ASSIGNMENT 3 - PANAROMIC STITCHING

EE17B047 - KOMMINENI ADITYA

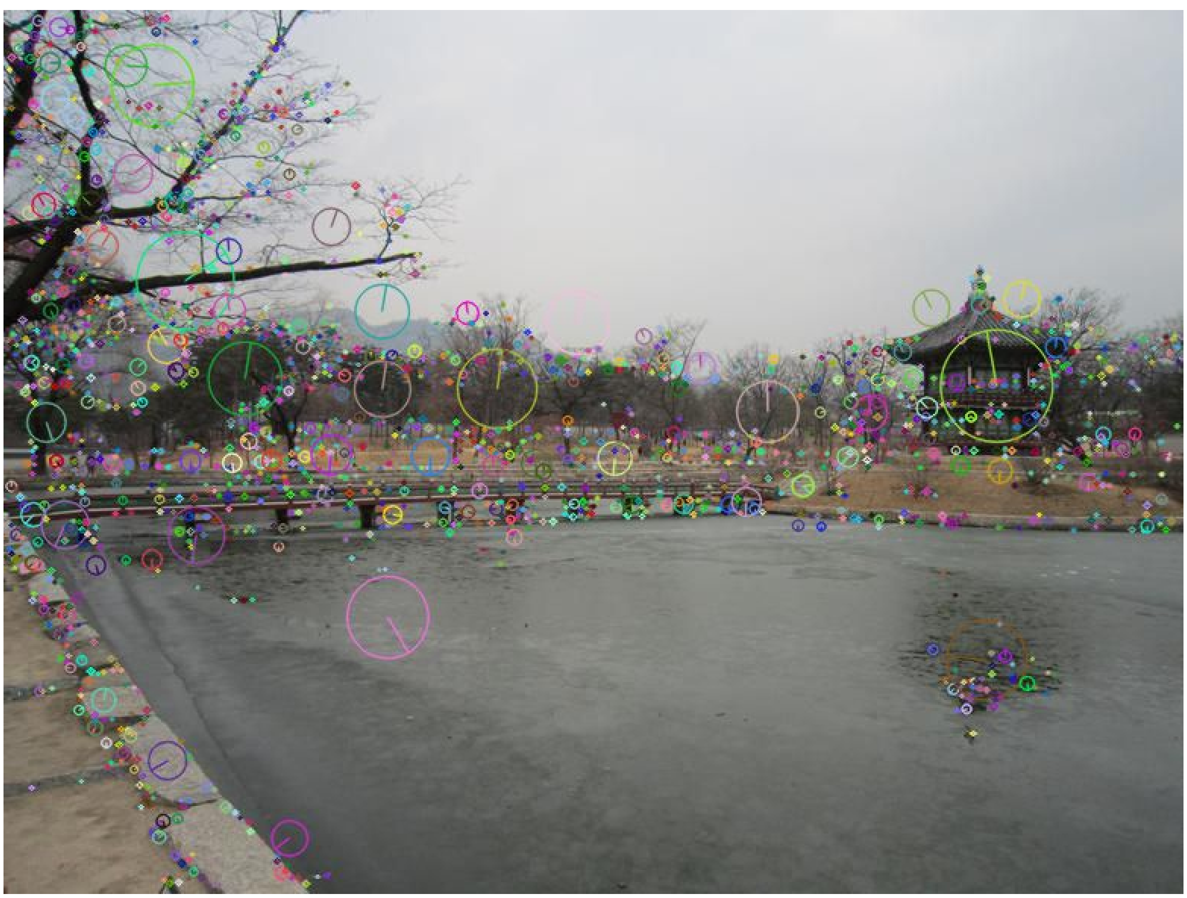
TASK 1 :

In this task, we are required to implement two functions, get_features and match_keypoints.

Firstly, the get_features functions computes the key points of a given image and computes the corresponding descriptors of the key points. In the code, I have employed cv2.sift function which calculates the SIFT descriptors around the key points.

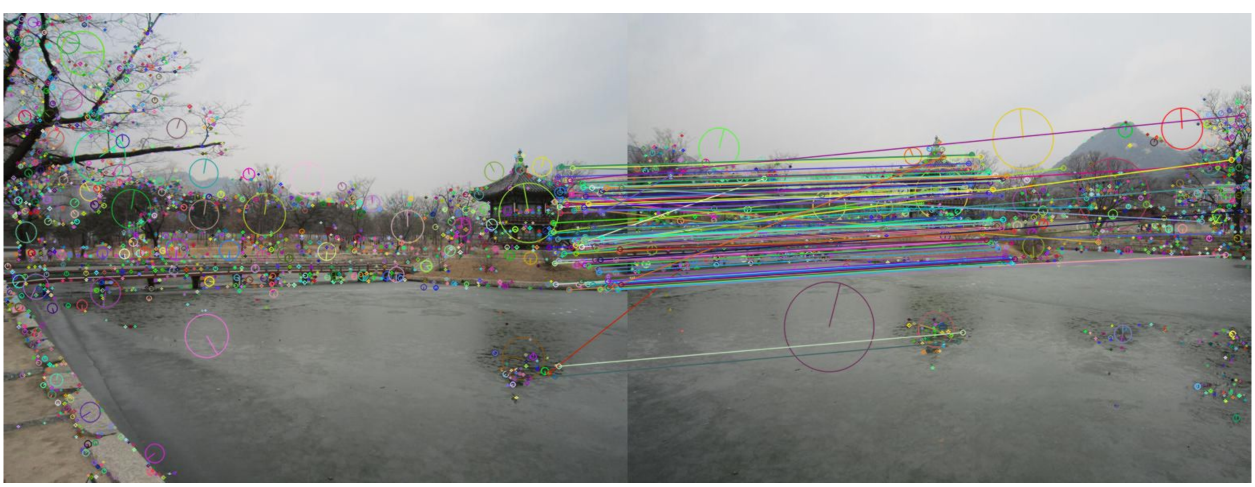
match_keypoints function takes both the descriptors for the images as the inputs and gives the corresponding matched key points as the output. In order to determine which keypoints are common to both the images, we employ a ratio test.

SIFT keypoints corresponnding to image 1 are as below :



SIFT key points corresponding to image 2 are as follows :

Below is the image which has the matching keypoints in both the images :



From the above image, we can see that most of the key point matches correspond to actual correspondence in the image features. However, there are certain keypoints which do not correspond to the same areas in the images.

TASK 2 :

In order to calculate the Homography matrix for the two image homography, we employ direct linear transform(DLT). Firstly, using the two sets of matching points, we compute the set of linear equations for the relation between the two sets of points.

At this points, we will have an equation Ah = 0 where A is an N*9 matrix, N is the number of matching key points. We can calculate the coefficients of the homography matrix(h) by computing the singular vector decomposition of A and taking the singular vector corresponding to the lowest singular value i.e.

$$A = U\boldsymbol{\Sigma} V^T$$

Here, we take the column of V corresponding to the lowest singular vector

We can calculate the SVD using the np.linalg.svd. This function orders the singular values in their descending order thereby, we can take the last column of V.

The homography matrix(H) is as shown below :

-1.36188238e-03	-3.21642529e-04	5.08913096e-01
-2.34705344e-03	-4.77032014e-04	8.60811869e-01
-4.42479267e-06	-1.00380011e-06	1.64088891e-03

TASK 3 :

In this, we implement the RANSAC algorithm. The steps to be followed are as follows :

1) Sample 4 points at random.
2) Compute the Ah = 0 equation.
3) Obtaing the homography matrix using the 4 points.
4) Now, compute the keypoints of image 1 in space of image 2 using homography. Then, compute the eucledian distance of the transformed points to the original key points of

image 1. If the transformed points are within 3 pixels of the original points, we declare them as inliers.

5) Repeat Step 1 and keep the Homography matrix which gives the maximum number of inliears.
6) Now, recalculate the homography matrix using all the inliers.

The Homography matrix after running RANSAC algorithm is :

2.85800039e-03	4.84060962e-04	-3.45108661e-01
-5.53513100e-06	3.25411931e-03	-9.38550940e-01
2.62200459e-07	1.79059950e-06	1.77555291e-03

Below are the images which show the inlier and outlier patterns for image1 and image2. From the images, we can see that the most of the inliers lie in the parts of the image which are common to both. Showing that RANSAC does a decent job in identifying the inliers.

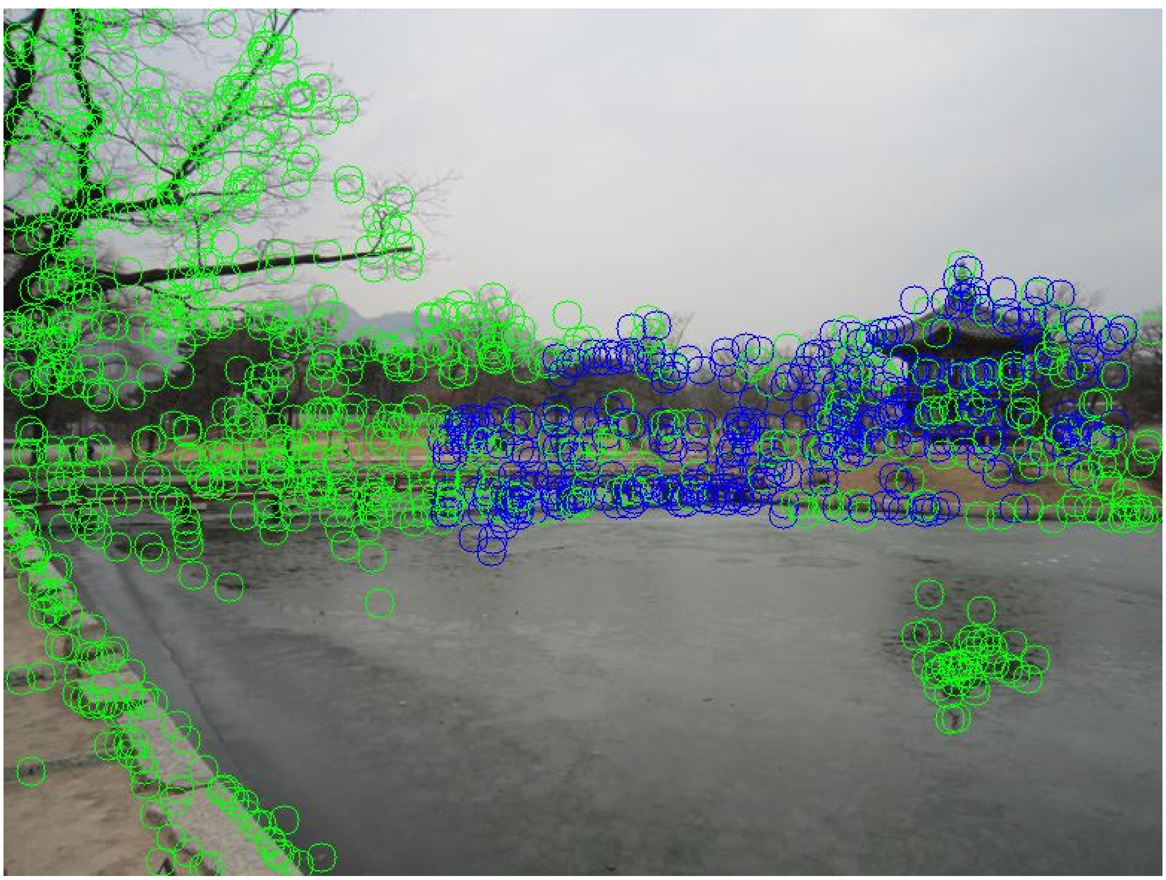


IMAGE 1(BLUE - INLIERS , GREEN - OUTLIERS)

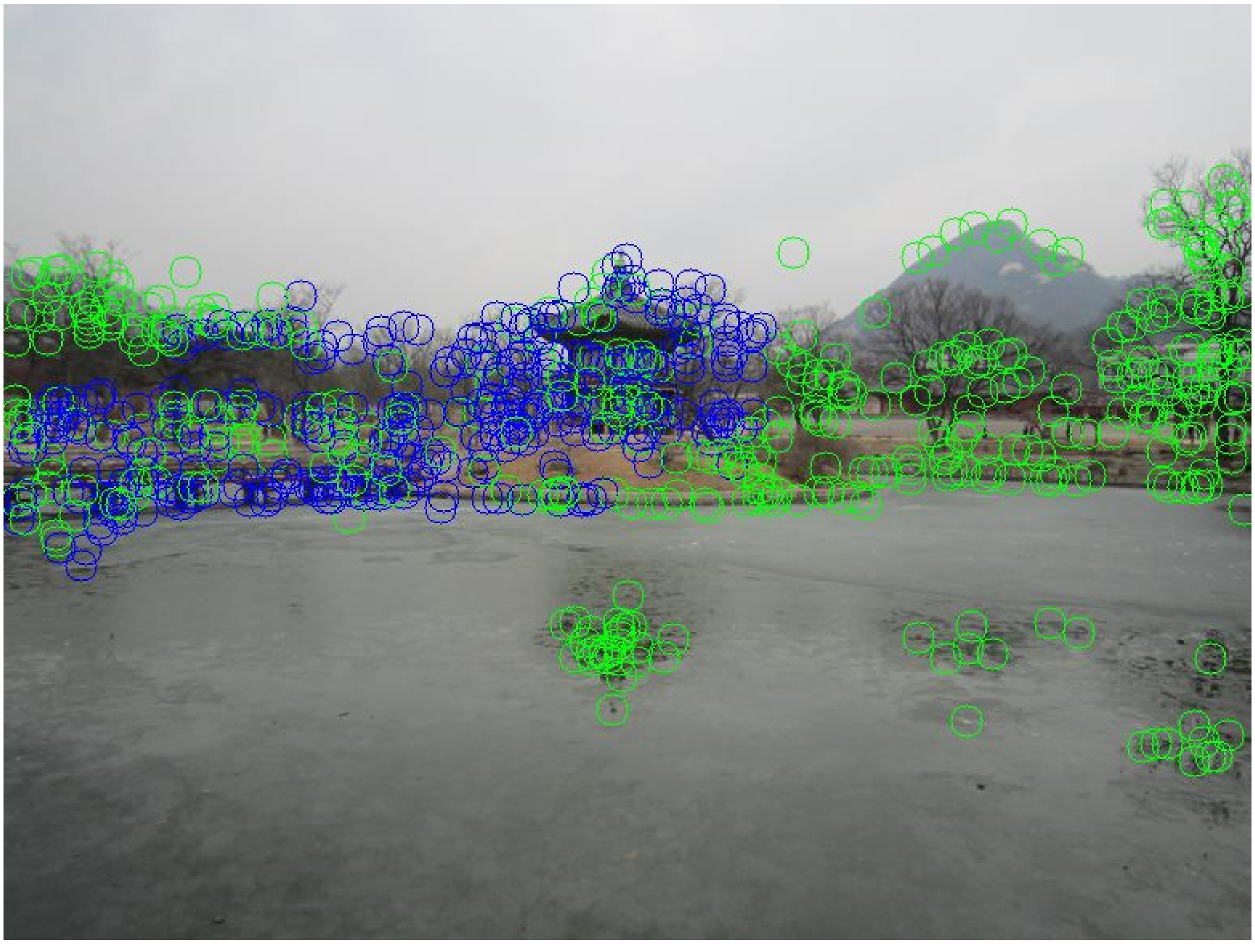


IMAGE 2(BLUE - INLIERS , GREEN - OUTLIERS)

TASK 4 :

Here, we stitch the images together employing the homography matrix computed in TASK 3. The procedure to stitch the two images is as given below :

1) We compute the inverse of the point in image 2 which corresponds to image 1.
2) If the point has an actual intensity in image1, then we assign the output image the same intensity of the pixel as in image1.
3) In this manner, we loop over all the pixels of the output image2.
4) After this, we copy image2 to the output image.

The output image is as given below :

We observe that instead of emplying this method i.e. inverse mapping, if we try to implement forward mapping, we will observe black patches in the stitched image. This is because in the forward mapping case, there might be pixels for which the original image may not have an exact correspondance since we approximate the pixels to their nearest integer rather than interpolating them. Hence, it will lead to such patches of black.

TASK 5 :

In order to stitch 3 images together, we keep the central image a constant and compute the homography matrices corresponding to image 1 and image 3. Then, we implement the porcedure as implemented in TASK 4 in order to stitch them together.

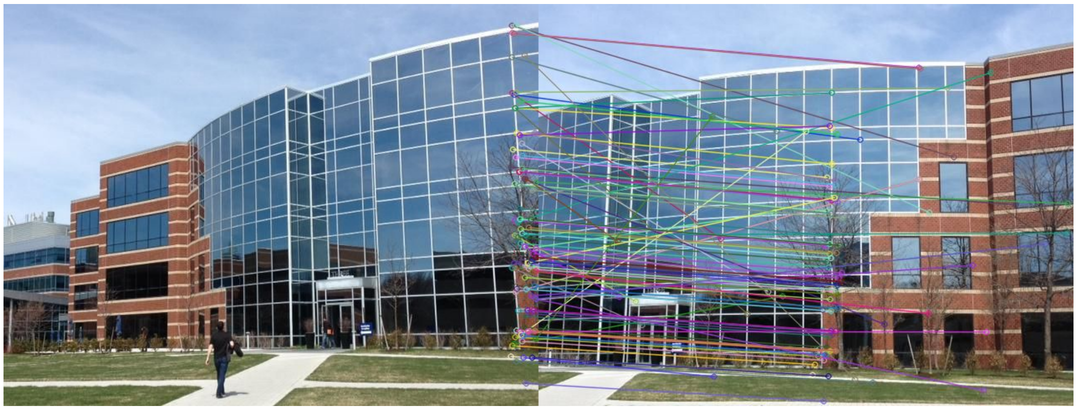
The SIFT keypoints corresponding to image1, image2, image3 are as follows :

SIFT FEATURES FOR IMAGE 1

SIFT FEATURES FOR IMAGE 2

SIFT FEATURES FOR IMAGE 3

The SIFT features key points matches are as given below :



FEATURE CORRESPONDENCE FOR IMAGES 1, 2

FEATURE CORRESPONDENCE FOR IMAGES 2,3

The inliers and outliers obtained from the RANSAC algorithm for image1, image2 are as follows

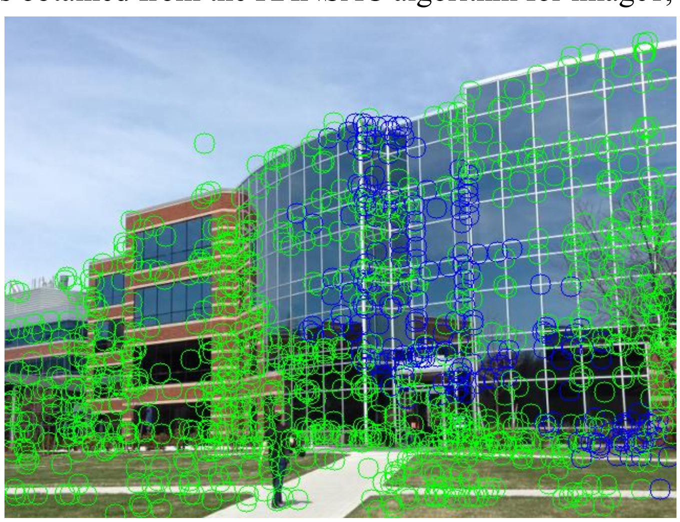


IMAGE 1 (BLUE - INLIERS , GREEN - OUTLIERS)

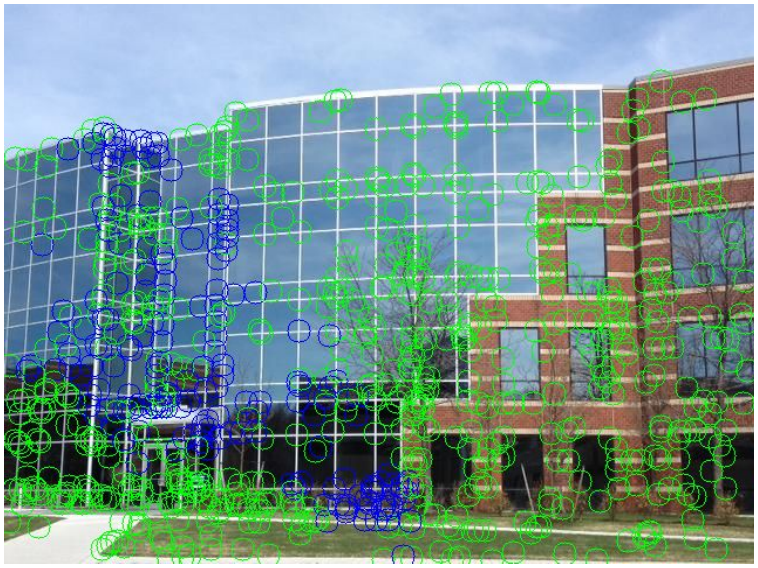


IMAGE 2(BLUE - INLIERS , GREEN - OUTLIERS)

The inliers and outliers obtained from the RANSAC algorithm for image3, image2 are as follows

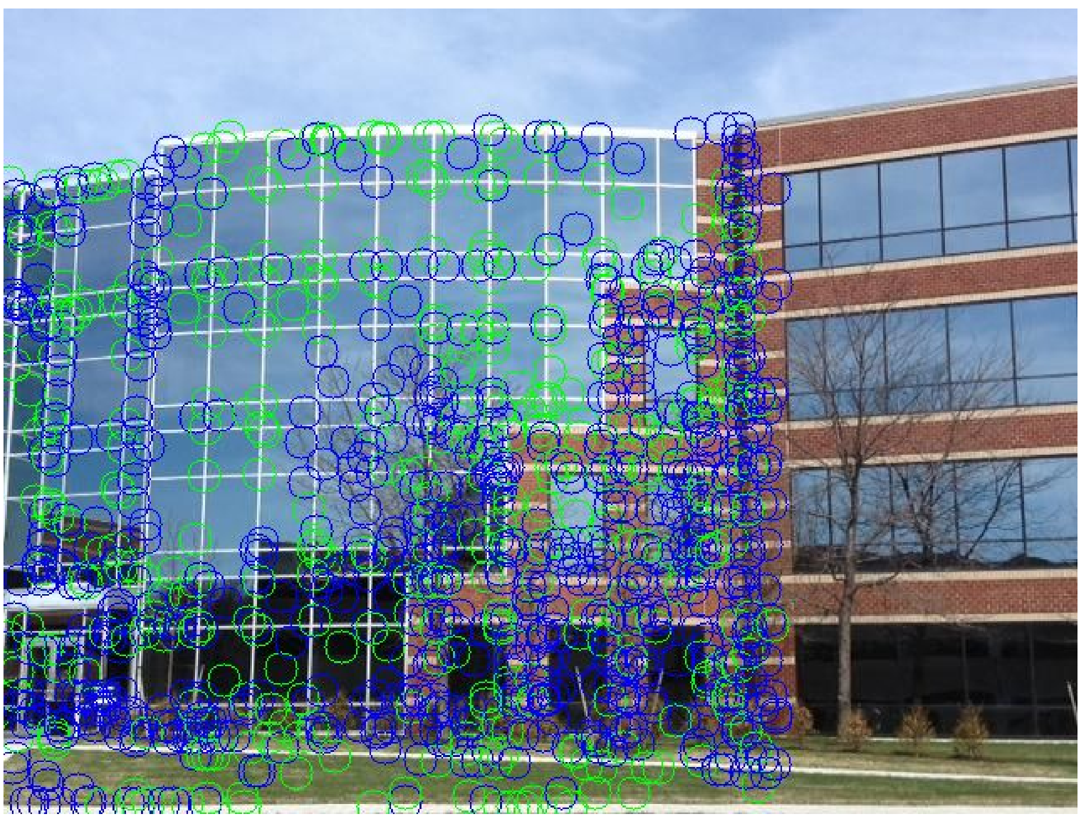


IMAGE 3(BLUE - INLIERS , GREEN - OUTLIERS)

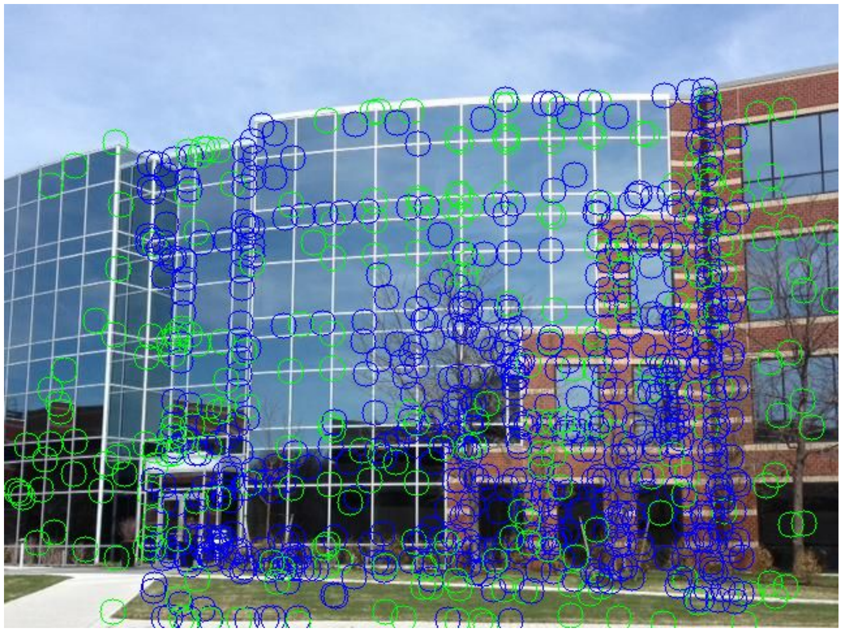


IMAGE 2(BLUE - INLIERS , GREEN - OUTLIERS)

The homography matrices corresponding to the two transformations are as given below :

HOMOGRAPHY MATRIX FOR IMAGE 1,2

-3.75419267e-03	5.89914923e-04	-2.99640778e-02
-3.00533429e-04	-3.12338906e-03	-9.99530837e-01
-5.79864575e-08	1.42168158e-06	-3.99657955e-03

HOMOGRAPHY MATRIX FOR IMAGES 3,2

9.17396617e-03	4.94698110e-04	-1.97818729e-01
-1.48081830e-04	9.52834711e-03	-9.80110908e-01
1.56318280e-08	1.36630445e-06	8.66839426e-03

The final stitched image is as shown below :