Computer Vision

Assignment 5 : Image Segmentation

Autumn 2018

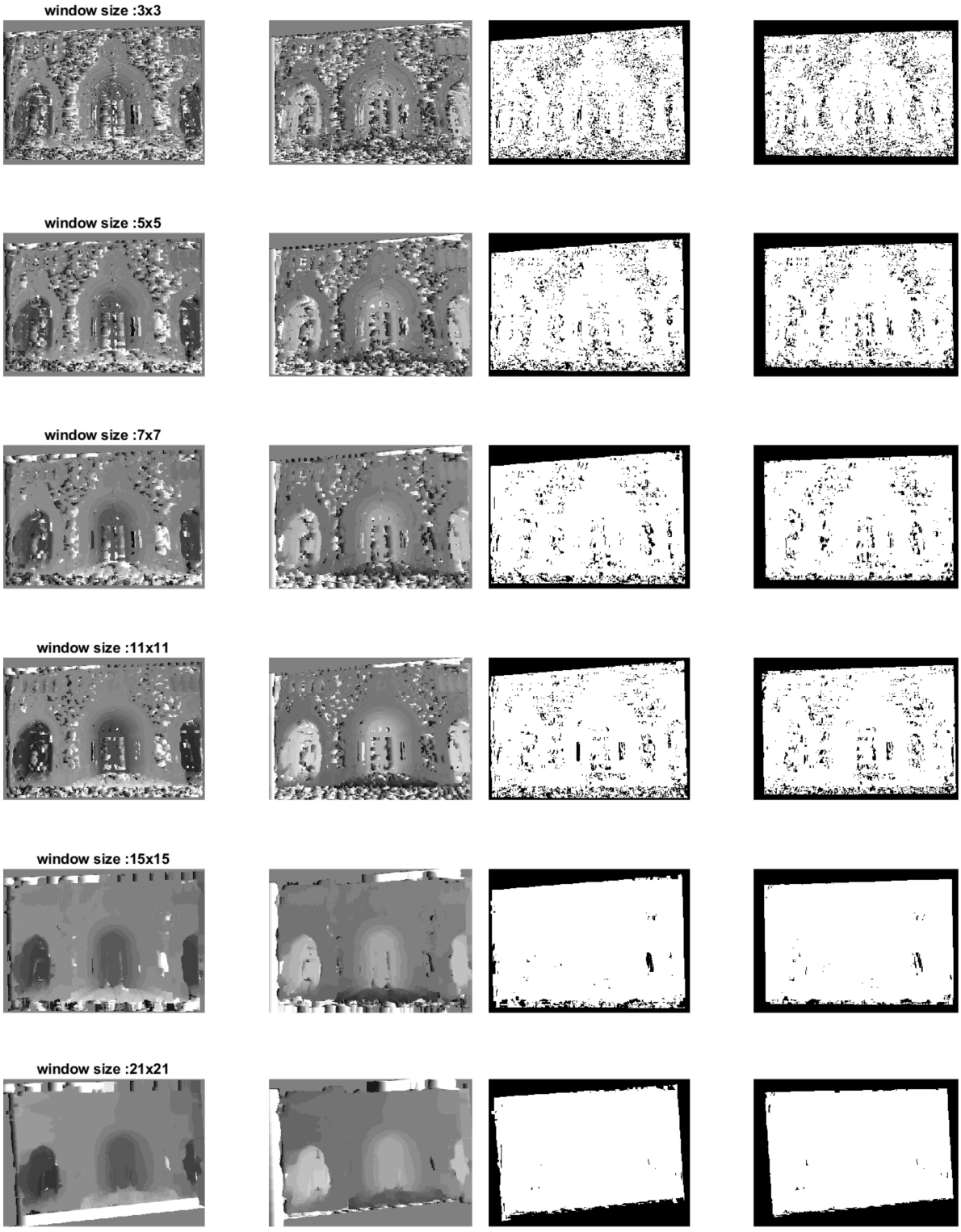
Nikita Rudin

November 15, 2018

1 Stereo Disparities

In the first part of this lab we implement the computation of disparities between two rectified images. We know that after rectification the match of a point should be on the same horizontal line. Therefore for a range of disparities (-15:15 for scale = 0.5^2) we compute the SSD over a window around each pixel. The disparity that provided the minimum SSD is recorded for each pixel. We can do this going from the left to the right image and the other way. We then compare the results and remove points have very different values in the two cases. The range of disparities was calculated using clicked points and taking the minimum and maximum of the differences on x. This could be done automatically by getting the appropriate matches with SIFT for example. The following images present the results for different window sizes:

We can see that for small window sizes the results are way too noisy. This is due to the fact that a wrong match can be selected easily. For to large sizes however, we start to loose some of the details (the stairs for example)



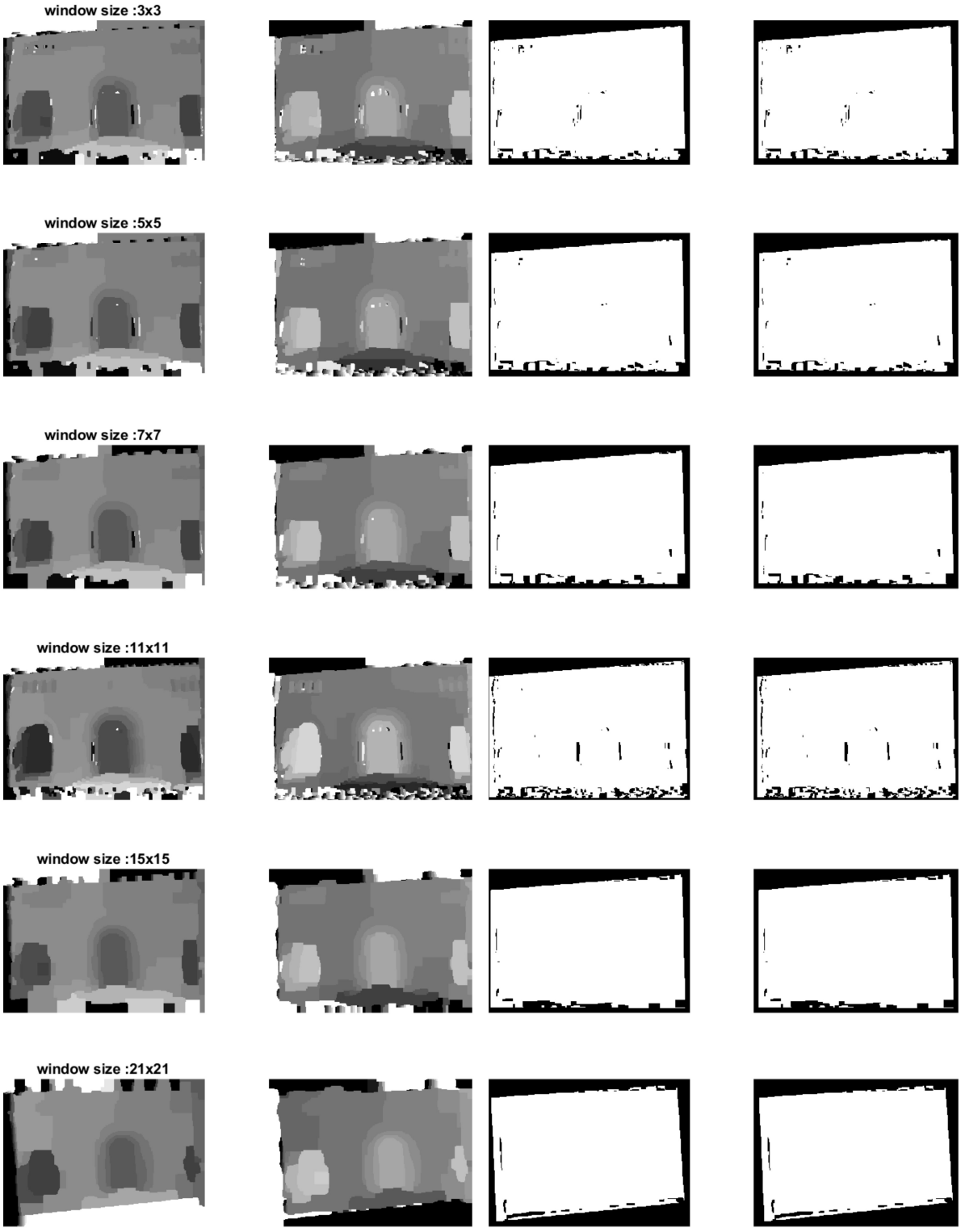
(a) Disparities (b) Errors in black

Figure 1: Comparison of disparities computation for different window sizes

2 Graph Cut

In this part we try to improve the results obtained before by using the graph cut algorithm to assign disparities to pixels. In addition to the SSD cost, which is computed as before, we add a smoothness cost. This means that there is a cost to assign a disparity different than the one of the neighbouring pixels. We can balance the two cost by adjusting parameters. This is trade-off between details and smoothness.

This the results are generally better. We see however that for large window sizes we are loosing a lot of details. The best size seems to be around 7x7. If we want to avoid mismatches between left and right we can go to 11x11. This will be done for the 3D reconstruction in order to limit the holes.



(a) Disparities (b) Errors in black

Figure 2: Comparison of disparities computation With Graph Cut for different window sizes

3 3D reconstruction

Finally we will use the disparities calculated in section 1 and 2 to triangulate our points and get a 3D model of our scene. In order to triangulate correctly we need to take into account the scale. For this we can multiply our PR and PL matrices with the inverse of S.

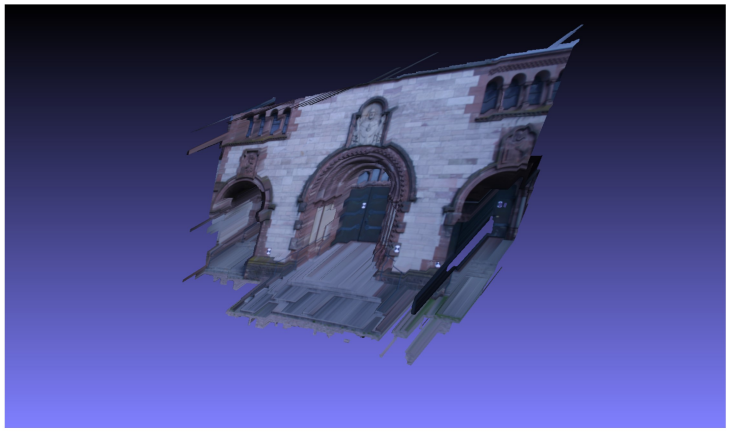


Figure 3: Reconstruction with Graph Cut

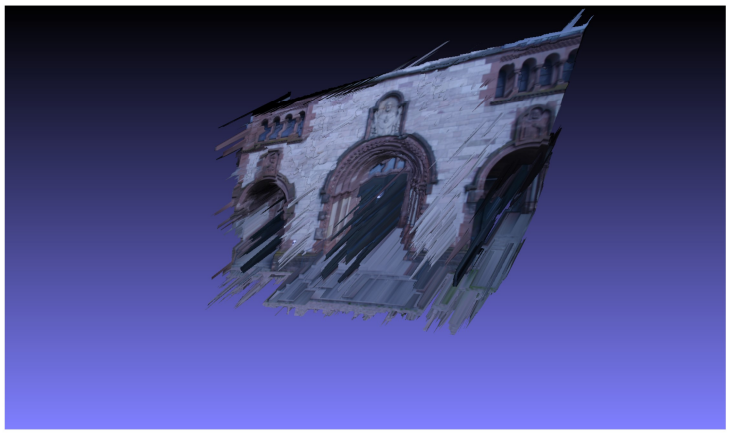


Figure 4: Reconstruction without Graph Cut

As expected the Graph Cut algorithm enforced smoothness and therefore eliminated a lot of noise. We still see some errors, but the global shape is well reconstructed. The result is of course still far from ideal and it can be that a better tuning of the different parameters of the cost functions could provide a better result.