

## Create multi-Image mosaic

### Algorithm:

First, my algorithm will pick a pair image. Then I use ORB detector to extract keypoints and using symmetric criteria to find the best match.

When I have the best match, my alg. will judge if it have at least 10 keypoints match. If it has more than 10 keypoints, then it will calculate the Fundermental matrix. And In order to get an accurate F matrix, I use RANSC to pick some outliers. If inliers take up at least 20% keypoints, then I consider it can match. And if inliers below 20%, then for this pair, it will not do anything.

If it can match, then my algo. will calculate the Homography matrix and also it will eliminate some outliers with RANSC. Same with F matrix, if at 20% keypoints left, then we will map this two image with cv2.warpperspetive.

When I transform one image to another, the shape will change. So the warped image which I get is not complete. Thus I calculate the 4 corner coordinates to find the bound of the new warped image and find its new origin. Then I add the origin to Homography which we have already calculated. Then I warp image again to get a complete warped image. I have the transition coordinates, thus I add anchor size of image to this coordinates. And I have the size of mosaic image. In order to map two image, I will fix one image and map another onto it.

Finally, I use the average blending method to create a mosaic. The average blending method means that I use 0.5 as a weight for two images in overlap regions.

### Multi-image mosaic:

My algorithm can find all the pair images. And for this part, I consider each image as a node and if it is in a pair images, then i will think it has one link. And if a node has the most link, then my algorithm will consider it as anchor image. The anchor image will fix. Then my algo. will find all direct connected nodes. For all direct connect nodes, my algorithm will judge if there is another image connecting with them. If another image exists, then my algorithm will map them first and then map it onto the anchor image.

When all connect nodes are found, my algorithm will map each of them onto the anchor image.

### Results and discussion:



Figure.1



Figure.2

In my algorithm, I use the percentage of inliers to as the criteria. As you can see, for these image, it is a good match. But from the percentage, we may get a conclusion that it is very hard to match. The reason is that these two images have small overlap parts. Therefore, the outliers will take up most. It can lead the result to a bad result.

In next figure, we can see that it also have about 20 percentage match. But they are not a good match.



Figure.3

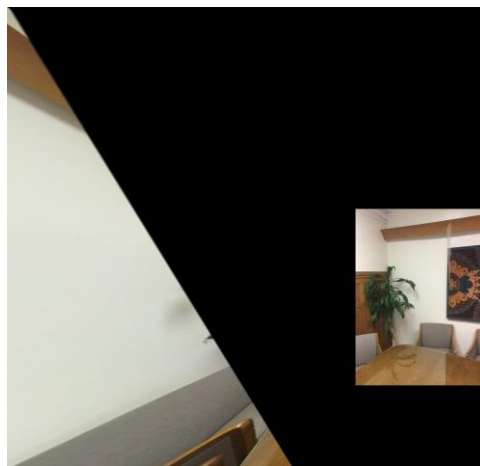


Figure.4

Because the shape of object is circle, thus, according to my algorithm, it still has minimum percentage to match. Thus for some object which has circle shape and symmetric, it might have totally wrong match.

My algorithm also has lots of good results. As shown as followed.



Figure.5

Figure.6



Figure.7



Figure.8



Figure.9

For these good results, I find that if two scene do not have angle or have very little angle, it can create a nice mosaic. If these scene have a large angle, them it is very hard to match. As shown in figure.10.

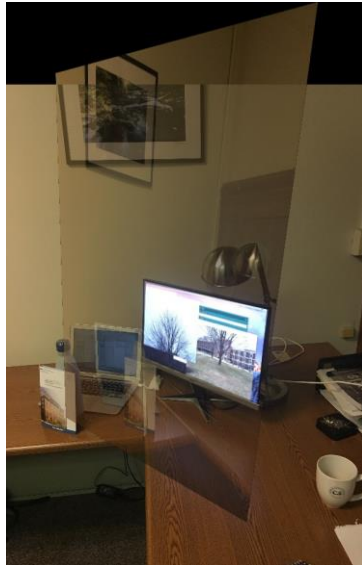


Figure.10

And from the keypoints, we can know that most keypoints are focus on computer, but other things are not a nice match.

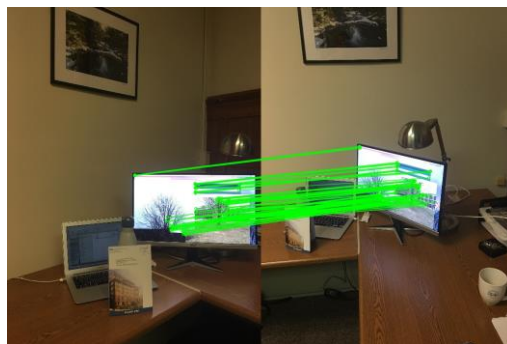


Figure.11

In this project, my blending algorithm is not very sophisticated but it very simple. And it can have a nice blending effect. It is not like adding some image on another directly. It emerges one into another.

And my algorithm can create a nice mosaic. It can find a proper anchor, thus, the mosaic looks like very peaceful.



Figure.12