

CSE 565 HW 3 REPORT

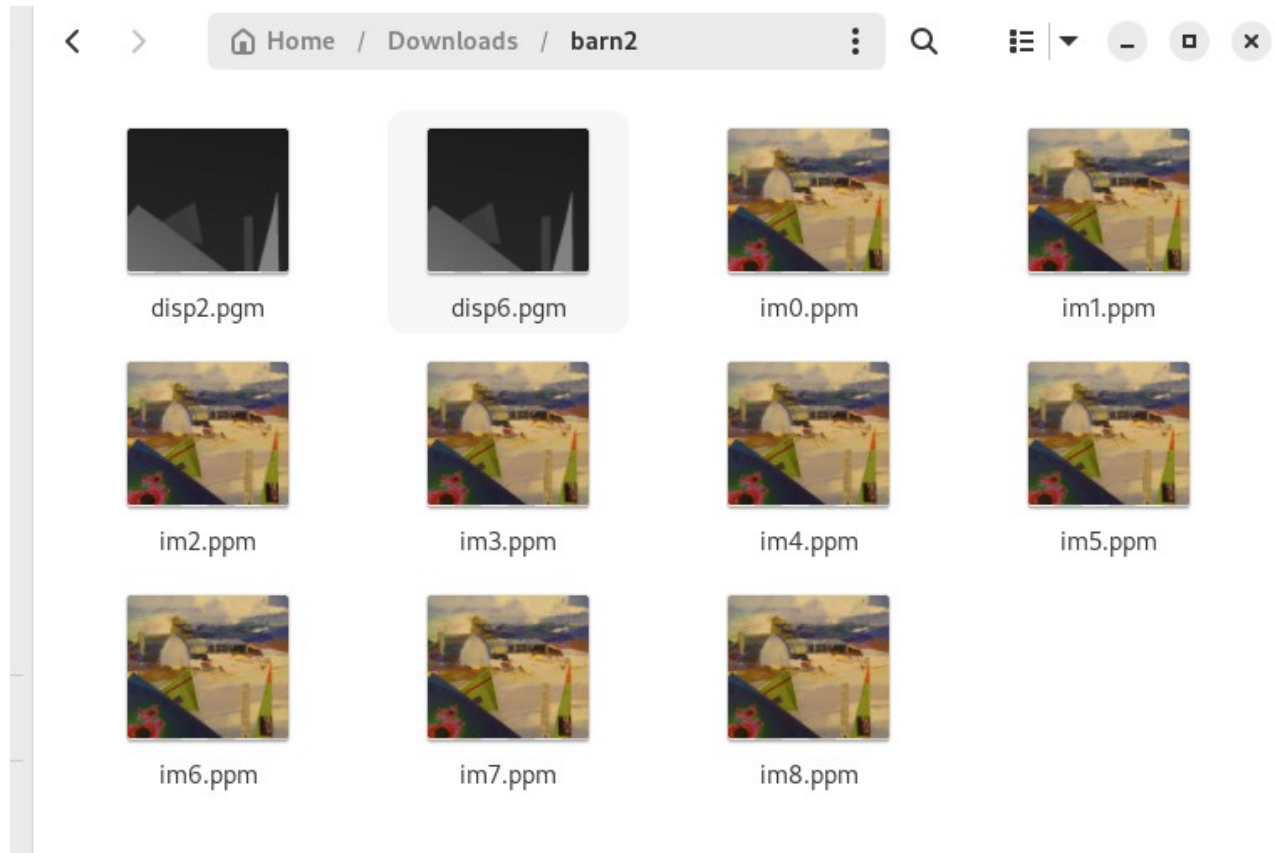
244201001033 AKCAN ERCAN

Contents

- Choosing stereo images,
- My over segmentation method / correspondence algorithm,
 - *Add Padding*
 - *Loop*
 - *Searching*
 - *Shifting*
 - *Block Comparison*
 - *Calculate*
- Results,
- Comparing,
- Discussion

➤ Choosing stereo images

I selected *Barn 2* images from [Middlebury](#) page. As we can see, the **barn2** folder includes nine stereo images and two disparity images.



P.S. *Please don't forget to change barn2 folder location with respect to your local folder.*

Load Images

```
5] 1 left_image = cv2.imread('docs/barn2/im0.ppm', cv2.IMREAD_GRAYSCALE)
    2 right_image = cv2.imread('docs/barn2/im1.ppm', cv2.IMREAD_GRAYSCALE)
    3
    4 ground_truth_1 = cv2.imread('docs/barn2/disp2.pgm', cv2.IMREAD_GRAYSCALE)
    5 ground_truth_2 = cv2.imread('docs/barn2/disp6.pgm', cv2.IMREAD_GRAYSCALE)
```

Executed at 2024.06.04 23:20:27 in 2ms

Disparity Maps

➤ My over segmentation method

- Firstly; for over segmentation; we need to disparity map which we use depth difference between right and left image based on block searching. General steps for finding disparity map;
 1. **Add Padding** For run our method running on image edges, I will add some padding to left and right images. The padding size depends on block size.
 2. **Loop** For each block, find best shift value using SSD (Sum of Squared Differences) method,
 3. **Searching** On each pixel blocks for defining search bounds,
 4. **Shifting** If right shift is true; the bounds are determined accordingly,
 5. **Block Comparison** Get a block in left image and compare all blocks of right image and calculate SSD and update,
 6. **Calculate** Disparity map for smallest shifting value.

1. Add Padding;

I set block size = 27. Many of times trying, I found optimal number block size = 27.

Because our images' size $\rightarrow 430px * 381px$. Padding size $\rightarrow 27 // 2 \sim 13$. This value is good enough for $430px * 381px$ size.



Figure 1: Right Image With 13px Padding



Figure 2: Left Image With 13px Padding

2. Loop

- row: 1 col: 197 i: 141 best_distance: 244673.0 shift: 140 ssd: 241750.0
- row: 5 col: 371 i: 369 best_distance: 392034.0 shift: 368 ssd: 207117.0
- row: 10 col: 201 i: 157 best_distance: 374462.0 shift: 150 ssd: 395361.0
- row: 15 col: 154 i: 129 best_distance: 18136.0 shift: 88 ssd: 26906.0
- ...
- ...
- ...
- row: 369 col: 322 i: 307 best_distance: 457334.0 shift: 285 ssd: 556516.0
- row: 374 col: 208 i: 194 best_distance: 6831.0 shift: 169 ssd: 11226.0
- row: 379 col: 117 i: 74 best_distance: 259053.0 shift: 46 ssd: 494437.0

3. Searching

```
[_] 1 def search_bounds(column, block_size, width, rshift):  
    2     disparity_range = 75  
    3     padding = block_size // 2  
    4     right_bound = column  
    5  
    6     if rshift:  
    7         left_bound = column - disparity_range  
    8         if left_bound < padding:  
    9             left_bound = padding  
10     step = 1  
11     else:  
12         left_bound = column + disparity_range  
13         if left_bound >= (width - 2 * padding):  
14             left_bound = width - padding * 2 - 2  
15         step = -1  
16     return left_bound, right_bound, step
```

4. Shifting (if rshift == True)

For example, $col = 20$ and $disparity_range = 75$:

$left_bound = 20 - 75 = -55$. but because of padding; $left_bound = 13$.

$right_bound = 20$.

$step = 1$.

5. Block Comparison

```
right_pixel = right_img[row:row + block_size, i:i + block_size]

if euclid_dist(left_pixel, right_pixel) < best_distance_infinity :
    ssd = np.sum((left_pixel - right_pixel) ** 2)
    # print('row:',row, ' col:',col, ' i:',i, ' best_distance:',best_distance_infinity, ' shift:',shift, ' ssd:',ssd)

    if ssd < best_distance_infinity:
        best_distance_infinity = ssd
        shift = i

if rshift:
    disparity_map[row, col] = col - shift
else:
    disparity_map[row, col] = shift - col
print('Calculated disparity at ('+str(row)+' '+str(col)+' ) :', disparity_map[row,col])
```

6. Calculate

Calculated disparity at (319,306) : 2.0

Calculated disparity at (319,307) : 2.0

Calculated disparity at (319,308) : 2.0

Calculated disparity at (319,309) : 2.0

Calculated disparity at (319,310) : 2.0

Calculated disparity at (319,311) : 2.0

...

...

Calculated disparity at (320,9) : 9.0

Calculated disparity at (320,10) : 10.0

Calculated disparity at (320,11) : 11.0

Calculated disparity at (320,12) : 12.0

Calculated disparity at (320,13) : 13.0

Calculated disparity at (320,14) : 1.0

Calculated disparity at (320,15) : 2.0

...

Calculated disparity at (380,348) : 2.0

Calculated disparity at (380,349) : 2.0

Calculated disparity at (380,350) : 2.0

Calculated disparity at (380,351) : 2.0

Calculated disparity at (380,352) : 68.0

Calculated disparity at (380,353) : 3.0

Calculated disparity at (380,354) : 4.0

Calculated disparity at (380,355) : 4.0

➤ Results,



Figure 3: Disparity Map lr block27



Figure 4: Disparity Map rl block27

➤ Comparing



Figure 5: Consistency Map block 27 disp2



Figure 6: Consistency Map block 27 disp6

➤ Discussion

- **Without Consistency Check**

- ✓ RMSE for left-right (block_size=27) 53.284250746797234
- ✓ RMSE for right-left (block_size=27) 61.22323398963955

- **With Consistency Check**

- ✓ RMSE for disp2 (block_size=27) after Consistency check: 51.28792009553811
- ✓ RMSE for disp6 (block_size=27) after Consistency check: 50.491453621857595

- **Reasons for failures**

- ✓ **block_size**; Larger block sizes can produce smoother but less accurate disparity maps. Smaller blocks can give more accurate but noisier results. So I set block_size=27. It must check for optimum value.
- ✓ **disparity_range**; I set 75, this wont be suitable for the maximum disparity in the images.
- ✓ **shift**; If the value is not correctly calculated, it can result in incorrect disparity values in the map.

- **General Checks and Improvements**

- ✓ Test different block sizes,
- ✓ Try methods other than SSD, for example SAD (Sum of Absolute Differences),
- ✓ Examine Shift Values.