COMPUTER LAB 4: STEREO

Ethelbert Uzodinma (S3886026)
Viljami Linna (S4169921)

Department of Artificial Intelligence
University of Groningen

ABSTRACT

Index Terms— stereo, camera calibration, disparity map, image calibration, SIFT, perspective projection, computer vision.

1. INTRODUCTION

The image data set provided for this exercises is compressed in a zipped folder named: **stereoprakt.zip**. Using this Image file, our task is to develop and compare methods to measure disparity in a pair of calibrated(or rectified) images taken from two different camera positions as shown in Figure 1

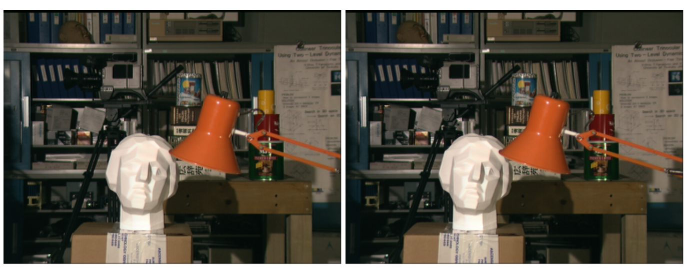


Fig. 1: The Tsukuba stereo pair.

2. METHOD

For the practical we first download the file "inference.zip" into the work directory of MatLab.
Then enter the following command in the command window:

```
unzip stereoprakt.zip

cd stereoprakt
```

The image contained in the `stereoprakt` folder for this experiment includes:

`tskuba1.png, tskuba2.png, tskuba_gt.png`

2.1. Exercise 1

In this exercise we manually picked feature points from the Tsukuba stereo pair images and computed the disparities of these points. Feature point positions and corresponding disparity value reference points are represented in Figure 2

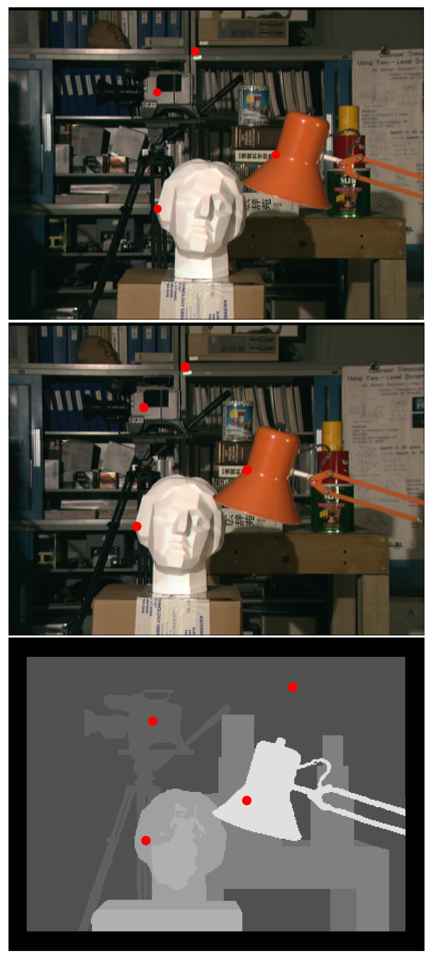


Fig. 2: Picked feature points and corresponding reference disparities

According to our experiments, the screen size of the computer and placement of the feature point have an influence of the accuraty of manually determined feature points. Best results can be obtained picking features points which human can perceive accurately(for example shape corners).

We picked four feature points on shape corners of different objects. We used laptop 17 inch screen. Disparities for the four objects determined selecting the feature points manually were really accurate. Maxium error of the disparity was smaller than 8 which means under 1 pixel error of the feature point location. Error values are represented in Figure 3.

Object	Manual dx	Reference dx	Bias
Head	162,56	160	1,57 %
Lamp	218,83	224	2,36 %
Camera	100,04	96	4,04 %
Bookself	78,15	80	2,37 %

Fig. 3: Manually determined disparities compared to the reference image values.

Fig. 4: Function `makeDispImage.m` output with window size 9 and maximum shift 28 for x and y

2.2. Exercise 2

In this exercise we developed a matlab function to compute the disparity based on a windowing method. Our function takes stereo image pair (left and right image) as an input parameter. With other input parameters we define window size n, and restrict the maxium shift of the window in y and x dimensions.

In this exrecise we only used square windows because time limit. Matching windows is determined by using normalized cross correlation. We first tried the sum of absolute difference values but it did not lead to satisfying results.

First, we read two image files and fit the window left up corner to every possible position $w(x_l, y_l)$ of the left image. For each window position we picked corresponding window of the right image and shifted the window in area defined by using input parameters.

For each shifted window $w(x_s, y_s)$ we calculated normalized cross correlation values with the $w(x_l, y_l)$. We chose shifted window with maximum value of the normalized cross correlation and thus determined the matching position (x_w, y_w). Disparity $d(x_s, y_s)$ is then calculated as follows:

$$d(x_s, y_s) = 8|x_s - x_w| \quad (1)$$

The value of the pixel $(x_s + floor(n), y_s + floor(n))$ in the disparity image is then set equal to $d(x_s, y_s)$. Because we can not fit the window outside of the original image range there always exist zero padding near by disparity image borders.

We applied Tsukuba stereo pair to our algorithm with window size 7. We set maximum x and y shift to 28. The output image is represented in Figure 4

With window size 7 our output image was noisy but main objects appeared to the output image. To improve results we tested to blur the input images. That did not improve the results. In further research we could try different combinations of the measurements (sum of absolute differences, squared differences and cross-correlation) to reduce noise.

2.3. Exercise 3

In this exercise we varied window size and compared results with the ground truth image. We did not had enough time to test different shapes so we just used square window. Using circle or rectangle shaped windows would maybe have improved our results.

We applied Tsukuba stereo pair to our `makeDispImage.m` function with $n = 5, 7, 9, 12, 15, 19, 23, 25$. For each output image we constructed a difference image between the ground truth image. Then, we calculated the sum of difference image pixel values and use that value as a measurement of the error. We plot error as a function of window size. This function is represented in Figure 5. According to the results using bigger window size leads to better results.

Output images of every n value is displayed in Figure 1. We see from the output images that bigger window size does not extract details well but reduces noise. Instead, with small window we got better results in places where sharp shapes occurred, but the amount of noise increased.

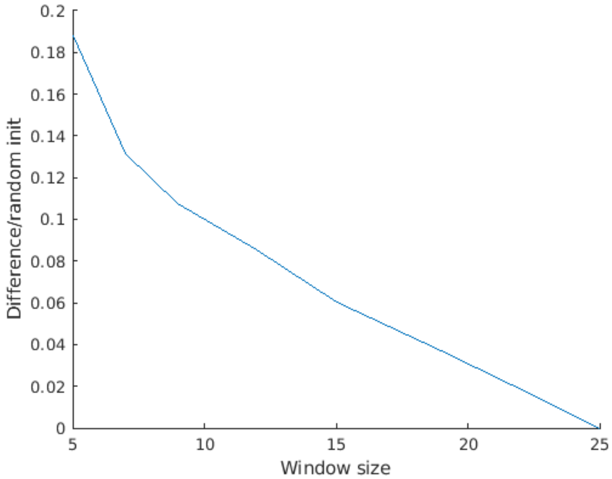


Fig. 5: Error as a function of n. Error values is divided by error of random image compared to ground truth image.

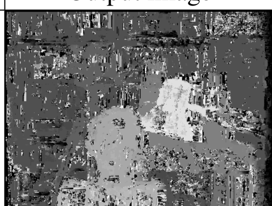
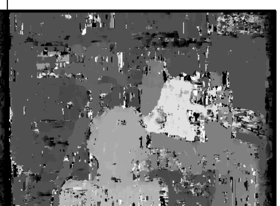
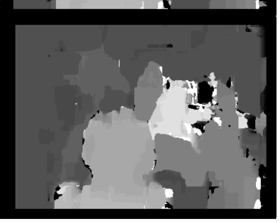
n	Output image	
5, 7		
9, 12		
15, 19		
23, 25		

Table 1: Output images with different window sizes.

2.4. Exercise 4

In this exercise we used SIFT(Scale Invariant Feature Transform) method to extract feature points in both Tsukuba stereo pair images. We matched these keypoints and calculated disparity of every matching keypoint pair. Next, we compared these disparities to the corresponding(= same pixel location) disparity values in the ground truth image. The result of the keypoint matching is represented in Figure 6

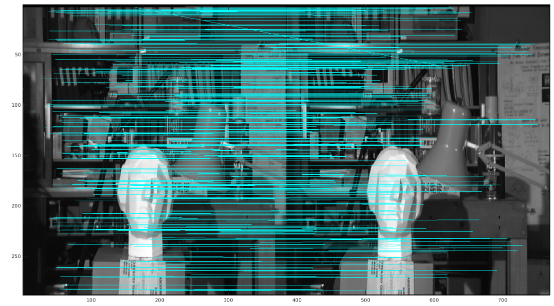


Fig. 6: Matched keypoints of Tsukuba stereo pair.

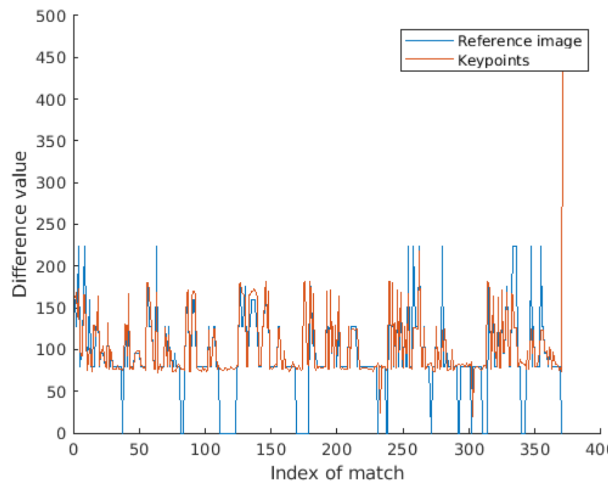


Fig. 7: Ground truth and SIFT based disparities as a function of match index.

We plot SIFT disparities and corresponding ground truth disparities as a function of match index. This function is displayed in Figure 7. According to the results SIFT based method seems to be more reliable and quicker method than our window based implementation. Generally we can not say which methods is better. To make reliable conclusions we would need more data. In addition we would have to test different combinations of shapes, window sizes and matching score functions when using window based methods.

In this exercise disparities of the keypoint pairs were determined as difference of two keypoints x-coordinate value. However, a keypoint pair y-coordinate values also differ a bit.

We calculated this difference for every matched keypoint pair. The results are displayed in Figure 8. According to the results most of the keypoint pairs have same

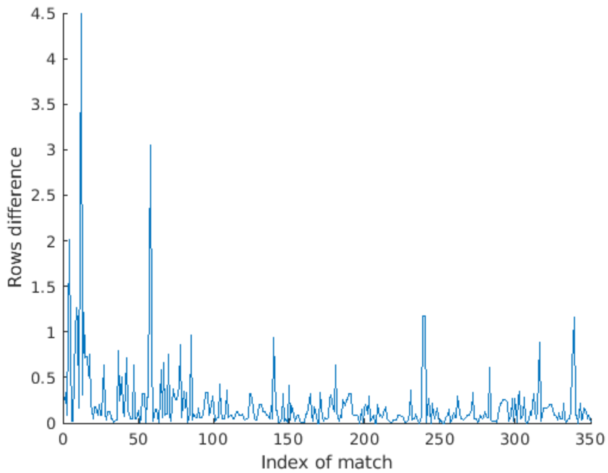


Fig. 8: Y-coordinate differences between keypoints of matches.

y-coordinate(difference < 0.5). Maximum value of the y-coordinate difference was 3.5.

3. CONCLUSION

4. REFERENCES