Advanced Computer Vision – HW4

Stereo Matching

宋體淮, R09921135, Electrical Engineering

Part 1: Depth from Disparity

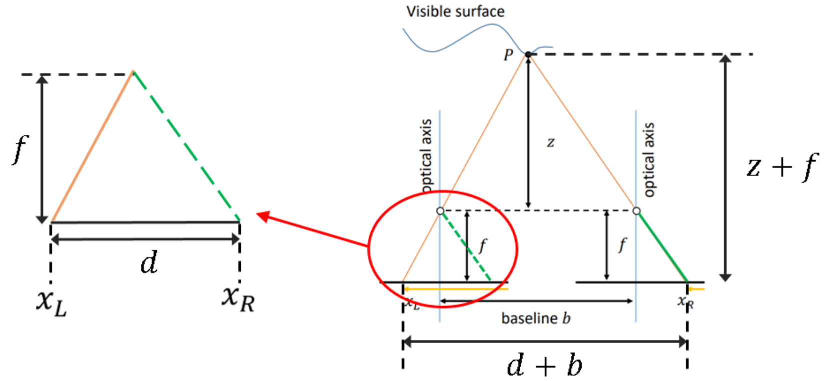


Figure 1: Similar triangles in the stereo geometry.

From the Figure 1, we can obtain the following equation by the property of similar triangles:

$$\frac{f}{d} = \frac{z+f}{d+b}$$

After rearranging the equation, we can get the final result:

$$f(d+b) = d(z+f) \rightarrow d = \frac{bf}{z}$$

Part 2: Disparity Estimation

I implement the methods from this paper [1]. I will briefly introduce the algorithm used in this work, and also what I modified to get a better performance.

- **Cost Computation**

Absolute Difference Cost + Census Cost (AD-Census):

Absolute difference cost (AD) measures the absolute color difference between the windows scanned in the left image and the right image. However, such pixel-based calculation is sensitive to noises or outliers.

Census cost encodes local image structures with relative orderings of the pixel intensities, and therefore tolerates outliers. But this could introduce matching ambiguities in image regions with repetitive or similar local structures.

As a result, the combination of them (AD-Census) could benefit from each other and could provide more accurate results.

- **Cost Aggregation**

Cross-based Cost Aggregation:

This method was first proposed from [2]. For every pixel, it will first construct a support region where the pixels in the region have similar color and location. For the cost aggregation step, the costs are aggregated alternatively for several iterations.

What this paper modify includes the construction rules of support region and also the aggregation order, which leads to better results.

However, I found that this method will take too much time. As a result, considering the runtime issue, I abandon this method and replace it with a bilateral filter. Inspired by [3], cost aggregation can be think as filtering on a cost volume, so I apply bilateral filter for every cost map at different disparity:

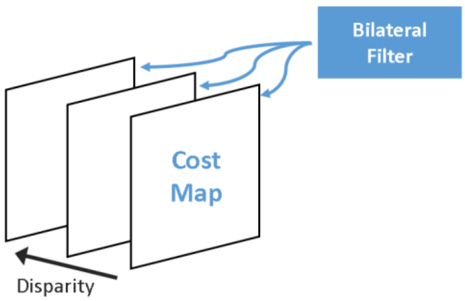


Figure 1: The usage of the bilateral filter.

— Experiments

	* Runtime (s)				
Name	Tsukuba	Venus	Teddy	Cones	Avg.
With Cross-based Cost Aggregation	610.5	1023.3	1912.1	1232.2	1194.5
With Bilateral Filter	87.1	132.4	105.6	64.1	**97.3**

Table 1: The comparison of different cost aggregation methods in runtime performance. * Including the runtime of constructing the support region.

	Bad Pixel Ratio (%)				
Name	Tsukuba	Venus	Teddy	Cones	Avg.
Without Bilateral Filter	3.58	1.41	9.45	8.70	5.79
With Bilateral Filter	3.38	1.41	9.19	8.93	**5.73**

Table 2: The effectiveness of using bilateral filter.

- **Disparity optimization**

Scanline Optimization:

This paper employs a multi-direction scanline optimizer based on semi-global matching method. The basic idea is that it will use dynamic programing to globally minimize a cost function which enforces the smoothness constraint to cost volume. Lastly, it will choose the least-cost disparity by WTA method.

- **Disparity Refinement**

Outlier Detection:

The outliers are first detected with left-right consistency check and are further classified into occlusion and mismatch points, since they require different interpolation strategy in later stage. for outlier p at disparity $D_L(p)$, the intersection of its epipolar line and D_R is checked. If no intersection is detected, p is labelled as 'occlusion', otherwise 'mismatch'.

Iterative Region Voting:

The detected outliers should be filled with reliable neighboring disparities. With the help of constructed support regions from the early stage, we can find the reliable disparities by building a histogram for each support region. The disparity with the highest bin value will be selected as the reliable one as long as it has enough amount of reliable information.

Proper Interpolation:

The remaining outliers are filled with a interpolation strategy that treats occlusion and mismatch points differently. For outlier p, we find the nearest reliable pixels in

16 different directions. If p is an occlusion point, the pixel with the lowest disparity value is selected for interpolation, since p most likely comes from the background; otherwise the pixel with the most similar color is selected for interpolation.

Depth Discontinuity Adjustment:

After processing the outliers, we should consider about the edges in the image. We find all the edges by sobel filter. For the pixel which is on the edge in the image, we take pixels from both sides of the edge, and pick the one with lower matching cost. Finally, we replace with the pixel which has lower cost.

Sub-pixel Enhancement:

This process based on quadratic polynomial interpolation is performed to reduce the errors caused by discrete disparity levels. The final disparity results are obtained by smoothing the interpolated disparity results with a 3×3 median filter.

Segmentation-based Refinement:

This step is what I propose inspired by [4]. After all the processes mentioned above, we can observe that there are still some obvious mismatched pixels. This segmentation-based disparity refinement method is based on the assumption that the pixels belonging to the same segment have similar disparity values. Using this assumption, we can find the outliers for each segment and use proper strategy to replace all the outliers. Below is the flowchart of this step:

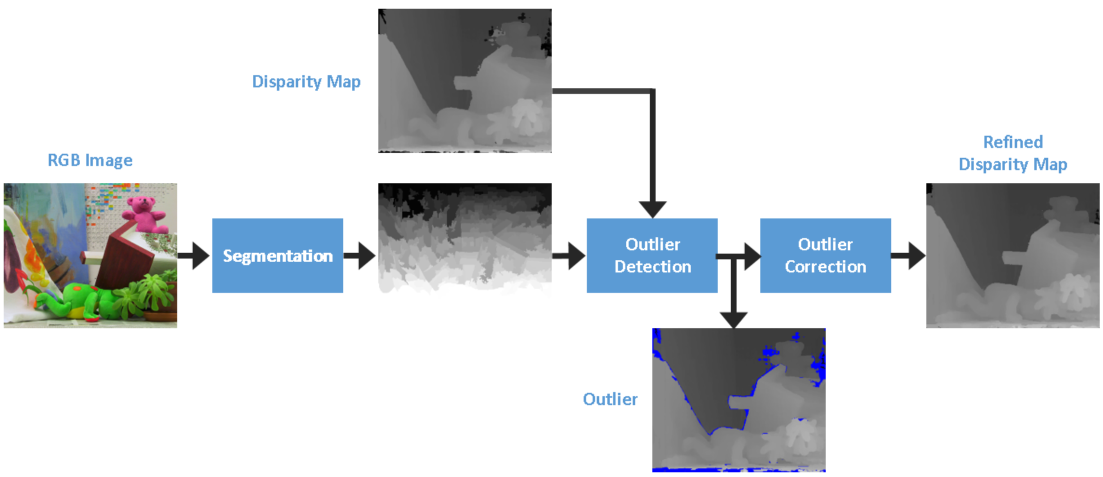


Figure 2: The flowchart of segmentation-based refinement.

The outliers are detected as follows:

$$p^i = Outlier, \quad \text{if } \left(p^i - mean^i\right) > Threshold$$

where p^i is a pixel in segment i. $mean^i$ is the mean disparity in segment i. $Threshold$ is set to be 6 in the experiment. The segmentation method I use is graph-based segmentation.

Finally, the outliers are corrected by the refinement steps mentioned above: *Iterative Region Voting, Proper Interpolation, Depth Discontinuity Adjustment.*

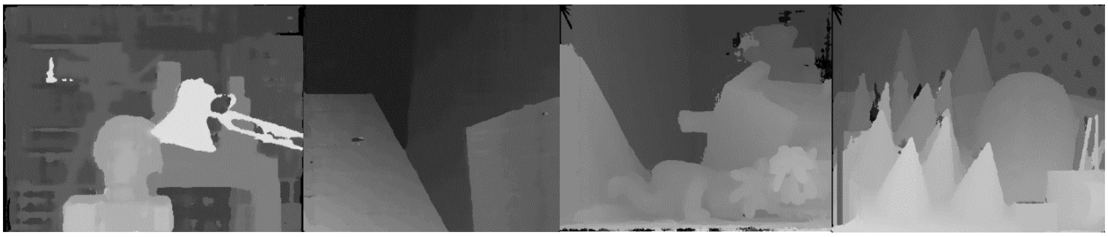
— Experiments

	Bad Pixel Ratio (%)				
Name	Tsukuba	Venus	Teddy	Cones	Avg.
Without Segmentation-based Refinement	3.53	1.47	10.08	9.54	6.15
With Segmentation-based Refinement	3.38	1.41	9.19	8.93	**5.73**

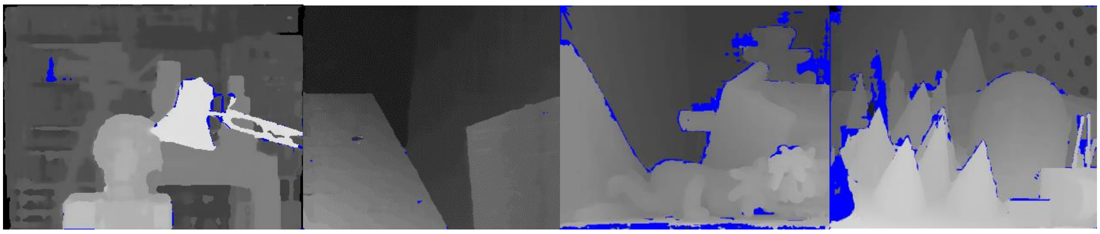
Table 3: The effectiveness of using segmentation-based refinement.

— Visualization

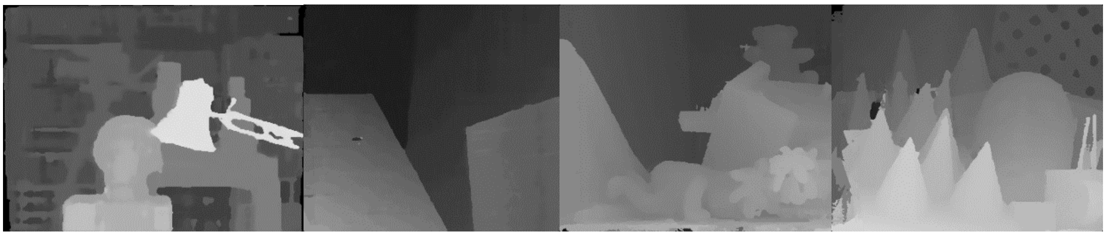
Input Disparity Map:



Outlier Detection:

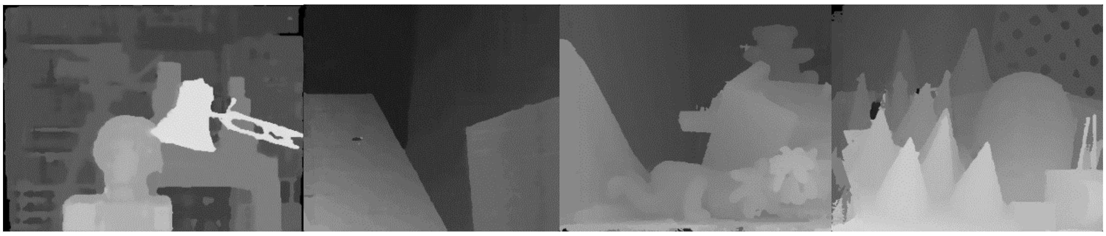


Output Refined Disparity Map:



- **Results:**

— **Final Results:**



— **Bad Pixel Ratio**

Bad Pixel Ratio (My result / Baseline) (%)				
Tsukuba	Venus	Teddy	Cones	Avg.
3.38 / 8	1.41 / 5	9.19 / 18	8.93 / 15	**5.73** / 11.5

Table 4: The final result of bad pixel ratio.

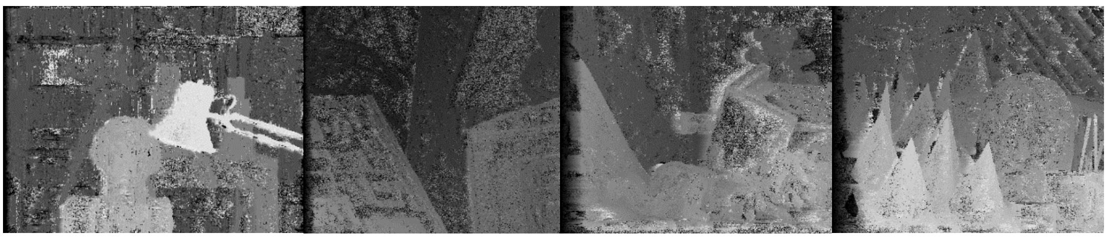
— **Runtime**

Runtime (s)				
Tsukuba	Venus	Teddy	Cones	Avg.
272.8	439.9	884.3	840.9	**609.5**

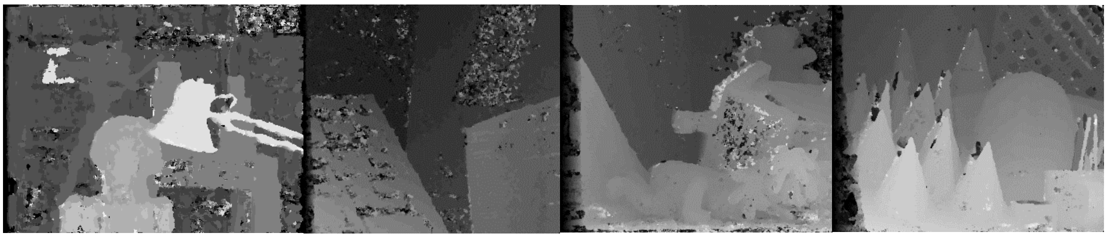
Table 4: The final result of runtime.

— **Visualization**

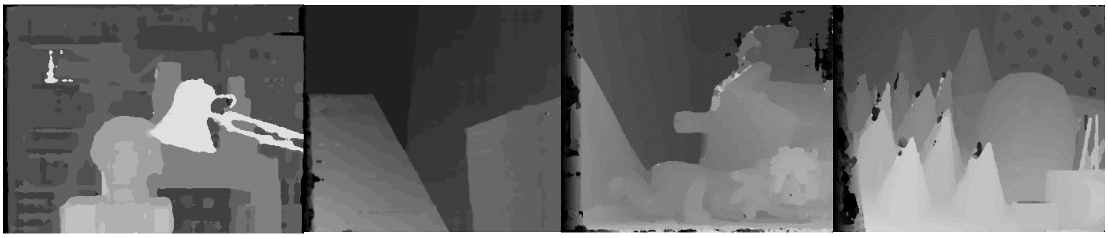
AD-Census Cost:



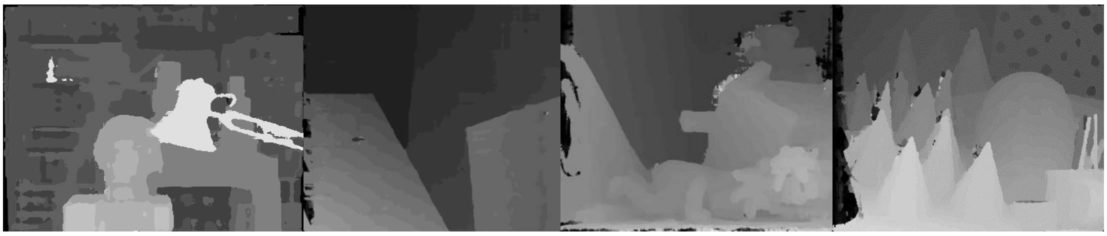
Cost Aggregation:



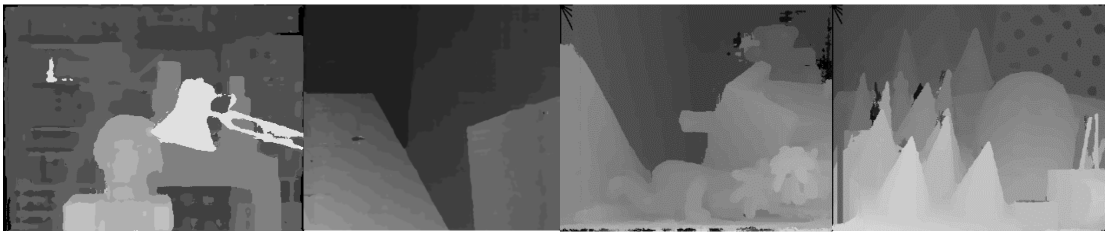
Scanline Optimization:



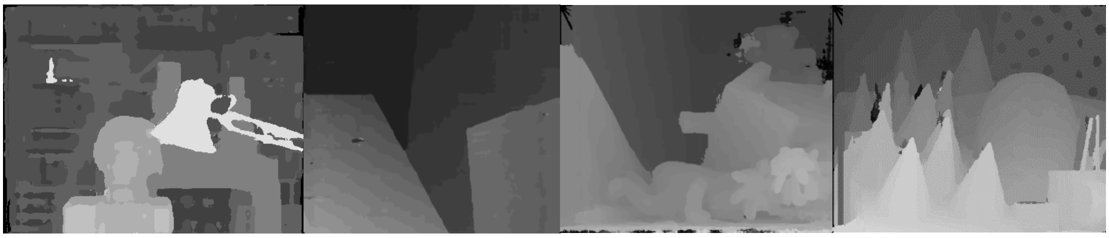
Iterative Region Voting:



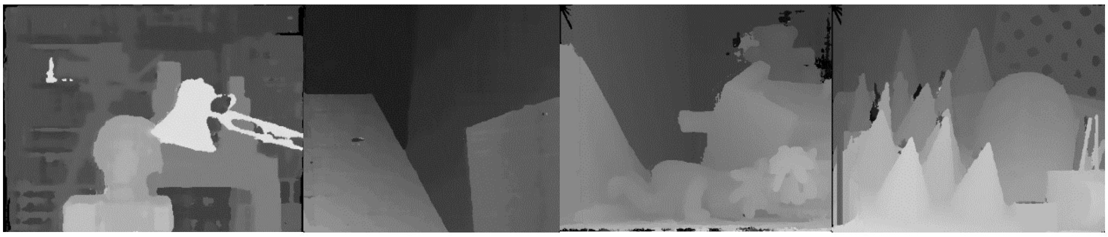
Proper Interpolation:



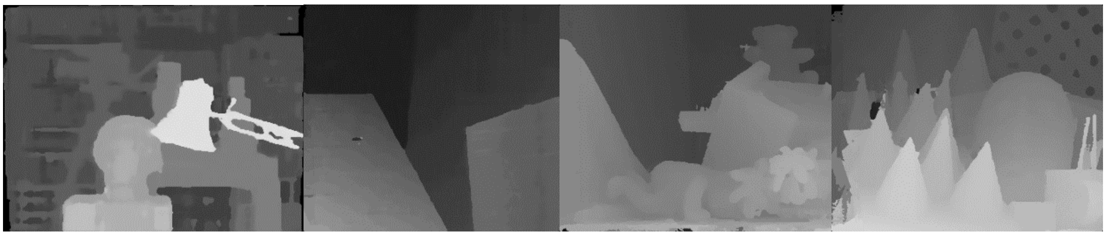
Depth Discontinuity Adjustment:



Sub-pixel Enhancement:



Segmentation-based Refinement:



- **Discussion:**

The runtime of *Scanline Optimization* can be improved. The entire process spent the most time on this step, so I will manage to speed up the performance or find some other optimization strategy in the future.

The *Segmentation-based Refinement* has harmful effect on Cones data. Based on my observation, this refinement step largely depends on the result of image segmentation, while the graph-based segmentation method I used is not that preferable. Therefore, I will try some deep learning based segmentation instead in the future.

- **Reference:**

[1] Mei, Xing, et al. "On building an accurate stereo matching system on graphics hardware." *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*. IEEE, 2011.

[2] K. Zhang, J. Lu, and G. Lafruit. Cross-based local stereo matching using orthogonal integral images. *IEEE TCSVT*, 19(7):1073–1079, 2009.

[3] Hosni, Asmaa, et al. "Fast cost-volume filtering for visual correspondence and beyond." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.2 (2012): 504-511.

[4] G. Bae and Y. H. Kim, "Segmentation-based disparity refinement," *2018 International SoC Design Conference (ISOCC)*, 2018, pp. 74-75, doi: 10.1109/ISOCC.2018.8649977.