Stereo Obstacle Detection Method for a Hybrid Omni-directional/Pin-Hole Vision System

Giovanni Adorni[2], Luca Bolognini[1], Stefano Cagnoni[1], and Monica Mordonini[1]

[1] Department of Computer Engineering, University of Parma
[2] Department of Communication, Computer and System Sciences, University of Genoa

Abstract. This paper describes an application of a vision system based on the use of both an omnidirectional vision sensor and a standard CCD camera. Such a hybrid sensor permits implementation of peripheral/foveal vision strategies, that can be very useful for navigation tasks. This paper focuses particularly on the use of the device as a stereo vision system to locate obstacles in a semi-structured environment through a perspective removal algorithm.

1 Introduction

Omni-directional vision systems are usually based on a catadioptric sensor, consisting of an upwards-oriented camera that acquires the image reflected on a convex mirror hanging above it [1, 2]. Such systems are very efficient as concerns target detection, but critical from the point of view of the accuracy with which the target is detected. For these reasons, in several cases, the omni-directional vision sensor has been integrated with different kinds of sensors (see, for example [3, 4, 5]). In this paper we present results obtained using a vision system prototype (called HOPS, Hybrid Omnidirectional/Pin-hole Sensor), that consists of an omni-directional sensor coupled with a standard CCD camera. HOPS was designed with the main intent to assist navigation tasks. This paper focuses on an obstacle detection method that was implemented for use with the hybrid sensor.

HOPS (see figure 1) integrates omnidirectional vision with traditional pin-hole vision, to overcome the limitations of the two approaches. It can be best described as a color CCD camera mounted on top of an omnidirectional catadioptric sensor. Its lower part consists of an omnidirectional sensor that consists of a color CCD camera pointing upwards at a convex mirror (see [6] for details). The omnidirectional sensor provides a base for the more traditional CCD-camera based sensor that leans on it. The CCD camera is fixed, looks down with a tilt angle of 60^o with respect to the ground plane and has a field of view of about 80^o. For the application described in this paper, to allow for a higher stereo disparity between the two images, it is positioned slightly off the center of the device. An example of the images that can be obtained by the two sensors is provided in fig. 1.

2 Hybrid Stereo Vision for Obstacle Detection

Inverse perspective mapping for obstacle detection was first introduced in [7]. If one applies the inverse perspective mapping transform with respect to the same plane to a

A. Birk, S. Coradeschi, and S. Tadokoro (Eds.): RoboCup 2001, LNAI 2377, pp. 244–250, 2002.

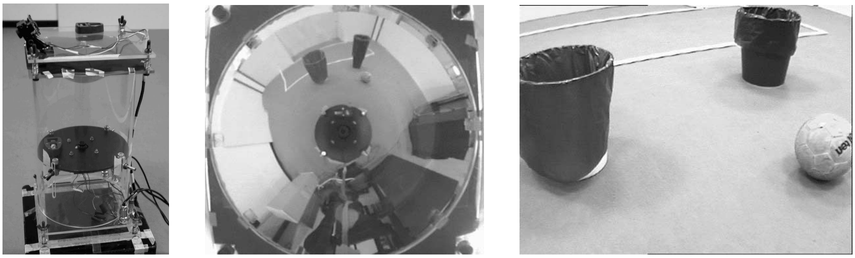

Fig. 1. The HOPS prototype (left) and an example of images that can be acquired through the omni-directional sensor (center) and the top CCD camera (right).

pair of stereo images, everything that lies on that plane looks the same in both views, while everything that does not is distorted differently. This property is particularly useful for tasks in which a relevant reference plane can be easily found. The steps required to detect obstacles based on stereo vision include: i) applying the inverse perspective transform to the two images; ii) subtracting one transformed image from the other one to detect differences; iii) labelling pixels of at least one of the acquired images as free space or obstacles.

Stereo vision is usually obtained by two cameras slightly displaced from each other or by a single moving sensor that can move to obtain the same effect. In the case of HOPS such a displacement is not constant, since the angle between the optical axis of the camera and the plane that is tangent to the mirror surface is different for each point of the mirror. If the traditional camera were positioned on the axis of the omnidirectional sensor, disparity would be close to zero in a few points. Positioning the pin-hole sensor off the omnidirectional sensor axis ensures that a different perspective be obtained from the two sensors for all points.

The problem dealt with by the inverse perspective transform (IPT) consists of computing a correspondence function $P_{x,y} = \mathcal{C}(I_{i,j})$ that maps the pixels belonging to image I onto points of a new image P that shows a bird's view of a reference plane. From the view obtained by applying $\mathcal{C}(I)$ information about the relative positions of objects (e.g., distances) that lie on the plane can be extracted.

If all parameters related to the geometry of the acquisition systems and to the distortions introduced by the camera were known, the derivation of $\mathcal{C}$ could be straightforward [7, 8]. However, this is not always the case, most often because of the lack of a model of the camera distortion. However, assuming an ellipsoidal model for the camera lens usually allows for an easy empirical derivation of the related parameters anyway.

In computing $\mathcal{C}_o$, the IPT for the catadioptric omnidirectional sensor, while, on one side, the problem is complicated by the non-planar profile of the mirror, on the other side the simmetry of the device is such that it is enough to compute the restriction of $\mathcal{C}_o$ along a radius of the mirror projection on the image plane to be able to compute the whole function. However, in doing so, one must take into account possible manufacturing flaws, that affect both the shape of the mirror and the smoothness of the mirror surface. In addition to "regular" flaws, that can be found all over the surface and can be included in the radial model of the mirror, a few other minor "random" flaws can be found. These

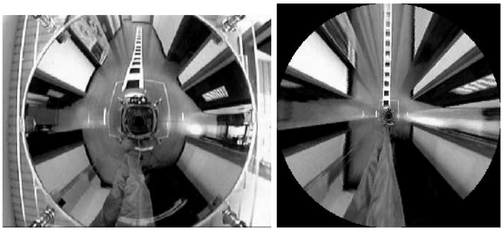

Fig. 2. Results of perspective removal (right) from the omnidirectional image (left).

require that the IPT be derived empirically, or at least that the ideal model be corrected in flawed points.

A first empirical derivation of $\mathcal{C}_o$ can be made by considering a set of equally-spaced radii, on each of which values of $\mathcal{C}_o$ are computed for a set of uniformly-sampled points. To obtain the values of the function for a generic point P_i located anywhere in the field of view of the sensor, a bi-linear interpolation is made between the four points among which P_i is located, within a uniformly-sampled polar grid. This makes reconstruction accuracy better close to the robot, as the area of the cells used for interpolation increases with radial distance. Reconstruction quality increases with the number of radii, as the angular sampling step of the grid decreases. However, distortions can still be noticed. To limit the problem, a sixth-grade approximating polynomial is used to make the restriction of $\mathcal{C}_o$ along each radius smoother. The approximating models are "manually corrected" in the neighborhood of interpolating points in which the approximation error is above a tolerance threshold, taking care not to affect smoothness. Results obtained using eight radii are shown in figure 2.

After applying the IPT, two "bird's view" images are obtained that can be compared to detect obstacles. However, differences in the two sensors require that the two IPT images be pre-processed. In particular, the two images differ in their pictorial features. Since the region of interest is the same for both sensors, after identifying the region $\mathcal{R}$ of the omnidirectional image O that matches the field of view of the image P acquired by the upper camera, a punctual transformation is applied to O, such that

$$\sum_0^n H_{o,col}(l,\mathcal{R}) = \sum_0^n H_{p,col}(l,P) \quad \forall n, col,$$

$H_{o,col}(l,\mathcal{R})$ and $H_{p,col}(l,P)$ being the discrete histograms of the *col* (*col* $\in R, G, B$) component for region $\mathcal{R}$ of O and for P, respectively. The variable l represents the intensity level of the color component. Since the input images are color images, the result of the difference operator must combine information from all three color components. We use a max operator followed by a thresholding to do so. Each pixel value of the resulting binary image D is computed as:

$$D_{i,j} = \begin{cases} 0 \text{ if } \max(|R_{i,j}(\mathcal{R}) - R_{i,j}(P)|, \\ \qquad\quad |G_{i,j}(\mathcal{R}) - G_{i,j}(P)|, \\ \qquad\quad |B_{i,j}(\mathcal{R}) - B_{i,j}(P)|) \geq T \\ 1 \text{ otherwise} \end{cases}$$

To make the algorithm less sensitive to light changes, the threshold T is a linear function of the average lighting intensity of the acquired images. The difference image

D computed as described is still affected by the presence of noise, either in the form of isolated white pixels and thin lines, or of small gaps in the significant regions. These artifacts are corrected using a majority-vote morphologic filter.

3 Obstacle Detection with HOPS

The white regions in D derive from two kinds of discrepancies. If they derive from different pan angles or from a lateral displacement of the two cameras, they are located to the left or right of obstacle projections in the IPT transformed images. With different tilt angles or vertical displacements of the cameras, they are above and/or below the obstacle projections. For HOPS aplications the reference plane is the ground plane on which a robot that mounts the sensor is moving.

Thus: i) lateral differences are almost symmetrical and not very large; ii) no differences can be detected at the base of obstacles, as obstacle bases lie on the reference plane and their transforms are identical there; iii) vertical differences are more relevant near obstacle top, and somehow proportional to obstacle height.

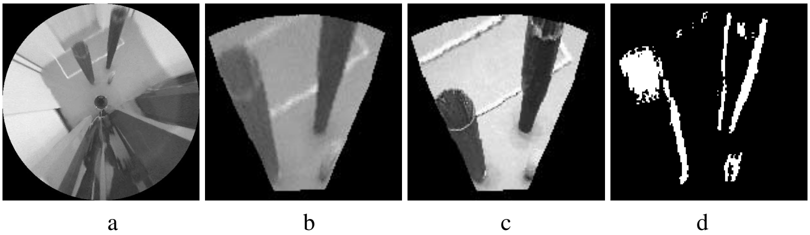

Fig. 3. IPT images acquired by the omni-directional sensor (a) and the standard CCD camera (c). The region of the IPT omnidirectional image corresponding to the field of view of the standard camera (b) and the thresholded difference image (d) are also shown.

Therefore, in the difference image, obstacle candidates are represented by regions bounded by two thin roughly-vertical regions laterally and by a rather large blob vertically. Sometimes one or more of the three region delimiters may be absent. Typical exceptions are small roundish obstacles, for which a single blob is usually detected, or cases in which the distortion of the matter that is located above the ground is such that the obstacle (pseudo)projection extends beyond the common field of view of the two cameras. This is the case shown in figure 1. The results of perspective effect removal and the difference image D are shown in figure 3.

The difference image D is the input of the obstacle-detection method, along with color information from the input images. Segmentation is performed to detect image regions having the features listed above. The analysis of D is a bottom-up process in which segments of regions of interest are matched and merged together to obtain larger regions corresponding to obstacles. First, low-level segmentation is performed

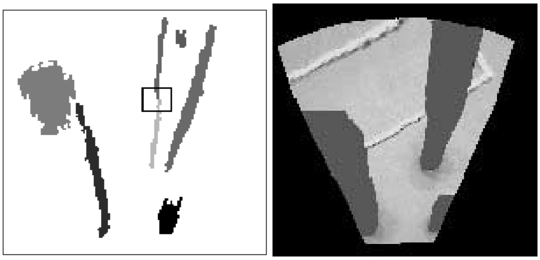

Fig. 4. Left: the results of the blob-coloring algorithm. The two regions evidentiated with a square belong to class 3. The two elongated ones belong to class 2; all other regions belong to class 4. Right: the obstacle regions detected by "closing" the regions surrounded by, or corresponding to, the obstacle segments detected in the previous steps.

to detect connected regions in D and label them using a blob-coloring technique. The regions whose area is below a pre-set threshold are discarded. For all other regions, a set of geometrical features is computed, to classify them into four possible categories of interest, namely:

1. long vertically-elongated thick regions with two apices in their lower part, that directly represent obstacles;
2. long vertically-elongated thin regions with a single lower apex, that may represent one of the segments of the typical obstacle "footprint" in D and need merging to form a class-1 region;
3. short vertically-elongated regions, a sub-class of class 2, to which lower significance is attributed in the matching phase;
4. small roundish regions (i.e., characterized by a height/width ratio close to 1) that may represent small obstacles, part of them or the upper blob of typical obstacle representations in D.

Classification is based on simple empirically-derived rules, implemented as boolean functions of the geometrical features. Although in most cases they can be interpreted as traditional crisp rules, they also allow for fuzzy interpretation in more complex cases. Figure 4 (left) shows the result of the blob-coloring algorithm.

The next step is aimed at merging regions that do not belong to class 1 into larger ones. The matching between regions is based on orientation and color of the neighboring background pixels in the IPT images. More precisely, regions are merged if they are close enough, their orientation differ by less than .2 *rad*, and have regions of the same colors as neighbors. Color similarity is evaluated according to a set of fuzzy rules.

The final step can be divided in two phases: in the first one a preliminary "closure" of the obstacle regions is made. Since the typical obstacle footprint in image D consists of two vertically elongated regions connected to a roundish one on top, the closure is obtained by connecting the two lower apices that can be detected in such a configuration. When some parts of the typical configuration lack, detecting two vertically-elongated regions or one vertically-elongated region and the boundary of the region of interest common to both cameras can be enough for the obstacle to be circumscribed. In the second phase, segmentation is refined, based on uniformity of color or, more generally, of pictorial features in the input images. A horizontal scan is made along the right and left boundaries of the closed regions, that are made coincident with the pixel in which the highest gradient between obstacles and background is found. If the maximum gradient falls below a preset threshold in one of the two IPT images the line is discarded: that is why the large round region on the left (the top of the trash can on the left in the IPT image of the upper camera) has almost completely disappeared in figure 4 (right). Results obtained in indoor environments are described in [9].

4 Final Remarks

In the application described in the paper, HOPS has been used as a stereo sensor. This was certainly not among the primary goals for which HOPS was designed. Actually, omni-directional vision is a complicating factor for inverse-perspective based obstacle-detection. However, results show that, besides the immediate advantages provided by HOPS (the joint availability of omni-directional vision and high-resolution information about a region of interest), HOPS provides also the capabilities of a stereo system by which navigation sub-tasks can be accomplished in real-time: our prototype application can process about 10 frames/s on recent high-end PCs.

Acknowledgements

This work has been partially supported by CNR under the ART, Azzurra Robot Team (http://robocup.ce.unipr.it) grant, ASI under the “Hybrid Vision System for Long Range Rovering” grant, and by ENEA under the “Intelligent Sensors” grant.

References

[1] S. K. Nayar. Omnidirectional vision. In *Robotics Research. 8th International Symposium*, pages 195–202, 1998.

[2] T. Svoboda and T. Pajdla. Panoramic cameras for 3D computation. In *Proc. Czech Pattern Recognition Workshop*, pages 63–70, 2000.

[3] L. Delahoche, B. Maric, C. Pégard, and P. Vasseur. A navigation system based on an omnidirectional sensor. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 718–724, 1997.

[4] J. S. Gutmann, T. Weigel, and B. Nebel. Fast, accurate, and robust selflocalization in polygonal environments. In *Proc. 1999 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 1412–1419, 1999.

[5] A. Clérentin, L. Delahoche, C. Pégard, and E. Brassart-Gracsy. A localization method based on two omnidirectional perception systems cooperation. In *Proc. 2000 ICRA. Millennium Conference*, volume 2, pages 1219–1224, 2000.

[6] A. Bonarini, P. Aliverti, and M. Lucioni. An omnidirectional vision sensor for fast tracking for mobile robots. *IEEE Trans. on Instrumentation and Measurement*, 49(3):509–512, 2000.

[7] H. A. Mallot, H. H. Bülthoff, J. J. Little, and S. Bohrer. Inverse perspective mapping simplifies optical flow computation and obstacle detection. *Biological Cybernetics*, 64:177–185, 1991.

[8] G. Adorni, S. Cagnoni, and M. Mordonini. An efficient perspective effect removal technique for scene interpretation. In *Proc. Asian Conf. on Computer Vision*, pages 601–605, 2000.

[9] G. Adorni, L. Bolognini, S. Cagnoni, and M. Mordonini. A non -traditional omnidirectional vision system with stereo capabilities for autonomous robots. In *Proc. 7th AI*IA Conference*, 2001. in press.