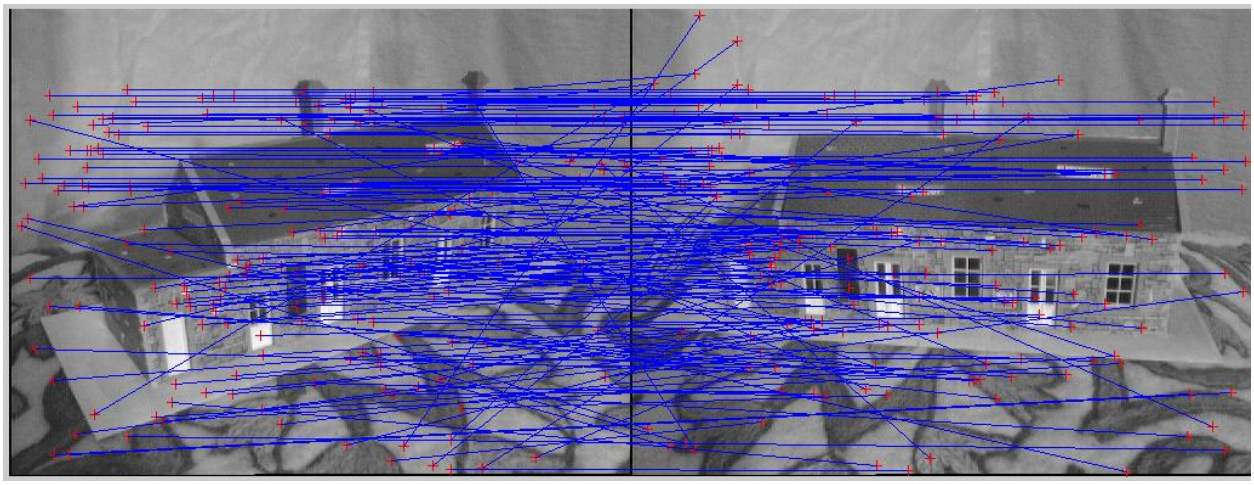
Computer Vision
Exercise 8

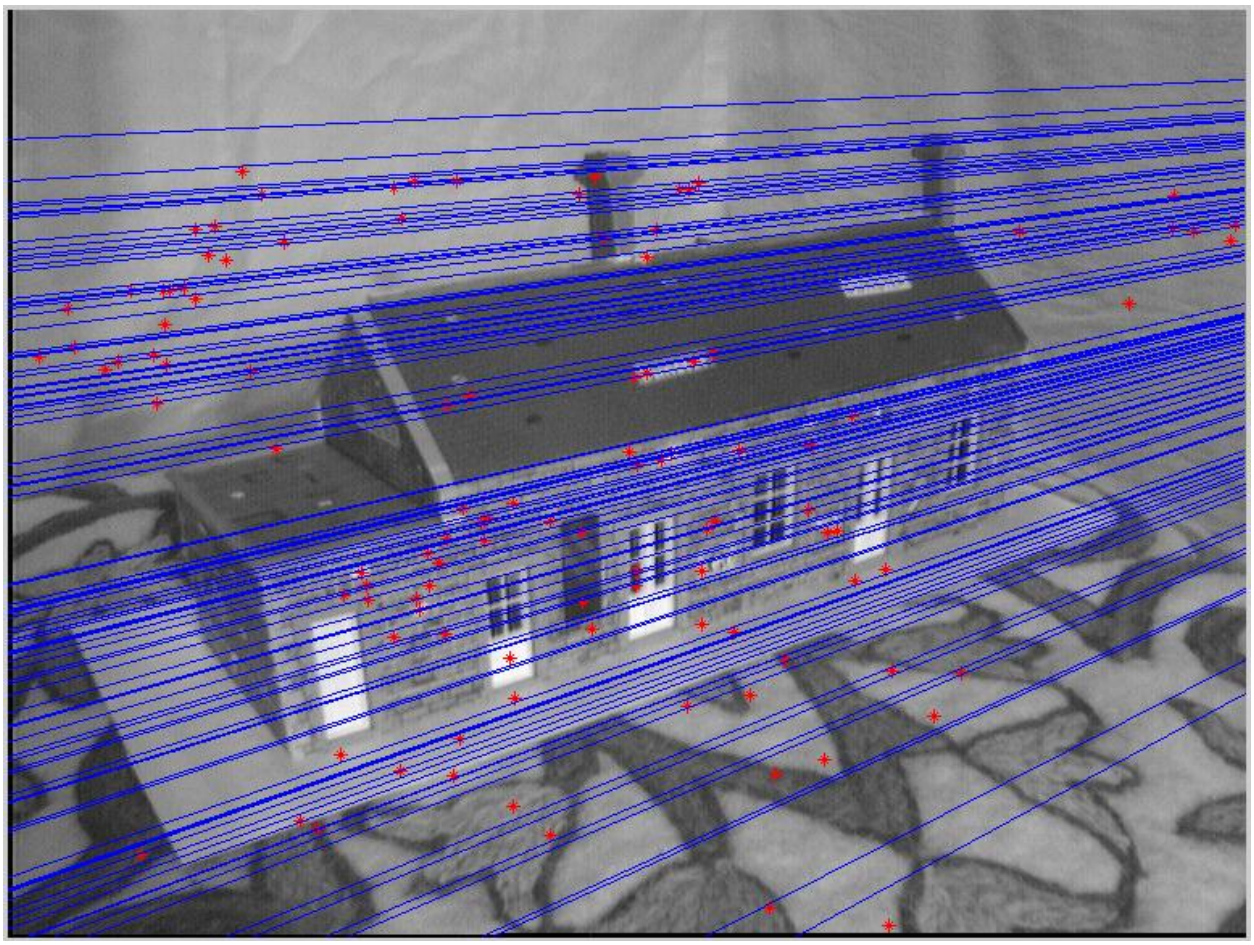
Amrollah Seifoddini

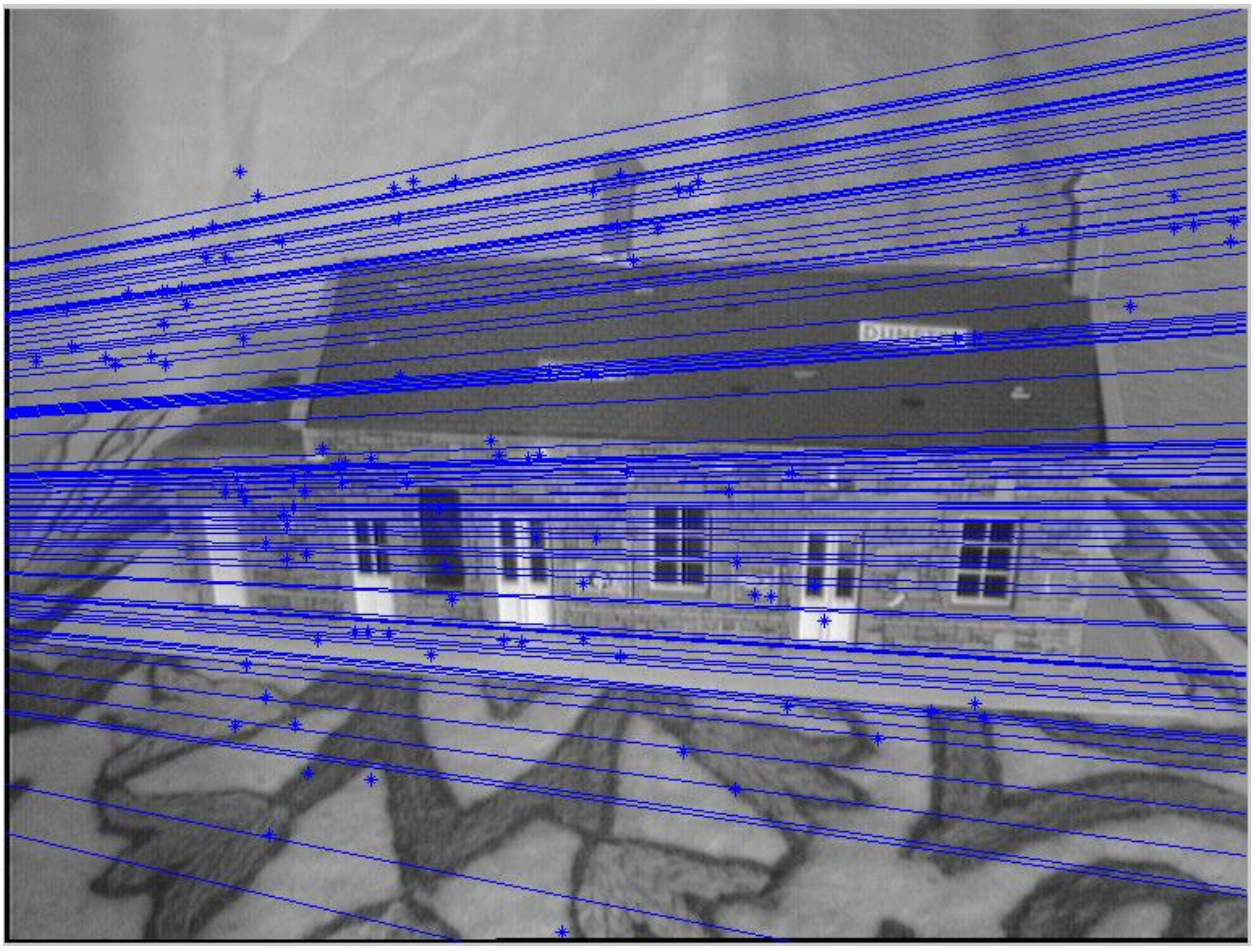
Task 1)

For this part, I use images 0 and 4 for initialization. Firstly, I compute SIFT features with VLFeat library and find matches in two images. Then, I use ransacfitfundmatrix function to estimate Fundamental matrix with RANSAC. The SIFT matches are shown below:



From the inliers of RANSAC, I draw epipolar lines in two images:





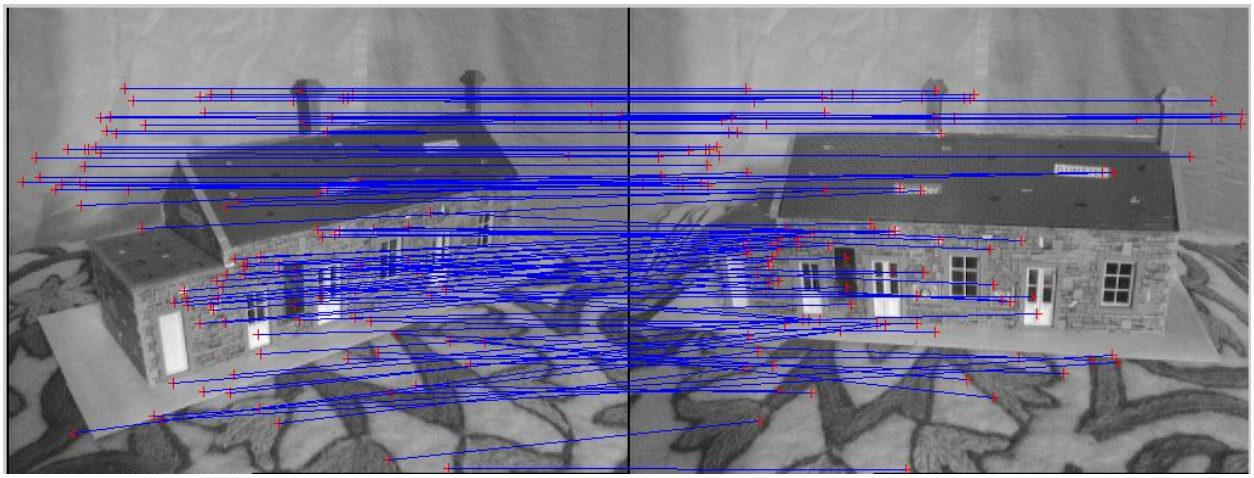
Afterwards, I compute essential matrix with K and F. Then, I calibrate the 2d points with K matrix and by using decompose function, the Projection matrix for these two images is extracted. Then I use linearTriangulation function to triangulate 2d point correspondence and get the 3d points.

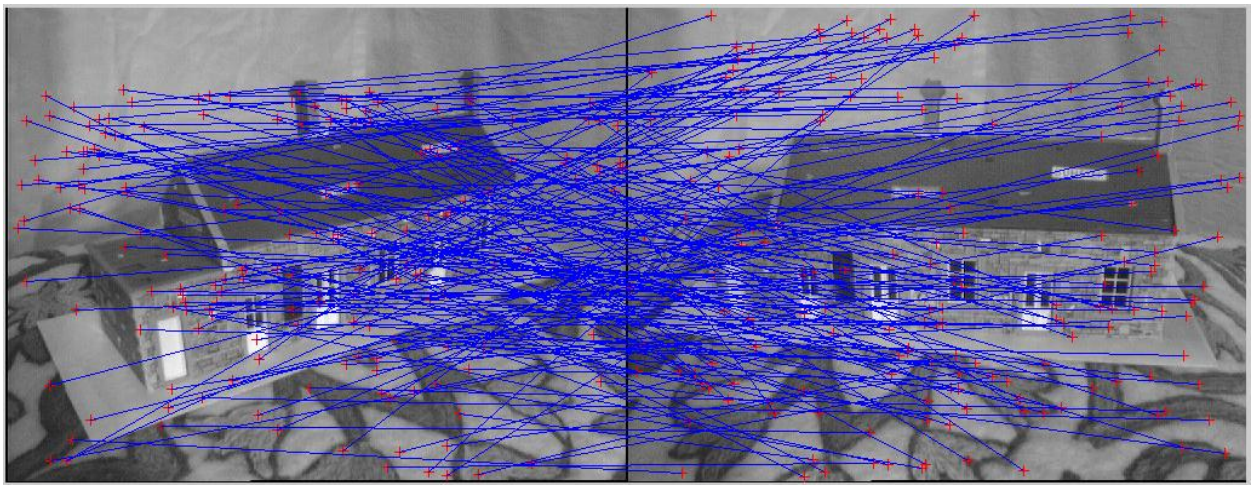
Task 2)

In this task, I load next image, find SIFT feature for it, match them against feature of first image. Then, I calibrate 3d points with K, and use ransacfitprojmatrix with calibrated points and 3d point correspondence from first task to extract projection matrix for this camera view. I realized that sometimes, projection matrixes are reflective (like the points are in a mirror). So, after some searching, I found that if determinant of projection matrix is minus, we should change the sign of its elements to get the right projection matrix. So, I did that for any new view I am adding. Then, I use linearTriangulation with 2d inlier points from image 0 and the current image to compute the 3d corresponding points.

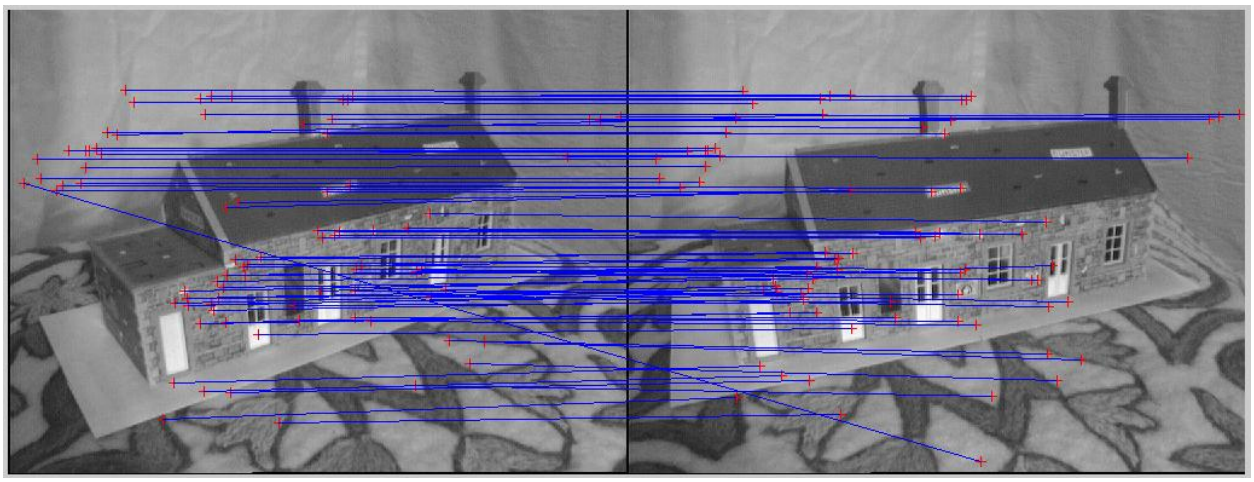
I repeated the same approach for other two images and computed the 3d points.

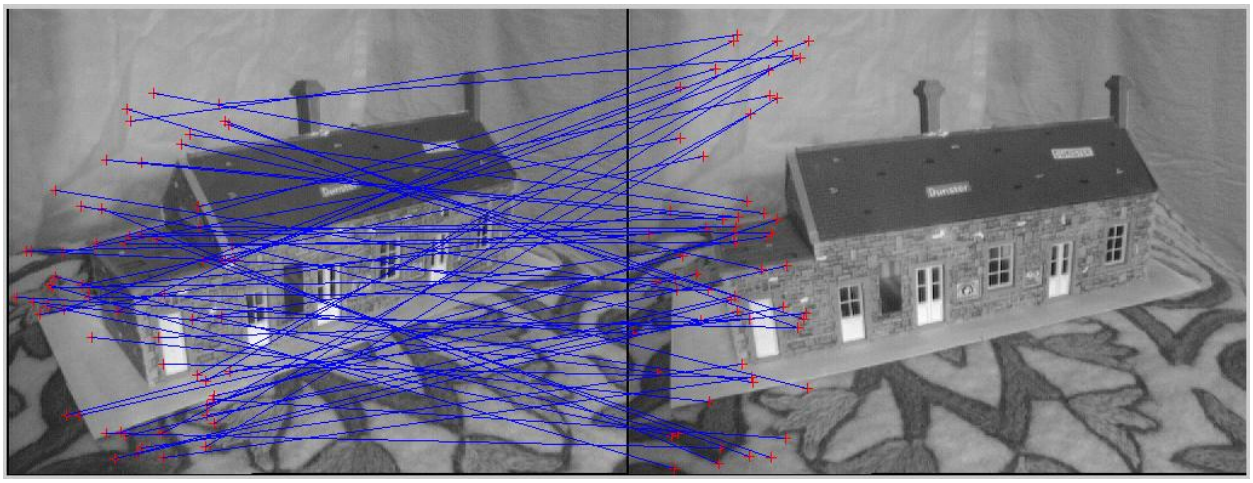
The inliers and outliers for initialization images are shown below:



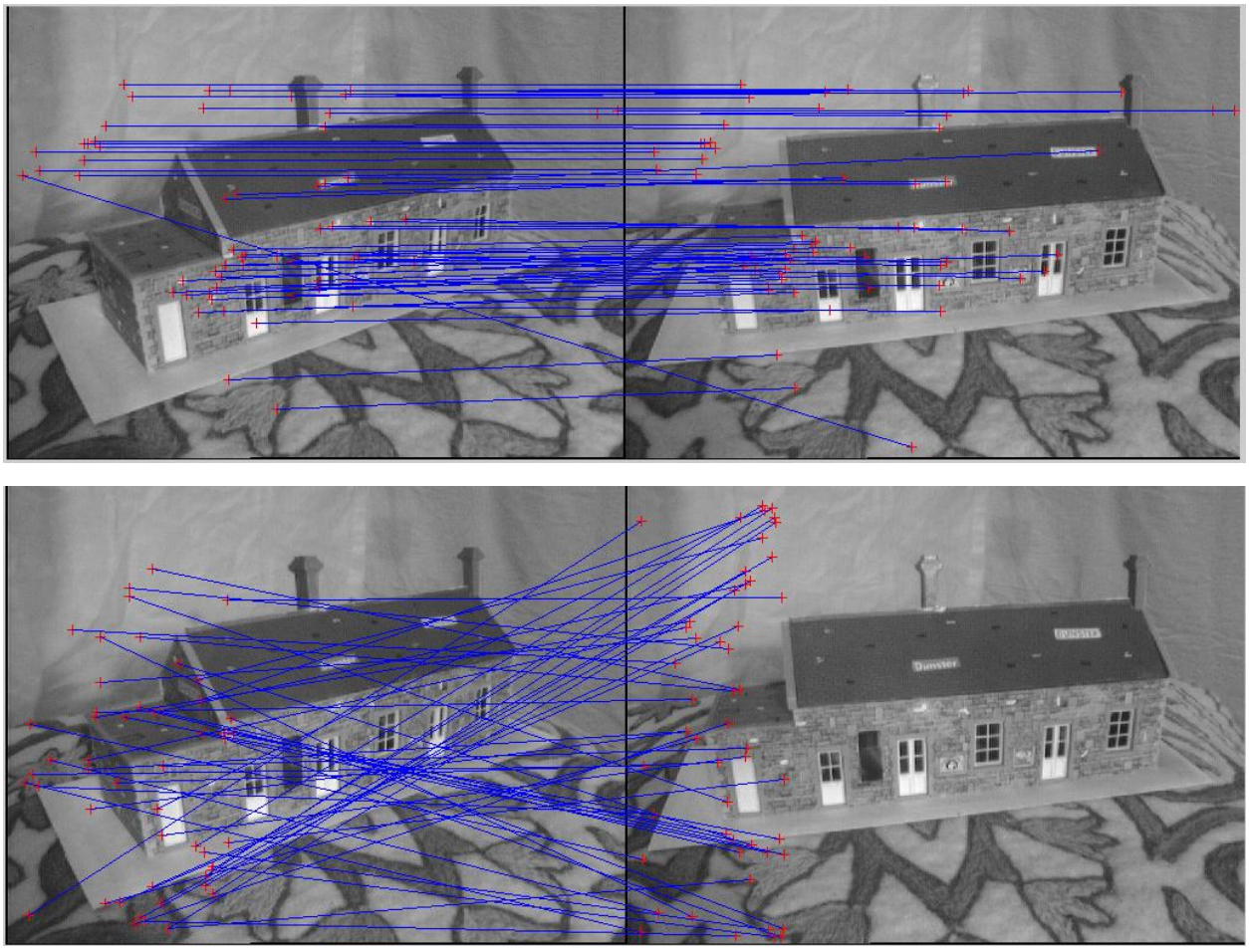


The inliers and outliers for first additional view are shown below:

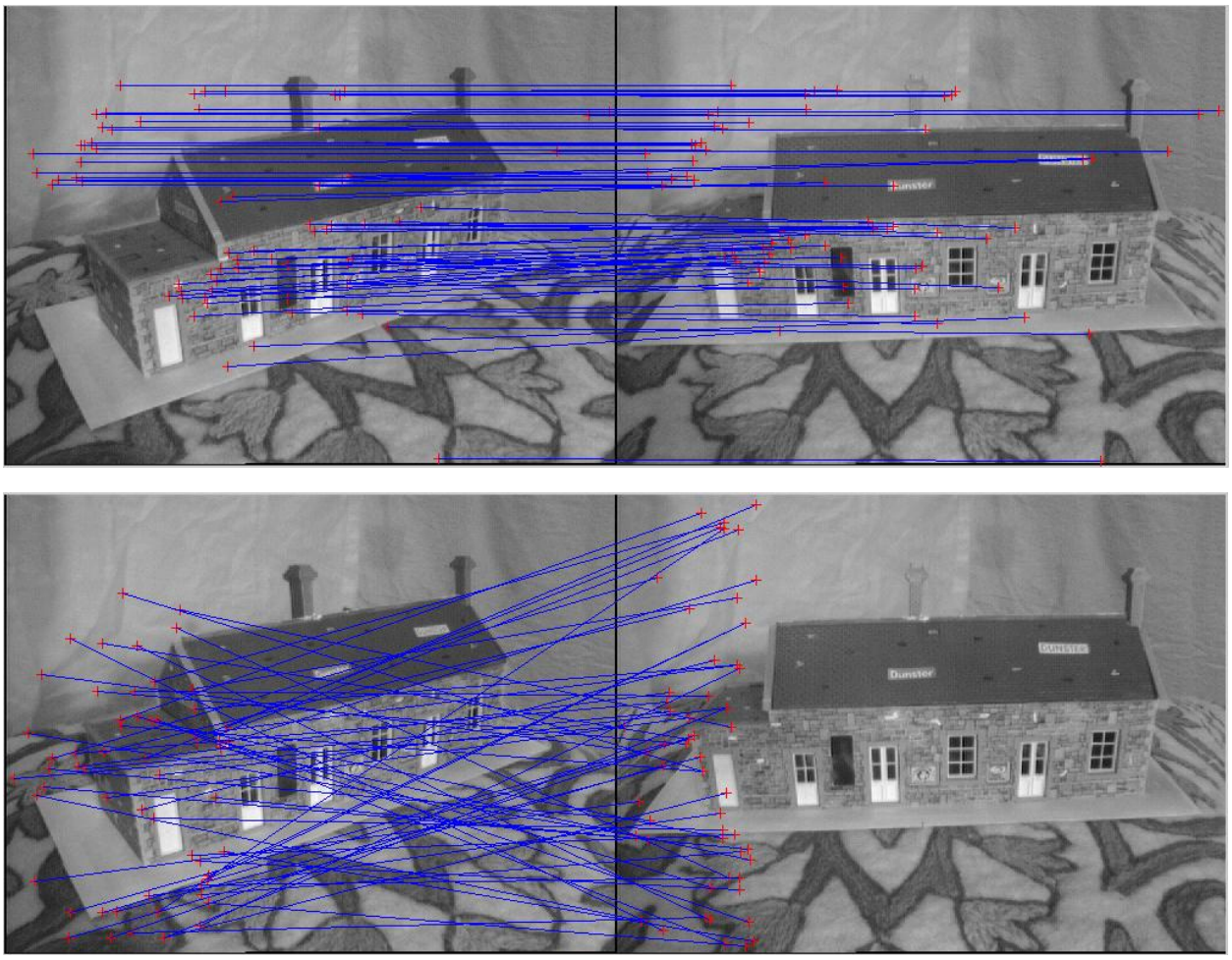




The inliers and outliers for second additional view are shown below:



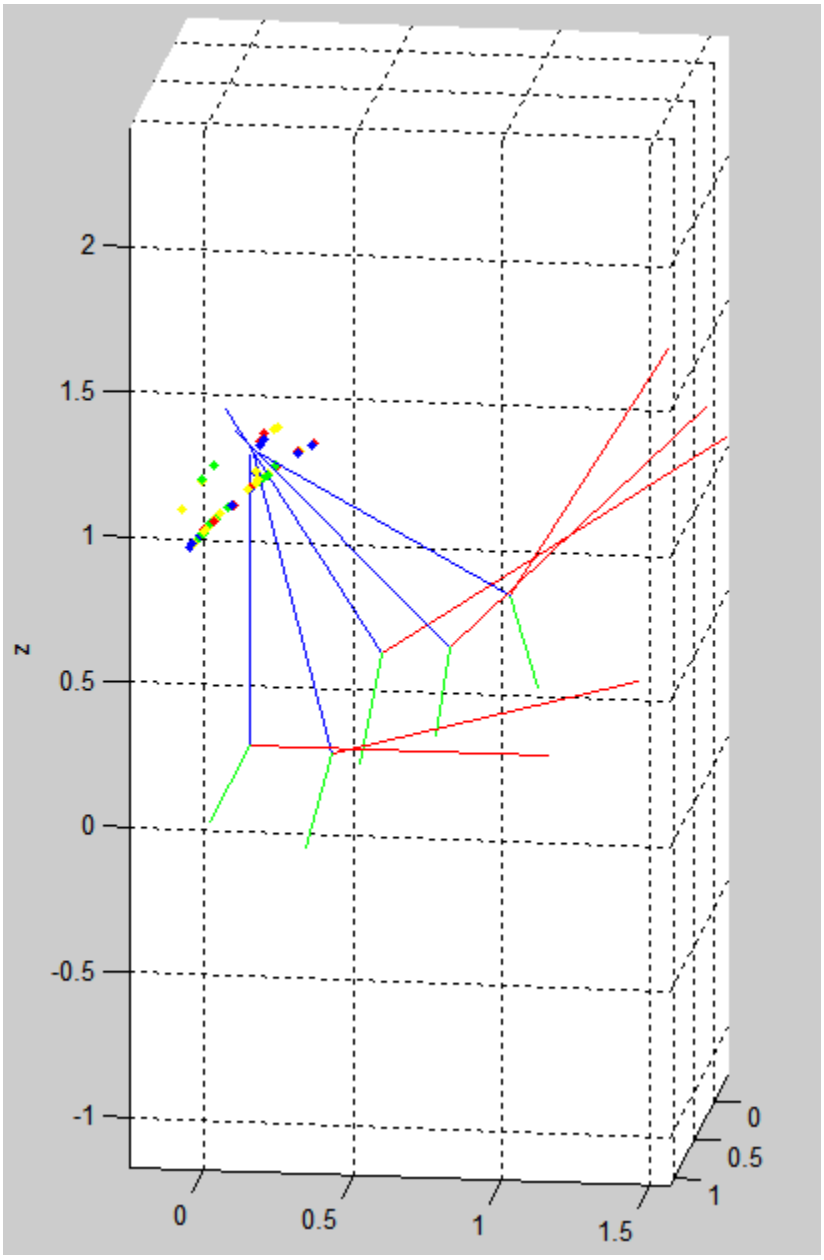
The inliers and outliers for third additional view are shown below:



Task 3)

In this part, I used the provided code to plot the 3d points extracted from each camera view. Then, I added camera poses to the plot.

Here is the result:



Although it is not perfect, I guess it is because of RANSAC threshold that match points in a planar surface rather than scatter over the 3d space.