Proceedings of the 2004 IEEE
International Conference on Networking, Sensing & Control
Taipei, Taiwan, March 21-23, 2004

Spatial Grouping of 3D Points from Multiple Stereovision Sensors

S. Nedevschi, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol
Computer Science Department
Technical University of Cluj-Napoca
Cluj-Napoca, ROMANIA
{Sergiu.Nedevschi, Radu.Danescu, Tiberiu.Marita}@cs.utcluj.ro

Abstract - *This paper will present a method for grouping 3D points into cuboids. The 3D points are extracted using multiple stereovision sensors, and the sensor fusion module performs the fusion of the data sets and the grouping of the points in a single algorithm. The fusion/grouping algorithm is scalable, being able to work using any number of sensors, including a single one. The grouping method relies on a method of transforming the 3D space so that the density of the points is kept constant, and all the points belonging to a single object are adjacent, making the grouping of points into cuboids a simple labeling problem.*

Keywords: stereovision, data fusion, feature grouping, communication, distributed computation.

1 Introduction

Stereovision is a well-established technique for extracting 3D information from images. This technique is a passive approach, which does not interfere with the observed environment, unlike other ranging sensors (radar, laser), and provides a much richer information set. The limitations of stereovision are the ranging precision and the cumbersome calibration methodology. The stereovision sensor provides a set of 3D points, which are not noise-free. The most difficult problem is to group these points into meaningful objects – a problem which affects all ranging sensors.

Each stereovision sensor is limited by its own field of view. A complete description of a 3D scene is difficult to be achieved by only one sensor, due to the sensor's position, occlusions of the objects, the limited range of operation, etc. For that reason, fusion of information from several sensors, placed strategically around the scene is necessary.

Several approaches for 3D points grouping are available in literature. In [8] the point grouping algorithm works mainly in the 2D image space, stereovision being used to extract the depth of the image points. In the image space, points are connected if they are in the 4-nearest neighbors relationship. In the depth space, the points are connected if the difference in depth is less than the depth uncertainty, which is a variable error resulted from a fixed disparity error of 0.5 pixels.

Another approach for 3D points grouping is presented in [3]. They use spatial coherence to identify regions from the depth information. First, they are looking for connected components in the 8-neighborhood of the image dimensions. In the depth dimension a neighboring pixel is connected if the difference in depth is less than a threshold. In the grouping process an object could be split into different disjoint regions. If two regions belong to the same object, they must be close to each other in 3D space. For each pair of regions a probability measure gives the likelihood that the regions belong to the same object.

Specific object shapes fitting algorithms as the L-Fit [2] are trying to group objects using L-shapes matching. The assumption in this case is that the correlated 3D points of an object with rectangular horizontal section, viewed from one side have an approximately L-shape in the depth map (ground plane projection).

The grouping algorithm presented in this paper works by compressing the 3D space in a way that the point density is preserved with the distance, and the points corresponding to the same object are neighbors. The compressed space map is different for each sensor, but then the points are mapped in a common uncompressed 3D space, preserving the adjacencies in the compressed space, and fused together, and then grouped into objects by a simple labeling algorithm. In this way, both of the problems, the grouping of the points and the fusion of sensor data, are solved in the same algorithm.

Another sensor fusion algorithm was proposed in [6]. The point grouping is performed at sensor level, and the fusion algorithm fuses the intermediate cuboids into final ones, on the basis of a confidence measure of the cuboids' corners.

2 The sensorial system architecture

The architecture of the sensorial system is presented in fig. 2.

The system consists of "*n*" Stereovision Sensors linked by TCP connection to the Sensor Fusion Module (SFM).

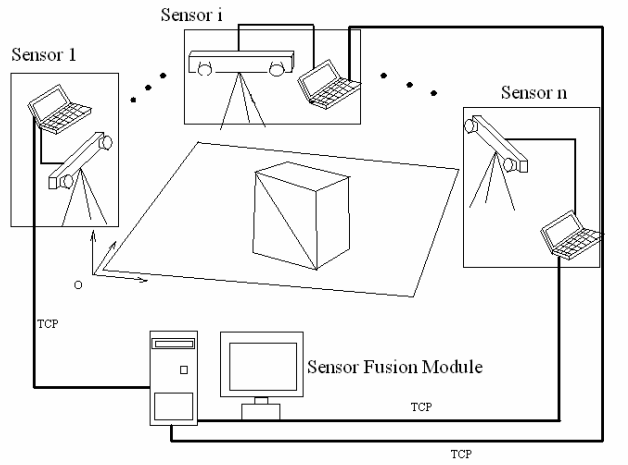


Figure 1. The architecture of the sensorial system

The Stereovision Sensors must be placed around the space of interest in such a way that a good coverage of the scene is accomplished. This way, each sensor has a different view of the 3D scene, and issues as hidden object facets or object occlusions are easier to treat.

A Stereovision Sensor consists of a pair of cameras, mounted on a rig, linked to its image processing computer. The image processing computer performs stereo 3D reconstruction cycles on the synchronously acquired image pairs. The reconstructed 3D points represent the sensor's output.

Each stereovision sensor must be calibrated before operation. The calibration will be performed with respect to a unique world coordinate system, so that each set of 3D points delivered by a sensor expressed in the same coordinate system, thus making the fusion easier.
The Sensor Fusion Module (SFM) is the central module of the method. This computer receives the information from all stereovision sensors, in the form of 3D point sets, and performs their fusion and grouping into objects. The SFM knows the camera parameters of each stereovision sensor. This information is needed in the point fusion algorithm.

The SFM acts like the synchronization master for the sensor array, ensuring that all sensors capture the scene at the same time. It is also responsible for delivering the information to client applications. The result is delivered in the form of 3D cuboids expressed in the unique world coordinate system.

3 Stereovision sensors calibration

In order to reconstruct and measure the 3D environment using stereo cameras, the cameras must be calibrated. The calibration process estimates the camera's intrinsic parameters (which are related to its internal optical and geometrical characteristics) and extrinsic ones (which are related to the 3D position and orientation of the camera relative to a global world coordinate system).

The intrinsic parameters of each camera are calibrated individually. The estimated parameters are the focal length and the principal point coordinates and the lens distortions. The parameters are estimated by minimizing the projection error from multiple views of a set of control points placed on a coplanar calibration object with known geometry. For a stereo system of two cameras, the obtained intrinsic parameters can be refined by inferring the stereo information available. This is done by introducing a new constraint in the estimation process which considers also the projection error of the control points image coordinates from one image to another [1].

For the benefit of the point fusion algorithm the calibration of the extrinsic parameters must be performed in the same coordinate system (a unique coordinate system belonging to the scene), and must be very precise. If the precision requirements are not met, the set of points from different sensors will have different meaning, and their fusion will be erroneous.

The extrinsic parameters of the cameras are estimated by minimizing against the extrinsic parameters the projection error for a set of 3D control points with measured coordinates in a world reference system [4, 5]. For the specific setup of the current application having multiple stereovision sensors, each stereo pair of cameras is calibrated using a set of control points measured in a unique world coordinate system - the coordinate system of the scene (Figure 2).

The obtained extrinsic parameters for each camera *"j"* are a translation vector of the camera in the world coordinate system (**T**j) and a rotation vector (**R**j) relative to the same coordinate system. This approach in the calibration process allows us to measure the coordinates of the reconstructed 3D object in the same world coordinate system, which is essential for the sensor fusion algorithm.

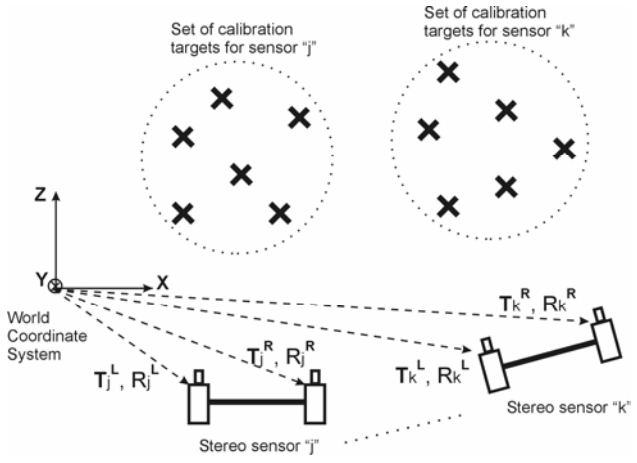


Figure 2. Calibration setup for calibrating the extrinsic parameters

4 Stereo 3D reconstruction

The stereo reconstruction algorithm used is mainly based on the classical stereovision principles available in the existing literature [7]: find pairs of left-right correspondent points and map them into the 3D world using the stereo system geometry determined by calibration.

Constraints, concerning real-time response of the system and high confidence of the reconstructed points, must be used. In order to reduce the search space and to emphasize the structure of the objects, only edge points of the left image are correlated to the right image points. Due to the cameras horizontal disparity, a gradient-based vertical edge detector was implemented. Non-maxima suppression and hysteresis edge linking are being used. By focusing to the image edges, not only the response time is improved, but also the correlation task is easier, since these points are placed in non-uniform image areas.

Area based correlation is used. For each left edge point, the right image correspondent is searched. The sum of absolute differences (SAD) function [9] is used as a measure of similarity, applied on a local neighborhood (5x5 or 7x7 pixels). Parallel processing features of the processor are used to implement this function. The search is performed along the epipolar line computed from the stereo geometry. Two modes are used: image rectification, search along the horizontal line or without rectification and the search is performed along the epipolar line determined by the system geometry.

To have a low rate of false pairs, only strong responses of the correlation function are considered as correspondents. If the global minimum of the function is not strong enough relative to other local minimums, the current left image point is not correlated. Repetitive patterns are rejected and only robust pairs are reconstructed.

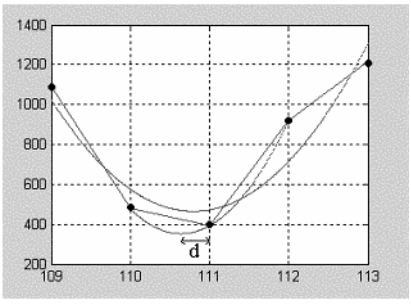


Figure 3. Linear piecewise approximation of the correlation function for 5 points. Two parabolas fitted to 3 and 5 neighbors are presented. The sub-pixel displacement d for the 3-neighbors parabola is shown.

To achieve a better 3D depth resolution, the sub-pixel right correspondent is computed by fitting a parabola to the correlation function [9]. The parabola is fitted to a local neighborhood (3 or 5 points) of the global minimum. The accuracy obtained is about 1/4 to 1/6 pixels. This accuracy is dependent of the image quality (especially noise and contrast). Our tests proved that the 3-neighbors parabola works better than the other one.

After this step of finding correspondences, each left-right pair of points is mapped into a unique 3D point [7]. Two 3D projection rays are traced, using the camera geometry, one for each point of the pair. By computing the intersection of the two projection rays, the coordinates of the 3D point are determined. The reconstruction formulas are simple, when image rectification is used, or complex, if the original images are used for correlation.

While image rectification provides a simple search area for correspondents and straightforward 3D reconstruction, the general geometry mode, without rectification, provides a better resolution since no image re-sampling is done.

5 Sensor fusion algorithm

The sensor fusion algorithm takes care of two problems: merging the set of points into one global view of the scene, and grouping those points into objects. The reason that prevents a simple union of the point sets is that the density of the 3D points reconstructed by stereovision is not constant with the distance from the camera. Therefore, the points in a region can have different densities, depending of the stereovision sensor that generated them.

The first step of the grouping algorithm is to select the points that are above the ground level. This selection can be direct, if the parameters of the ground surface are known (if the sensor is fixed with respect to the scene), or it can follow a ground detection routine, if the sensors are mobile.

For each sensor we'll map the points into a space of even density. However, because the density depends of the position of the point with respect to the stereovision sensor that generated it, we have to make the transformation in the sensor's coordinate system. Without losing the generality, we'll consider the coordinate system of the sensor j to be defined by the rotation matrix $\mathbf{R_j^L}$ and $\mathbf{T_j^L}$ of the sensor's left camera. The transformation that maps the world point *(X,Y,Z)* into the coordinate system of the sensor *j* is:

$$\begin{pmatrix} X_j \\ Y_j \\ Z_j \end{pmatrix} = (\mathbf{R_j^L})^{\mathbf{T}} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} - \mathbf{T_j^L} \qquad (1)$$

The points in the coordinate system of the sensor *j* will then be mapped into a compressed space, which accounts for the difference in point density with the distance from the camera.

The formulas used to find the position *(row, col)* in the compressed space, of a point *(X,Z)* in the uncompressed space, are:

$$row = \log_{1+\frac{k}{f}} \frac{z}{z_{\min}} \quad (2)$$

$$col = X \cdot Scale(Z) \quad (3)$$

$$Scale(Z) = f \cdot \frac{1}{Z} \cdot k \quad (4)$$

where Z_{min} is the lowest distance boundary for the space of interest, f is the focal length of the cameras, and k is a factor which depends on the richness of 3D reconstructed points with the current reconstruction method. For the X and Z axes the values for k can be different. The k factors are chosen to satisfy two conditions of the found objects:
- to not divide a real object into more smaller objects;
- to not unify more real objects into a bigger object.

The *XZ* plane of the 3D space is mapped into a compressed space, as shown in the figure 4:

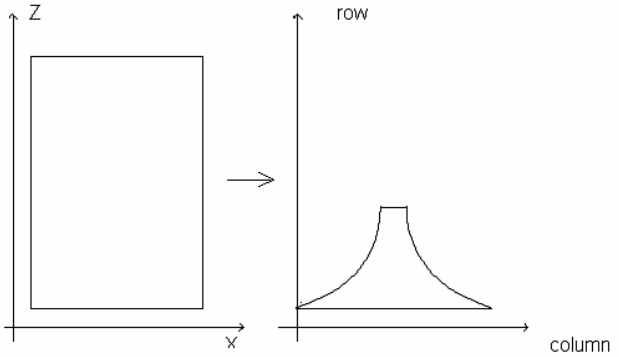
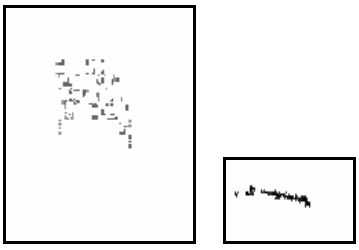


Figure 4. Compressing the space to obtain an even point distribution

Figure 5 shows the results of applying the compression on the cluster of points belonging to the same object. The points in the 3D space are very sparse, while the points in the compressed space are joined together.



a) b)

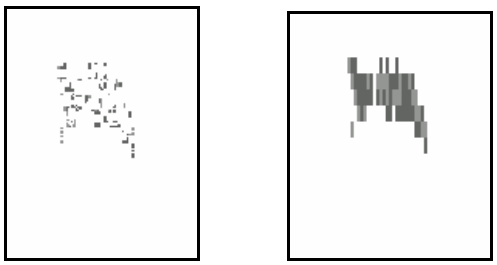
Figure 5. Results of applying compression on a cluster of points belonging to the same object

The grouping of the points in the compressed *XZ* plane can be viewed as a simple labeling: the connected clusters of points are grouped into top-view objects. However, we cannot fuse together the points of different sensors, represented in the compressed spaces, because of the different coordinate systems. Therefore, we take the points from the compressed space and map them back in the real world *XZ* plane, but this time we take care to fill the gaps.

The algorithm that does that is the following:

- Define a grid of *n* rows and *m* columns which is a linearly scaled and discrete representation of the *XZ* plane. Let's call this space the C*onnected 3D space*
- For each *r* and *c* in this space, perform the following:
  - Compute the corresponding *X* and *Z*
  - Compute the corresponding *Xj* and *Zj*
  - Find the position in the compressed space of sensor *j*
  - If the position in the compressed space is not null, mark the position in the grid

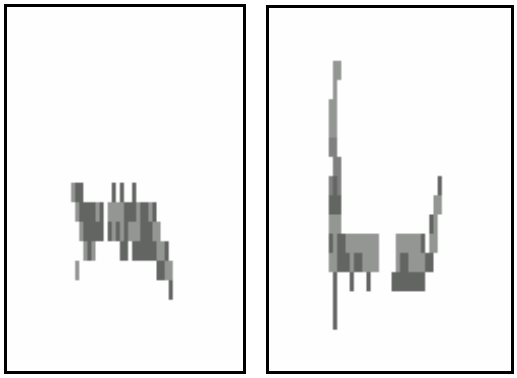
This algorithm obtains a top-viewed scaled 3D space, in the unique world coordinate system, but having the important property that the points maintain the connection property.



a) b)

Figure 6. Comparison of the point density between the original point distribution in the world coordinate system (a) and the point distribution in the *Connected 3D Space* (b). View (b) is a linearly scaled down and discrete representation of the space in a bird-eye view

The connected 3D space allows us to use the same simple labeling algorithm for joining the points into distinct objects. However, this space has one more important property: the entries in the connected 3D spaces of each sensor can be simply added up to form the fused connected 3D space of the whole scene, if the camera calibration procedure was properly performed.



a) b)

Figure 7. The connected 3D space view of the same object from two different sensors

Each of the sensor's view of the scene is incomplete. Each sensor can bring supplementary information for the scene reconstruction, enlarging the field of view of the vision system. If an object is viewed by more than one sensor, the data from each sensor is added into a better description of the object. By combining the connected space results of each sensor into a single connected space description of the scene, we achieve both objectives: the different parts of the scene, visible to only some of the sensors, are combined together, and the information about the same object, coming from different sensors, is added up. Figure 7 shows two views of a single object.

We may notice that none of them views the object as a single cluster of connected points – and thus we'll obtain incorrect results. Figure 8 shows the result of adding up the data from the two sensors. The points which form the object are now all connected, and they will receive the same label.

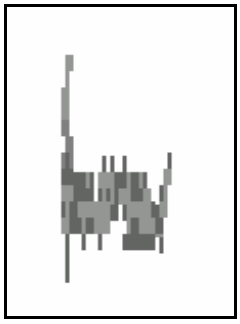


Figure 8. Result of adding the connected space information from two sensors in the case of the same object

The regions which result from labeling are size filtered so that only the ones large enough are kept. This way we increase the robustness of the result.

The extraction of the 3D characteristics of the cuboids is performed using the original sets of points (in the real 3D coordinate system). The sets of 3D points obtained from all the stereovision sensors are united in a single set. For each point the algorithm computes its coordinate in the connected 3D space, and checks the label present at these coordinates, which gives the identity of the object to which the 3D point belongs. Each object will have its own list of 3D points. By computing the minima and maxima of each coordinate of these points, the limits of the 3D cuboid are extracted $(X_{min}, Y_{min}, Z_{min}, X_{max}, Y_{max}, Z_{max})$.

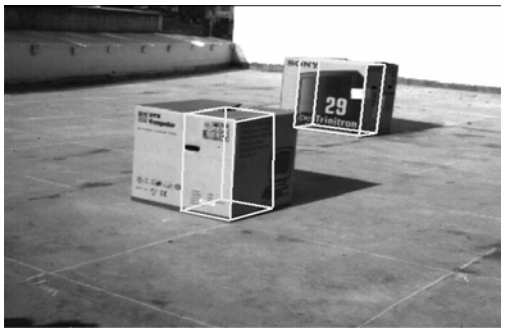
6 Results

For testing of the algorithm we have used two stereovision sensors, which were calibrated using the method described in the calibration section, using a common coordinate system.

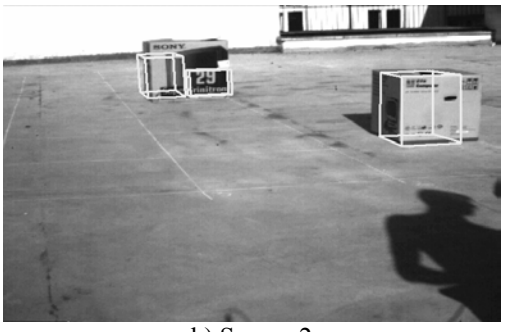
The stereovision-extracted 3D points were sent to a fusion computer, using a standard networking protocol. The algorithm of sensor fusion and point grouping was applied in three cases: using only the first sensor, using only the second sensor, and using both sensors. Because of the generality of the algorithm, there was no need to program these three types of operations differently.

The results of point grouping using only one sensor are presented in the figure 9. The objects are only partially reconstructed, due to the limits of the point of view and the partial occlusions. Splitting of an object in two components can be also a problem, as shown in figure 9, b).

The results of the sensor fusion algorithm are shown in figure 10. The same results are projected in the left image plane of both stereovision sensors, for a qualitative comparison and analysis. The results of the fusion show a clear improvement of the detection.



a) Sensor 1



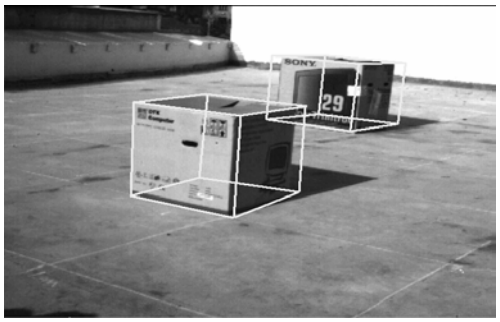
b) Sensor 2

Figure 9. Object detection using a single stereovision sensor.

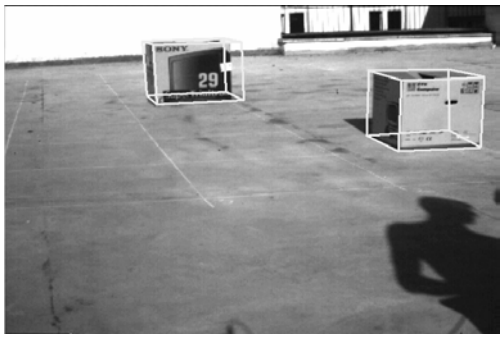
7 Conclusions

A method for spatial grouping of 3D points resulted from multiple stereovision sensors was presented. The algorithm treats the problem of point grouping and of point fusion in the same time. The main problem that needed to be faced by the system was the uneven distribution of the 3D points generated by stereovision, distribution which needed to be compensated before joining the point sets and grouping them into objects. After the distribution was

compensated and the sets fused, the grouping became a simple problem of point labeling.



a) Fusion result projected on the left camera of sensor 1



b) Fusion result projected on the left camera of sensor 2

Figure 10. Results of the sensor fusion

The results were conclusive: multiple views of a scene generate better detection, and a larger field of view. The accuracy of the calibration process, which is performed with respect to a unique coordinate system, is essential to the success of the fusion routine.

The algorithm has a high degree of generality. The same grouping routine works the same with any number of sensors, including a single one. This provides a better resistance to errors, and also a great possibility for system extension.

References

[1] J.Y. Bouguet, "Camera Calibration Toolbox for Matlab", *MRL-Intel Corporation*, USA, August, 2003, http://www.vision.caltech.edu/bouguetj/calib_doc/.

[2] N. Kämpchen, U. Franke, R. Ott, "Stereo vision based pose estimation of parking lots using 3D vehicle models", *Proceedings of the IEEE Intelligent Vehicle Symposium*, Versaille, France, June 2002.

[3] E. B. Meier, F. Ade, "Object Detection and Tracking in Range Image Sequences by Separation of Image Features", *IEEE International Conference on Intelligent Vehicles IV'98*, Stuttgart, Germany, pp.176-181, October 1998.

[4] S. Nedevschi, T. Marita, M. Vaida, R. Danescu, D. Frentiu, F. Oniga, C. Pocol, D. Moga, "Camera Calibration Method for Stereo Measurements", *Journal of Control Engineering and Applied Informatics (CEAI)*, Bucuresti, Romania, Vol.4, No. 2, pp.21-28, July 2002.

[5] S. Nedevschi, T. Marita, R. Danescu, F. Oniga, D. Frentiu, C. Pocol, "Camera Calibration Error Analysis in Stereo Measurements", *microCAD International Scientific Conference,* Miskolc, Hungary, pp. 51-56, March 2003.

[6] S. Nedevschi, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, "Real-Time Extraction of 3D Dynamic Environment Description Using Multiple Stereovision Sensors", *Proceedings of International Conference on Computer, Communication and Control Technologies CCCT'03*, Orlando, Florida, USA,Vol.3, pp.520-524, Aug. 2003.

[7] E. Trucco, E. Verri, *Introductory techniques for 3D Computer Vision*, Prentice Hall, New Jersey, 1998.

[8] J.Weber, D. Koller, Q.-T. Luong and J. Malik, "An integrated stereo-based approach to automatic vehicle guidance", *Fifth International Conference on Computer Vision, Collision Avoidance and Automated Traffic Management Sensors*, Cambridge, Massachusetts, pp.52-57,June 1995.

[9] T. Williamson, *A High-Performance Stereo Vision System for Obstacle Detection*, doctoral dissertation, Robotics Institute, Carnegie Mellon Univ., Pittsburgh, 1998.