Object detection with single-camera stereo

J. McBride, M. Snorrason, R. Eaton, N. Checka, A. Reiter, G. Foil, M. R. Stevens
Charles River Analytics, 625 Mount Auburn St., Cambridge, MA 02138
[jmcbride,mss,mrs]@cra.com

ABSTRACT

Many fielded mobile robot systems have demonstrated the importance of directly estimating the 3D shape of objects in the robot's vicinity. The most mature solutions available today use active laser scanning or stereo camera pairs, but both approaches require specialized and expensive sensors. In prior publications, we have demonstrated the generation of stereo images from a single very low-cost camera using structure from motion (SFM) techniques. In this paper we demonstrate the practical usage of single-camera stereo in real-world mobile robot applications. Stereo imagery tends to produce incomplete 3D shape reconstructions of man-made objects because of smooth/glary regions that defeat stereo matching algorithms. We demonstrate robust object detection despite such incompleteness through matching of simple parameterized geometric models. Results are presented where parked cars are detected, and then recognized via license plate recognition, all in real time by a robot traveling through a parking lot.

Keywords: Surveillance, UGV, mobile robot, vehicle detection, text detection, object detection, 3D reconstruction, 3D shape, stereo, structure from motion, SFM, license plate recognition

1. INTRODUCTION

Mobile robot platforms with onboard vision systems have the potential to revolutionize parking lot surveillance and security. Such platforms will eventually be capable of automatically building maps of vehicle locations and performing rudimentary inspection to assess potential threats. The motivation for the following system was to provide a flexible, low-cost, real-time solution to automated parking lot surveillance while staying within the size, weight, and power constraints of even the smallest commercial robot capable of roaming a parking lot. In particular, our system does not require any expensive specialized 3D sensing equipment. Instead, it uses only a single low-cost video camera to perform 3D stereo reconstruction of the parking lot scene.

1.1 Previous work in this field

The system is based upon previous work performed in many areas. Image *feature tracking* has been studied at length, particularly by Shi, Tomasi and Kanade [1;2]. The field of multiple view geometry, particularly *structure from motion*, has been covered in extensive works by Hartley and Zisserman [3-8]. Stereo geometry and reconstruction has had a great number of contributors [9-12] and is a still evolving field. In addition, the epipolar rectification techniques of Fusiello and Pollefeys [13;14] have made single-camera stereo practical and effective. License plate detection is inspired by text detection techniques described in [15;16]. Finally, previous work on this system has been published in [17;18].

1.2 System Overview

The system consists of a USB "web-cam" connected to a miniature PC, both of which are mounted on a mobile robot. The camera is mounted so that it points in a direction normal to the direction of motion for the robot. This ensures that as the robot moves, the positioning of the camera emulates the disparity of a stereo rig (see Figure 1-1).

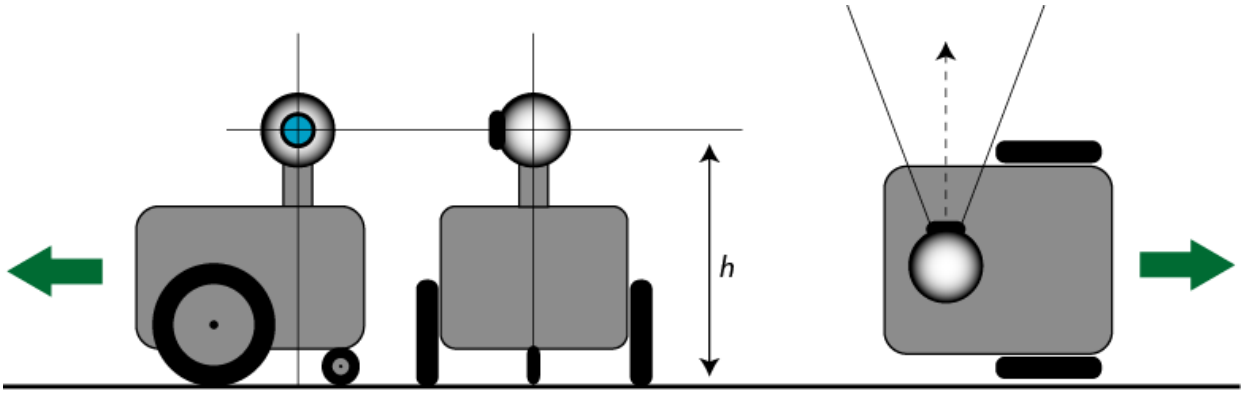

Figure 1-1: Camera mounting diagram

As the robot drives along a bank of parked cars in the parking lot, the computer reads images from the camera as well as odometry from the robot. The system begins by selecting an image from the video sequence.

A feature tracker then measures disparity in the scene caused by the robot's motion. Once sufficient disparity has been achieved, the system selects the current image and performs stereo reconstruction. This current image now becomes the previous image for the next stereo pair and the process continues. In order to mitigate the high volume of data produced by stereo reconstruction, the data is converted into a compact, low-resolution format, which gets added to a global model of the scene.

As the scene model is built, the system processes the data and identifies vehicles. The system also detects any license plates in the selected images and uses the stereo reconstruction to reject false alarms. Evidence from these two detection algorithms is then combined to complete a robust vehicle detection system, which is summarized in Figure 1-2.

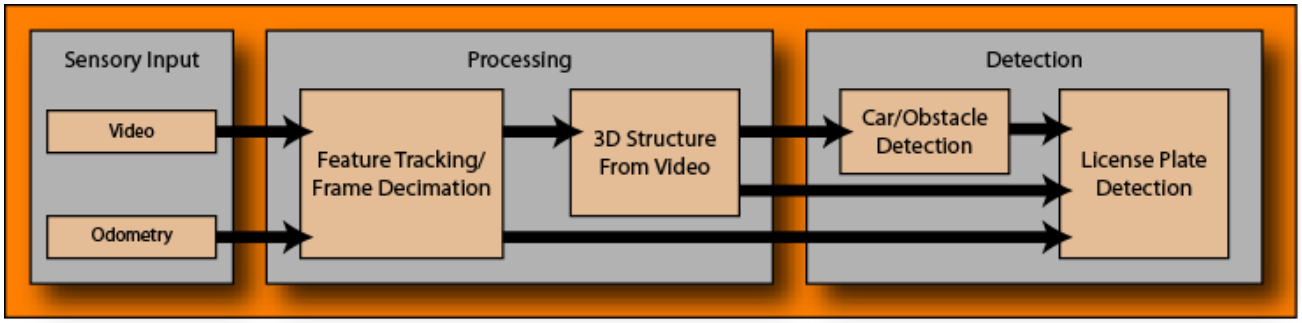

Figure 1-2: Main system diagram

2. ESTIMATING 3D STRUCTURE FROM VIDEO

The process of recovering the 3D structure of a scene from a video sequence is based on the idea of single camera stereo. A video camera is mounted sideways on a moving platform so that it roughly emulates the disparity of a stereo camera. An example result is shown in Figure 2-1.

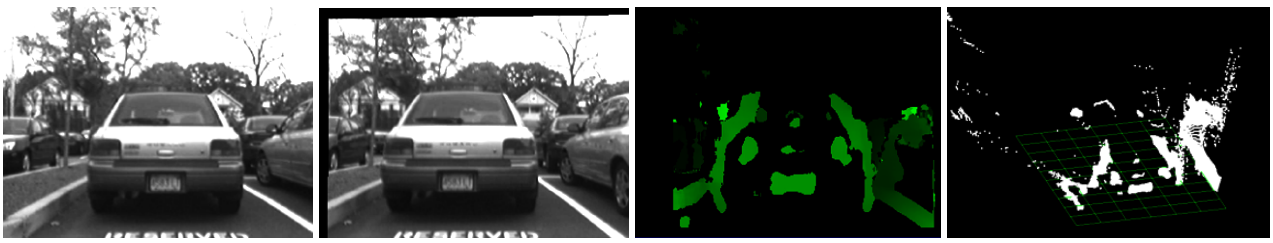

Figure 2-1: Stereo reconstruction from the two images on the left taken from a single camera in motion

This technique has been published in a previous paper [19] and is summarized here.

The first component in the single camera stereo pipeline is the feature tracker [1;2;20] which tracks image features across the image sequence which is captured from the mounted video camera. This provides the dual purpose of both decimating the image sequence into a series of images separated by a wide baseline, and also providing the basis for image alignment. We refer to the decimated images as *keyframes*.

Every pair of consecutive keyframes is then warped into an ideal stereo pair using an epipolar rectification technique described in [13]. This requires an exact measurement of the relative motion between the camera positions, which correspond to the pair of keyframes. Odometry measurements from the robot can provide a close approximation to this motion but the resulting image alignment is usually poor. Using corresponding features in the two images tracked by the aforementioned feature tracker, a *Levenberg-Marquardt* minimization technique [21] is used to iteratively adjust the motion parameters until the features become aligned in the y-direction. This ensures that the images can be passed on to a COTS stereo matching algorithm.

Finally, for each pixel in the image with a stereo correspondence, a metric 3D point is constructed using triangulation. A final transformation is applied to the 3D points to reverse the effects of the image rectification.

Figure 2-2 illustrates the single camera stereo pipeline.

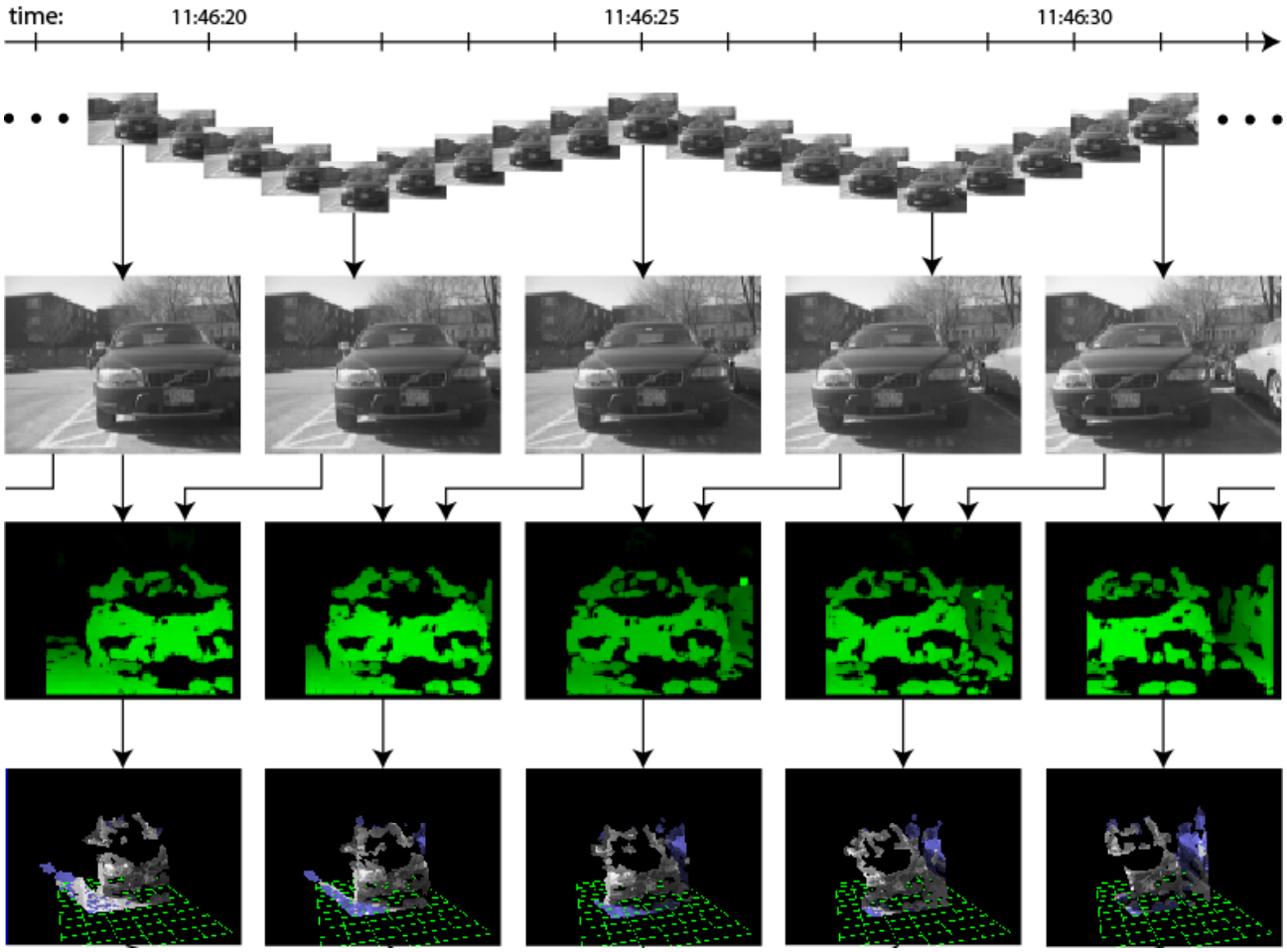

Figure 2-2: The 3D structure-from-video pipeline

3. STRUCTURE AGGREGATION/MANAGEMENT

One of the major challenges in dealing with dense stereo reconstruction is the sheer volume of data it creates. A 320x240 disparity image contains 76,800 pixels, 30-40% of which typically yield a valid 3D point in x, y and z creating a data output of roughly 80,000 floating point values. With the system generating a stereo frame every few seconds, this data can quickly add up, and the "slice" of the world that is reconstructed typically overlaps significantly with the one previous to it. This brings up the issue of registration, for while each individual stereo reconstruction can be highly accurate, proper placement of the points in world-space can be thrown off greatly by small errors in camera orientation.

Both of these issues are mitigated by converting the 3D data to a compact, low-resolution format. The scene structure is stored in a two dimensional histogram over the ground plane with a set bin size β. By reducing the data to bin counts and effectively discarding the vertical component, the data volume is constrained by the physical area over which the reconstruction takes place. And, since the bins are stored in sparse format, the system must only allocate enough memory to hold the structure generated so far. Additionally, points subject to small registration errors are still likely to fall into the same bin. This is ensured by setting the bin size to match the expected registration error of the system. Even though the vertical coordinate, or "height" is discarded, when a point is added to a bin, the bin keeps track of the minimum, maximum, mean and variance of the heights as well as the number of points added. It also tracks how many individual stereo frames contributed to that bin. These statistics preserve a great deal of descriptive information about the compressed 3D data while maintaining a constant storage overhead.

The process begins by translating each frame of stereo structure data from camera coordinates to world coordinates. In camera coordinates, the camera is located at the origin and is pointing in the direction of the Z-axis. To register, the points $\mathbf{M}_i$ are transformed by the camera's position $\mathbf{T}_c$ and orientation $\mathbf{R}_c$ in world-coordinates.

$$\hat{\mathbf{M}}_i = \mathbf{R}_c \mathbf{M}_i + \mathbf{T}_c \qquad 3\text{-}1$$

The bin location for the transformed point is then identified by scaling the point by the bin size and rounding to the nearest integer.

$$B(\mathbf{M}_i) = \text{round}\left(\frac{1}{\beta} \cdot \mathbf{M}_i\right) \qquad 3\text{-}2$$

If no bin has yet been created at that location, then a new bin is created and initialized using that point. If the bin already exists, then the statistics for that bin are updated with the incoming point. Figure 3-1 shows the binning process.

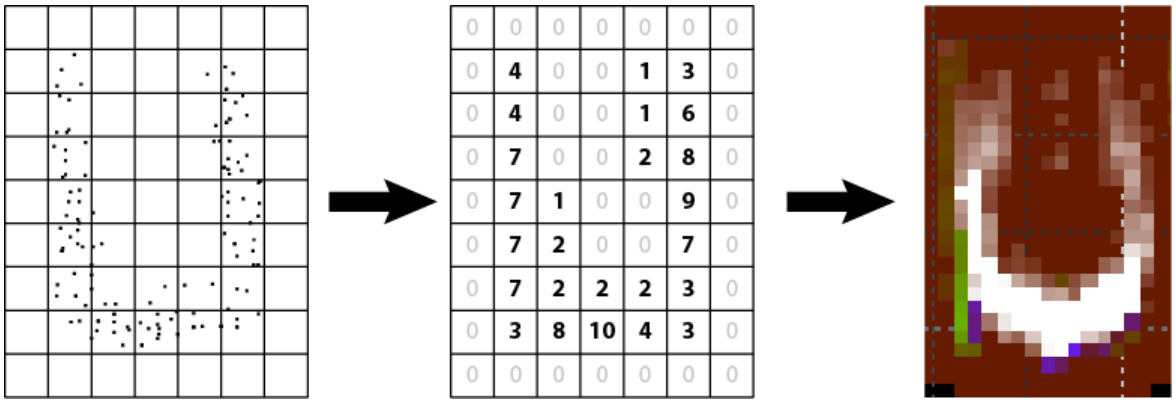

Figure 3-1: 2D histogram conversion process

Because the data is compressed vertically, vertical surfaces become most prominent in the histogram, which is ideal for detecting vehicles. Also featured more prominently are portions of the reconstruction that overlap many times over several consecutive stereo frames. This tends to make the true structure stand out even amidst uncertainty in the registration. In addition, the count of contributing frames to each bin provides another measurement of this property, but unbiased by total bin counts. The resulting model of an entire bank of vehicles can be seen in Figure 3-2.

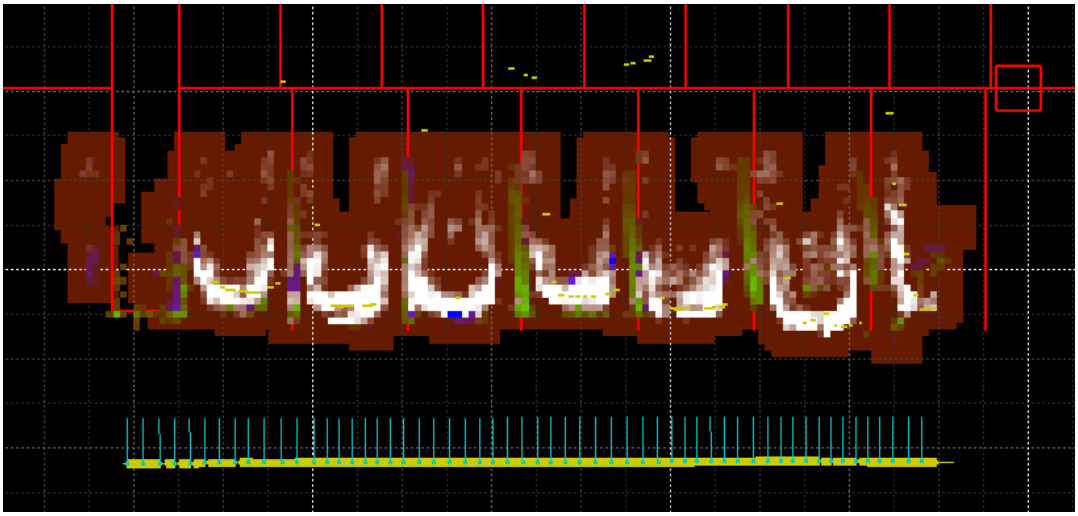

Figure 3-2: Example results of compact global reconstruction overlaid on a parking lot map

4. DETECTING VEHICLES AND OBSTACLES

The function of this component of the system is to take the compact representation of the 3D scene structure and segment out vehicles and any other objects which should be avoided (such as curbs, light posts, trees, etc.). A logic-based signal processing approach is used to analyze and segment the data. As described above, the compact data structure consists of a sparse 2D histogram with each bin storing statistics about the original 3D points that went into it. Because the bins are constantly being updated as the robot travels along its path, care must be taken not to analyze bins which may still change. Thus the detection algorithm only processes data that is out of the "visual range" of the camera.

This detection technique was formed after observing the output of the scene reconstruction component. Images of the vehicles only ever show the sides and visible end of the vehicle and never the top, bottom or hidden end. It follows that only these three visible parts of a vehicle could be reconstructed. This explains the characteristic "U" shape of most reconstructed vehicles. Given this expected shape, the detection technique attempts to detect vehicles by identifying and then assembling the simple components of left side, right side and front.

4.1 Structure Culling

The first step in this logic-based detection technique is to divide up the bins according to what kind of structure they represent. This is done by applying a set of thresholds to the maximum height value $h_{\max}$ of each bin.

$$\text{structure type} = \begin{cases} [\text{vehicle}], & h_{\max} > h_v \\ [\text{obstacle}], & h_o < h_{\max} < h_v \\ [\text{ground}], & h_{\max} < h_o \end{cases} \qquad 4\text{-}1$$

Bins labeled as "ground" are completely ignored. Bins labeled as "obstacle" or "vehicle" are clustered together to form a "no-go" region for the robot. This makes it easier for the robot to avoid curbs and parked vehicles in future navigation. Bins labeled "vehicle" are the only ones processed in the remainder of the detection technique.

4.2 Filtering

The next step is to create a set of 1D signals that represent the vehicle data. Two filters are used with one filter response for every keyframe. At each position, a mask is applied to sum over the bins. The first is an edge detector with an excitatory region on the right and an inhibitory region on the left. The positive and negative extrema of this function tend to correspond to the left and right side of vehicles respectively. Figure 4-1 shows examples of applying the edge detector mask with a positive peak shown on the left and a negative peak shown on the right.

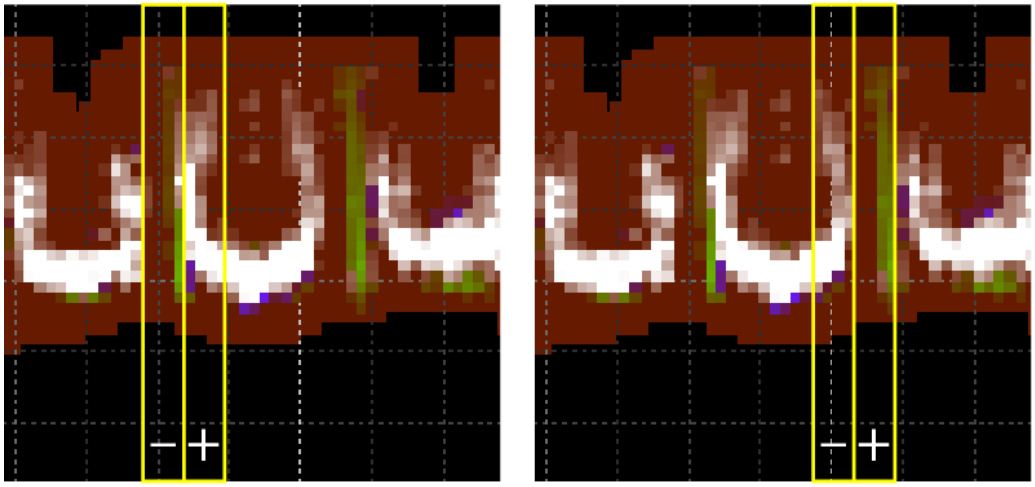

Figure 4-1: Convolution with edge detector mask

The second filter is a simple box filter which helps to distinguish between the sides and the front of a vehicle and also the space in between vehicles. It is applied in a similar manner as the edge detector mask. Plots of these two response functions are shown in Figure 4-2 along with edge detector peaks.

Figure 4-2: Filter Response Plots

4.3 Detection

Detection is based on the assumption that a vehicle is usually represented as the region between a rising (positive) edge peak and a falling (negative) edge peak as this generally corresponds to pairing the left side of a vehicle with the right side. Making no prior assumptions about which edges are paired with which, every rising edge is paired with every falling edge that is located further along the robot path. Then, the distance between the two edges is computed for every pair and those pairs which are too big or too small to be a vehicle are rejected. The remaining pairs are considered possible vehicles unless they overlap in which case, one must be rejected. This often occurs when one of the pairs spans a gap between two vehicles. These can then be rejected by comparing the total box filter response between the two edges for each pair and choosing the one with the greater. Since one of them usually spans a vehicle and the other one spans

the gap between vehicles, this will results in the correct detection being chosen. Figure 4-3 shows the process of pairing rising and falling edges.

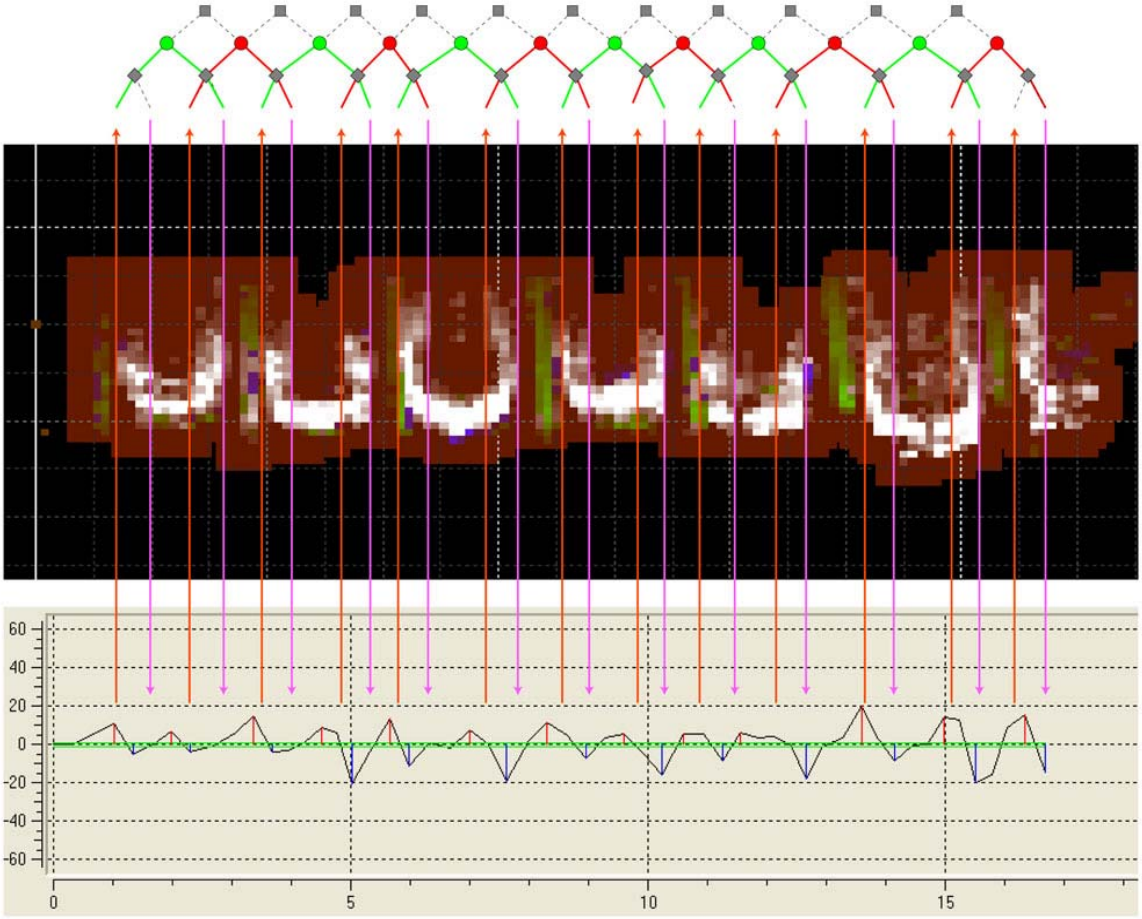

Figure 4-3: Vehicle detection process

Pairs marked by grey squares are too large and grey diamonds are too small. The remaining pairs in red and green are an acceptable size but they overlap. The red pairs span a gap between vehicles and will be rejected when compared to the box filter response between the red pairs. Figure 4-4 shows the final detection results.

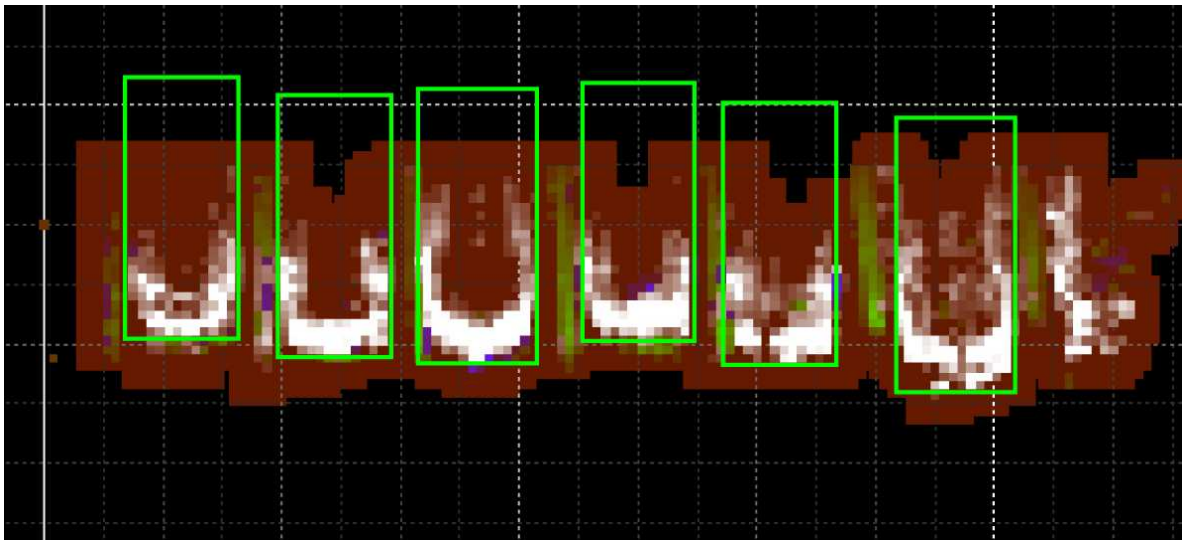

Figure 4-4: Final vehicle detections

5. DETECTING LICENSE PLATES

One application that usually accompanies vehicle detection is license plate detection and identification. One way to detect license plates is to use a text detector. However, there is usually a tradeoff between detection rate and false alarm rate when there are many distractors in the scene such as trees or other text such as road signs. Fortunately, we can use the results of the 3D stereo reconstruction to introduce a context-based verification component to the process. Thus, we use a two-stage approach. First, a text detector is applied to each selected image to identify text-like regions. Second, the 3D structure (see Section 2) of each text-like region is examined. If the 3D structure is consistent with the geometry of a license plate, then the region is accepted as a license plate.

5.1 An Overview of Our Text Detector

The text detector used here finds text features based on a series of fast 1D operations performed on horizontal scan lines. The algorithm runs on a grayscale image (for road signs, the image consisting of the minimum of the RGB components is effective) and is designed to run in real-time. It is based on the premise, inspired by [15], that scan lines cutting though areas of text have intensity frequencies marked by a series of similar peaks or valleys. The first part of the algorithm detects such frequencies and groups them into candidate text regions. In the second part, these regions are checked based on their shape, compactness, and orientation with respect to neighboring regions. Regions not meeting the required criteria are filtered out and the remaining regions are tagged as text. This technique was published in a previous work [16] and some example results are shown below in Figure 5-1.

Figure 5-1: Example results from the text detector

On one set of 119 images taken in urban environments, the text detector achieved an average detection rate of about 80% with a false alarm rate of about 53%. This high false alarm rate is dealt with as explained in the next section.

5.2 False Alarm Rejection

As discussed above, the text detector can be liberal in its hypotheses because an examination of the 3D structure at the hypothesis will reject false alarms. The 3D structure of a given stereo image is used to determine if the structure within the detected region is consistent with the geometry of a license plate. For example, the depth of the region should be uniform (because license plates are flat), the size and aspect ratio of the region should be consistent with the actual measurements of license plates, and the distance of the region from the camera should not be more than a few meters in the current operating scenario. Figure 5-2 shows the results of the license plate detector. The red box on the license plate indicates a true license plate, while the blue box in the upper right corner shows a false alarm from the text detector that was eliminated during the structural examination.

Once the license plates have been detected, it would be trivial to hand the chips off to a commercial OCR algorithm to read and identify the license plate number.

Figure 5-2: Stereo imagery rejects false alarm from text detector

6. RESULTS

To evaluate the performance of our system, we collected data during 10 separate robot runs through the parking lot. Each run consists of the robot driving by one bank of vehicles. The runs were collected at different times of day, under different lighting conditions (flat light vs. direct sunlight and glare). The data from each run was stored for subsequent processing.

Using ground truth measurements it was observed that reconstruction accuracy was not symmetric in all directions. In the direction of viewing, there was observed a maximum error of about 0.5 meters (1.64 feet). In the lateral direction on the other hand, there was observed a maximum error of only 0.3 meters (0.98 feet).

Over the 10 runs, the robot passed a total of 130 vehicles. Our detection algorithm found 106 vehicles correctly (each side of the bounding box within 0.3 meters, (~1 foot), of a visible surface of the vehicle), for a correct detection rate of 81.54%. There were 6 false alarms (bounding boxes more than .3 meters from the visible surface of a vehicle) over the same dataset—a false alarm rate of just 4.62%. Passing images of the 106 detected vehicles through the license plate detector, 82 plates were correctly detected, resulting in a correct detection rate of 77.36%. Only 3 false alarms occurred, resulting in a false alarm rate of 2.83%.

Under ideal conditions (minimal glare, cars not too close together) the system performed extremely reliably, detecting nearly every vehicle correctly. Figure 6-1 shows an example series of successful detections.

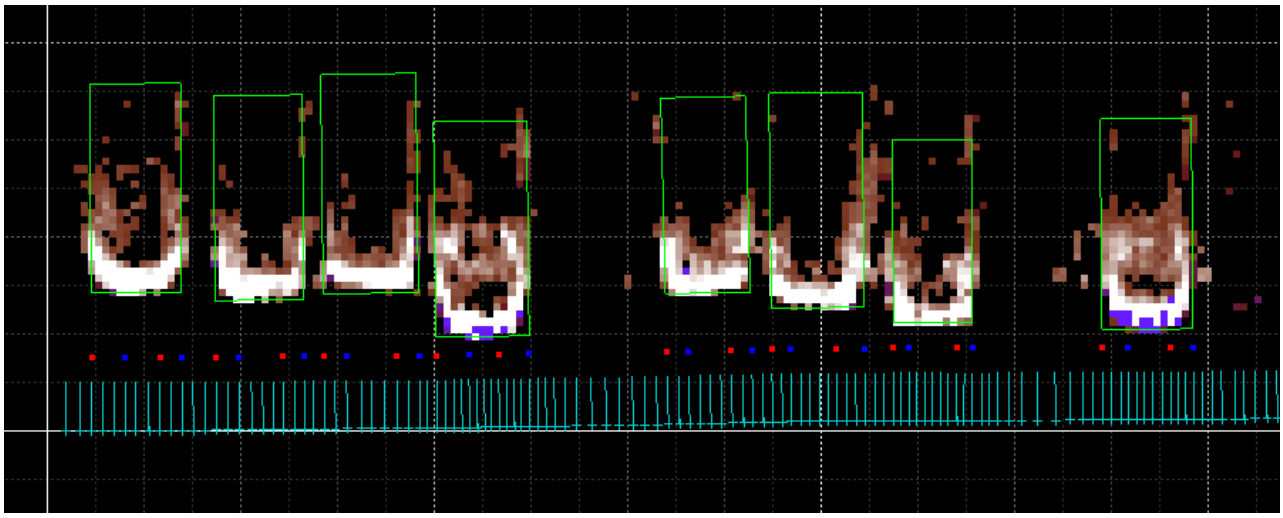

Figure 6-1: Successful vehicle detections

License plate detection also performed very well under these conditions. Figure 6-2 shows the locations of detected license plates in blue overlaid on vehicle detection bounding boxes. Note that almost every license plate is detected near the back center of each vehicle, where one would expect to find a license plate.

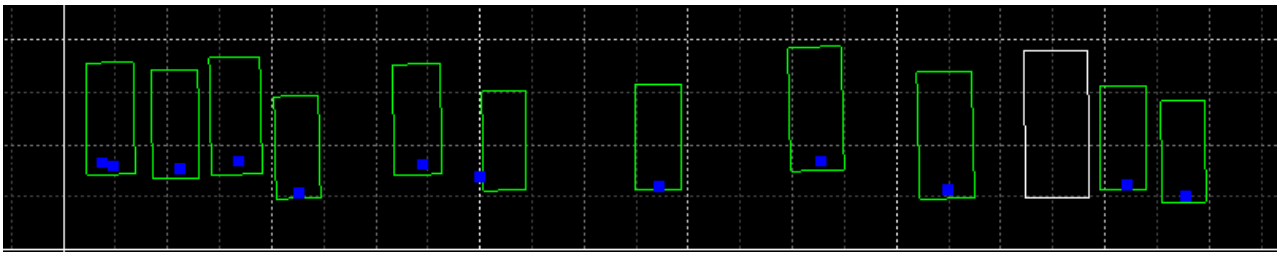

Figure 6-2: License plate detection locations overlaid on vehicle detection boxes

One type of failure condition is when vehicles are parked so close together that the space separating them falls below the bin size of the vertical histogram representation of the data. When this occurs, there is no way for the system to distinguish between the two vehicles. This is the case in Figure 6-3 where the two vehicles seen in the left image cause the false detection seen in the center of the right image.

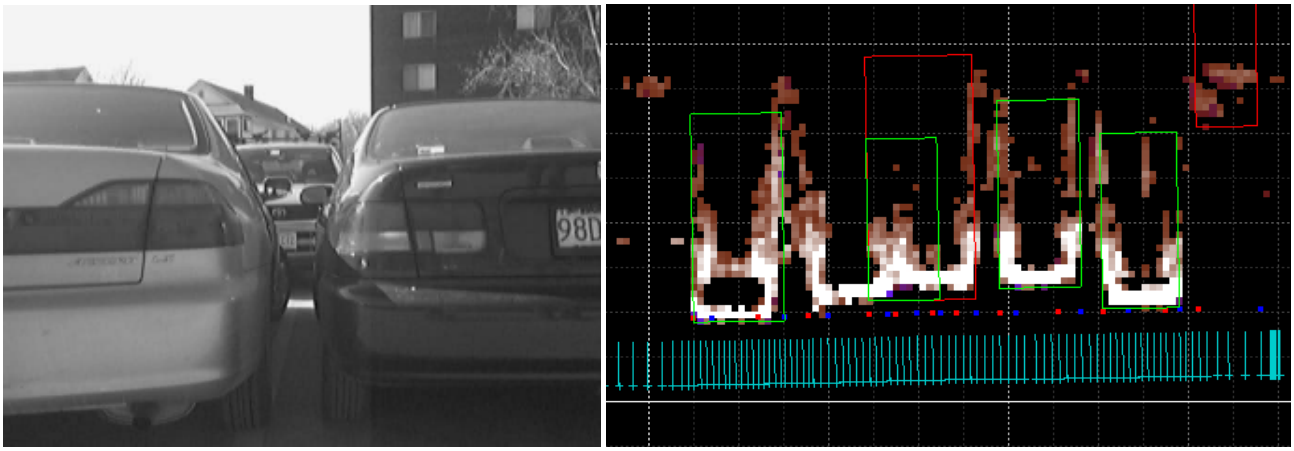

Figure 6-3: Two vehicles parked too close together (left) cause detection failure (right)

Finally, generally poor imaging conditions can cause the system to be less reliable by reducing the quality of the reconstruction. Figure 6-4 shows an example image on the left from a sequence which exhibited a significant amount of solar glare. The reconstruction and detections shown on the right suffer as a result.

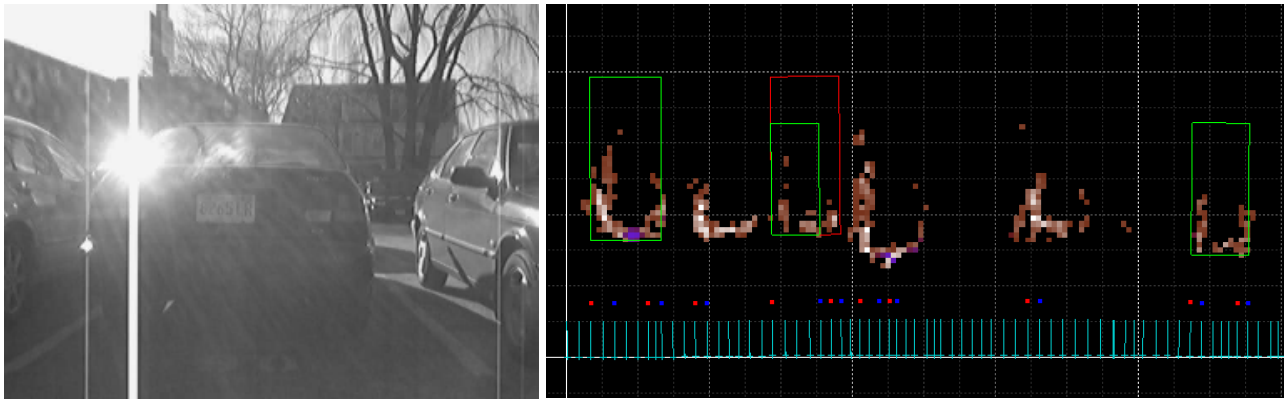

Figure 6-4: Solar glare causes problems for stereo reconstruction

7. CONCLUSIONS

We have created an effective system for parking lot monitoring. The accuracy of scene reconstruction is impressive and the detection rate for vehicles and license plates shows promise for a viable commercial system.

However, there were also many shortcomings revealed in some of the third party software packages used in this system. In particular, it would make sense to develop a custom stereo matching algorithm specifically geared to single camera stereo rather than for fixed baseline stereo.

Finally, because the detection component of the system was developed last, there exists a significant amount of information contained in the reconstruction model that does not get used. A more sophisticated and statistics based detection algorithm that makes full use of all the information available would most likely result in greatly improved detection results. Further development could also possibly yield greater robustness against commonly encountered adversarial conditions.

REFERENCES

1. C. Tomasi and T. Kanade, "Detection and Tracking Point Features," Carnegie Melleon University,1991.
2. J. Shi and C. Tomasi, "Good Features to Track," IEEE Conference on Computer Vision and Pattern Recognition 1994, pp. 593-600.
3. P. Beardsley, P. H. S. Torr, and A. Zisserman, "3D Model Aquisition from Extended Image Sequences," in *European Conference on Computer Vision* Springer--Verlag, 1996, p. 683.
4. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision* Cambridge Press, 2000.
5. W. Triggs, P. F. McLauchlan, R. I. Hartley, and A. Fitzgibbon, "Bundle adjustment for structure from motion," in *Vision Algorithms: Theory and Practice*. W. Triggs, A. Zisserman, and R. Szeliski, Eds. 2000.
6. R. Hartley, "Cheirality invariants," DARPA Image Understanding Workshop Morgan Kauffman, 1993, pp. 743-753.
7. R. I. Hartley, "In Defense of the Eight--Point Algorithm," *PAMI*, vol. 19, pp. 580-593, 1997.
8. R. I. Hartley and P. Sturm, "Triangulation," *Computer Vision and Image Understanding*, vol. 68, no. 2, pp. 146-157, 1997.
9. O. Faugeras, B. Hotz, H. Mathieu, T. Vi'eville, Z. Zhang, P. Fua, E. Th'eron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, and C. Proy, "Real-Time Correlation-Based Stereo: Algorithm, Implementations and Applications," INRIA,Technical Report 2013, 1993.
10. T. Kanade and M. Okutomi, "A Stereo Matching Algorithm With an Adaptive Window: Theory and Experiment," *PAMI*, vol. 16, pp. 920-932, 1994.
11. R. Szeliski and D. Scharstein, "Symmetric Sub-Pixel Stereo Matching," ECCV, II ed 2002, p. 525.

12. M. Z. Brown, D. Burschka, and G. D. Hager, "Advances in Computational Stereo," *PAMI*, vol. 25, no. 8, pp. 993-1008, 2003.
13. A. Fusiello, E. Trucco, and A. Verdi, "Rectification with unconstrained stereo geometry," British Machine Vision Conference 1997.
14. M. Pollefeys, "A simple and efficient rectification method for general motion," International Conference on Computer Vision (ICCV) 1999, pp. 496-501.
15. B. K. Sin, S. K. Kim, and B. J. Cho, "Locating Characters in Scene Images using Frequency Features," Int. Conf. on Pattern Recognition 2002.
16. S. Holden, M. Snorrason, T. Goodsell, and M. R. Stevens, "Autonomous Detection of Indoor and Outdoor Signs,", vol 5804 ed 2005.
17. T. Goodsell, M. Snorrason, M. R. Stevens, B. Stube, and J. McBride, "Ego-location and situational awareness in semi-structured environments," Unmanned Ground Vehicle Technology V, 5083 ed Orlando, FL, USA: SPIE, 2003.
18. J. McBride, M. Snorrason, T. Goodsell, R. Eaton, and M. R. Stevens, "3D scene reconstruction: why, when, and how?," SPIE Defense & Security, 5422 ed Orlando, FL: 2004.
19. J. McBride, M. Snorrason, T. Goodsell, R. Eaton, and M. Stevens, "Single Camera Stereo for Mobile Robot Surveilance," IEEE Conference on Computer Vision and Pattern Recogniton, Workshop on Advanced 3D Imaging for Safety and Security 2005.
20. B. K. P. Horn and B. G. Schunck, "Image Flow Segmentation and Estimation by Constraint Line Clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 10, pp. 1010-1027, 1989.
21. D. M. Bates and D. G. Watts, *Nonlinear Regression and Its Applications*. New York: Wiley, 1988.