Lab IV Computer Vision: Shape from Stereo

Amit Bharti
University of Groningen
Groningen, The Netherlands
a.bharti.1@student.rug.nl

Pedro Rodriguez de Ledesma Jimenez
University of Groningen
Groningen, The Netherlands
p.rodriguez.de.ledesma.jimenez@student.rug.nl

I. Introduction

The computer stereo vision is the process of extracting 3D information from digital images about a scene from two different vantage points. 3D information of the scene can be obtained by examining the relative position of the objects in the two image planes. This framework simulates human binocular vision and therefore gives it the ability to perceive depth. Hence, computer stereo is the process of finding the 3D scene point P from the point P1 and P2 in the image plane as shown in Figure 1. In the setup, the camera's lens are located at a distance b between them also known as baseline, generating two coplanar image planes at a distance f in front of each camera lens. Figure 1 displays the setup of the binocular stereo used in the experiments.

Let (u_L , v_L) be the co-ordinate of the point P1 in the left image plane and (u_R , v_R) be the co-ordinate of the point P2 in the right image plane. Since we know that, the 3D scene point will be lying on the ray connecting the image plane point and camera co-ordinate (0,0,0), so point of intersection of the one ray passing from O_L and P1 and the 2^{nd} ray passing from O_R and P2 will be the 3D scene point. From perspective projection equation, the relation between image plane point and world co-ordinate point is as below equation.

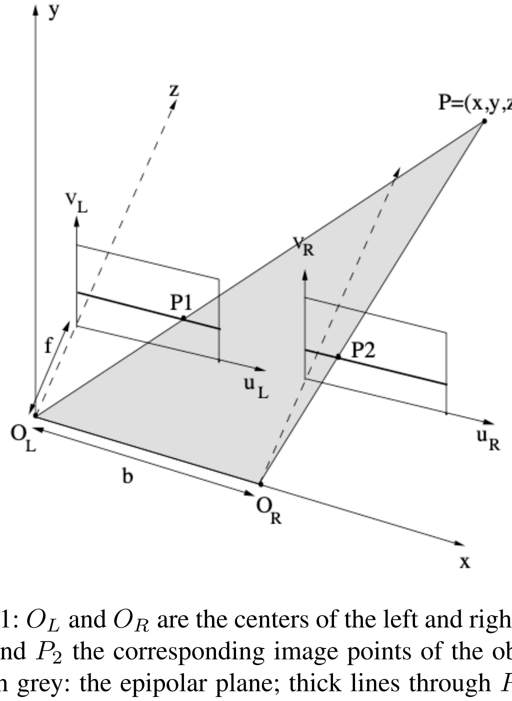

Fig. 1: O_L and O_R are the centers of the left and right cameras; P_1 and P_2 the corresponding image points of the object point P. In grey: the epipolar plane; thick lines through P_1 and P_2: the epipolar lines.
Source: University of Groningen Computer Vision slides

$$u = f_x * \frac{x}{z} + o_x \tag{1}$$

$$v = f_y * \frac{y}{z} + o_y \tag{2}$$

where f_x and f_y are the focal length in x and y direction respectively, and o_x and o_y are the centre of the image plane in x and y direction respectively.
Referring to the above equation, the value of the points P1(u_L,v$_L$) and P2(u_R,v$_R$) are as below.

$$u_L = f_x * \frac{x}{z} + o_x \tag{3}$$

$$v_L = f_y * \frac{y}{z} + o_y \tag{4}$$

$$u_R = f_x * \frac{x-b}{z} + o_x \tag{5}$$

$$v_R = f_y * \frac{y}{z} + o_y \tag{6}$$

Rearranging the equation 3, 4, 5 and 6 to obtain the value of x, y and z is as follows:

$$x = b * \frac{u_L - o_x}{u_L - u_R} \tag{7}$$

$$y = bf_x * \frac{v_L - o_y}{f_y(u_L - u_R)} \tag{8}$$

$$z = \frac{bf_x}{u_L - u_R} \tag{9}$$

The difference between the u co-ordinate(u_L - u_R) of the point P1 and P2 is known as Disparity. And from Equation 9, we could observe that depth of the 3D scene point P is inversely proportional to the disparity of the setup. In addition, the disparity along x-axis and y-axis is shown in the Figure 2. We could observe that there is no disparity along y-axis, so points will be on same horizontal line, while disparities in x-axis are $\frac{f}{z} * b$. Therefore, the setup as shown in Figure 1 allows us to obtain the depth of the 3D point in a world

$$
\begin{aligned}
x-\mathit{disparity} &: u_L - u_R = \frac{f}{z}b,\\
y-\mathit{disparity} &: v_L - v_R = 0
\end{aligned}
$$

Fig. 2: Disparities in each of the axis and the relation with the distance of the point.

Points	u_L	v_L	u_R	v_R	Disparity
1	191	127.23	177.31	127.23	13.7
2	291.14	195.28	283.0	195.28	8.14
3	93.43	46.78	83.16	46.35	10.27
4	72.46	220.1	66.9	219.67	5.56

TABLE I: A table containing points from both images that were selected manually for calculating disparsity.

co-ordinate system based on the setup and camera calibration parameters.

In the following section of this paper, four exercises are presented, where we are going to compute the disparities of the images captured by the Tsukuba stereo pair. Firstly, the points are being selected manually, and we are going to evaluate the accuracy of the model. Secondly, more accurate values of disparities are computed computationally through matching image's blobs captured in 2 different images. Thirdly, different sizes of blobs are matched and evaluated. Finally, the SIFT code is used to extract feature points in both images and then disparities are computed.

II. Exercises

A. Exercise 1

In this exercise, disparities of the 2 image plane points are calculated through selection of feature point manually. 4 feature points were selected in both the tsukuba1 and tsukuba2 image, for which co-ordinates in u and v direction are given in the Table I according to the point's location shown in Figure 3. We could find the disparities value in the Table I which is quite accurate as compared to the ground truth. We could see that Figure 3 illustrates the ground truth which is generated using an active illumination method. The illumination is done such that the closure the point to the camera, brighter the image will be. Since we know from the equation 9 that disparity is inversely proportional to the depth of the point in the 3D scene. Therefore, the brighter the image in the ground truth, greater will be its disparity and smaller will be its depth. Hence, disparity of the point 1 is greatest and point 4 has the least value. The accuracy of manual selection depends on the correctness of the point picked in both the images. Since we know that v_L and v_R value should remain same while calculating disparity but due to human error in point 3 and 4, they are slightly different. Therefore, the human error influences the accuracy of manual selection. Point 1 seems to be the closest point in the tsukuba since the object on which point lies is the brightest in the ground truth, while point 4 is the darkest, so point lying on the object is the farthest. Hence, range of the disparity should be between 15 and 5.

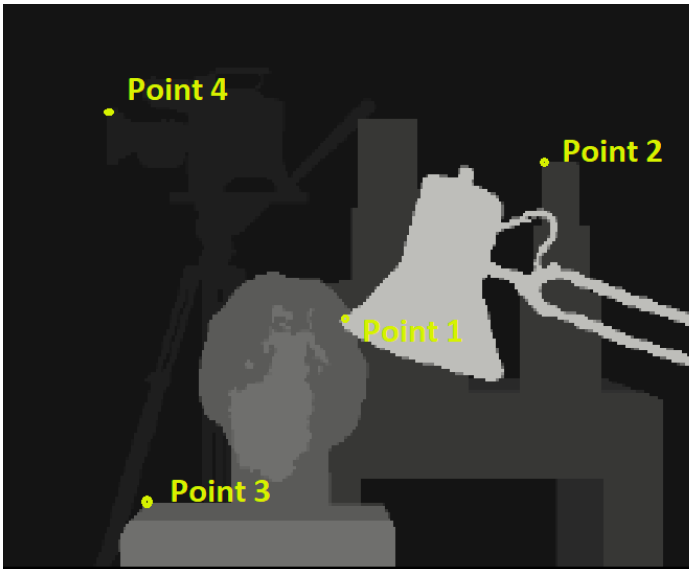

Fig. 3: Position of points that were selected manually for calculating the disparity

B. Exercise 2

To compute the disparities of the Tsukuba images, a script was developed that compare the surrounding of a pixel P_1 in the left image to slightly translate position P_2 in the right image to estimate the disparity of the pixel P_1(Figure 4).

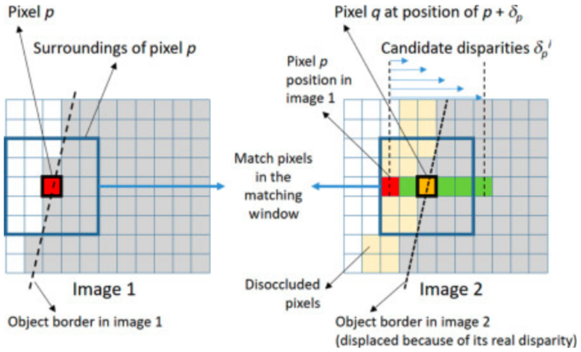
Pixel p
Surroundings of pixel p
Image 1
Object border in image 1
Match pixels in the matching window
Pixel p position in image 1
Disoccluded pixels
Pixel q at position of $p + \delta_p$
Candidate disparities δ_p^i
Image 2
Object border in image 2 (displaced because of its real disparity)

Fig. 4: Representation of the process of matching windowns in stereo matching.
Source:Multiview video: Acquisition, process- ing, compres- sion, and virtual view ren- dering

The fit score is computed using two different algorithms, which are the sum of squared differences and the sum of absolute differences. Results obtained through both the algorithms were same. To achieve higher computer efficiency, all the operation were done in matrix. Figure 5 shows the result in which the blob chosen has image plane co-ordinate $(u_l, v_l) = (180, 150)$ with a size of 40x40 pixels. The matching blob in the right image fits with the one selected based on human judgment. The disparity obtained by calculating the difference between the x-values of the blobs is 10.

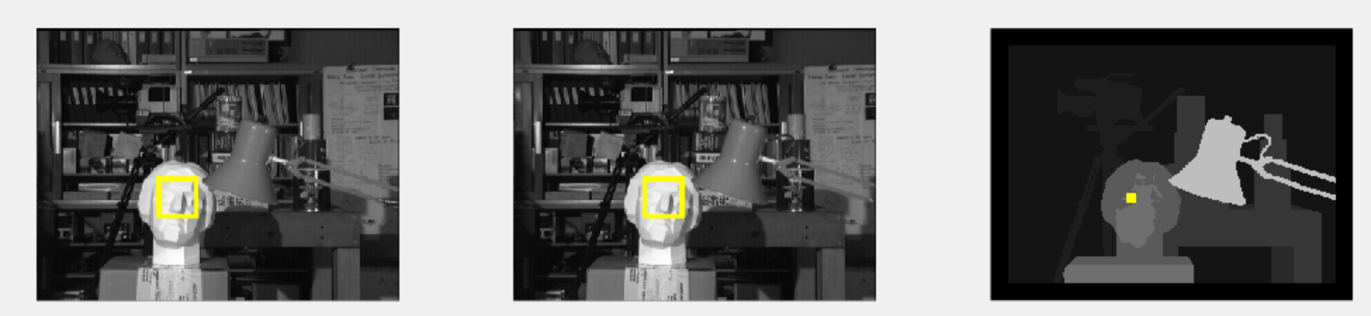

Fig. 5: Disparities in each of the axis and the relation with the distance of the point.

C. Exercise 3

Is important while windowing to well define the shape of the surrounding matching pixel, so all the pixels inside have the same disparity. As previously mentioned, the disparity is inversely proportional to the distance of the object. Therefore, if the matching window lies half over one object and covers for the other half another object lying at a very different depth, pixels that have dissimilar disparity values will be matched together. In the Figure 6 are displayed different window's shapes that contain points of one object only. To obtain a good shape local stereo approaches make coinciding the windows borders with objects borders. This could be done by calculating the gradient of the image and therefore the windows will never cross an object border, and therefore it provides a more reliable depth estimation.

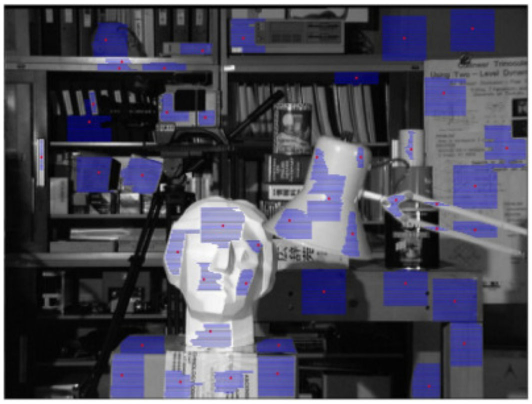

Fig. 6: Display of different windows shapes that contain points of the same objects.
Source:Multiview video: Acquisition, process- ing, compression, and virtual view ren- dering

In the next run of the script, a rectangular window of 20x40 was defined with centre in the point (330,90)(Figure 7. This shape was selected to just contain point of one object (the blackboard).

D. Exercise 4

SIFT stands for Scale Invariant Feature Transform algorithm [1]. In SIFT, we obtain local invariant features that deal with searching SIFT key-points which are invariant to image translation, rotation and scaling and partially invariant to illumination changes and are unique and highly descriptive of the object in order to easily identify them and match them in other images where the object appears. As a consequence, the algorithm is able to recognize object in complex appearances. Therefore, we used the function ”matches = match(’tsukuba1.png’,’tsukuba2.png’);” to get all the matching keypoints in both the images. We got 783 and 751 as individual keypoints(feature points) while 446 matching keypoints in both of them. For the matching keypoints, we computed the difference between the u_L and u_R to get the disparity. As compared to the window based method, the number of feature points to be computed for disparity is less, since we are looking for matching features instead of doing a random search along the horizontal axis with the same y value. So, we are considering the context of the figure using SIFT while performing simple stereo operation. No, matching features are not always on the same image row. In general, variation is not more than a blob of window size 20, but for key points that were incorrectly matched by the SIFT, the value of disparity will be very high.

REFERENCES

[1] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the seventh IEEE international conference on computer vision*, vol. 2. Ieee, 1999, pp. 1150–1157.

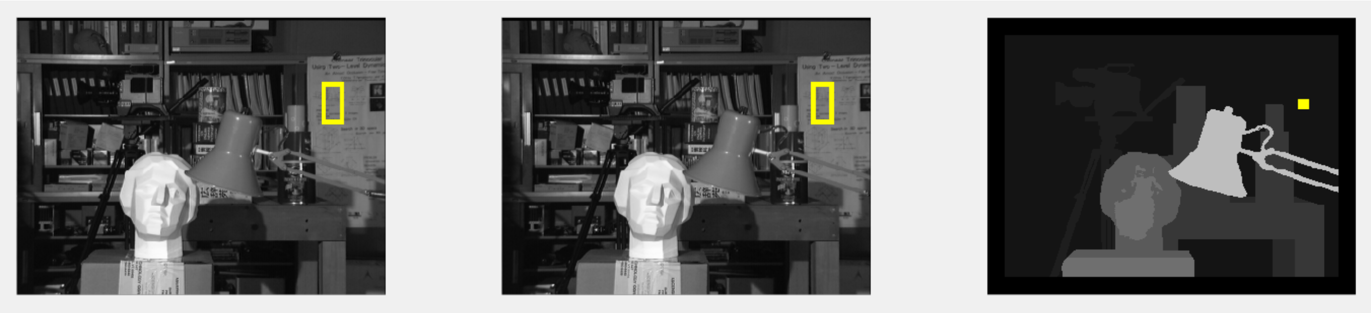

Fig. 7: Disparities in each of the axis and the relation with the distance of the point.