

## Panorama

### 1. Extraction of interest points using SIFT and mapping the points between the two images

The OpenCV implementation of SIFT was used to first extract the key points from the two images. The correspondence between two key points is established by calculating the Euclidean distance between their descriptor vectors. The Euclidean distance between the Euclidean distance is calculated using the following formula

$$D = \sqrt{\sum_i^{128} (f_i - g_i)^2},$$

Where  $f$  is the descriptor vector from one image and  $g$  is the descriptor vector from another image. The smaller the Euclidean distance between the descriptors the more likely they are a match. Thus, to determine point correspondence I first sort all the correspondences based on the ascending order of Euclidean distance and then I take the first 100 points with the least Euclidean distances.

### 2. RANSAC & Least squares to estimate homography

The least squares method is used a set of correspondences  $n$  to determine the homography between two images. However, if the correspondences contain outliers the least square method will not be able to accurately create the homography. Thus, the RANSAC algorithm is used to remove the outliers from the correspondence that was obtained using the SIFT image matching. The RANSAC algorithm uses the following set of parameters :

1.  $p$  : The probability that one of the  $N$  trials will be free of outliers. I have chosen it to be 0.99
2.  $n$ : The set of correspondences for calculating the homography during each trial. I have chosen it to be 6
3.  $\epsilon$ : Estimate of the percentage of outliers. I have chosen it to be 0.1.
4.  $\delta$ : The decision threshold to construct the inlier set. It is the maximum accepted distance between the correspondence point in one image and its mapped value for it to be accepted as an inlier. I have chosen it to be 20 pixels.
5.  $M$ : The minimum value for the size of the inlier set for it to be acceptable.  
 $M = n_{total} (1 - \epsilon) = 90$ ,  $n_{total}$  is the total number of correspondences obtained from SIFT.
6.  $N$ : Number of trials.  $N = (\ln(1 - p)) / (\ln(1 - (1 - \epsilon)^n)) = 6$

During each trial of the of the RANSAC randomly choose  $n$  correspondences between two images and calculate the homography between the correspondences. The homography can be calculated using the method specified below

$$\begin{bmatrix} x'_1 \\ y'_1 \\ \vdots \\ x'_n \\ y'_n \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1y'_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1y'_1 \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix}$$

Here  $h_{33}$  is assumed to be 1.

Thus the homography matrix  $H = B^{-1}A$ , where

$$B = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1y'_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1y'_1 \end{bmatrix}, A = \begin{bmatrix} x'_1 \\ y'_1 \\ \vdots \\ x'_n \\ y'_n \end{bmatrix}$$

Once the homography is calculated apply the homography to the correspondences in the in one image. Determine the distance between the mapped correspondences to the correspondences in the other image. If the distance is lesser than the threshold,  $\delta$  it is considered as an inlier set. The total number of inliers should be greater than M. Repeat this process N times and accept the homography that gives you the largest inlier set.

### 3. Homography refining using the Levenberg Marquardt (LM) Method

The LM algorithm is used to refining the homography by combining both the gradient descent and gauss newton methods. In the LM algorithm the step size in each iteration is given by the formula

$$\vec{\delta}_p = \left( J_{\vec{f}}^T J_{\vec{f}} + \mu I \right)^{-1} J_{\vec{f}}^T \vec{\epsilon}(\vec{p}_k)$$

$\vec{\epsilon}(\vec{p}_k) = \|X'_{actual} - F(P_k)\|$  is the error function, where  $P(k)$  is the homography estimation

$[h_{11} h_{12} h_{13} h_{21} h_{22} h_{23} h_{31} h_{32} h_{33}]$  and  $F(P_k)$  is a vector of  $i$  points  $\begin{bmatrix} x'_i \\ y'_i \end{bmatrix}$

$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + 1}$  and  $y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + 1}$ .  $J_{\vec{f}}$  is the Jacobian matrix of  $\vec{\epsilon}(\vec{p}_k)$  given by  $\frac{\delta F(P_k)}{\delta P_k}$ .  $\mu$ , is the damping coefficient.

Thus, the new step is given by,  $\vec{p}_{k+1} = \vec{p}_k + \vec{\delta}_p$

The LM algorithm switches between the GD and GN based on the value of  $\mu$ . When  $\mu = 0$ , the solution follows the GN method while when the value of  $\mu$  is much larger than the diagonal elements of  $J_{\vec{f}}^T J_{\vec{f}}$  the solution follows GD.

The main challenge of the LM is to determine the value of the damping coefficient for each iteration. The initial value of  $\mu_0 = \Gamma \times \max(\text{diag}(J_{\vec{f}}^T J_{\vec{f}}))$ . After one iteration is completed a quality test is done if done to determine where to accept the change or not. The quality test is done on  $\vec{\delta}_p$  using the ratio of the actual changes in the cost function to the changes in the cost function predicted by a choice for  $\mu_k$ . The ratio is given by

$$\rho_{k+1}^{LM} = \frac{C(\overrightarrow{p_{k+1}}) - C(\overrightarrow{p_k})}{\vec{\delta}_p \vec{\epsilon}(\overrightarrow{p_k}) + \vec{\delta}_p \mu_k \mathbf{I} \vec{\delta}_p}$$

where,  $C(\overrightarrow{p_{k+1}}) - C(\overrightarrow{p_k}) = \overrightarrow{\epsilon^t}(\overrightarrow{p_{k+1}}) \vec{\epsilon}(\overrightarrow{p_{k+1}}) - \overrightarrow{\epsilon^t}(\overrightarrow{p_k}) \vec{\epsilon}(\overrightarrow{p_k})$ . The value of  $\rho_{k+1}^{LM}$  is used to calculate the damping coefficient for the next iteration using the following formula.

$$\mu_{k+1} = \mu_k \times \max\left(\frac{1}{3}, 1 - (2 \times \rho_{k+1}^{LM} - 1)\right)$$

When  $\rho_{k+1}^{LM}$  is negative,  $2 \times \rho_{k+1}^{LM} - 1$  is negative and its cubed power remains negative with

its magnitude greater than one. Thus  $\mu_{k+1} \geq 2 \times \mu_k$  which means that the solution moves towards the direction of GD. Instead if  $\rho_{k+1}^{LM}$  is positive like 2, then  $\mu_{k+1} = \mu_k/3$ , which means that the solution moves towards GN.

In the following experiment I have implemented the LM algorithm to optimize the homography matrix using the `scipy.optimize.root` library. In order to implement the library I had to create a function that would calculate the error as well as the Jacobian matrix which was then used as an input to the library. `Scipy.optimize.root` then outputs the optimized homography matrix

#### 4. Image Mosaicing

Calculate the homography matrices between the five adjacent images using the RANSAC method. Thus, the homography between image 1 and image 2 is H12, between image 2 and image 3 is H23, between image 3 and image 4 is H34 and between image 4 and image 5 is H45. To combine the 5 images, we need to first calculate the homography matrices of the image 1, image 4 and image 5 with respect to image 3.

$$H_{13} = H_{23} \times H_{12}$$

$$H_{43} = H_{34}^{-1}$$

$$H_{53} = H_{34}^{-1} \times H_{45}^{-1}$$

Once the homography matrices are calculated we need to translate them to by twice the width of the input images to ensure that the third image is in the center of the page. This is done by multiplying the matrices by  $H_{tran}$ .

$$H_{tran} = \begin{bmatrix} 1 & 0 & 2 \times \text{width} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Next create a blank image and apply the inverse of the homography matrices to create the panorama.

## 5. Results

Using SIFT we were able to successfully able to extract the correspondences between the adjacent images. Next using RANSAC, least squares and the LM algorithm we were able to successfully calculate the adjacent homography matrices between images. We have created panorama using the homography matrices that were not refined using the LM algorithm as well as homography matrices that were refined using the LM algorithm. The panorama created using the homography matrices refined by LM are seen to be slightly better than the ones created with out using the LM algorithm. The following are the parameters that were used in this experiment.

	$p$	$n$	$\epsilon$	$\delta$	$M$	$N$
Value	0.9	6	0.1	20	180	6

Input Images



Figure 1: Input image 1



Figure 2: Input image 2





Figure 3: Input image 3



Figure 4: Input image 4



Figure 5: Input image 5



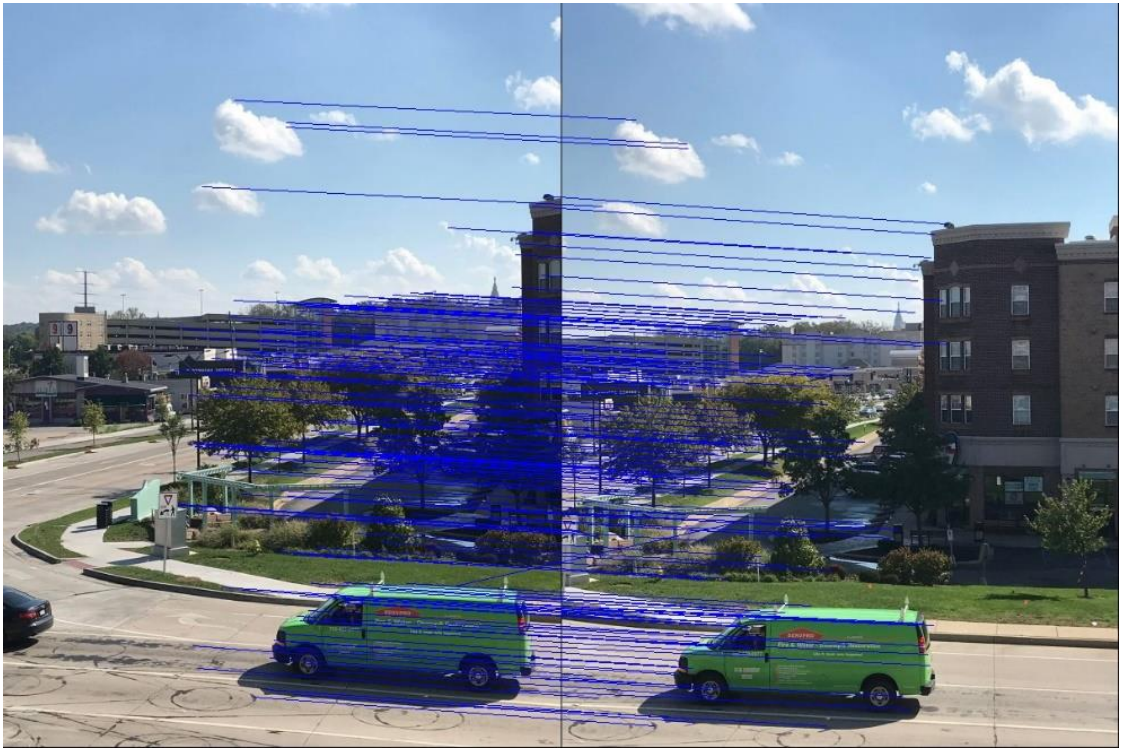


Figure 6: SIFT correspondences between images 1 and 2

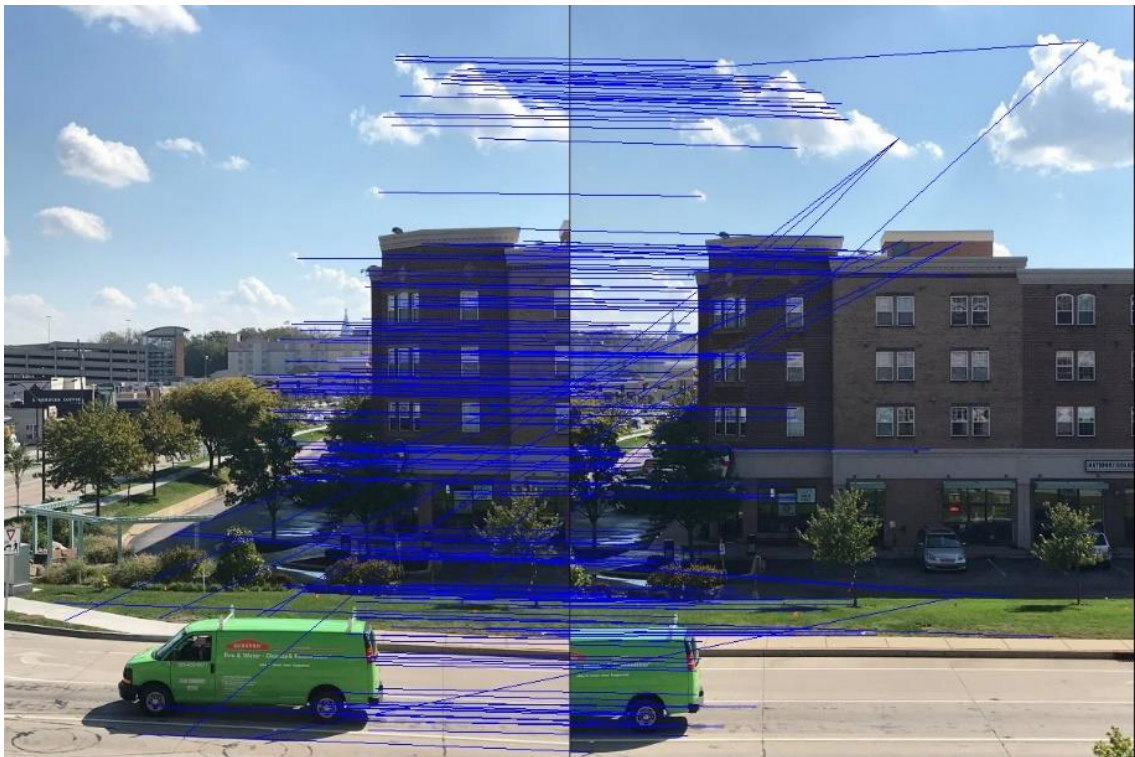


Figure 7: SIFT correspondences between images 2 and 3





Figure 8: SIFT correspondences between images 3 and 4



Figure 9: SIFT correspondences between images 4 and 5





Figure 10: Inliers(green) and outliers(red) between images 1 and 2

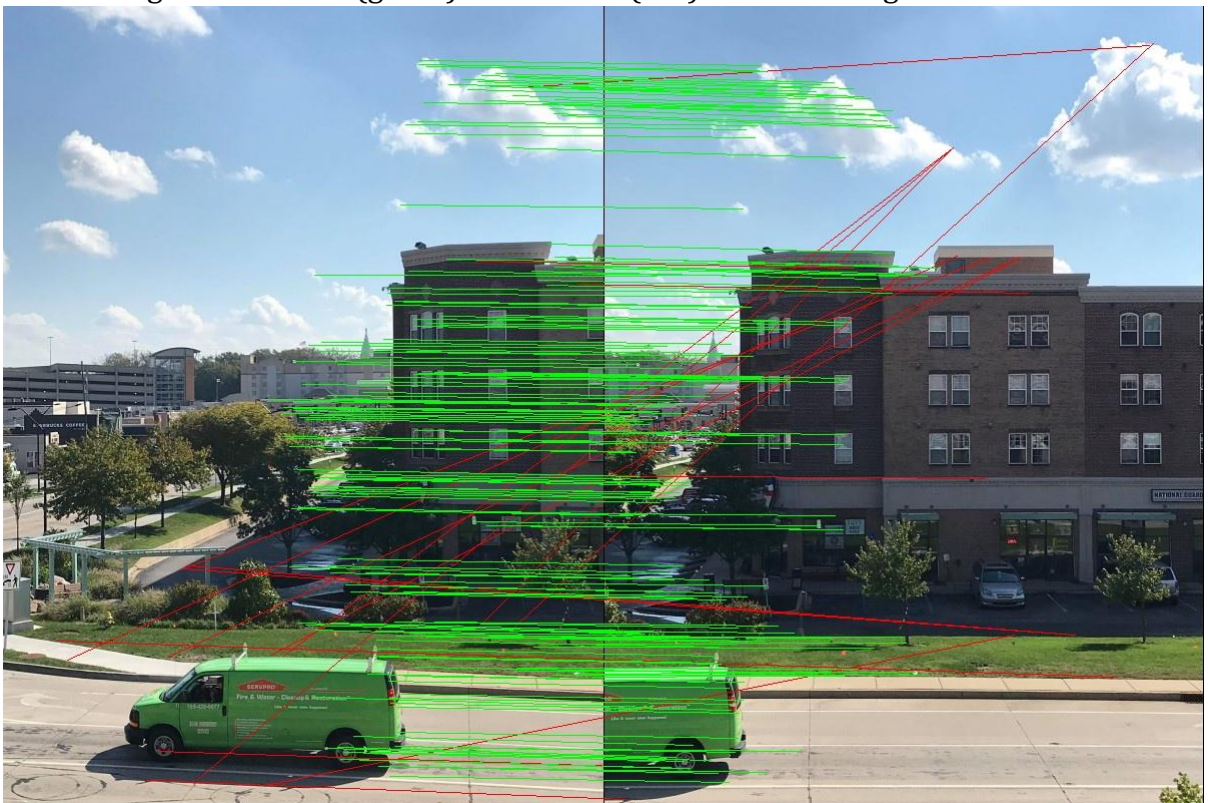


Figure 11: Inliers(green) and outliers(red) between images 2 and 3





Figure 12: Inliers(green) and outliers(red) between images 3 and 4

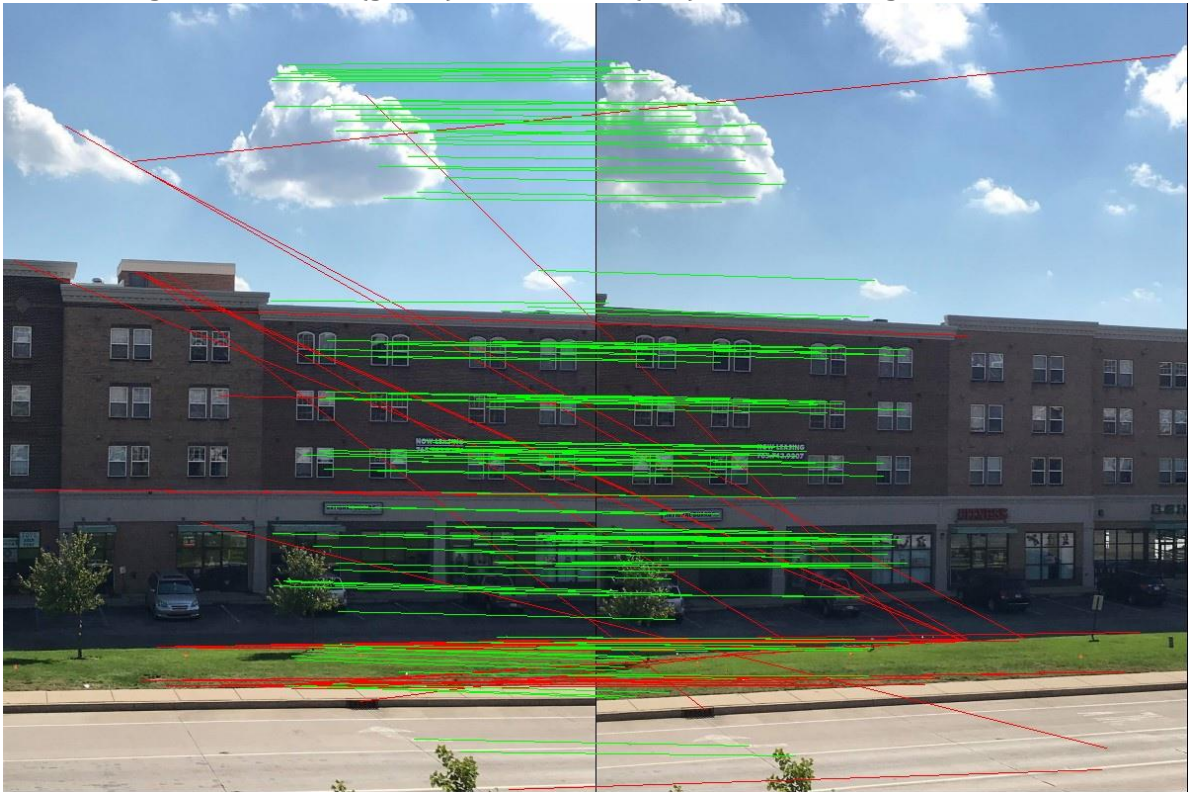


Figure 13: Inliers(green) and outliers(red) between images 4 and 5



Figure 14: Panorama created without using the LM algorithm



Figure 15: Panorama created using the LM algorithm