**Computer Vision - 217**

**Homework 2**

**Chen Naveh**

**Elad Wasserstein**

**Answers for questions:**

**Section A:**

* No questions in this section to answer

**Section B:**

* No questions in this section to answer

**Section C:**

**Question E:**

Since we calculated the disparity map only on X axis we defined D2d(:,:,1)=0 because we don’t want to change anything in Y axis.

**Question H:**

If we increase the the size of the patch the algorithm will miss more changes in depth that are smaller than the patch. Moreover, it demands more computing power. Because of the assumption of order preserving the algorithm sometimes finds a pixel with high similarity however it is a mistake.

**Question I:**

The regions where there is a change in depth have more errors since there is more noise and uncertainty

**Documentation of the function**

**Section B:**

* Stereo:
  + Function sign:
    - [P] = stereo\_list(ps1,ps2, ML,MR)
  + Input parameters:
    - Ps1 – list of points in 2d
    - Ps2 – list of points in 2d
    - ML – projection matrix for left image
    - MR – projection matrix for right image
  + Output parameters:
    - List of points in 3d that represent the corresponding points in 2d
  + The function is located in the file stereo\_list .m

**Section C:**

* Disparity calculation:
  + Function sign:
    - [D] = disparityCalc(im1, im2,Sx,Sy,d\_min, d\_max)
  + Input params:
    - im1 – input image #1
    - im2– input image #2
    - Sx – patch size in x axis
    - Sy – patch size in y axis
    - d\_min – the beginning of the range to find corresponding pixel
    - d\_max – the end of the range to find corresponding pixel
  + Output params:
    - matrix D that consists of the disparity of each pixel
  + The function is located in the file disparityCalc.m