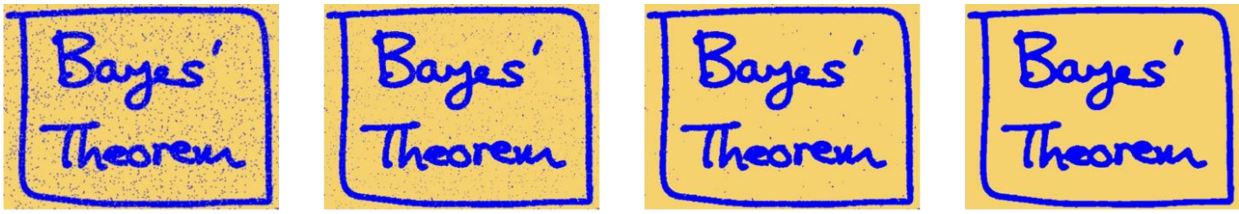
EE5731 CA2 REPORT

Xin Zhengfang E0427425/A0206597U

CONTENTS

PART-1 Noise Removal

There are some results of my part-1 with different lambda values.



(a) Original input (b) Lambda = 1 (c) Lambda = 10 (d) Lambda = 100

Fig. 1. Some results of PART-1

As lambda increases, the prior constraint become stronger. We can see that the pictures becomes pure and smooth from Fig.1. Moreover, I use 8 neighbors to construct 8 edges of one pixel, therefore, my result is more anti-aliasing.

(a) 8 edges of 1 pixel (b) 4 edges of 1 pixel

Fig. 2. Anti-aliasing with more edges

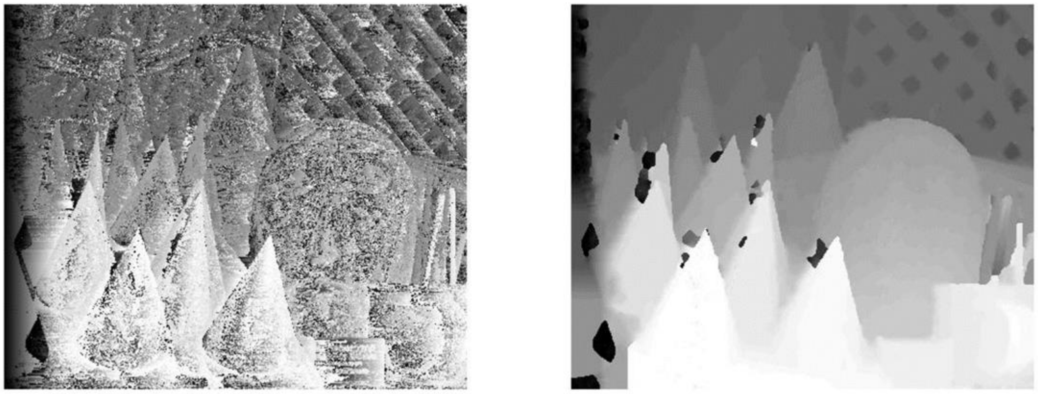
PART-2 Depth from Rectified Stereo Images

The original two input images show in Fig. 3.

(a) Left input image (b) right input image

Fig. 3. Input images

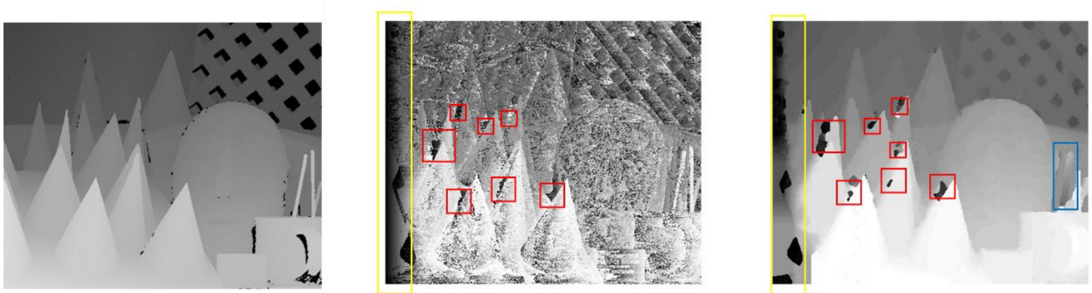
My results show in Fig. 4.



(a) Before graph cuts (b) After graph cuts

Fig. 4. Results of PART-2

Comparing with ground truth of depth map, my result has some unhappy regions that shows in Fig. 5.



(a) Ground truth (b) Before graph cuts (c) After graph cuts

Fig. 5. Unhappy cases comparing with ground truth (1) The yellow region is caused by no information in two images. (2) The red region is caused by occlusion. (3) Blue region is caused by treated as noise. (Because some noise has the same scale as these thin sticks. I want the picture to be beautiful, therefore, I have to sacrifice these sticks)

PART-3 Depth from Stereo

The original two input images of PART-3 show in Fig. 6.

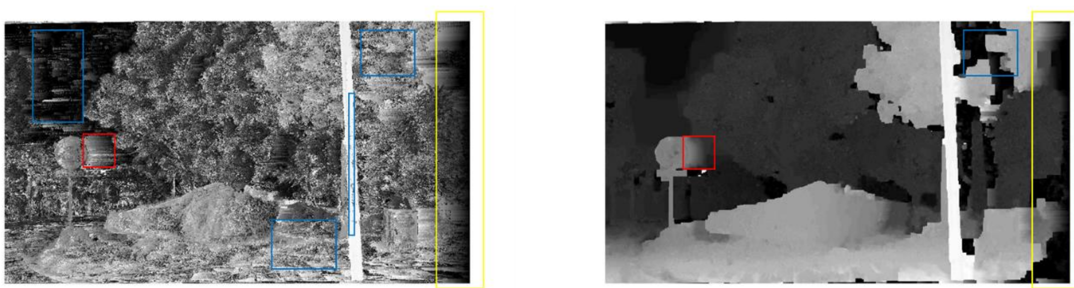
(a) Left input image (b) Right input image

Fig. 6. Input images of PART-3

My best results of PART-3 show in Fig. 7

(a) Before graph cuts (f) After graph cuts

Most of the noises come from ambiguity, occlusion and information loss. The noise impacts show in Fig.8.



(a) Before graph cuts (f) After graph cuts

Fig.8 The noise impacts (1) Yellow region: information loss (2) Red region: Occlusion (3) Blue region: ambiguity

PART-4 Depth from Video – Basic

Some my results of PART-4 show as following figures.

road_17_0.100.jpg

road_90_0.100.jpg

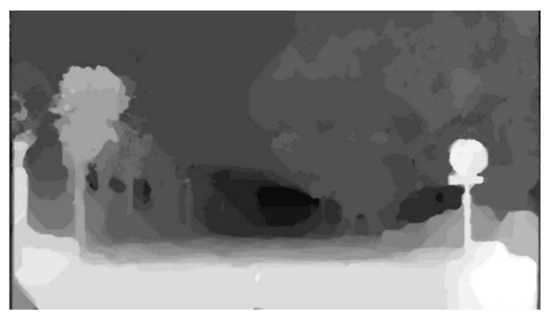
Fig. 9 Results of PART-4 (frames = 5, object = middle, sigma_d = 0.1)

road_55_0.250.jpg

road_55_0.250.jpg

Fig. 10 Results of PART-4 (frames = 5, object = left, sigma_d = 0.25)

road_54_0.100.jpg



road_139_0.100.jpg

Fig. 11 Results of PART-4 (frames = 5, object = middle, sigma_d = 0.1, edges = 8)

In this part, because we use multi-frames to estimate the depth, the ambiguity problem almost was solved. Information loss is improved if we use the middle frames as the main object to be estimated. However, annoying occlusion still exists, although it can be improved a little by bundle optimization.

Why occlusion influence is still there even we use multi-frames? Because frame will have certain occlusion which can leave the occlusion ghost on the object when we comparing the object with each frame. Finally, we add these frames' occlusion ghost together, which makes my result bad especially on the light pole, the traffic pole and the tree pole. That is also the reason why if I use more frames, the result will be worse. Because we have already solved the ambiguity problem with only 5 frames. If more frames are used, they only contribute to the occlusion ghost and time-consuming.

Bundle optimization is not magic

Why I say so, because bundle optimization is just a constraint that makes images tend to be black. We can see the reason from the line check figures and p_v .

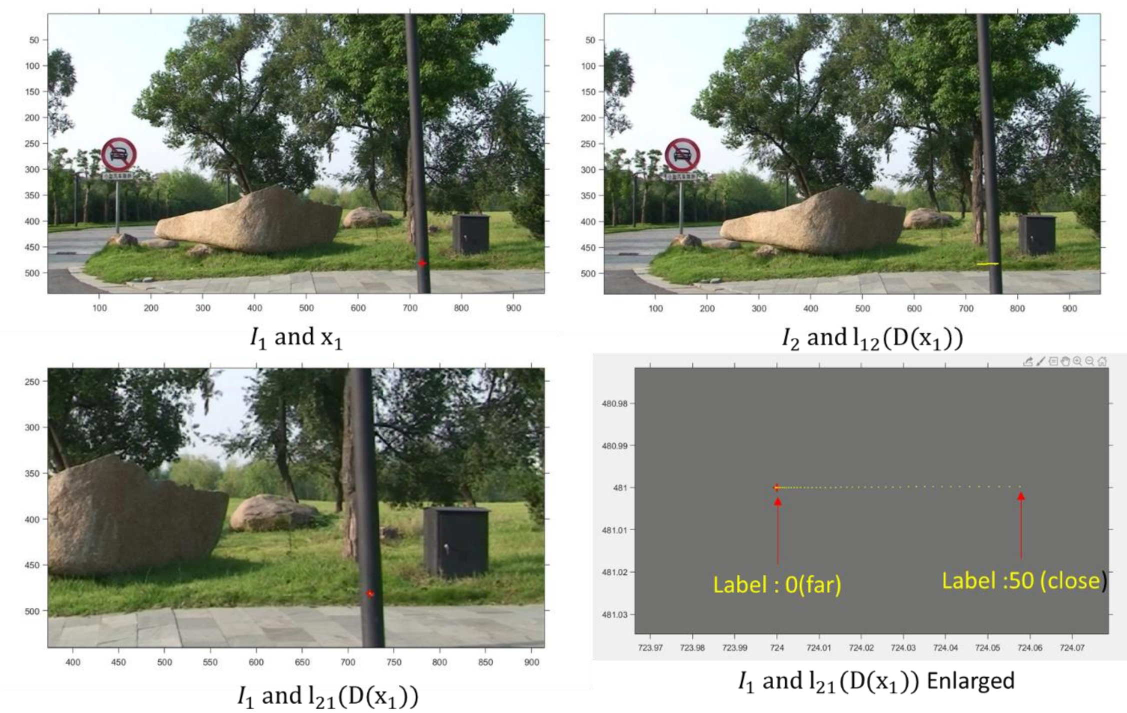
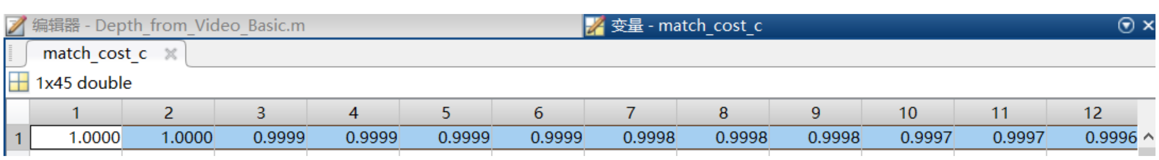


Fig. 12. Line check

When we project x' from frame t ' to t, the one with d = 0 will always no difference with x.



	1	2	3	4	5	6	7	8	9	10	11	12
1	1.0000	1.0000	0.9999	0.9999	0.9999	0.9999	0.9998	0.9998	0.9998	0.9997	0.9997	0.9996

Fig. 13. Some of p_v

This will always weak the close labels. In another words, it always makes images tend to be label 0 (black).