

# Robot Eyes Report

David Kenny, Jamie Kerr  
BSc (Hons) Applied Computing

March 2016

## Abstract

The team was given the task of creating a disparity map from a pair of stereo images using techniques that were talked about in class.

## 1 Introduction

The assignment required the team to acquire new knowledge of computer vision by producing their own disparity map from scratch using Matlab. We were encouraged to experiment with variables and publish our results of the findings. This report details the experiments that the team performed and the notable results that were acquired from them.

## 2 Background

A disparity map [4] is used to show the difference between a pair of stereo images. The disparity is used to calculate depth information from a combination of the images.

The disparity value is the value of the shift required to get the minimum sum of squared differences [1] for that pixel or section of the image.

Stereo pairs are images typically taken with only a slight difference in the distance of the cameras. Preferably the camera settings should be the same and the hardware should be the same as well to ensure that the images are similar.

Another reason that this image was selected for testing was due to their not being many pixels present to test.

## 3 Description of Algorithm

The team used a brute force method to gain their results using this method meant that the run time was extremely high due the fact that a disparity value had to be calculated for every pixel of the image.

This was a problem when the team was testing their algorithm. Also since we were learning at the same time it didn't help as a simple error could take in excess of 10 minutes to produce a result even on a 100 x 100 image.

## 4 Description of Images Used

The team used the stereo images provided on the VLE to create their program. The pair that was mainly used was the bookcase image shown in figure 1. It was selected as there are clear levels of depth between the different books and the team thought that they would produce a noteworthy result.



Figure 1: Bookcase

The other pair that the team used was the "Scene" pair which we had seen used a lot in the computer vision community [5] [3] for various benchmark tests including disparity maps. We believe that this is due to there being an obvious deviation of layers in the image. The image is also has a small resolution too which means that it won't take long to produce a result.

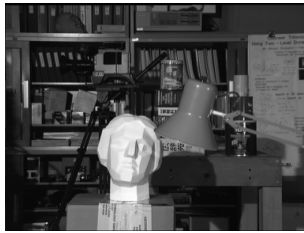


Figure 2: Scene

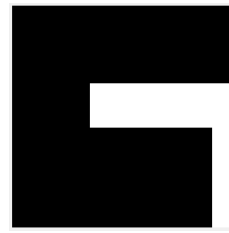


Figure 4: First Result

We used our own image that we took with an iPhone 6S as well with an iPad box and a bottle of coke in front of it to indicate clear layers of depth in the stereo pair.



Figure 3: Our Image

The team found that the SSD values were not correctly working since it was either producing a value of 0 or 255 as the final output this was confirmed when the image in figure 5 was produced.

We believe that the black band shown in figure 5 is due to the padding that was added to image.

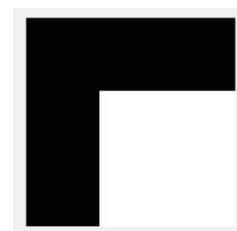


Figure 5: Second Result

The image was scaled down from a resolution of 3024 x 4032 to 150 x 200 using photoshop for testing purposes as it would take significantly less time to compute. On the left of figure 3 is the left and the right image overlaid. When the image was taken the team used a table to place the camera on to ensure that the position of the camera remained constant from taking the left and right image.

## 5 Results

The result shown in figure 4 is the result of running the test on the bookcases image taking the top 20 x 20 pixels of the image.

We discovered that the displacement values had to be converted into an 8 bit unsigned integer to be correctly displayed as a greyscale image. This then resulted in figure 6 being produced.



Figure 6: Third Result

After figure 6 was produced the team then ran a test using a 200 x 200 image. This test took over two hours to run on a modern laptop and the end result wasn't what the team were hoping to produce.

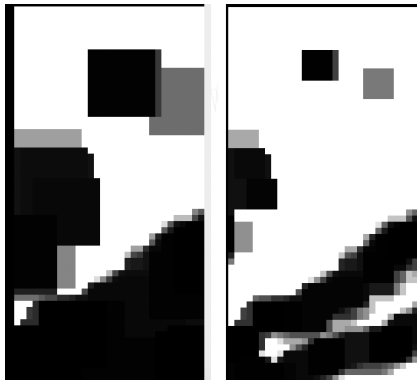


Figure 7: Search Window

Since the team couldn't find out much information from the corner of the books we decided to try using the scene image (figure 2) and focus on the section of the image where the lamp is in front of the head. The result that was produced from this was the image on the left of figure 7. The team were pleased with this result as there was a larger variation in values.

The result seemed pixelated the team believed that this was because of the search window size since we knew that decreasing the size of the window would increase the resolution and provide more clarity in the disparity

map however the trade off is that noise isn't handled as much.

The window size was reduced from 15 to 9 and the image on the right of figure 7 was produced. This seemed to be hopeful.

## 6 Conclusion

The team had never worked with matlab before and were pleased with what they were able to achieve within this short space of time.

The full source code for the teams work can be found in a git hub repository [2].

## 7 Future Work

If the team had more time available we would like to have been able to perform research on 3D imaging techniques and work on a way of creating an optimised algorithm that minimises computation.

## Acknowledgements

The team would like to thank Emanuele Trucco for his support throughout this module.

## References

- [1] Alessandro Verri Emanuele Trucco. 1998. *Introductory techniques for 3D - Computer Vision*. Prentice Hall.
- [2] Jamie Kerr. 2016. Matlabrobot. (2016). <https://github.com/jkerr123/matlabrobot>
- [3] Shawn Lankton Online. 2007. 3D Vision with Stereo Disparity. (2007).

- [http://www.shawnlankton.com/2007/12/  
3d-vision-with-stereo-disparity/](http://www.shawnlankton.com/2007/12/3d-vision-with-stereo-disparity/)
- [4] Jay Rambhia. 2013. Disparity Map. (2013).  
[http://www.jayrambhia.com/blog/  
disparity-maps/](http://www.jayrambhia.com/blog/disparity-maps/)
- [5] Stanford. 2016. Correlation-Based stereo. (2016).  
[http://ai.stanford.edu/~mitul/cs223b/  
dp.html](http://ai.stanford.edu/~mitul/cs223b/dp.html)