Robust Object Recognition using Wide Baseline Matching for RoboCup Applications*

Patricio Loncomilla[1,2] and Javier Ruiz-del-Solar[1,2]

[1] Department of Electrical Engineering, Universidad de Chile
[2] Center for Web Research, Department of Computer Science, Universidad de Chile
{jruizd, ploncomi}@ing.uchile.cl

Abstract. As the RoboCup leagues evolve, higher requirements (e.g. object recognition skills) are imposed over the robot vision systems, which cannot be fulfilled using simple mechanisms as pure color segmentation or visual sonar. In this context the main objective of this article is to propose a robust object recognition system, based on the wide-baseline matching between a reference image (object model) and a test image where the object is searched. The wide baseline matching is implemented using local interest points and invariant descriptors. The proposed object recognition system is validated in two real-world tasks, recognition of objects in the RoboCup @Home league, and detection of robots in the humanoid league.

1 Introduction

In the RoboCup soccer leagues robot vision systems are mostly based on basic color segmentation algorithms, and in some cases on the use of visual sonar (analysis of scan lines) for detecting lines. The main advantage of these vision mechanisms is their high processing speed. However, as the soccer leagues evolve, higher requirements are imposed over the vision systems, which cannot be fulfilled using those simple vision mechanisms. For instance, nowadays some teams are looking for advanced features such as: use of natural landmarks without geometrical and color restrictions, pose independent detection and recognition of teammates and opponents, detection of the teammates and opponents pose, automated refereeing tools, etc. Neither of those features can be achieved by pure color segmentation and/or using a visual sonar. Moreover, some non-soccer leagues (e.g. @Home) require robust, fast, easy trainable and general-purpose object recognition methodologies for recognizing complex objects like newspapers, bottles and soda cans (see @Home 2007 rules definition in [18]). In some tests, the object detector must be trained in runtime using only a few images as it cannot be trained before the test starts (i.e. the "lost & found" @Home test).

In this context, the main objective of this article is to propose a robust and versatile object recognition system, based on the wide-baseline matching between a reference image (object model) and a test image where the object is searched. Under this paradigm, local interest points (local maxima/minima in a filtered image set) are

* This research was funded by Millennium Nucleus Center for Web Research, Grant P04-067-F, Chile.

extracted independently from both the test and the reference image, then characterized using invariant descriptors (each one describes the gradient distribution in a region around an interest point), and finally several matches between similar descriptors from both images are used to get an affine transformation between the two images. Several verification stages are introduced to test the correctness of the transformation. If the object model has a known pose in the reference image, the obtained transformation allows determining the object's pose in the test image.

Object recognition based on wide-baseline matching has the following desired features: (i) no training requirements: only one image for each relevant view of the object is required; (ii) general purpose: any given object can be recognized, given that an example image of that object is available; and (iii) near real-time operation: depending on the exact characteristics of the implemented system and in the number of object classes, a processing speed of up to 3-9 frames per second can be achieved.

In the paper we describe the implemented object recognition system (section 2), and we show its use for recognizing objects in the RoboCup @Home league (section 3), and for detecting robots in the humanoid league (section 4). Finally, some conclusions of this work are given in section 5.

2 Object Recognition based on Wide Baseline Matching

In the wide baseline matching problem formulation, the images to be compared are allowed to be taken from widely separated viewpoints, so that a point in one image may have moved anywhere in the other image, generating a hard matching problem.

Wide baseline matching approaches have become increasingly popular, experiencing an impressive development in the last years [1][4][9][12][16]. Local interest points are extracted independently from both a test and a reference image, characterized using invariant descriptors, and finally the descriptors are matched. By processing the matches, a transformation between the images is obtained.

The most employed interest point detectors are the single-scale Harris detector [2] and the multi-scale Lowe's sDoG+Hessian detector [4]. The best performing interest point detectors are the Harris-Affine and the Hessian-Affine [11], but they are slow for runtime applications. In the other hand, the most popular and best performing descriptor [10] is the SIFT (Scale Invariant Feature Transform) [4].

Lowe's system [3][4] uses the SDoG+Hessian detector, SIFT descriptors, a Hough transform to accumulate evidence from the matches for the possible similarity transformations, and a probability test to discard Hough transform bins which have few votes (then they could be generated only by random matches). This system has great recognition capabilities and near real-time operation. However, Lowe's system main drawback is the use of just a simple voting-based probabilistic hypothesis rejection stage, which cannot successful reduce the number of false positives when the true positive detection rate is prioritized. This is a serious problem in real world applications as, for example, robot self-localization [14], robot head pose detection [5] or image alignment for motion detection in video [15]. In [6][7] we proposed a system that reduces largely the number of false positives by using several hypothesis-based rejection stages. In this work, we extend this system by including the following new features: a fast probabilistic hypothesis rejection stage, a new linear correlation verification stage, a better organization of the hypothesis rejection tests

into several stages, and the use of the RANSAC algorithm and a semi-local constraints test. Although RANSAC and the semi-local constraints tests have being use by many authors, Lowe's system does not use them. The proposed system is described in the following subsections.

2.1 Generation of the matches between SIFT descriptors for each image pair

Local descriptors (SIFT descriptors) are extracted from both images, and matches between pairs of these descriptors belonging to different images are generated. This process is described in detail in [5][4]

2.2 Transformation Computation and Hypothesis Rejection Tests

This computation method (L&R – Loncomilla & Ruiz-del-Solar) considers several stages that are described in the next paragraphs.

1. Similarity transformations are determined using the Hough transform (see description in [3]). After the Hough transform is computed, a set of bins, each one corresponding to a similarity transformation, is determined. Then:
   - **a.** Invalid bins (those that have less than 4 votes) are eliminated.
   - **b.** Q is defined as the set of all valid candidate bins, the ones not eliminated in 1.a.
   - **c.** R is defined as the set of all accepted bins. This set is initialized as a void set.

2. For each bin B in Q the following tests are applied (the procedure is optimized for obtaining high processing speed by applying less time consuming tests first):
   - **a.** If the bin B has a direct neighbor in the Hough space with more votes, then delete bin B from Q and go to 2.
   - **b.** Calculate r_{REF} and r_{TEST}, which are the linear correlation coefficients of the interest points corresponding to the matches in B that belong to the reference and test image. If the absolute value of any of these two coefficients is high, delete bin B from Q and go to 2. This numerical-robustness verification stage is explained in detail in the appendix.
   - **c.** Calculate the fast probability P_{FAST} to B. If P_{FAST} is lower than a threshold P_{TH1}, delete bin B from Q and go to 2. This probability test is described in [7].
   - **d.** Calculate an initial affine transformation T_B using the matches in B.
   - **e.** Compute the affine distortion degree of T_B using a geometrical distortion verification test (described in [5]). If T_B has a strong affine distortion, delete bin B from Q and go to 2.
   - **f.** Top down matching: Matches from all the bins in Q who are compatible with the affine transformation T_B are cloned and added to bin B. Duplication of matches inside B is avoided.
   - **g.** Calculate Lowe's probability of bin B (see description in [3]). If this probability is lower than a threshold P_{TH2}, delete bin B from Q and go to 2.
   - **h.** Apply RANSAC for finding a more precise transformation. In case that RANSAC success, a new transformation T_B is calculated.
   - **i.** Accept the candidates B and T_B, what means delete B from Q and include it in R (the T_B transformation is accepted).

3. For all pairs (B_i, B_j) in R, check it they may be fused into a new bin B_k. If the bins may be fused and one of them is RANSAC-approved, do not fuse them and delete the other in order to preserve accuracy. If the two bins are RANSAC-

approved, delete the least probable. Repeat this until all possible pairs (including the new created bins) have been checked. This fusion procedure is described in [5].

4. For any bin B in R, apply semi-local constraints procedure to all the matches in B. The matches from B who are incompatible with the constraints are deleted. If some matches are deleted from B, T_B is recalculated. This procedure is described in [13].
5. For any bin B in R, calculate the pixel correlation r_{pixel} using T_B. If r_{pixel} is below a given threshold t_{corr}, delete B from R. This correlation test is described in [6].
6. Assign a priority to all the bins (transformations) in R. A more probable bin (in the Lowe's probability sense) has better priority than a less probable one, but any RANSAC-approved bin has better priority than any non RANSAC-approved one.

3 Solving RoboCup @Home Tests

The RoboCup @Home league defines seven tests to be solved in the 2007 competitions [18]. In three of them, complex and versatile visual object recognition abilities are required:

- In the "Lost & Found" test, an object is shown just one time to the robot, then the object is hidden somewhere in the environment and the robot should be able to find it within a limited amount of time [18].
- In the "Manipulate" test the robot must manipulate some specified objects (open a door, a refrigerator, get a soda can, grab a newspaper, etc) [18].
- In the "Navigate" test the robot has to safely navigate toward some specified objects in a living room environment [18].

These three tests put the following requirements to the object recognition system:

- General purpose. The objects to be recognized are of different types and in general complex: a TV, a door handle, a newspaper, a soda can, a bottle. Therefore a general-purpose object recognition system is required.
- No/Less training. In at least one of the test ("lost & found"), the objects to be recognized are not known by the robot before the test starts, while in the other two cases, the objects are not known by the participants before the RoboCup competitions start. Therefore, just one or two images of each object should be enough for a fast training and a robust characterization of the objects.
- Near real-time processing. The tests need to be solved in a short time, and for solving them the object recognition system need to be applied several times (e.g. hundreds of frames before finding an object in an arbitrary position in a complex environment).Then, the images need to be analyzed in near real-time.

These three requirements can be fulfilled using an object recognition system based on wide baseline matching, as the one described in the former section. As mentioned, this object recognition system outperforms similar ones terms of recognition rate, number of false positives and speediness, as it is shown in [7]. Therefore it will be used for implementing object recognition in the RoboCup @Home tests.

We implemented the described object recognition system in our RoboCup @Home robot [19]. We have carried out several experiments for solving the "Manipulate", "Navigate", and "Lost & Found" tasks, concentrating ourselves in solving the corresponding object recognition subtask. Some examples of object recognition, when the robot looks for different objects in different frames, are shown in figure 1. As in can be observed, our object recognition systems can successfully recognize in

cluttered backgrounds a wide variety of objects which can appear in the *lost & found*, the *manipulate* and the *navigate* @Home tests.

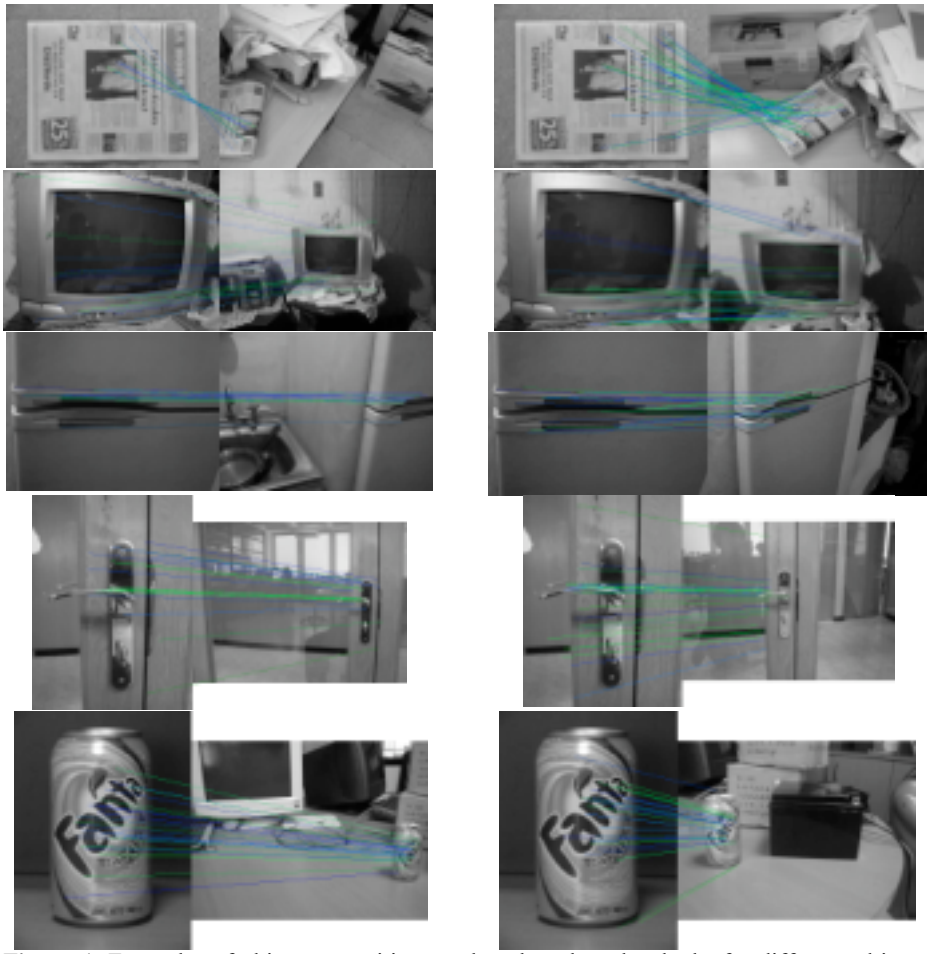

Figure 1. Examples of object recognition results when the robot looks for different objects in different frames. In each case is shown the pair reference (left) - test (right) image.

4 Robot Detection in the RoboCup Humanoid League

In the RoboCup soccer competitions, the detection of teammates and opponent robots present in the scene is a key skill for good playing (e.g. passing, robot avoidance, goal kicking). Most existing vision systems, which use colors and depend on the illumination conditions, are not robust enough for solving this task. We aim at reverting this situation by using the L&R system in the detection of soccer robots, specifically humanoid robots.

We carried out several tests using our humanoid Hajime HR18 robot [20], and real video sequences processed in a notebook. The results are summarized in table 1. As it can be observed acceptable detection rates are obtained, however the processing speed should be increased, because in the humanoid league most of the robots are equipped with low-speed Pocket PCs as main processors (not notebooks). One possibility for achieving this reduction is applying this detector not in each frame, or using features that can be evaluated in less time (e.g. SURF [16]).

For exemplifying the detection of humanoid robots, in figure 2 we show some video frames where the robot is successfully matched against the reference image.

Table 1. Detection of a humanoid Hajime HR18 robot, 221 frames. Results were obtained with the system running in a notebook core-duo @ 1.66 GHz, 1GB RAM, running Windows XP.

Flavor	DR (%)	Number of False Positives	Processing Speed (fps)
Original image size: 320x240,	80.1%	14	4.4
Sub-sampled image: 240x170	75.1%	7	4.7
Sub-sampled image: 160x120	64.3%	3	11.5

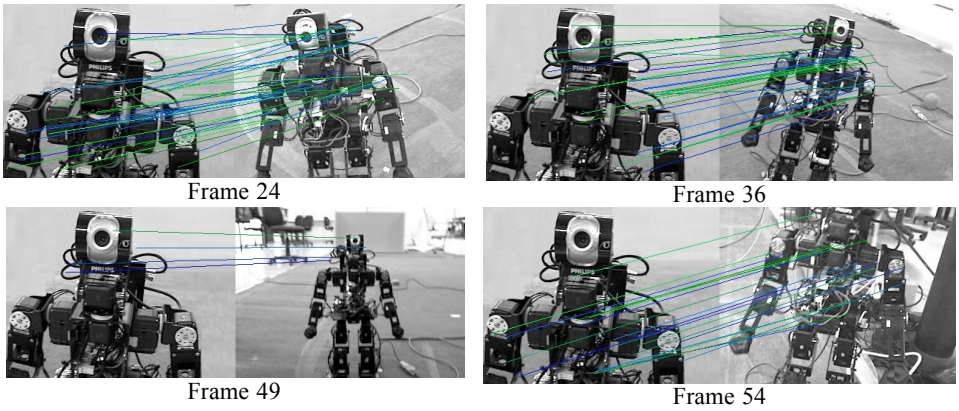

Figure 2. Some examples of humanoid robot detection in a video sequence. In each frame is shown the pair reference image (left) - test image (right).

5 Conclusions

In this article we have described a robust object recognition system, based on the wide-baseline matching between a reference image (object model) and a test image where the object is searched. The wide baseline matching is implemented using local interest points (sDoG+Hessian detector) and invariant descriptors (SIFTs). The main novelty of the described system is the inclusion of several hypothesis rejection tests that reduces largely the number of false positives, allowing the use of the system in real-world applications.

The proposed object recognition system is validated in two real-world tasks, recognition of objects in the RoboCup @Home league, and detection of robots in the humanoid league. The obtained results are satisfactory in terms of detection rate and number of false positives, although for an application in the humanoid league, where

most teams employ Pocket PCs as main processors, the processing speed of the system should be increased. We are working in this direction using some novel features that can be evaluated in less time, as for example SURF features [16].

References

1. Ferrari, V., Tuytelaars, T., Van Gool. L., 2004. Simultaneous Object .Recognition and Segmentation by Image Exploration. *Lecture Notes in Computer Science* 3021 (*ECCV 2004*), Springer, 40 - 54.
2. Harris, C., Stephens, M., 1998. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conf.*, 147-151, Manchester, UK.
3. Lowe, D., 2001. Local feature view clustering for 3D object recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, Hawaii, 682-688.
4. Lowe, D., 2004. Distinctive Image Features from Scale-Invariant Keypoints. *Int. Journal of Computer Vision*, 60 (2): 91-110, Nov. 2004.
5. Loncomilla, P., Ruiz-del-Solar, J., 2005. "Gaze Direction Determination of Opponents and Teammates in Robot Soccer". Lecture Notes in Computer Science 4020, pp. 230-242, Springer.
6. Loncomilla, P., Ruiz-del-Solar, J., 2006. "Improving SIFT-based Object Recognition for Robot Applications". Lecture Notes in Computer Science 3617, pp. 1084 – 1092, Springer.
7. Loncomilla, P., Ruiz-del-Solar, J., 2006. "A Fast Probabilistic Model for Hypothesis Rejection in SIFT-Based Object Recognition", Lecture Notes in Computer Science 4225, pp. 696-705, Springer.
8. Ruiz-del-Solar, J., Loncomilla, P., Vallejos, P., 2006. "An Automated Refereeing and Analysis Tool for the Four-Legged League". Lecture Notes in Computer Science, Springer.
9. Mikolajczyk, K., Schmid, C., 2004. Scale & Affine Invariant Interest Point Detectors. *Int. Journal of Computer Vision*, 60 (1): 63 - 96, Oct. 2004.
10. Mikolajczyk, K., Schmid, C., 2005. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Machine Intell.* 27 (10), 1615 – 1630.
11. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L., 2005. A Comparison of Affine Region Detectors. *Int. Journal of Computer Vision* 65 (1-2), 43-72.
12. Schaffalitzky, F., Zisserman, A., 2003. Automated location matching in movies. *Computer Vision and Image Understanding* 92 (2-3), 236 – 264.
13. Schmid, C., and Mohr, R., 1997. Local grayvalue invariants for image retrieval. *IEEE Trans. Pattern Anal. Machine Intell.* 19(5), 530-534.
14. Se, S., Lowe, D., Little, J., 2002. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *Int. Journal of Robotics Research* 21 (8) 735 – 758.
15. Vallejos, 2007. Left blank for blindness purposes.
16. Herbert Bay, Tinne Tuytelaars and Luc Van Gool 2006. SURF: Speeded Up Robust Features, *Proc. of the 9th European Conference on Computer Vision*, Springer LNCS 3951, 404–417,.
17. UCH100 database. Electronically available in http://vision.die.uchile.cl/
18. RoboCup @Home 2007 competition rule book. Electronically available in http://www.ai.rug.nl/robocupathome/documents/rulebook.pdf.
19. Javier Ruiz-del-Solar, Simón Norambuena, Fernando Bernuy, Sebastián Cubillos, Mauricio Mascaró, Ignacio Olavaria, Cristián Solís, Carlos Toro, Juan Vargas. UChile HomeBreakers 2007 Team Description Paper. Available in http://www.robocup.cl

20. Javier Ruiz-del-Solar, Paul Vallejos, Isao Parra, Javier Testart, Pablo Ravest, Rodrigo Briones, María Isabel Avilés. UChile RoadRunners 2007 Team Description Paper. Available in http://www.robocup.cl

Appendix: Linear correlation test

An affine transformation can be calculated from a set of matches between points (x, y) in the reference image and points (u, v) in the test image. The affine transformation can be represented in the following two ways:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_X \\ t_Y \end{pmatrix} \Rightarrow \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} x & y & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x & y & 1 \end{pmatrix} \begin{pmatrix} m_{11} & m_{12} & t_X & m_{21} & m_{22} & t_Y \end{pmatrix}^T \quad (1)$$

From the last expression, and using least squares, the parameters of the transformation can be calculated from matches between points (x_i, y_i) and (u_i, v_i):

$$\begin{pmatrix} m_{11} \\ m_{12} \\ t_X \\ m_{21} \\ m_{22} \\ t_Y \end{pmatrix} = \left(X^T X\right)^{-1} X^T \begin{pmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ \dots \end{pmatrix} \quad ; \quad X = \begin{pmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 0 \\ x_2 & y_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & y_2 & 1 \\ \dots & \dots & \dots & \dots & \dots & \dots \end{pmatrix} \quad (2)$$

The parameters are calculable only if the 6-by-6 $X^T X$ matrix is invertible, and this is possible only if X has rank 6. If the points in the reference image lay on a straight line, the relations $y_K = a\, x_K + b$ holds, then the second and fifth columns in X become linearly dependent, and the matrix X gets at most rank 4. Then, if the points in the reference image lay on a straight line, the parameters of a transformation from the reference to the test image cannot be successfully calculated. In the symmetric case, if the points in the test image lay on a straight line, a transformation from the test to the reference image cannot be calculated. Then, to get a numerically-stable and invertible transformation, the points in the reference and the test image cannot lie on a straight line, i.e., the correlation coefficients of the points in both images must be low. Then the following test can be done to reject numerically unstable transforms:

1. Calculate r_{REF}, the linear correlation of the interest points in the reference image
2. If r_{REF} > threshold, reject the transformation
3. Calculate r_{TEST}, the linear correlation of the interest points in the test image
4. If r_{TEST} > threshold, reject the transformation