Stereo Vision in Small Mobile Robotics

Bryan Hood

Abstract
The purpose of this stereo vision research was to develop a framework for stereo vision on small robotic platforms. The areas investigated were matching easily identifiable objects, pixel to pixel correspondence, and matching of image features. Finding the location of objects which could be found in each image separately by color filters or object recognition gave the best results but lacked information about the environment and background. Although useful for some task based applications, a dense disparity map containing information about all objects and background is useful for navigating an environment. Pixel to pixel matching gave significantly more information about all objects in an image, but was very susceptible to noise and was much more computationally expensive. Matching features was also computationally expensive but had less noise. The last two methods need further work to reduce noise in order to be used in practice.

Introduction
Stereo vision is a useful technique for gathering distances of objects and features in an environment. Applications are far ranging from robotics to astronomy. Stereo vision provides positional data that can be found from geometric relations among multiple unique viewing points. Generally two viewing frustums are used to gather enough information to create a disparity map of the target, however more may be used. A disparity is an image whose pixels' intensities correlate to depth. For robotic applications, disparity maps are particularly useful because they contain information required to identify and track features in an image. Past research involving stereo vision in robotics shows that it is possible to create full stereo maps but limitations include the inability to perceive long distances, intensive processing requirements, and variability due to lighting [1].

Two cameras looking forward with a finite spacing between them will see an object placed at infinity at the same location. As that object moves closer there will be a disparity in the x location of the object between the two images. With respect to the left image, the object in the right image will be further to the left. This disparity correlates to distance, and knowing the position of the cameras, the object's position in space can be measured relative to the cameras.

Multiple object scenarios must be taken into account for stereo vision to work properly. For example, a viewer standing at the base of a tree must observe the tree as multiple connected units in order to see a gradient in distance. If the tree was treated as a single object and its centroid positions were compared, there would be a significant amount of error in the distance to the top and the base of the tree. Therefore, it is useful to match horizontal scan lines of the tree but to preserve a connection between adjacent scan lines. Also, in terms of overlapping objects, it is not possible to determine if the objects are overlapping or connected unless there is sufficient texture in the image. This requires matching key features of each object between the two images and also finding features of one object that exist only in one image.

The goal of this research is to create a platform for small mobile robots to be able to identify locations and distances of target objects. This is particularly useful for task based robotics, where there is a defined problem with low variability. However, it is also the goal of this research to attempt to go beyond that to create disparity maps of an entire image to provide more functionality in dynamic environments. Funding was provided by the University Scholars Program.

Methodology

The setup for the project required two cameras and a computer with Microsoft Visual Studio, Matlab and Intel's Open CV. The particular cameras used were Creative Notebook Web cams (Model No. VF0250). Each camera was interfaced via USB to the computer.

The cameras each have manual focus rings for calibration. The focus calibration method used involved applying a first difference image filter to detect edges. The focus rings were then rotated in order to maximize the values of the filter output to obtain crisp edges. Each camera was fixed to a mounting panel via the clips on the camera housing. This ensured that the focal points rested in a line parallel to any line in the images being viewed. The tolerance of the factory edges on the camera housings was high enough that further calibration in that dimension was not required. Next, the cameras were focused on a dot and rotated until the dot existed at the vertical midpoint of the image. A horizontal line was then added to the dot and the cameras were again rotated until the line was horizontal in the image and the dot was still at the vertical midpoint. This procedure fixed the cameras in proper orientation and focus required to continue the experiment. No lens corrections were attempted as the camera quality was not high enough for it to be worthwhile.

The first step in the process was to apply stereo vision to easily identifiable objects on a plain background. For example, red tags on a white background. Images were taken of static objects from both cameras. Next, a simple color filter was used to isolate the objects from the background. A scan line system was used to compare the centers of the objects from each image to determine its location. This process was repeated with the same distinctly colored objects on a non-uniform background. Filtering and processing by Awcock was used to identify the objects [2].

Finally, images pairs of common household objects were taken in a household environment in order to produce full scale disparity maps. The resolution of the images produced by the web cam was 320x240 pixels. The data stream was in the JPEG format and had high lossy compression. To try to overcome loss, images were blurred in order to smooth out the sharp edges caused by the compression. However, images from the Stanford AI Lab were finally chosen [3]. The methods used for creating disparity maps included matching scan line segments, matching scan line segments with high edge values, matching blocks of pixels and matching blob features between the images.

For the scan line segments and blocks of pixels, the matching algorithm would calculate the magnitude of the match for each position along the horizontal scan. The match was defined as the minimum discrepancy between the translated pixels and the target pixels. The distance of translation in pixels was used as the disparity value and placed into the disparity map image in the location of the found match. The sizes of the scan lines and blocks of pixels were changed in order to vary the results.

To match blob features, the images were converted to gray scale. Then one image was translated over the other while taking differences. If the difference fell within a thresholded value, it was flagged. This information was used to produce a binary image of the difference between the static frame and the translated frame. Another binary image was created the same way but the translated image was one pixel ahead of the previous. These two images were then averaged to create a new image. The averaged image had three states: 0, ½ and 1. An example of this image created in the scanning process is shown in Figure 1. Every time a section of pixels in a horizontal cross section had the value of ½ surrounded by 0's the disparity map was marked in that location corresponding to the horizontal shift. At the very end the markings were connected together to fill in the gaps between edges.

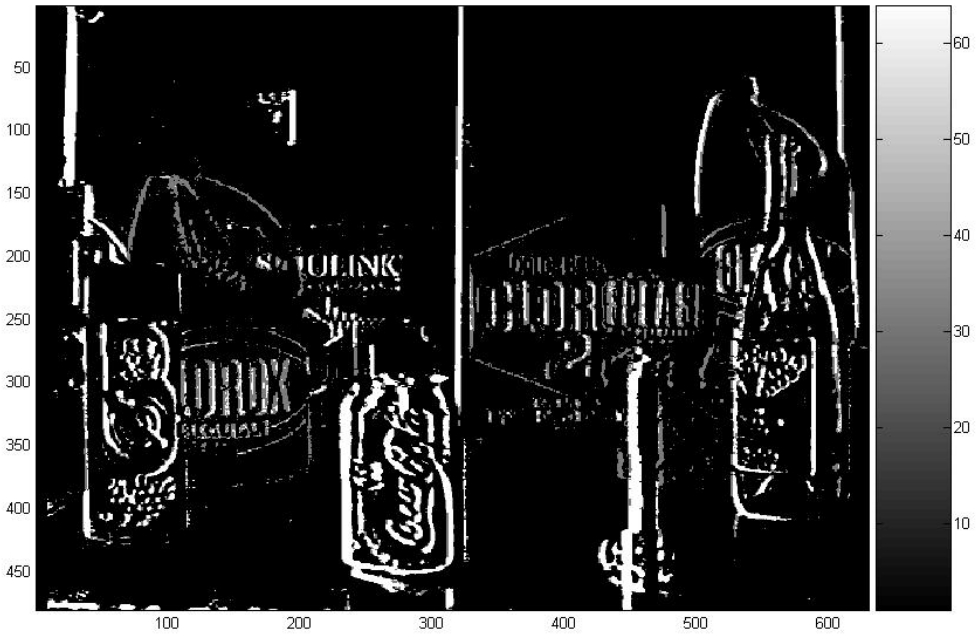

Figure 1. Image result from scanning process

Results and Discussion

Distinct objects of a specific color could correctly be ordered in terms of depth by isolating them by color, finding their location in each image and calculating how far the object was translated from the right frame to the left frame. Attention was not paid closely to exact values of position, only to relative position of objects. The key component to success for this type of matching was isolating the object from the background. As long as the color was distinct and could be filtered from the background with only discrepancies at the edges, the objects could always be matched and ordered correctly. However, using the scan line method, some scan lines of a single object would show a value differing values from a scan line above or below it. This allowed for gradients in proximity values along large objects but also gave errors in small objects when the distance to any point on the object was roughly the same. The highest difference between adjacent scan lines was a single disparity level. With that in mind, the results were easily smoothed and appropriated disparity levels were found for each object. This method worked on any range of backgrounds as long as the background did not interfere with the color filtering. Further use of this method could be applied to other filters that can isolate an object. These may include object recognition and blob detection. As long as the object can be recognized in each image, the disparity can be determined. The amount of processing required for this type of stereo is very low and can easily be accomplished on a small robotic platform.

The results were rather poor for matching segments and blocks of pixels to produce full disparity maps in non uniform environments. If the size of the segments or blocks was increased in order to get a better match, the resolution suffered. When the matched features were smaller, the matching error increased. This resulted in the matches coming in out of order and getting a wide range of disparities about the edges of features. A human viewer would in some cases be able to see a rough resemblance of the disparity map to the original image but only in few cases. Most cases resulted in the disparity values

being so jumbled that there appeared to be no correspondence between the original images and the disparity map.

Figure 2 shows the disparity map results from matching horizontal pixel blocks. Large pixel values represent close objects and small values represent far away objects. The background has significant amounts of noise where features were found in the image and matched incorrectly. The important features such as the lamp and table are present and identifiable as closer than the features on the wall, but the noise distorts the disparity map quite significantly. Figure 3 shows the “left” original image.

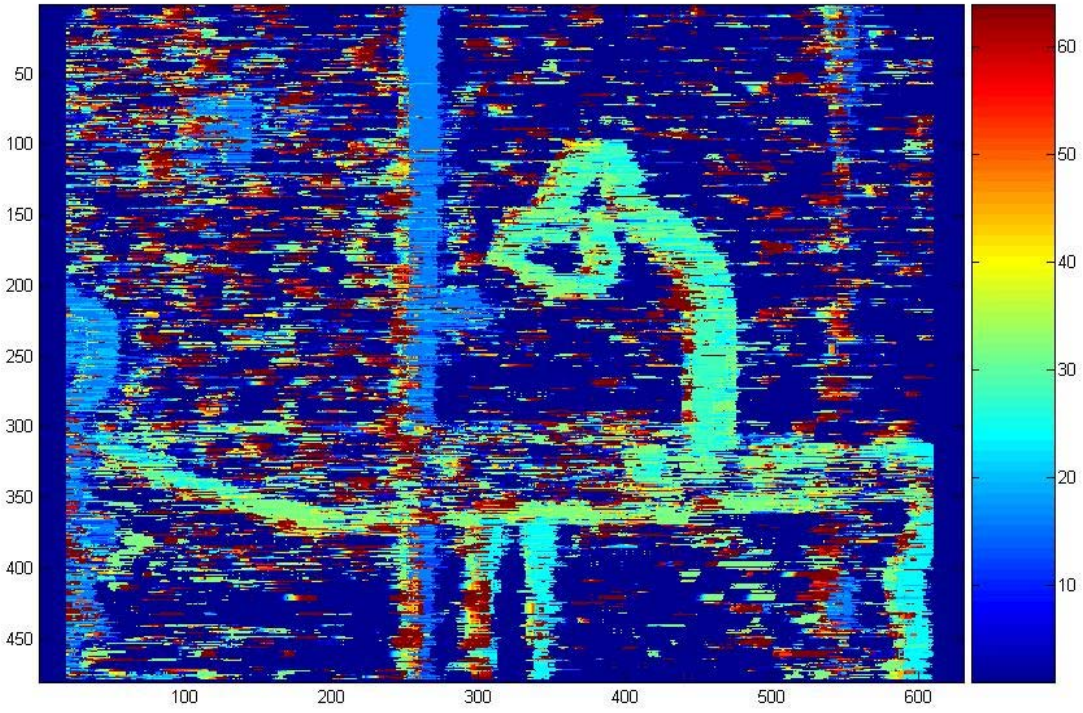

Figure 2. Pixel block matched disparity map

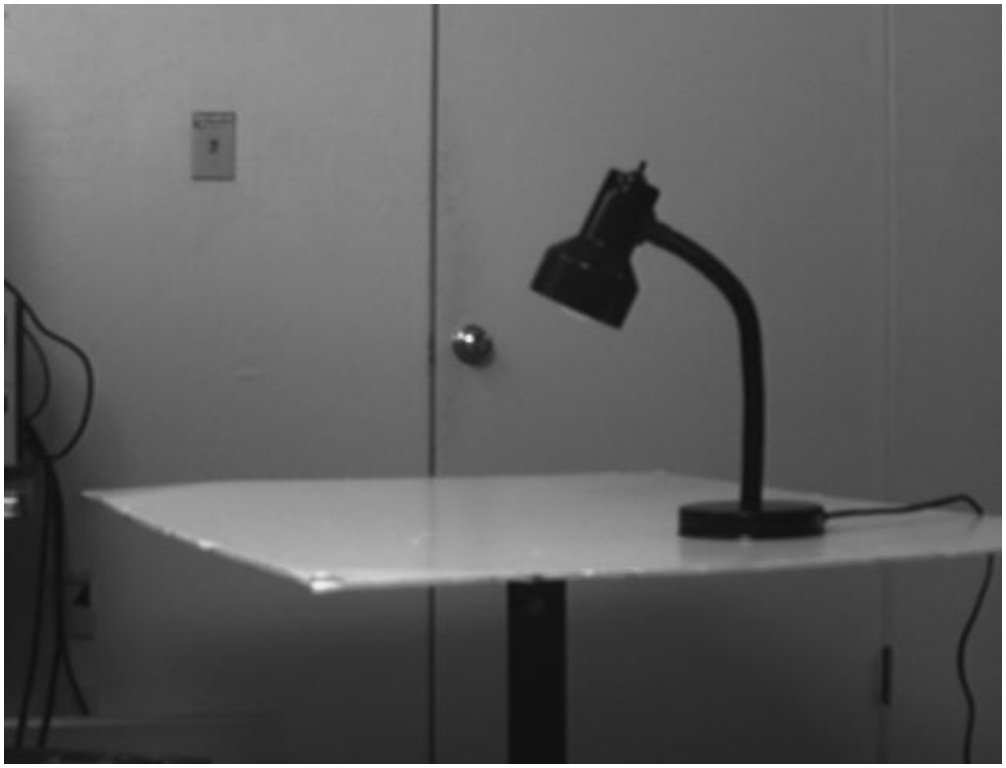

Figure 3. Original image
Source: S. Birchfield, "Depth Discontinuities by Pixel-to-Pixel Stereo”

Matching blob features however, had slightly better results. Although there was a significant amount of noise present in the disparity maps produced by this method, a human viewer could easily recognize the position of objects within the image. The most significant issue with this method was that filling in the blobs was difficult if its edge features were difficult to detect. This resulted in some scan lines extending very far beyond the edges of the blob and into others. Also, shifting the image distances of over 15 pixels started to produce disparities that did not actually exist in the image. The results given by this algorithm were far better than any of the previous methods for creating a full disparity map, however, they are still too noisy to be reliable on a robotic platform. The algorithm needs further development and the disparity map results need extensive filtering to be useful.

As seen in Figure 4, matching blob features was much cleaner than matching blocks of pixels. Objects can clearly be seen in order of position. However, there was still significant error introduced by not finding all of the blob boundaries. The difference can be seen in Figure 5 where the scan was done to the left as opposed to the right in Figure 4. These results verify Henkel's claim that feature based matching produces better results [4]. The “left” original image is shown in Figure 6.

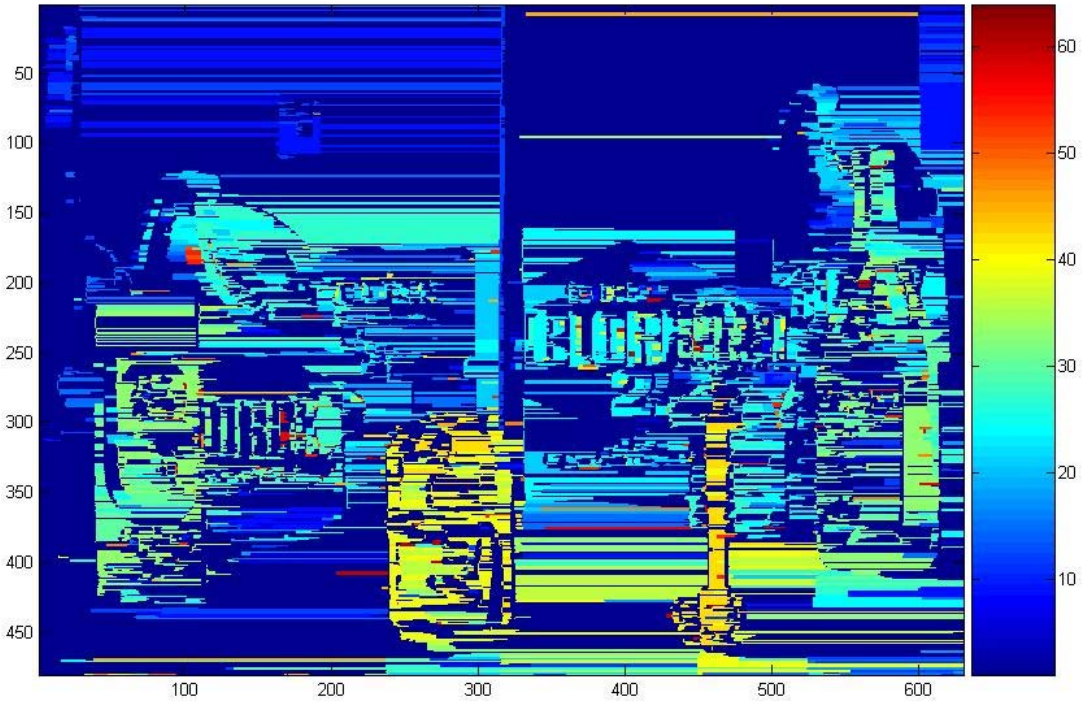

Figure 4. Disparity map using feature based mapping (Scan lines to right)

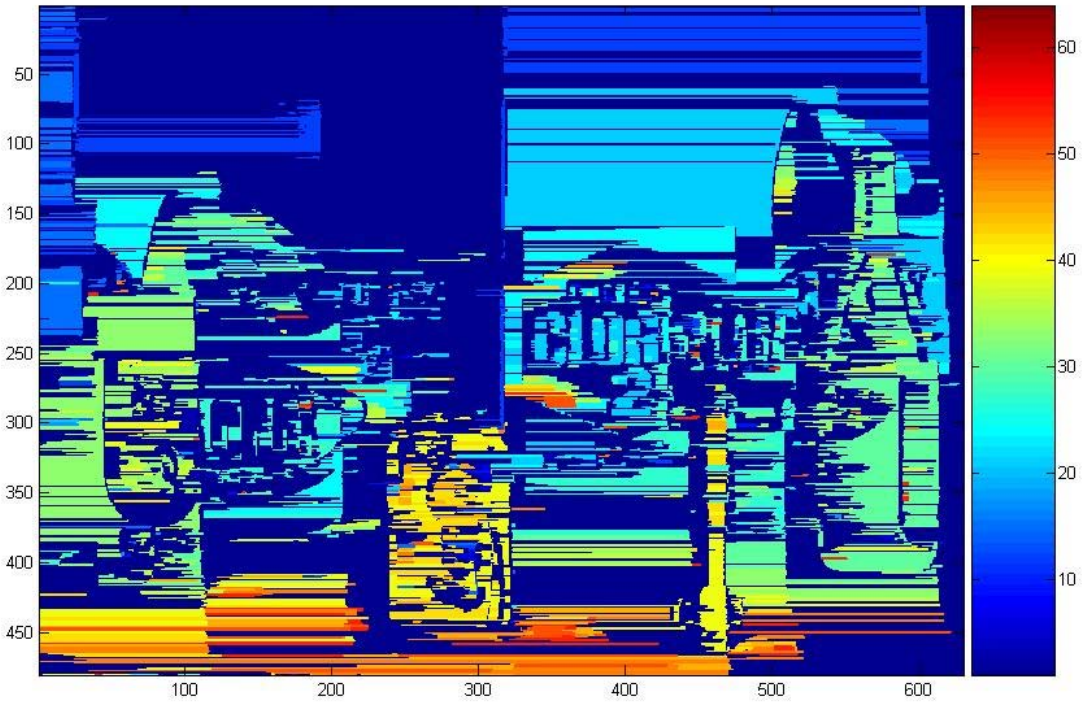

Figure 5. Disparity map using feature based mapping (Scan lines to left)

Figure 6. Original image
Source: S. Birchfield, "Depth Discontinuities by Pixel-to-Pixel Stereo"

Conclusion

The results confirm that finding a particular object's distance and isolating it by color, shape or other means is the most simple solution and gives the best results. However, in many cases a full disparity map is necessary for navigation of rooms or terrain. Lacking information about the position of obstacles is nearly the equivalent of being blind for a robot. This is where a trade off must be made and a user must evaluate the level of computational power that can be carried on the robot versus its goals. For full information of the cameras viewing frustum, the full disparity maps created by the pixel to pixel or feature to feature matching algorithms are necessary. Matching features gives a more dense disparity map with usually less noise. However, it is far more complicated and this may compromise frame rate [5]. The current results show that finding the distance to a specified object is the only part that can be used in practical means. Further experimentation is needed to reduce or filter noise to make the other algorithms useful.

References

[1] M. F. Ahmed, "Development of a Stereo Vision system for Outdoor Mobile Robots," M.S. thesis, University of Florida, 2006.

[2] G. W. Awcock and R. Thomas, *Applied Image Processing.* Houndmills: R. R. Donnelley & Sons Company, 1995.

[3] S. Birchfield, "Depth Discontinuities by Pixel-to-Pixel Stereo," [Online]. Available: http://vision.stanford.edu/~birch/p2p/. [Accessed Aug. 12, 2007].

[4] R. D. Henkel, “Fast Stereovision with Subpixel-Precision,” presented at IEEE 6th International Conference on Computer Vision.

[5] S. Florczyk, *Robot Vision: Video-based Indoor Exploration with Autonomous and Mobile Robots.* Weinheim: Wiley-VCH, 2005.