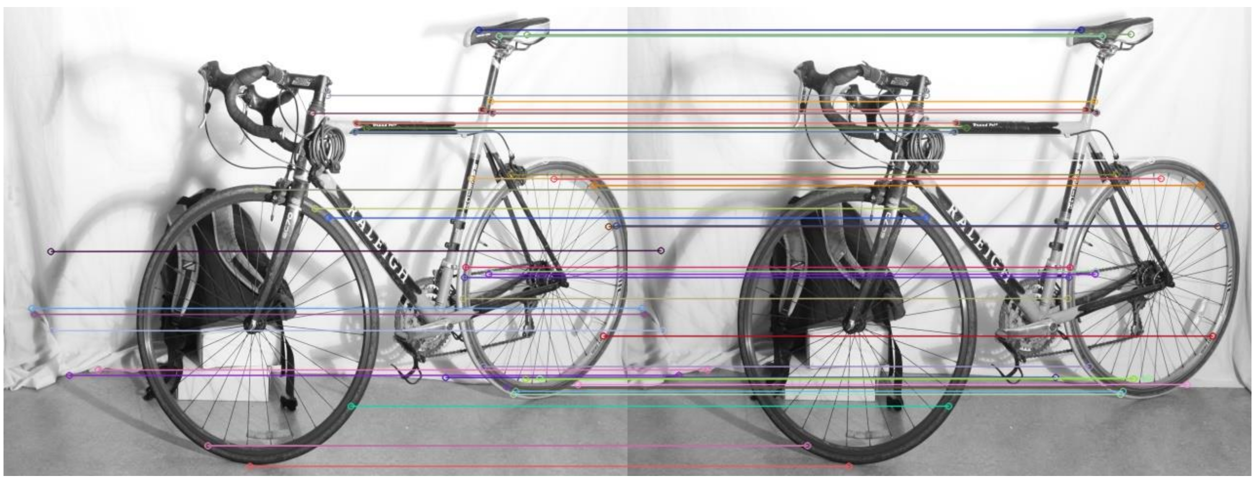
ENPM673
Project 2 Report

- ## **Calibration**:

- SIFT : This is a crucial starting step in the entire pipeline where Scale Invariant Feature Transform is used to detect and describe the features of the give images. SIFT provides the matching feature points good enough to be processed ahead.



Feature matching using SIFT

- Fundamental Matrix: Getting the fundamental matrix is the first step towards obtaining the camera center. It is a 3 X 3 matrix which contains intrinsic and extrinsic parameters of the camera and relates images from different views.

$$\mathbf{A}\,\mathbf{f} = \begin{bmatrix} x'_1 x_1 & x'_1 y_1 & x'_1 & y'_1 x_1 & y'_1 y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_n x_n & x'_n y_n & x'_n & y'_n x_n & y'_n y_n & y'_n & x_n & y_n & 1 \end{bmatrix} \mathbf{f} = 0$$

The equation $\mathbf{x'^T}\,\mathbf{F}\,\mathbf{x} = 0$ is the epipolar constraint equation which is a homogeneous linear system with 9 unknowns. Here each point only gives one equation as the equation is a scalar equation. This system can be solved by least squares method using SVD. But the Fundamental matrix needs to be reduced to a rank of 2 so that it does not have an empty null space.

Since the data from SIFT does have noisy feature points, RANSAC algorithm is used to reject the outliers. The pints are normalized by shifting the origin to the mean and placing them at a distance of Sqrt(2) from the new origin. RANSAC gives out the most number or inliers for the best fundamental matrix possible which is used afterwards.

- Essential matrix is the matrix which contains the extrinsic parameters rotation and translation of the camera. The essential matrix is simply calculated by the formula **E = k' F k** and it also has been constrained to rank 2.

- Rotation and Translation are estimated after decomposing E using SVD which gives out 4 possible solutions out of which one solution is picked after using the triangulation function. The matrix with the most possible numbers corresponds to positive depth.

➢ **Rectification**:

- Rectification is basically transforming images to make them parallel. This is required to use the epipolar constraint at an advantage, so as not to search for a pixel in the entire image.

- First, the image is translated to a center 0,0,1 which transformation is also applied to the epipole.

-After this the transformation matrices T1, T2 and T3 are obtained for the purposes of centering the image, shifting the epipole to infinity to make the images parallel and to translate the images accordingly. Resulting Transformation = H0 = $T1^{-1}$ T3 T2 T1.

- After this H1 transformation is calculated and with the help of H0 and H1 transformations, the images can be rectified.

- **Correspondence:**

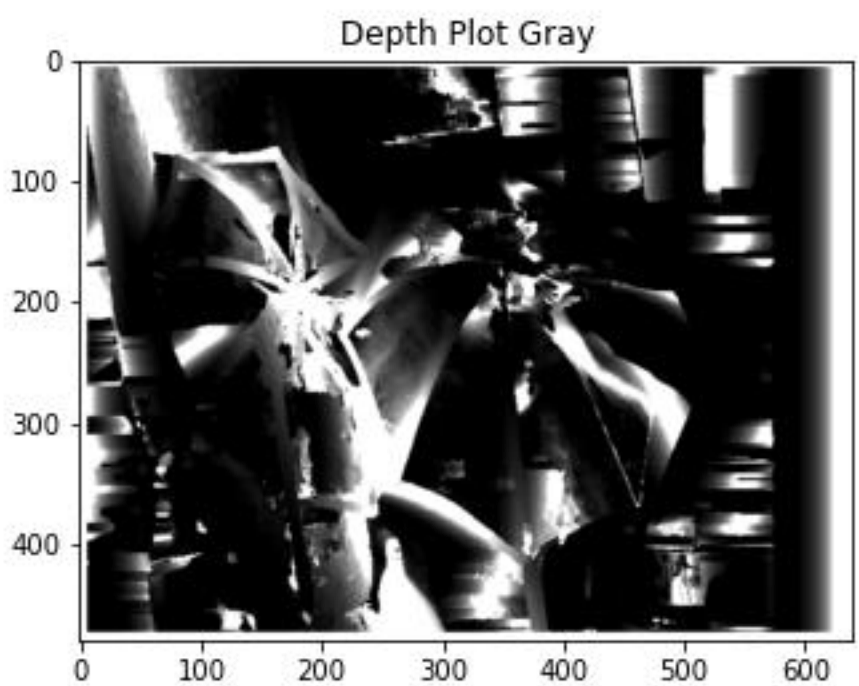
- Using the concept of Sum of Squared Differences and the matching windows concept on each epipolar line, the points on the left image are matched with that of the right image. The disparity is calculated by the difference in image pixels along width.

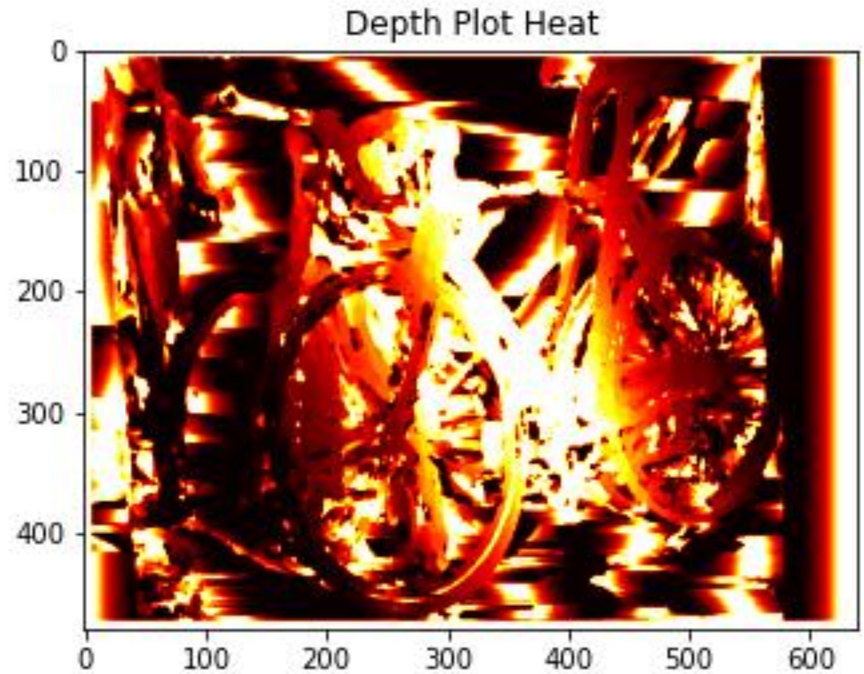
- It is scaled using matplotlib.

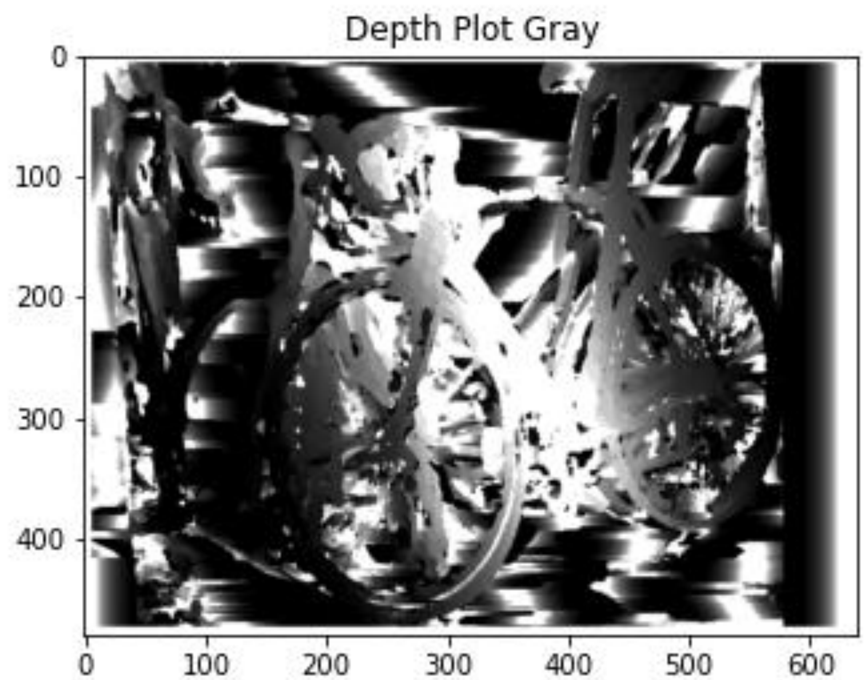
- **Depth:**

- Depth is calculated using the formula: d = baseline * f / disparity

- **Results:**




Depth Plot Heat
0
100
200
300
400
0
100
200
300
400
500
600


Depth Plot Gray
0
100
200
300
400
0
100
200
300
400
500
600