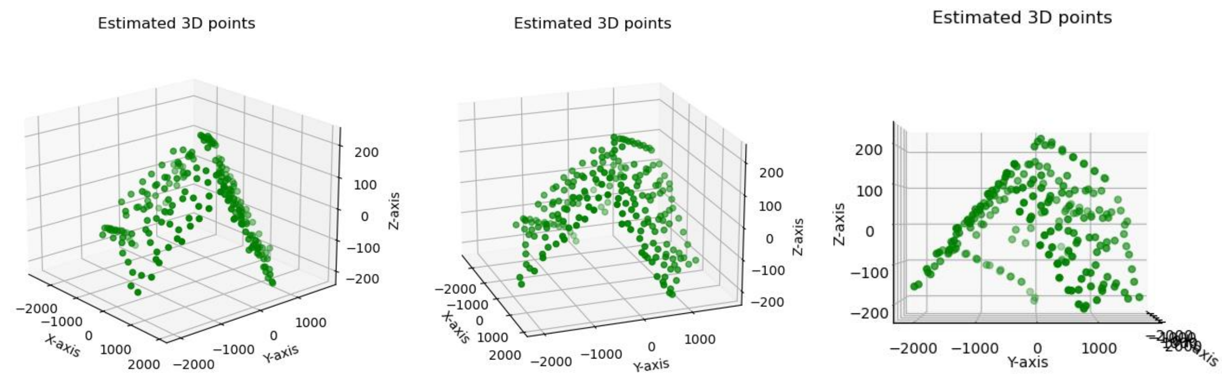
CS543/ECE549 Assignment 5

Name: Kaylin Chen
NetId: tc49

Part 1: Affine factorization

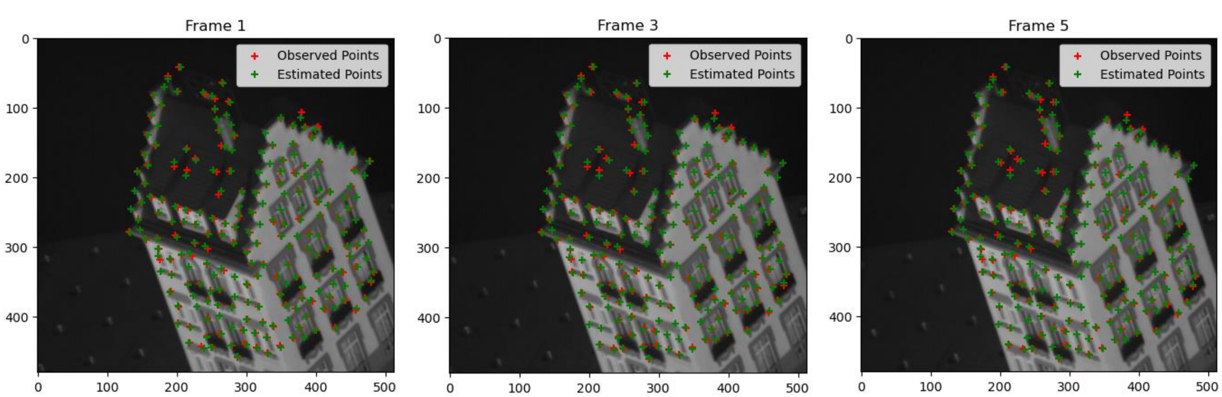
A: Display the 3D structure (you may want to include snapshots from several viewpoints to show the structure clearly). Report the Q matrix you found to eliminate the affine ambiguity. Discuss whether or not the reconstruction has an ambiguity.



Q matrix:
[[7.94778741e-03, 0.00000000e+00, 0.00000000e+00]
 [-7.14716488e-18, 8.53955215e-03, 0.00000000e+00]
 [8.52597488e-19, 1.42832718e-17, 2.53757864e-02]]

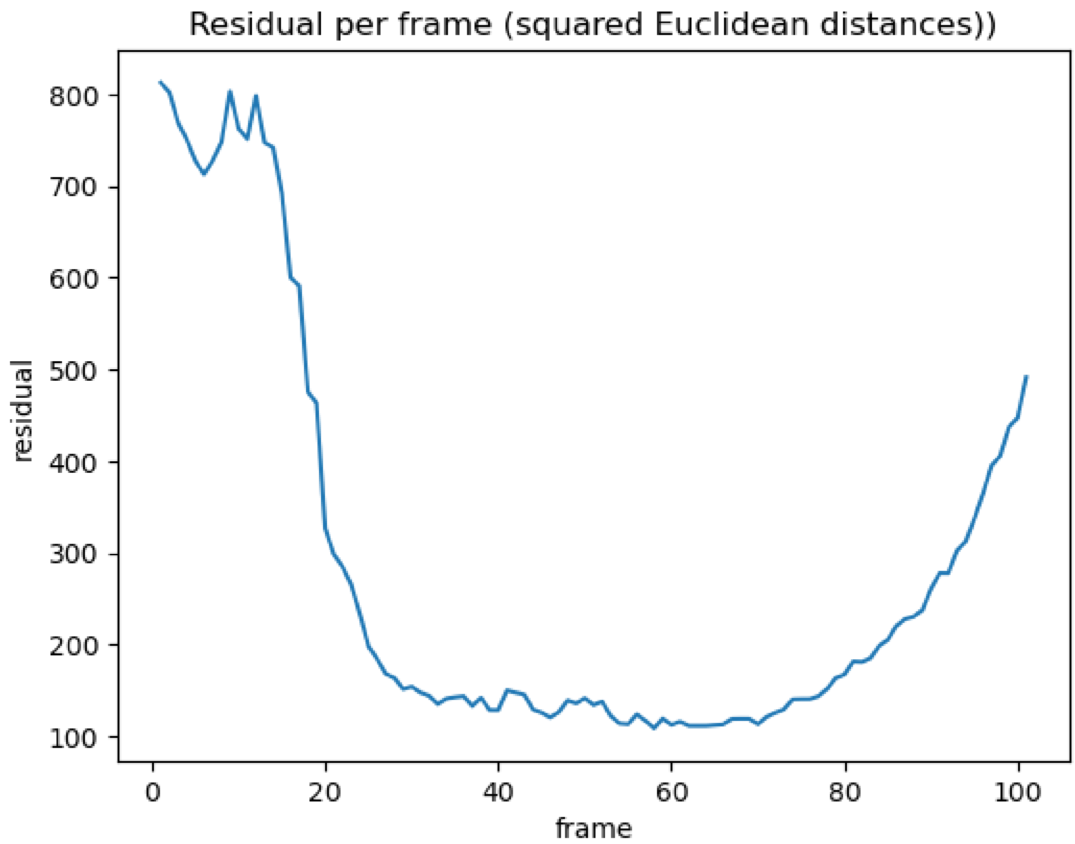
Though we eliminate the affine ambiguity by applying affine constraint, the construction still have the similar ambiguity which might output a structure with different scaling.

B: Display three frames with both the observed feature points and the estimated projected 3D points overlayed.



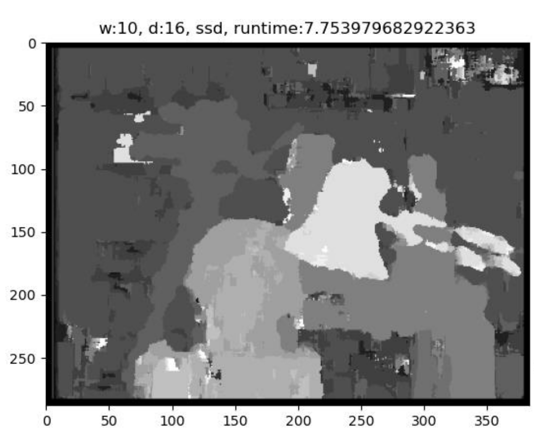
C: Report your total residual (sum of squared Euclidean distances, in pixels, between the observed and the reprojected features) over all the frames, and plot the per-frame residual as a function of the frame number.

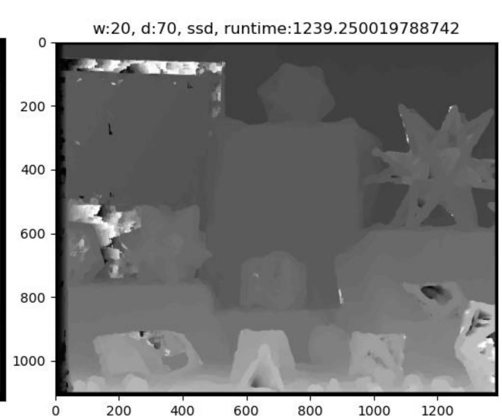
Total residual: 28583.41464098431



Part 2: Binocular stereo

A: Display best output disparity maps for both pairs.

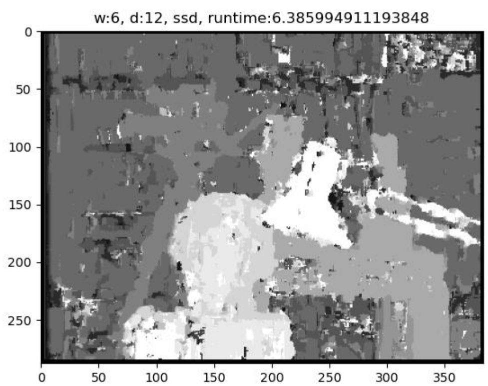


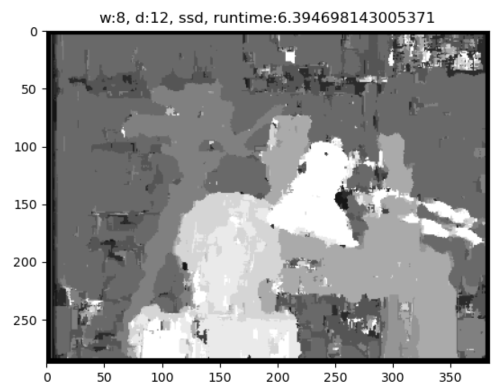


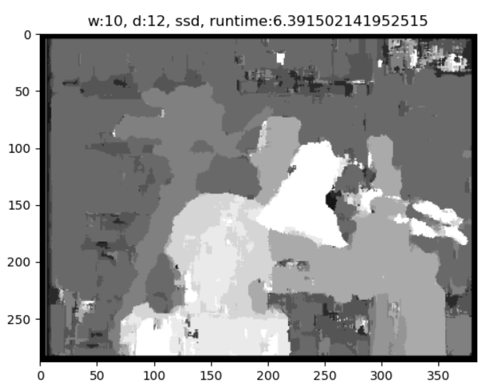
B: Study of implementation parameters:

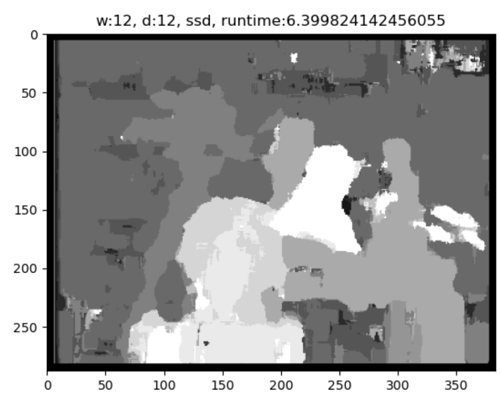
1. **Search window size:** show disparity maps for several window sizes and discuss which window size works the best (or what are the tradeoffs between using different window sizes). How does the running time depend on window size?

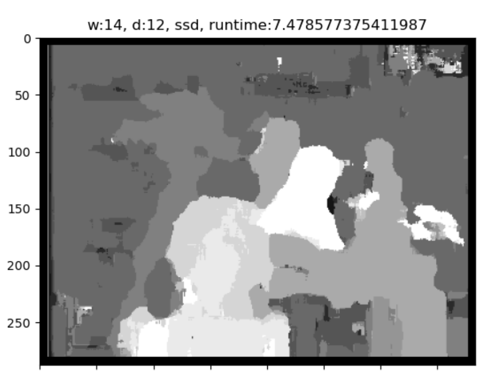
   Take the first pair of images as an example, the pictures below shows that when the window size is small, the running time is smaller, but the disparity map is also more coarsen. On the other hand, though the running time is higher when using larger window size, the output is smoother but with less details. Among these window sizes, I personally believe that window size 10 has the best balance between smoothness and capturing details.

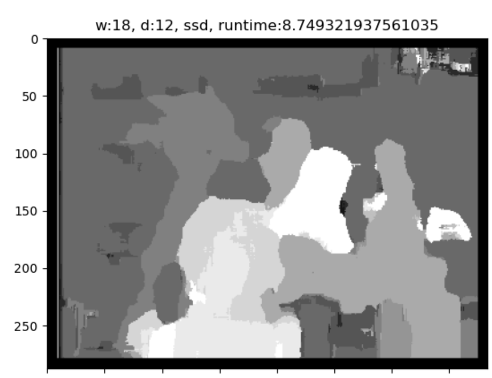








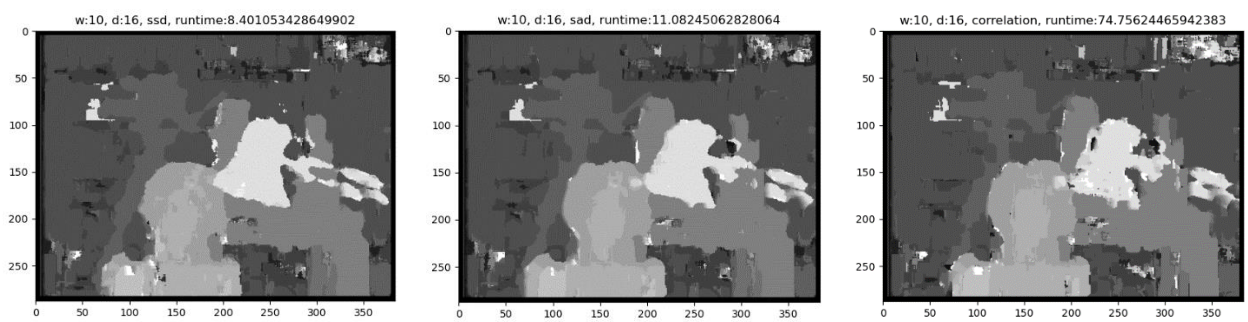




2. **Disparity range:** what is the range of the scanline in the second image that should be traversed in order to find a match for a given location in the first image? Examine the stereo pair to determine what is the maximum disparity value that makes sense, where to start the search on the scanline, and which direction to search in. Report which settings you ended up using.

   In both pairs of images, the objects in the second image are left shifted from the first image, thus when finding match for pixel at the j column in the first image, the reasonable range of the scanline should be j-disparity_range to j. We can find the disparity range from the shifted amount of the objects close to the camera, for example, the amount for the first stereo pair should be less than 25, while it should be less than 100 for the second stereo pair. I ended up using ranges of 16 and 70 for the pairs searching from left to right on the scanline to get the best result.

3. **Matching function:** try sum of squared differences (SSD), sum of absolute differences (SAD), and normalized correlation. Show the output disparity maps for each. Discuss whether there is any difference between using these functions, both in terms of quality of the results and in terms of running time.



Take the first stereo pair for example, when using the same window size and disparity range, SSD is the fastest with the best result. Though SAD has similar running time, the output tends to be more blurred. In contrast, normalized correlation requires significantly more time to complete the computation, and the output is coarser compared to the other methods.

C: Discuss the shortcomings of your algorithm. Where do the estimated disparity maps look good, and where do they look bad? What would be required to produce better results? Also discuss the running time of your approach and what might be needed to make stereo run faster.

The basic stereo matching algorithm works well on the textured surfaces as shown in the second pair but look bad on the texture less surfaces like the top right corner of the first pair, pixels where having moving specular effect such as the lamp in the first pair, and repetitive objects like the books in the background of the first pair. To produce better results, applying non-local constraints such as uniqueness, ordering, and smoothness as described in the lecture might be helpful. When using SSD and SAD method, the running time is around ten seconds for the first stereo pair but takes up to 1000 seconds for the second stereo pair due to the resolution of the images. Applying hardware acceleration or down sampling the images as we did in the first assignment might be needed to speed up the process.

Part 3: Extra Credit

Post any extra credit for parts 1 or 2 here. Don’t forget to include references, an explanation, and outputs to receive credit. Refer to the assignment for suggested outputs.