nearest neighbors from the database. Two interesting aspects of this approach are that a fast method to compute the scoring of new query histograms using inverted files is proposed in the article, and new images can be added to the database in real-time, which makes this method suitable for incremental learning.

3 Modifications

For the tests, we have used our own implementation of the Lowe schema and a modified version of Andrea Vedaldi's implementation of the Nistér and Stewénius bag of features method[2].

Because of the necessarily broad clusters of the Hough transform, some erroneous matches can still be present and need to be removed. In order to do so a RANSAC step is added to the Lowe approach. RANSAC labels non-coherent

[2] `http://vision.ucla.edu/~vedaldi/code/bag/bag.html`

matches as outliers and, additionally, estimates the most probable affine transformation for every hypothesis given its initial set of matches. Hypotheses that lose matches below three are discarded. The hypotheses that remain after the RANSAC step are reasonable outlier-free and a more accurate model fitting algorithm like IRLS can be used. One of the drawbacks of the bag of features method is that, in contrast to Lowe's constellation method, is designed to work with pre-segmented images. If one image contains more background than a certain threshold, the probability of miss-classification increases. Furthermore, if a particular image contains two objects, there is no way to recognize both. A straightforward solution is to define a grid of overlapping windows with different sizes and shapes covering the whole image. This can be done in an efficient way using a technique similar to that of integral images with the keypoint counts. As most windows from the grid will be selecting areas without any object known by the robot, some technique to reject the false positives is required. In this work we have tested two strategies to this end. The first one consists in introducing a *background* category to which windows selecting no object can be matched, and the second one consists in imposing some constraints to accept the result of a window as a true positive. Among the advantages of the background category method we have that no additional parameters are introduced, and the current schema can be used as is. The drawbacks are that the background category should be general enough to cover all the possible negative windows but specific enough to avoid losing important windows due to changes in point of view or small amounts of background clutter in a good window. Regarding the constraints method, an advantage is that the object database will remain smaller, but some new parameters will be introduced and, if not adjusted wisely, the number of correctly detected objects can decrease as miss-classifications remain at a similar level. To determine the significance of the k nearest neighbors, we have weighted the votes in the following way:

$$D = [\ d_1\ d_2\ d_3\ ...\ d_k] \tag{4}$$

$$W = [\ w_i\ =\ 1 - \frac{d_i}{max(D)}\ |\ \forall d_i \in D\ ], \tag{5}$$

being D the set of distances of the k nearest neighbors and W the set of weights applied to each vote. In our experiments we have used a grid of approximately 60,000 rectangular windows, with side sizes that range from 100 to 500 pixels with steps of 50 pixels, placed every 30 pixels both in vertical and horizontal axis. For the window rejection method we have used the following constraints:

- We are only interested in windows that clearly stand for a category. If the score of the second classified object category is more than λ times the votes of the first object category the window is discarded. In our experiments λ has been set to 0.8.

– Keypoint occurrence statistics can be used to reject windows. A window is only accepted if:

$$N(\tilde{x}_i + 2\sigma_i) > s \tag{6}$$
$$N(\tilde{x}_i - 2\sigma_i) < s \tag{7}$$

for a window labeled as class i, where N is the number of pixels of the window, $\tilde{x}_i$ is the mean number of keypoints per pixel of the class i, σ_i is the standard deviation of the number of keypoints per pixel of class i and s is the number of keypoints found in the current window.

An alternative to the grid of windows is to use a segmentation technique over the image. This strategy has been applied in [7] where the authors report notable improvement in the results. The segmentation technique can be based in pixel intensity or color and also, if stereo images are available, disparity can be used to improve the segmentation. An obvious benefit of using a segmentation technique is a relief in the computational effort of the object recognition step, as the number of considered regions will be much smaller than in the grid of windows approach. More meaningful regions will be used, and therefore the probability of a correct classification will be much higher. The main drawback of using a segmentation technique is that the performance of the object recognition step relies on the quality of the segmentation, and the computational cost of the segmentation step must be added.

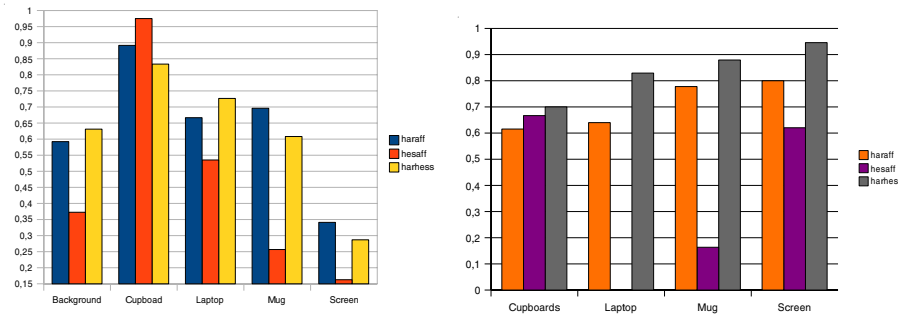
4 Experimental Results

To evaluate the performance of the two methods the following experiments were carried out: first, to validate the suitability of the bag of features method for our purposes, a preliminary test with a dataset of cropped images of objects from our lab has been done. Next, we evaluated both methods using the nine household objects dataset.

4.1 Preliminary Test

The dataset used consists of a hundred training and approximately 125 testing images for each of five categories: cupboard, mug, laptops, screens and background. On-line processing means that images acquired by a mobile robot hardly have a resolution greater than one megapixel, and the object to be detected will probably only occupy the image partially. Additionally, movement often implies blurred images. The capacity to maintain good performance under these circumstances is of key importance for an object recognition system intended to be used in a mobile robot. To test the strength of the bag of features method under these difficulties the test images have been acquired in low resolution and a significant portion of them are blurred.

In this test we compared the performance of three region detectors: Harris Affine (from now haraff), Hessian Affine (hesaff) and a combination of Harris



(a) Ratio of correct classifications with cropped images

(b) Ratio of false negatives among errors

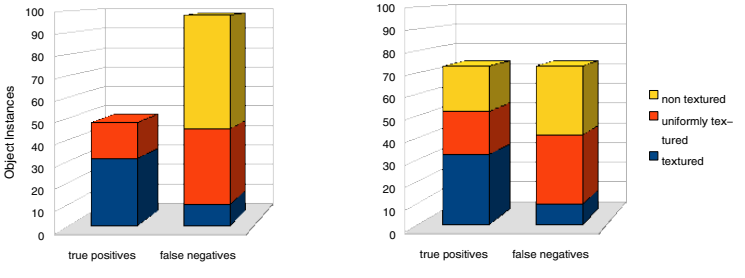
Fig. 2. Results of bag of features test with manually cropped images

Laplace and Hessian Laplace without rotation invariance (harhes). All these region detectors are described in [5] and are known to produce regions with a hyphenate repeatability rate. In Figure 2(a), it can be observed that the ratio of correct classifications is over 60% for all classes except for the screens category. Furthermore, as can be seen in Figure 2(b), the majority of the errors where false negatives (objects confused with background). On average, haraff classified correctly 63.8% of the test images, harhes 61.7% and hesaff only 45.4%. Based on these results it was decided to use Harris Affine in the subsequent experiments. Taking into account the characteristics of the test images used, the results obtained are good enough to take the method into consideration.

4.2 Comparison of Methods

The main purpose of this work is to evaluate two candidate object recognition algorithms to be used in an indoor mobile robot. To this end, we have created an image dataset of nine typical household objects. These objects are divided in three categories (three objects per category): textured, untextured and textured but with repetitive patterns. In Figure 4.2 one training object from each category can be seen. Additionally, 36 test stereo pairs were acquired with the robot cameras, a STHMDCS2VAR/C stereo head by Videre design. Each test image contains one or more instances of objects from the dataset, some of them with illumination changes or partial occlusions. The training images have been taken with a standard digital camera at an original resolution of 2 megapixels, although the region of interest has been cropped for every image. The test images where acquired at 1.2 megapixels, the maximum resolution allowed by the stereo head. In this experiment we are not testing neither for viewpoint invariance nor for intraclass variation as the Lowe constellation approach does not effectively handle these kinds of transformations. For comparison purposes, two versions of the training images have been considered: a rectangular cropped version of the object, that inevitably includes some background texture, and a precise

Fig. 3. Images from the dataset. First column corresponds to objects with repetitive texture, second to textured objects and third to non-textured objects.



(a) Lowe constellation method (b) Bag of features method

Fig. 4. Results of the comparison

segmentation of the object boundaries. Using the Lowe approach, the method that worked best is the rectangular cropped version of the training images. Almost all of the textured object occurrences and some instances of the uniformly textured book have been detected (Figure 4(a)). However none of the non-textured objects were recognized. The best performance of the bag of features approach has been achieved using the window rejection method (Figure 4(b)).

Table 1. Results for the two bag of words approaches evaluated. False positives and miss-classifications are combined.

	Considered Windows	True Positives	False Positives	False Negatives	True Negatives
Background category	43228	3.76%	17.02%	32.1%	47.12%
Window filtering	18825	12.79%	87.21%	–	–

As can be seen, the number of detected objects, especially for the categories of uniformly textured and non-textured, has increased using this approach.

As expected, the main drawbacks of the bag of features approach using the grid of windows is the amount of false positives, suggesting that posterior verification stages based, for example, in feature geometry inside the considered window should be introduced. In the background category approach, the best results were obtained with the manually segmented training dataset, while in the window filtering approach the cropped training images produced better results. The percentage of true positive and false positive windows for both approaches can be seen in Table 1. At a posterior stage, overlapping windows must be combined to yield object hypotheses. In spite of being more effective than the Lowe method in objects without texture, the bag of features has some drawbacks that need to be properly addressed. Its main problem is the pre-segmentation of the image that is required to recognize objects. Regarding computational complexity, our C++ implementation of the Lowe method takes approximately one second per image while the matlab implementation of the bag of features method complexity varies hugely upon scene content, and can take from 38 to 7 minutes. All tests where done in a P3 with 1Ghz of memory running a Linux operating system.

5 Conclusions

In this work we have addressed the problem of object recognition applied to mobile robots. Two state-of the-art object recognition approaches are compared. A dataset of nine typical household objects is used for the tests. The dataset incorporates typical problems that would be experienced by object recognition systems being deployed in home environments. This includes lack of texture and repetitive patterns. Further, the test images were acquired with the vision system of our robot instead of a good quality digital camera. The experiments presented in this report conclude that the bag of features method combined with a grid of windows over the image is able to detect poorly textured objects in low quality images typical of a mobile robot scenario. However, this method still lacks a fast yet robust way to reject false positives and a technique to reliably fuse multiple overlapping windows, each representing one potential hypothesis of the object occurring in the image, towards a single occurrence of the particular object. As an alternative to the computationally expensive windowing strategies, an image segmentation strategy is proposed. This method could improve results by reducing background clutter. However, this is a "two-edged sword" in that the object

recognition quality is greatly influenced by that of the image segmentation. A verification stage based on feature geometry is proposed as a method to reject unlikely hypothesis.

Future work includes evaluating different strategies for improving the grid of windows method as well as testing different image segmentation strategies. A fast implementation of the bag-of-features approach that includes efficient scoring methods (such as those employed in [3]) is being developed. Future work would also test the robustness of the bag of features method to intra-class variation and viewpoint change in this context.

Acknowledgements

This work has been partially funded by the FI grant and the BE grant from the AGAUR, the European Social Fund, the 2005/SGR/00093 project, supported by the Generalitat de Catalunya, the MIDCBR project grant TIN 200615140C0301, TIN 200615308C0202 and FEDER funds. The authors would also like to thank Andrea Vedaldi for making available his implementation of the Nistr and Stewnius bag of words algorithm.

References

1. Vasudevan, S., Gachter, S., Nguyen, V., Siegwart, R.: Cognitive maps for mobile robots - an object based approach. In: Robotics and Autonomous Systems, May 31, 2007. From Sensors to Human Spatial Concepts, vol. 55(5), pp. 359–371 (2007)
2. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Interantional Journal of Computer Vision 60(2), 91–110 (2004)
3. Nister, D., Stewenius, H.: Scalable recognition with a vocabulary tree. Conf. Computer Vision and Pattern Recognition 2, 2161–2168
4. Csurka, G., Bray, C., Dance, C., Fan, L.: Visual categorization with bags of keypoints. In: Workshop on Statistical Learning in Computer Vision, ECCV, pp. 1–22 (2004)
5. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A comparison of affine region detectors. International Journal of Computer Vision 65(1/2), 43–72 (2005)
6. Nowak, E., Jurie, F., Triggs, B.: Sampling Strategies for Bag-of-Features Image Classification. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 490–503. Springer, Heidelberg (2006)
7. Rabinovich, A., Vedaldi, A., Galleguillos, C., Wiewiora, E., Belongie, S.: Objects in context. In: Proceedings of the International Conference on Computer Vision (ICCV) (2007)