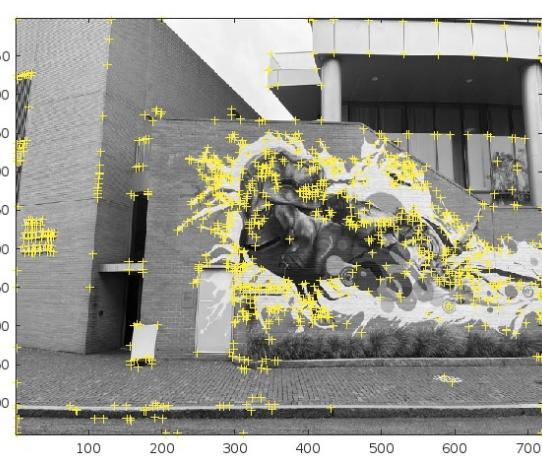
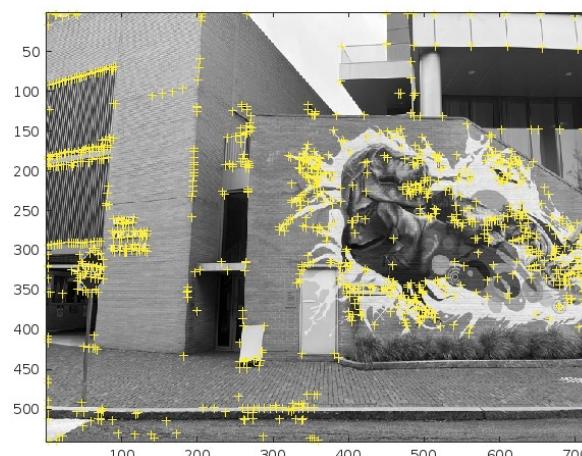
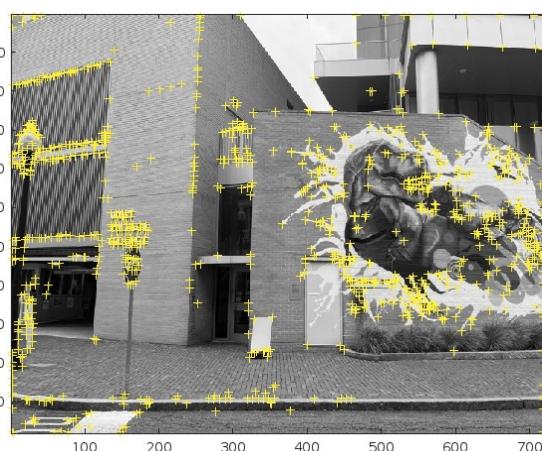
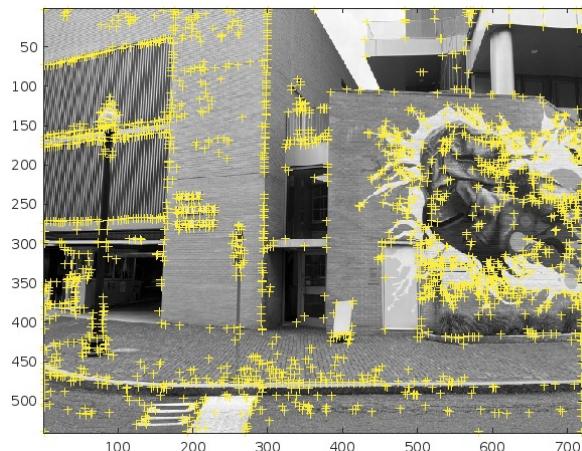
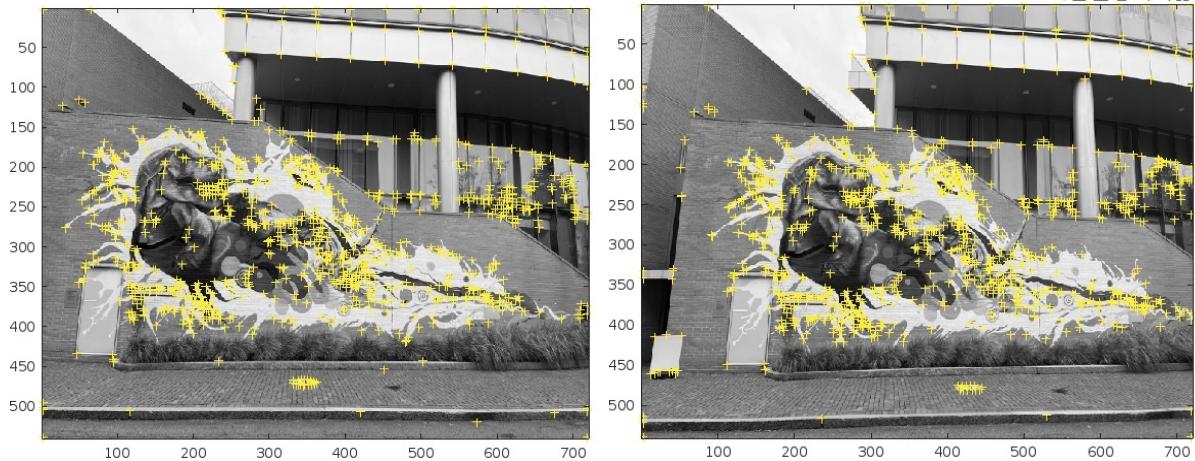


LAB 5

Mural 1- Trex





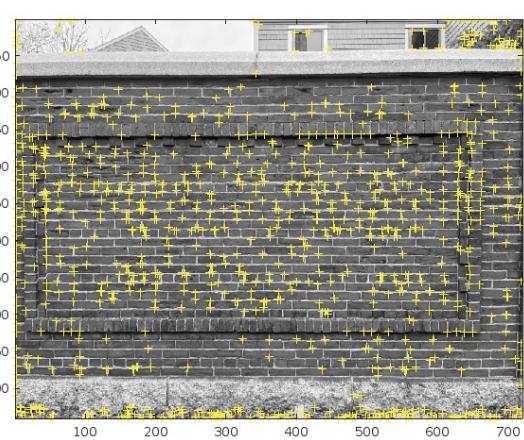
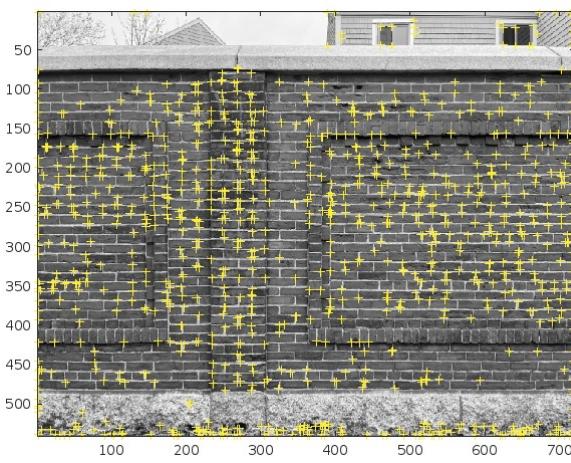
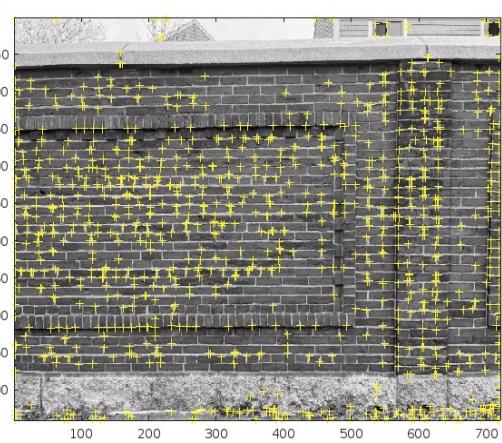
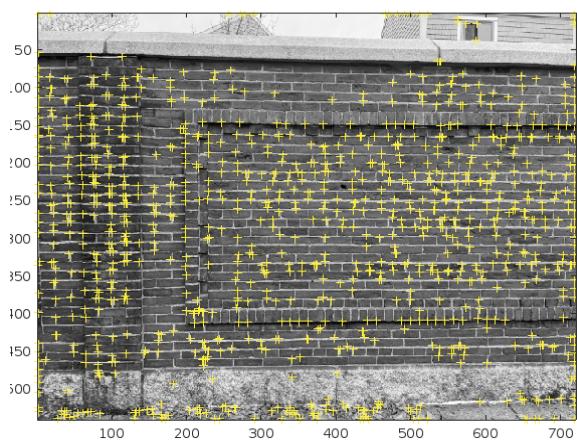
Comments:

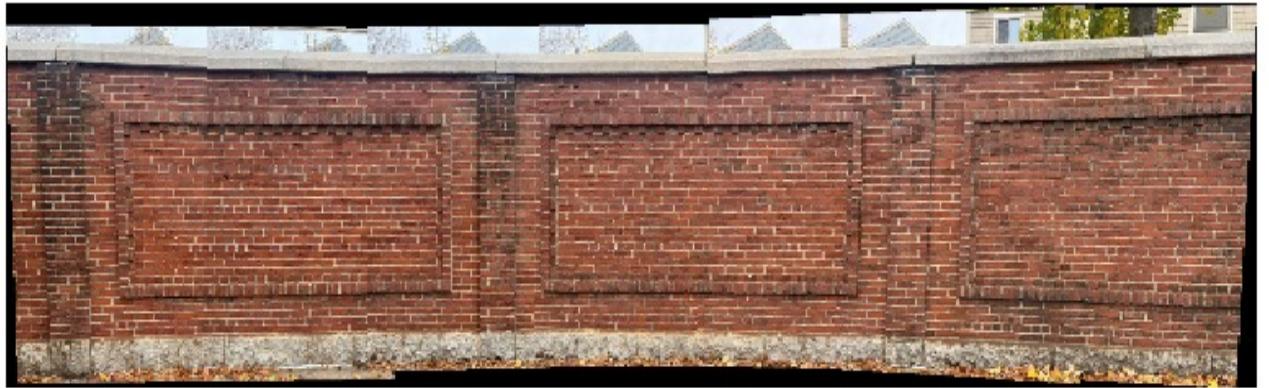
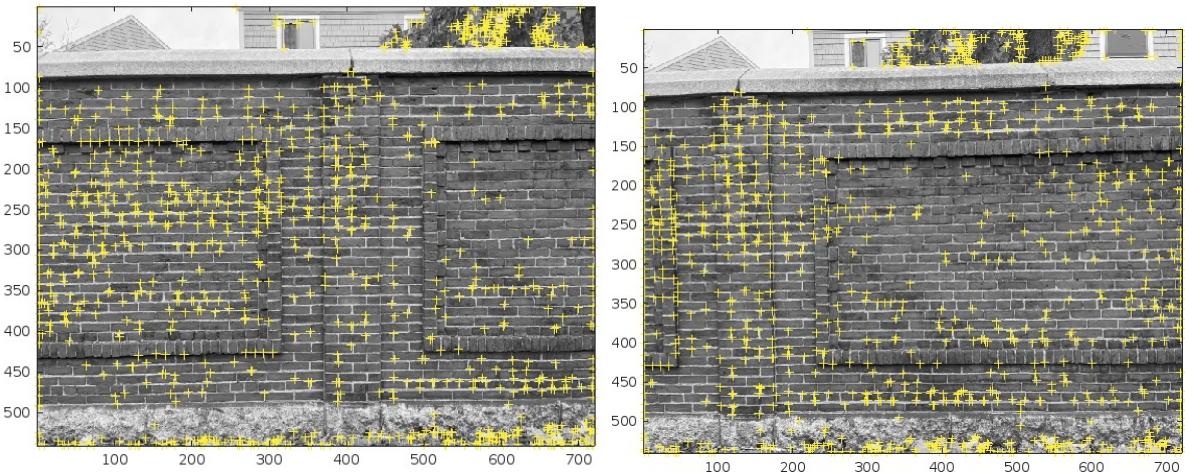
The code tried to match the best features from the pictures and that's why the mural appears clean whereas the buildings appear a little distorted. Also, the features detected in one image are not always detected in the next image. The camera had to also be perfectly aligned at all times. Also, it is harder to detect the 3D features of the image as we are using a 2D detector, that's why the mural appears better than the rest of the structures.

Cinder block

Comments:

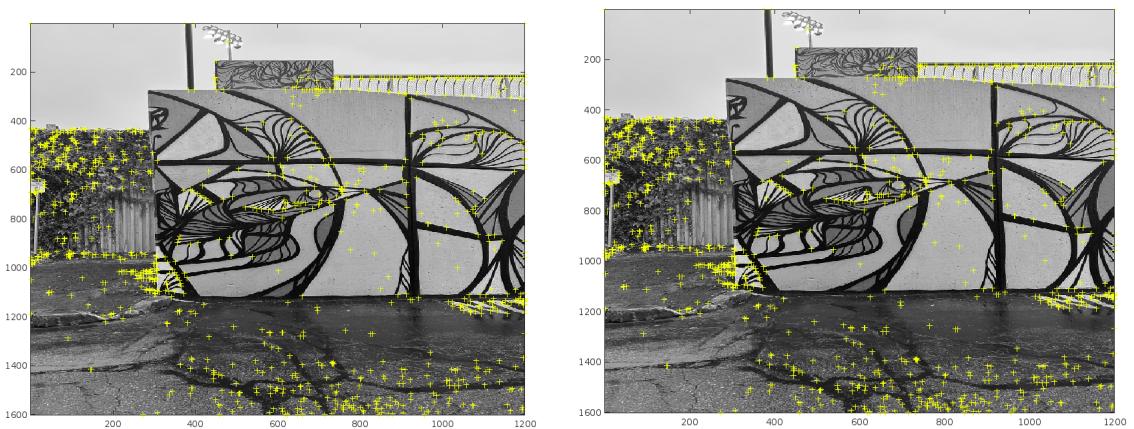
I tried to take the best pictures for this part and hence all my photos are perfectly aligned. The features across the photos are uniform and corners are well distinguished between all bricks which make it a really perfect match and a cleaner panorama. The overlapping of features because of the 50% overlap of the images make this dataset work good with the given code.

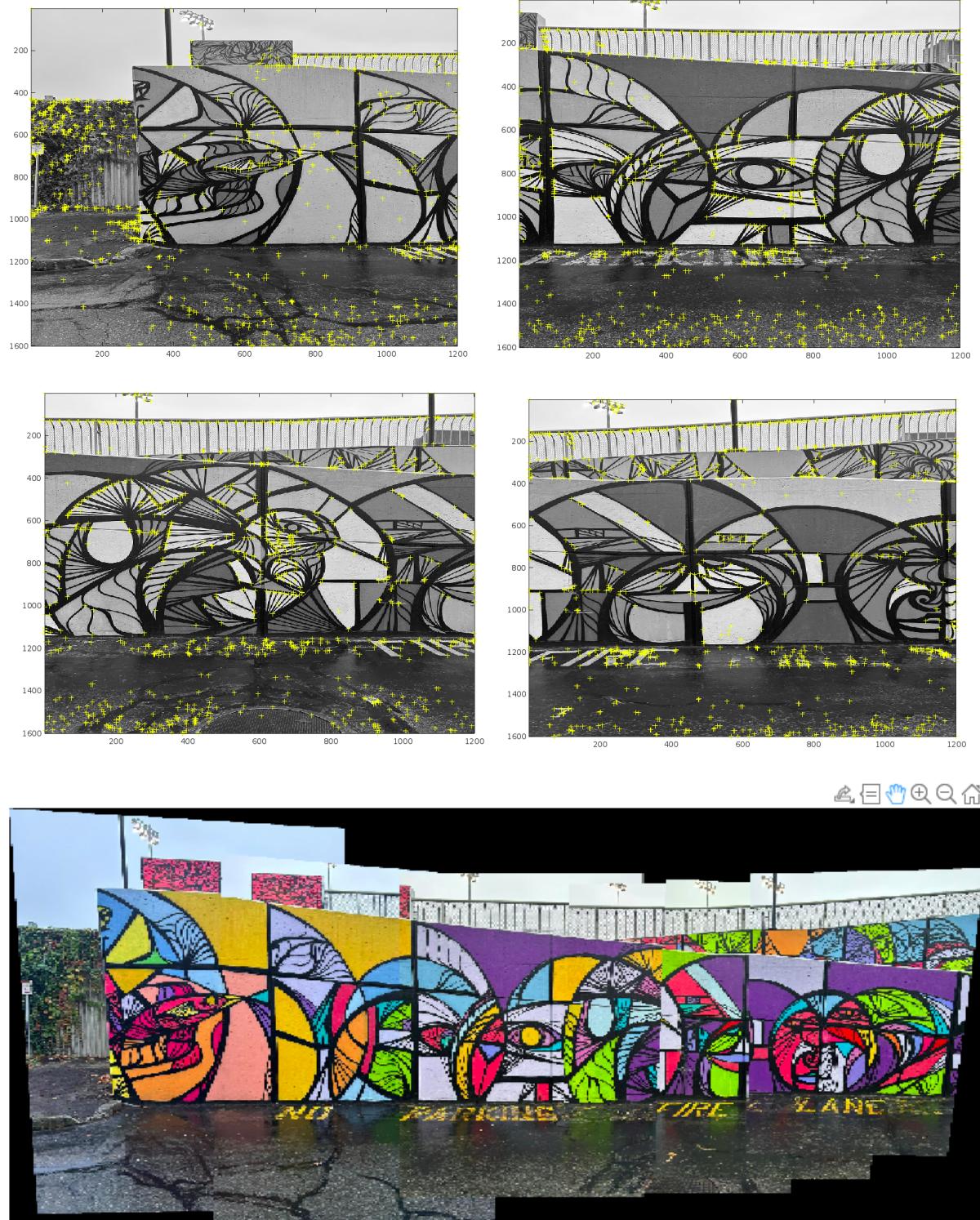




Mural 2 (behind Curry Student centre): 15% overlap

I feel like I messed up with this dataset with too many overlapping features... Please ignore this while grading.

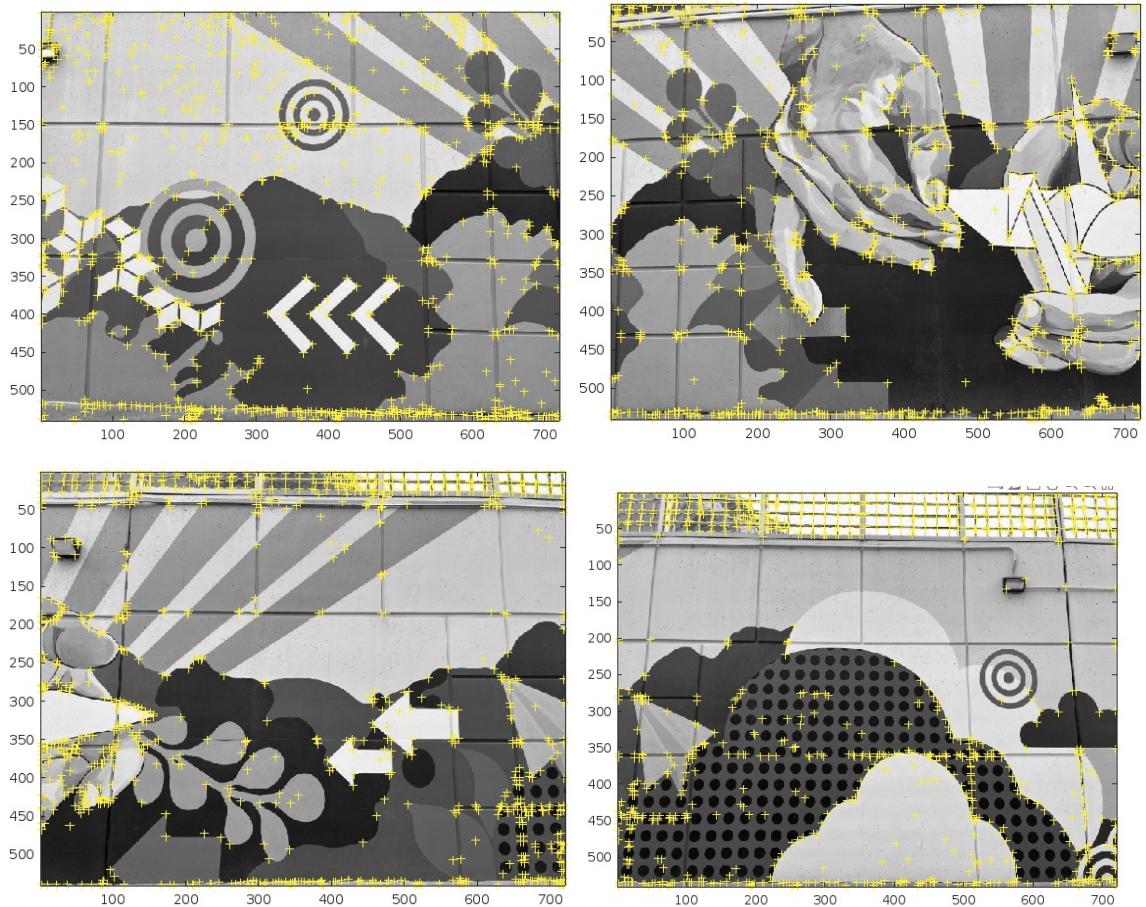


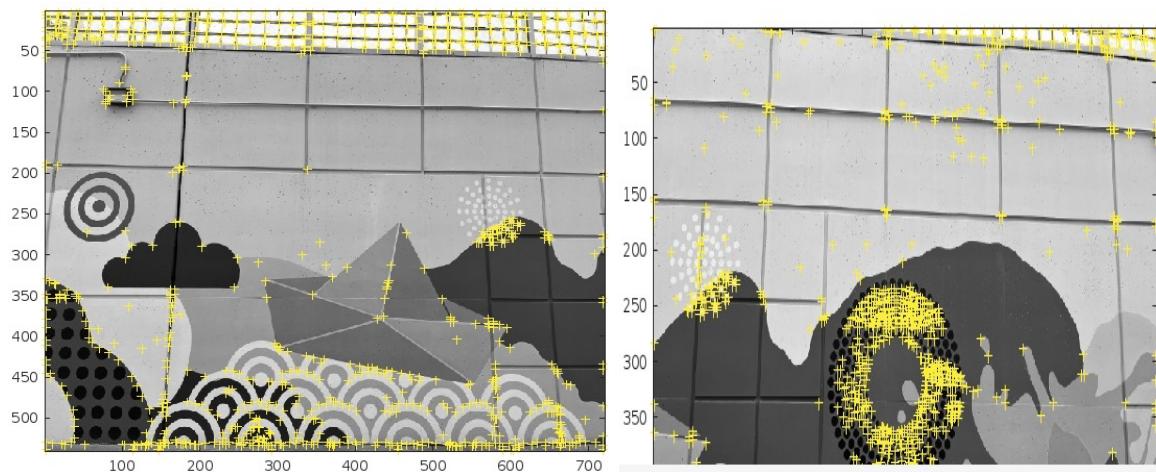


Comments: Too many distinct and common features and good overlapping makes a good panorama for this dataset

Final mosaic

Dataset: <15% overlap

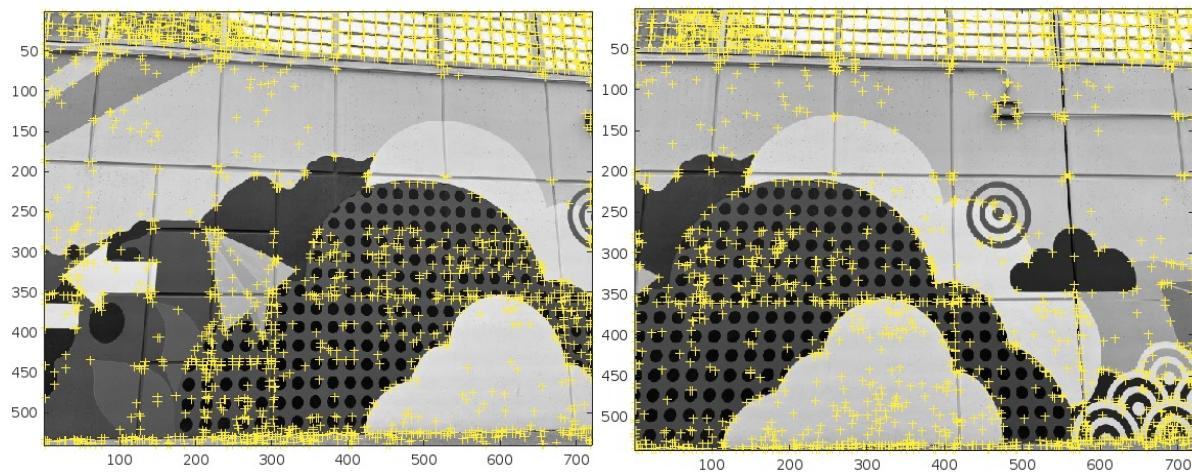




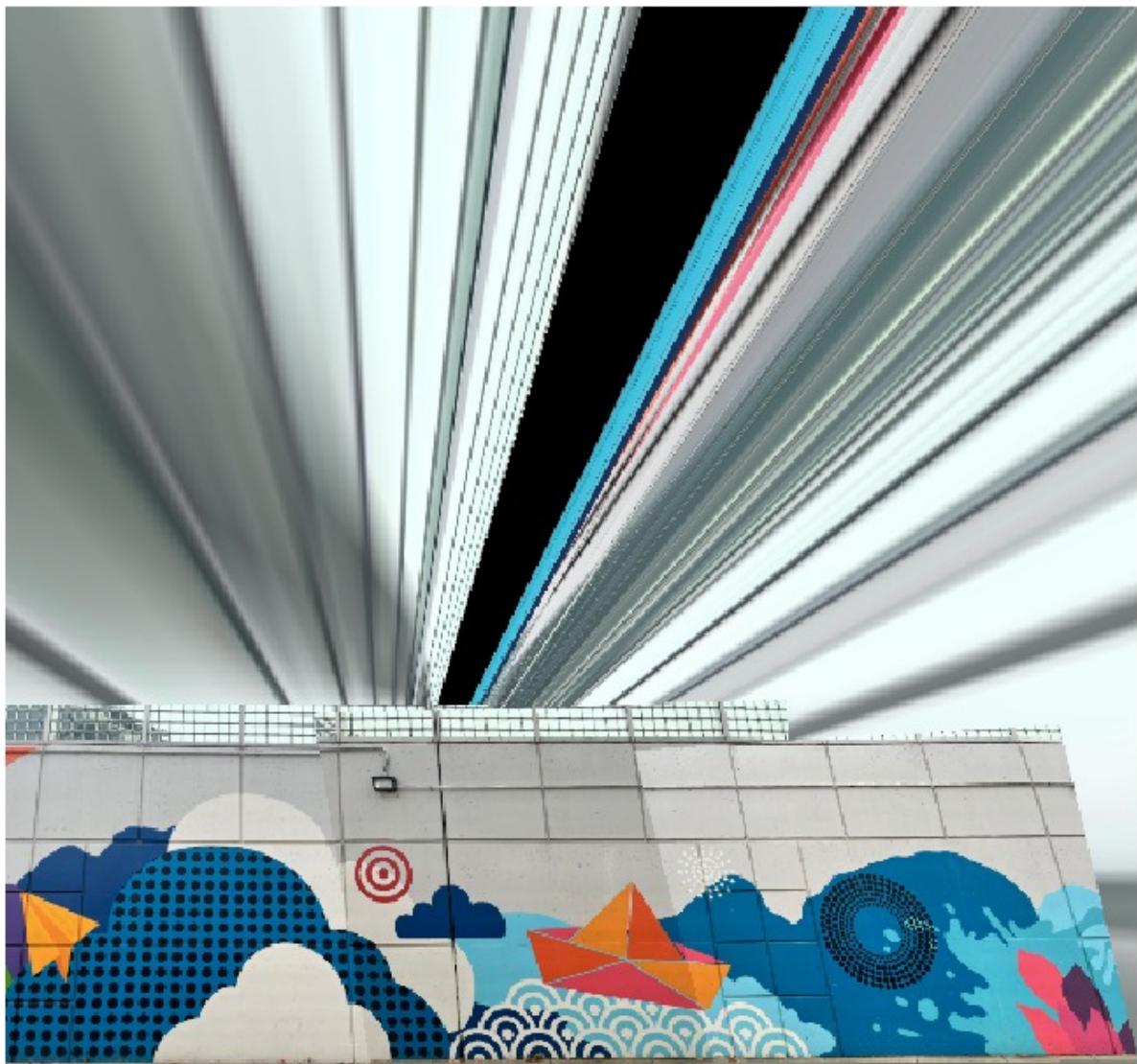
Dataset for 50% overlap







Panorama with <15% overlap



Panorama with 50% overlap



Comments: Since overlapping between images is very less it makes it very difficult to identify common features and thus the transformations are not precise which leads to misalignments. Also, projective transformations are not very accurate for images with smaller overlapping features. This is also why I got a fairly distorted and stretched picture as my final result. I have used the same code for all images and I had to change something within the code for it to give a better panorama.

With better overlap I have also gotten a better panorama as seen above with the same code because of better feature detection as seen by the number of yellow dots.

What have I changed?

Changed projective transformation to affine transformation in the code

Changed number of feature points from 1000 to 2000

Changed tile size from [2 2] to [1 1]

The change in transformation as affine transformation works better with parallel lines, plus application of more features and smaller tile size

New result with 15% overlap:

