

Using real-time stereo vision for mobile robot navigation

Don Murray

Jim Little

Computer Science Dept.
University of British Columbia
Vancouver, BC, Canada V6T 1Z4

Abstract

This paper describes a working stereo-vision-based mobile robot that can navigate and autonomously explore its environment safely while building occupancy grid maps of the environment. We present a method for reducing stereo vision disparity images to two dimensional map information. Stereo vision has several attributes that set it apart from other sensors more commonly used for occupancy grid mapping. We discuss these attributes, the errors that some of them create, and how to overcome them. This includes the idea of segmenting disparity images based on continuous disparity surfaces to reject “spikes” caused by stereo mismatches. Stereo vision processing and map updates are done at 5Hz and the robot moves at speeds of 150 cm/s.

1 Introduction

Perception is a crucial part of the design of mobile robots. We want mobile robots to operate in unknown, unstructured environments. To achieve this goal, the robot must be able to perceive its environment sufficiently to allow it operate with that environment safely.

Most robots that successfully navigate in unconstrained environments use sonar transducers or laser range sensors as their primary spatial sensor [6] [3, 4] [2, 1]. Computer vision is often used with mobile robots, but usually for feature tracking, or landmark sensing, and not often for occupancy grid mapping or obstacle detection.

In this paper, we present a working implementation of a robot that uses correlation-based stereo vision and occupancy grid mapping to successfully navigate and autonomously explore unknown and dynamic indoor environments. Stereo vision mapping is very sensitive to errors, as the process of collapsing the data from 3D to 2D encourages errors in the form of “spikes” to be propagated into the map. We examine the characteristics of correlation-based stereo which give rise to these



Figure 1: *José*, the mobile robot

errors and make some suggestions on how to overcome them and improve the reliability of the resultant maps. Several examples of stereo range sensing are given, as well as some autonomously generated occupancy grid maps.

2 Architecture

2.1 Mobile robot: *José*

We used a Real World Interfaces (RWI)¹ B-14 mobile robot, *José*, to conduct our experiments in vision-based robot navigation. *José* is equipped with a PentiumTM PC running the Linux operating system as its onboard processing. This robot is a significant improvement over *Spinoza*, our other mobile robot that was reported in [16]. *Spinoza* uses embedded processors exclusively and, although it is a powerful system, it proved to be a difficult development environment. *José* is equipped with an Aironet ethernet radio modem that allows communication to a host computer, as well as a Matrox Meteor RGB frame grabber connected to a Triclops trinocular stereo vision camera module.

The Triclops stereo vision module was developed at

¹www.rwii.com

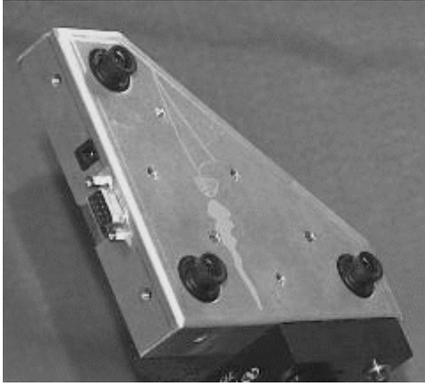


Figure 2: The Triclops stereo head

the UBC Laboratory for Computational Intelligence (LCI) and is being marketed by Point Grey Research, Inc.² The stereo vision module has 3 identical wide angle (90° degrees field-of-view) cameras. The system is calibrated using Tsai’s approach [15]. Correction for lens distortion, as well as misalignment of the cameras, is performed in software to yield three corrected images. These corrected images conform to a pinhole camera model with square pixels. The camera coordinate frames are co-planar and aligned so that the the epipolar lines of the camera pairs lie along the rows and columns of the images.

2.2 Trinocular Stereo Algorithm

The trinocular stereo approach is based on the multibaseline stereo developed by Okutomi and Kanade [14]. Each pixel in the reference image is compared with pixels along the epipolar lines in the top and left images. The comparison measure used is sum of absolute differences (SAD). The results of the two image pairs (left/right, top/bottom) are summed to yield a combined score. Multibaseline stereo avoids ambiguity because the sum of the comparison measures is unlikely to cause a mismatch – an erroneous minimum in one pair is unlikely to coincide with an erroneous minimum in another pair.

The disparity results are validated in two ways. First, there is a “sufficient texture” test. This test checks that there is sufficient variation in the image patch that is to be correlated by examining the local sum of the Laplacian of Gaussian of the image. Low texture areas score low in this sum. If there is insufficient variation the results will not be reliable, thus the pixel is rejected because there will be too much ambiguity in the matches. Secondly, there is a “quality of match” test. In this test, the value of the

score is normalized by the sum of all scores for this pixel. If the result is not below a threshold, the match is consider to be insufficiently unique and therefore a likely mismatch. This kind of failure generally occurs in occluded regions where the pixel cannot be properly matched.

These validation methods are both tunable via thresholds. The trade-off is between quality and quantity of data. If tuned too high, much of the disparity image is invalid and although the valid data is highly reliable, it may not give enough coverage to be useful. If the tuned values are too low, the map will be subject to errors.

3 Stereo vision and occupancy grids

3.1 Review of occupancy grids

Occupancy grid mapping, pioneered by Moravec and Elfes [12, 5], is the most widely used robot mapping technique due to its simplicity and robustness and also because it is flexible enough to accommodate many kinds of spatial sensors. It also adapts well to dynamic environments. We selected it for all these reasons. The technique divides the environment into a discrete grid and assigns each grid location a value related to the probability that the location is occupied by an obstacle. Initially, all grid values are set to a 50% value (i.e., equal probability for occupied and unoccupied). Sensor readings supply uncertainty regions within which an obstacle is expected to be. The grid locations that fall within these regions of uncertainty have their values increased while locations in the sensing path between the robot and the obstacle have their probabilities decreased.

3.2 Stereo sensor model

To apply the occupancy grid method one must have a sensor model. A rigorous investigation of range sensing with stereo vision was provided by Matthies and Grandjean in [11]. They found disparity estimates to have Gaussian distributed random errors with standard deviations as small as 0.05 pixels. These standard deviations were consistent over different resolutions of images. While it is true that they used a different comparison score (sum of squared differences), cameras and calibration, their results show the magnitude of accuracy that can be achieved through careful correlation stereo vision. Our own experiments with sub-pixel interpolation indicate that the Triclops stereo vision module produces results with standard deviations well below one pixel. However, for real-time considerations, we have not used sub-pixel interpolation in our stereo algorithm. Due to the resulting quantization of the disparity we can approximate our

²www.ptgrey.com

stereo model by the following:

$$P(d|Z) = \begin{cases} 1 & \text{for } Z = Z(d + 0.5) \rightarrow Z(d - 0.5) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

For our stereo vision system, with aligned optical axes and therefore focus at infinity, the relation of disparity to depth is given by

$$Z(d) = \frac{fB}{d}$$

where Z is the depth, f is the focal length of the cameras, B is the baseline between the cameras and d is the disparity. The basis for expanding this 1D approximation into 2D is illustrated in Figure 3. This figure

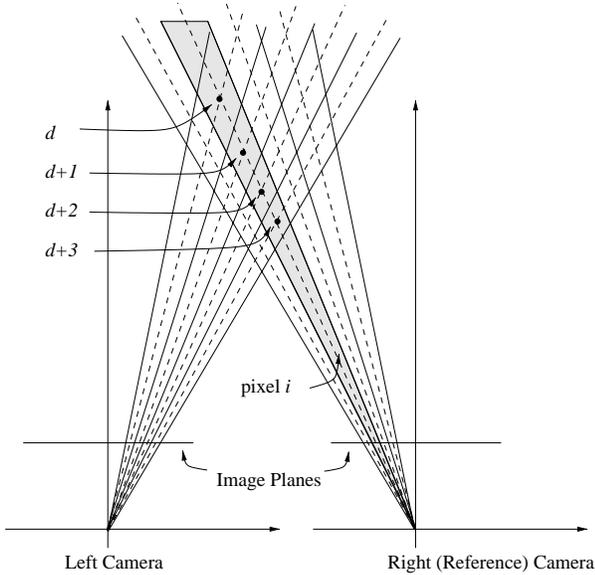


Figure 3: Stereo triangulation: dashed lines indicate lines-of-sight from the centre of the pixels, solid lines indicate lines of sight at the edges of the pixels

represents the intersection of the lines-of-sight (LOS) for individual pixels in stereo pair of cameras. One can see for a given pixel i in the right (or reference) camera and a disparity result d from stereo matching, a 2D position can be determined by triangulation. In addition to the 2D position at the intersection of the LOS of the centres of the two pixels, there is also a “diamond” around this position which can be taken as an error bound. Given Equation 1 we determine the region of uncertainty for a given pixel and disparity as shown in Figure 4. The corners of the trapezoid region of uncertainty can be found by calculating $Z = Z(d \pm 0.5)$ and then determining $X = \frac{(x \pm 0.5)Z}{f}$ where x is the image plane coordinate along the rows of the image

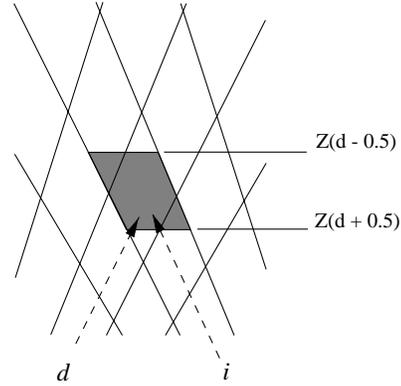


Figure 4: Region of uncertainty for a given pixel i and disparity d

and f is the focal length. The clear region in which an obstacle should not appear is the triangle formed by the robot’s position and the closest two corners of the trapezoid.

3.3 Constructing top-down views from stereo images

Although occupancy grids may be implemented in any number of dimensions, most mobile robotics applications (including ours) use 2D grids. This may be viewed as unfortunate as stereo data provides information about the world in 3D. Much of this data is lost in the construction of a 2D occupancy grid map. This reduction in dimension is justified since indoor mobile robots fundamentally inhabit a 2D world. The robot possesses 3 DOF (X , Y , heading) within a 2D plane corresponding to the floor. The robot’s body sweeps out a 3D volume above this plane. By projecting all obstacles within this volume to the floor, we can uniquely identify free and obstructed regions in the robot’s space.

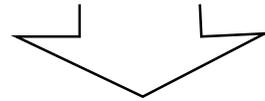
Figure 5 shows the construction of the 2D occupancy grid sensor reading from a single 3D stereo image. Figure 5(a) shows the reference camera grayscale image (160x120 pixels). The resulting disparity image is shown in Figure 5(b). The white regions indicate areas of the image which were invalidated and thus not used. Darker areas indicate lower disparities, and are farther away from the camera. Figure 5(c) shows a column-by-column projection of the disparity image, taking the maximum valid disparity in each column. The result is a single row of maximum disparities. These represent the closest obstacle in each column. Figure 5(d) shows this column values converted into distance, and (e) shows these distance values converted into an occupancy grid representation, with black indicating the uncertainty region around



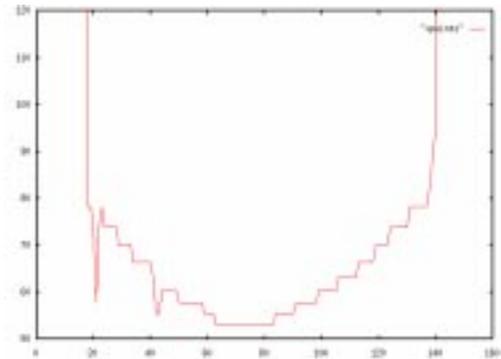
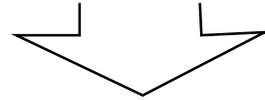
(a)



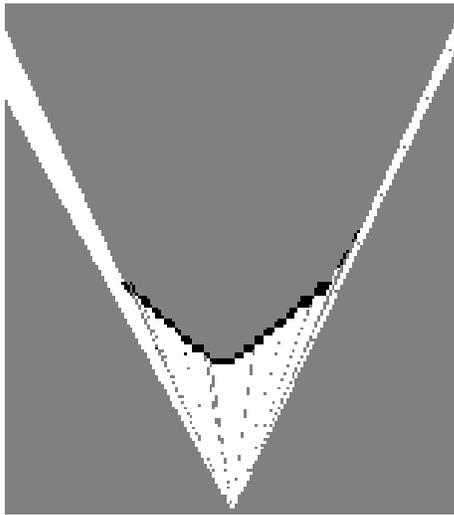
(b)



(c)



(d)



(e)



Figure 5: From stereo images to top-down views: (a) grayscale image (b) disparity image (white indicates invalid, otherwise brighter indicates closer to the cameras) (c) the maximum disparity in each column of the disparity image (d) depth versus column graph (depth in centimetres) (e) the resultant estimate of clear, unknown and occupied regions (white is clear, black is occupied and gray is unknown)

the object, and white indicating regions that are clear. Notice the two “spikes” on the left hand portion of Figure 5(d). These were caused by mismatches in the stereo algorithm and their causes and removal will be discussed in Section 4.1.

The process illustrated in Figure 5 generates the input into our stereo vision occupancy grid. The mapping system then integrates these values over time, to expand the map, and keep it current in the changing world.

3.4 Map updating

The theory for updating occupancy grid maps according to a probabilistic framework is given in [5][9] and others. We adopted a much simpler approach for a few reasons:

1. computationally simpler
2. stereo errors that affect mapping are mostly systematic (as discussed in Section 4.1) and not modeled well with a probabilistic framework
3. the error in the sensor readings mainly depend upon the robot’s position, the surface texture, as well as other effects. As a result the errors are not independent between sensor readings.
4. we decided to try a simpler approach first and move to the more complicated one only if the simple one failed

The update rule was simply

$$\begin{aligned} \text{if } i \in \text{OCC}(r) \quad G(i) &= G(i) + K \\ \text{if } i \in \text{CLEAR}(r) \quad G(i) &= G(i) - K \end{aligned}$$

where i is an occupancy grid location, r is the sensor reading, $\text{OCC}(r)$ represents the region of uncertainty around the sensed obstacle, $\text{CLEAR}(r)$ represents the clear region between robot and the sensed obstacle, $G(i)$ represents the current grid value at location i and K is some constant. The values of $G(i)$ are clipped between some G_{max} and G_{min} values. In our implementation, $G(i)$, the occupancy grid value, ranges from 0 to 255. Our 50% value indicating an unknown grid cell is determined by $(G_{max} - G_{min})/2$. We selected $K = 20$ to have sufficient speed in map updating. For planning purposes, we selected conservative values of $G(i) < 50$ being free of obstacles, and $G(i) > 150$ being definitely obstructed.

This update rule provides a simple linear transition between occupied and unoccupied values. It is not overly sensitive to erroneous readings, yet transitions quickly in the presence of dynamic objects. An unfortunate side-effect is that good quality data tends to be

smothered by poor quality data when an area in the map that was viewed from close range is later viewed from long range. We apply a heuristic to remove this side-effect that is given in Section 4.3.

4 Improving stereo vision mapping

4.1 Characterizing stereo errors

The mapping approach described so far is very sensitive to noise. As described in Section 3.2, the depth estimate distribution about the true depth of properly matched pixels is of sufficiently low deviation to be swallowed up by our quantization. However, noise due to stereo mismatches is a serious problem. Indoors scenes containing specular surfaces, repetitive patterns, and time-varying light sources can cause errors that are more or less uniformly distributed across the disparity range of the stereo system. These mismatches can be reduced by validation through comparing left-to-right and right-to-left best matches [7], the validation approaches described in Section 2.2 or by increasing the number of cameras in a multibaseline system [8]. However, even with a trinocular stereo system, these errors will appear. They generally appear as “spikes” in the disparity image and can drastically affect the quality of the map.

To overcome this we have tried common image filtering techniques such as median filtering or morphological filtering. These filters can improve the results, but not as significantly as we would like. These techniques fail because they are methods intended to remove or reduce noise from inputs that have unstructured or evenly distributed noise, i.e., noise that appears randomly and consequently will appear most often as single pixels in the image. In disparity images errors often do not have these qualities. Errors in stereo matching occur when two similar image features are in proximity to each other. The algorithm may match one of the features in one image to the wrong feature in the other image. A classic example of this problem is the so-called “picket fence” problem. When looking at a picket fence with binocular stereo, there will be regularly spaced, nearly identical image patches that can be mismatched. This causes several “ghost” fences to appear between the true fence and the robot. Three camera stereo largely overcomes this problem, but pathological situations still do occur regularly in the unstructured world, especially in man-made environments.

If these kinds of errors occur, they will generally happen over a patch of the incorrectly matched feature. Thus, noise does not appear in isolated pixels but in dense, connected regions. These coherent errors will confound the above filtering approaches as

they will appear as a stable signal, one to be preserved instead of rejected.

As well, filters such as described above may remove valid features if those features are insufficiently thick. This can be a problem when there are thin objects in the scene such as poles or table edges. For low resolution stereo these objects often appear only one or two pixels wide and may be removed as not sufficiently stable.

4.2 Spike removal

We developed an approach using surface segmentation to overcome the problem of noise rejection for coherent errors as described in the previous section. To remove spikes caused by feature mismatches, we take into account the attributes of these errors: they are locally stable, but not large and they have no support from surrounding surfaces; they are genuine spikes with sharp disparity discontinuities at all borders. True surfaces in the stereo image should be not only locally consistent, but globally part of a larger 3D surface. By segmenting the image into continuous disparity surfaces, we can establish a good hypothesis based on the size of the surface whether it is a real 3D surface or a noise artifact. To segment the image into surfaces of continuous disparity we apply the following logic:

$$i = L \quad \text{given} \quad \begin{array}{l} j \in N(i) \\ j = L \\ |d_j - d_i| \leq 1 \end{array}$$

where i is any given pixel, L is a surface label, $N(i)$ is a neighborhood of pixels around i and d_i is the disparity value at location i . Entire surfaces are invalidated from the disparity image if the number of pixels that have a given label do not pass a threshold.

This approach has two significant benefits: it can reject cohesive spikes that may fool noise rejection filters; and it can preserve thin structures that are part of a coherent structure. An example of the effectiveness of this approach is shown in Figure 6. As one can see in (b) and (e), the raw disparity image contains many disparity spikes that corrupt the resulting map. The morphologically filtered disparity image shown in (c) performs marginally better. The map shown in (g), constructed from the surface segmentation approach, has a cohesive and more accurate representation of the actual scene.

4.3 Accuracy Preservation

A problem with our update rule is that low quality data can obscure better data when obstacles are viewed at longer range. Initially, we applied a “horizon” or maximum range threshold on our sensor data to limit the use of low resolution data. While this results in cleaner maps, it is a wasteful loss of potentially

useful data and can seriously reduce the efficiency of exploration algorithms.

To overcome this problem we applied a hypothesis-explanation heuristic. Effectively, the occupancy grid is a model of our robot’s world. Each time an obstacle is sensed, we make a hypothesis that there is an obstacle within a specified uncertainty region. At this point, before applying our update, we can inspect the uncertainty region for better quality data. If we find more precise evidence of an obstacle within the region, this “explains” our hypothesis. Consequently, the “model” (or map) and the sensor data agree, and there is no need to apply this lower quality update.

To implement this, a second grid value is maintained. Each grid updated inside an uncertainty region also stores the value of the disparity from which this reading arose. By inspecting update regions for higher disparity readings before applying the update, we avoid the smearing or blurring affect of lower quality data on the map. This solution is simple, robust, sensitive to dynamic objects and applies no artificial horizon on the sensor data.

5 Navigation and exploration

We have implemented a path planning and autonomous exploration algorithm using the stereo vision-based occupancy grid mapping that was reported in [13]. The path planning algorithm is a mixture of shortest path and potential field methods. The robot has successfully explored building floors through several rooms and corridors without human intervention. Two examples of the maps generated by the exploration are shown in Figure 7.

6 Conclusion

We have shown the stereo vision is a viable alternative to other spatial sensors for constructing occupancy grid maps. Although stereo vision has several characteristics that can make mapping prone to errors, it also has great capability in localizing obstacles quickly and accurately. We showed a few techniques to improve the quality of stereo vision mapping results on a working system. Several example results were given.

6.1 Remaining challenges

One serious problem that remains unsolved in our current mapping and navigation implementation is the problem of seeing over, under or past objects. Certain objects are very difficult to see for the stereo vision, due to the large occlusions involved. The most dangerous are table tops that appear between the height of the top and reference cameras. These obstacles are dangerous because



(a)



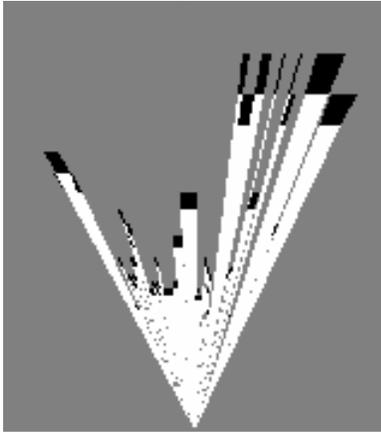
(b)



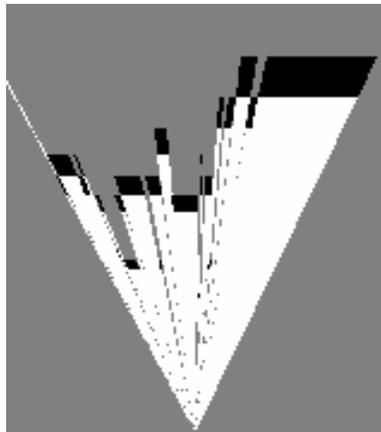
(c)



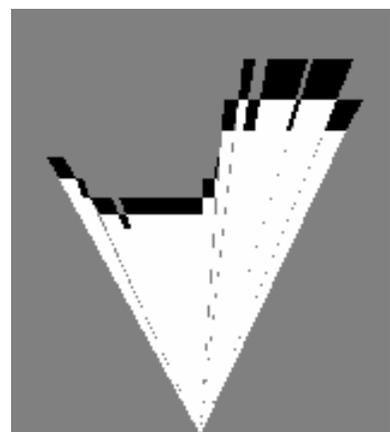
(d)



(e)



(f)



(g)

Figure 6: The results of a single stereo disparity image viewed from above with various filters applied: (a) shows the reference image, (b) and (e) show the raw disparity image (with invalid regions white) and the resultant top-down view, (c) and (f) show the same with the disparity image filtered using morphological erosion on regions of constant disparity, and (d) and (g) show the results using surface segmentation for noise rejection (the surfaces that have been rejected are shown in black in image (d))

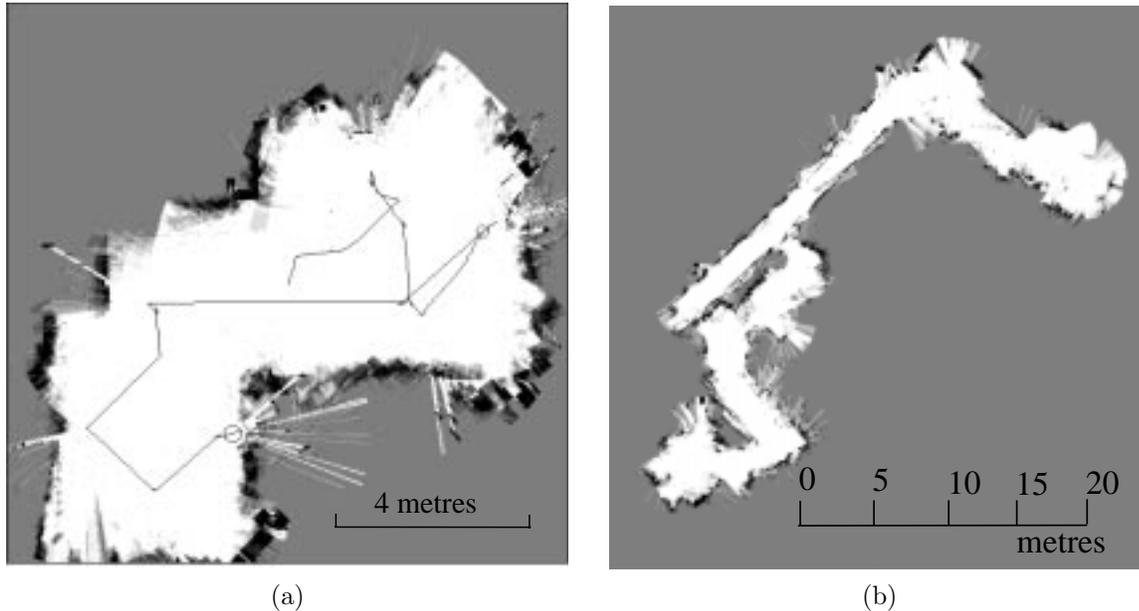


Figure 7: Two example maps generated through autonomous exploration

- they are difficult to successfully match in the left/right camera pair as the the features are aligned with the baseline
- they are difficult to match in the top/bottom camera pair as the views of the obstacle is very different between the two images due to occlusions
- they are positioned to cause the maximum damage to the robot if it should run into them

Unfortunately, even though they are difficult to see for the robot, they do not obscure surfaces behind them. Therefore the robot often can accurately resolve obstacles beyond the table, and since it sees “through” the table, does not add it to the map. This problem has been partially solved by the addition of surface segmentation to the the disparity image filtering. However, this only works when the table top, or portions of it, are successfully resolved by the stereo algorithm. We are looking into incorporating concepts from Konolige’s MURIEL system [9] to address this problem.

Another imposing challenge that remains is the localization problem. Regardless of the quality of the maps, their utility is limited because the robot cannot reuse them from power-up to power-up, and also as odometry drift accumulates, old map data becomes inaccurate. We are currently investigating automatic acquisition and use of 3D landmarks using a variety of

features such as corners, vertical lines and intersection of 3D lines. Some preliminary work has been reported in [10].

We have done some experiments in determining disparity images to sub-pixel resolution with good success. The challenge now is to tune the algorithm to run in near real-time with the onboard processing capability of a PC-based robot. By taking advantage of the MMX instruction set, and using Intel Pentium II processors, we hope to achieve sub-pixel resolution stereo with 320x240 pixel images at speeds of over 3Hz. This would dramatically increase the sensing resolution and provide many new opportunities (and new difficulties to be overcome) for stereo vision mapping.

Finally, there is the challenge of extending our 2D occupancy grids to 3D voxel-based representations of the world. With a texture-mapped voxel representation of the world we intend to investigate automatic generation of virtual reality models based on real-world scenes.

Acknowledgements

We would like to thank all the members of the LCI lab at UBC for their fruitful insights and commentary. A special thanks goes to David Lowe and Jochen Lang for their perseverance in the face of obstinacy.

References

- [1] Rodney A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2:14–23, 1987.

- [2] Rodney A. Brooks, John Connell, and Peter Ning. Herbert: A second generation mobile robot. A.I. Memo No. 1016, MIT AI Laboratory, January 1988.
- [3] Greg Dudek, M. Jenkin, Evangelos Miliotis, and David Wilkes. On building and navigating with a local metric environmental map. In *ICAR-97*, 1997.
- [4] Greg Dudek, Evangelos Miliotis, M. Jenkin, and D. Wilkes. Map validation and self-location for a robot with a graph-like map. *Robotics and Autonomous Systems*, to appear, 1996.
- [5] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, June 1989.
- [6] S. B. Nickerson *et al.* Ark: Autonomous navigation of a mobile robot in a known environment. In *IAS-3*, pages 288–293, 1993.
- [7] Pascal Fua. A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications*, 6(1):35–49, 1993.
- [8] T. Kanade, A. Yoshida, K. Oda, H. Kano, and M. Tanaka. A video-rate stereo machine and its new applications. In *Proc. IEEE Int'l Conf. on Computer Vision and Pattern Recognition*, San Francisco, California, USA, June 1996.
- [9] Kurt Konolige. Improved occupancy grids for map. *Autonomous Robots*, (4):351–367, 1997.
- [10] James J. Little, Don Murray, and Jiping Lu. Identifying stereo corners for mobile robot localization. In *IROS-98*, 1998.
- [11] L. Matthies and P. Grandjean. Stochastic performance modeling and evaluation of obstacle detectability with imaging range sensors. *RA*, 10(6), 1987.
- [12] H. Moravec and A. Elfes. High-resolution maps from wide-angle sonar. In *Proc. IEEE Int'l Conf. on Robotics and Automation*, St. Louis, Missouri, March 1985.
- [13] Don Murray and Cullen Jennings. Stereo vision based mapping for a mobile robot. In *Proc. IEEE Conf. on Robotics and Automation, 1997*, May 1997.
- [14] M. Okutomi and T. Kanade. A multiple-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(4):353–363, 1993.
- [15] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, 3:323–344, 1987.
- [16] Vladimir Tucakov, Michael Sahota, Don Murray, Alan Mackworth, Jim Little, Stewart Kingdon, Cullen Jennings, and Rod Barman. Spinoza: A stereoscopic visually guided mobile robot. In *Proceedings of the Thirteenth Annual Hawaii International Conference of System Sciences*, pages 188–197, January 1997.