Project – Report

Omkar. A. Chittar

*** The examples given are for the ladder dataset**

Pipeline - Calibration:

1. Import the req libraries - **NumPy**, **cv2, matplotlib.pyplot**.
2. Import the images from the datasets provided based on the choice (1. Artroom, 2. Chessboard, 3. Ladder).
3. Use SIFT to detect features from the two images and then use brute force matcher to match these features.
4. Select only top 500 of the total matches obtained.
5. Draw these matches on our images using the **'drawMatches()'** function.

6. Estimate the fundamental matrix using the features obtained using the epipolar constraint:

$$\mathbf{x}_i'^{\mathbf{T}}\mathbf{F}\mathbf{x}_i = 0 \text{ where } i = 1, 2, \ldots, m.$$

Since, F is a 3×3matrix, we can set up a homogeneous linear system with 9 unknowns:

$$[x_i' \quad y_i' \quad 1] \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = 0$$

$$x_i x_i' f_{11} + x_i y_i' f_{21} + x_i f_{31} + y_i x_i' f_{12} + y_i y_i' f_{22} + y_i f_{32} + x_i' f_{13} + y_i' f_{23} + f_{33} = 0$$

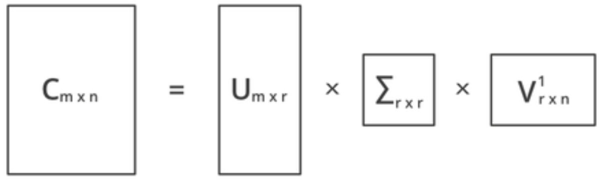
Upon simplifying for 'm' correspondences, we get:

$$\begin{bmatrix} x_1 x_1' & x_1 y_1' & x_1 & y_1 x_1' & y_1 y_1' & y_1 & x_1' & y_1' & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_m x_m' & x_m y_m' & x_m & y_m x_m' & y_m y_m' & y_m & x_m' & y_m' & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{21} \\ f_{31} \\ f_{12} \\ f_{22} \\ f_{32} \\ f_{13} \\ f_{23} \\ f_{33} \end{bmatrix} = 0$$

7. In the estimation of the fundamental matrix, each point only contributes one constraint as the

epipolar constraint is a scalar equation. Thus, we require at least 8 points to solve the above homogenous system.

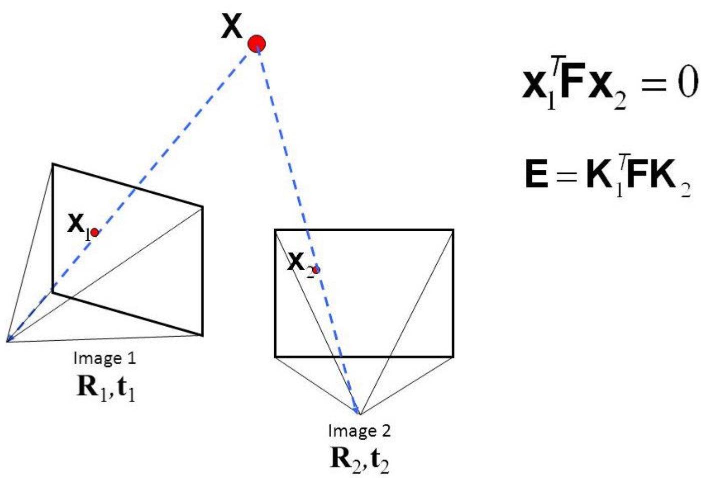
8. Stack the equation in a matrix A and solve the system of equations Ax=0 using SVD to obtain a decomposition **USV^T**.



9. Estimate the fundamental matrix which has inliers with minimum error using RANSAC.
10. Further decompose the fundamental matrix to obtain another set of **USV^T**. Update the fundamental matrix by equating the last element of the **'S'** matrix to 0 and then composing this new S into **USV^T.**

```
Fundamental Matrix:
 [[ 4.95307124e-11 -3.74278117e-08  1.12177302e-04]
 [ 3.66108510e-08 -8.09429642e-11 -1.36632030e-03]
 [-1.11581550e-04  1.36621053e-03  2.74095969e-03]]
```

11. Estimate the essential matrix by using the Fundamental matrix.



12. Decomposing this estimate using SVD. Replacing the S matrix obtained with [1, 1, 0] and plug it back into the **USV^T** composition to obtain the Essential matrix.

```
Essential Matrix:
 [[ 6.46960354e-05 -4.78488438e-02  5.62459826e-02]
 [ 4.67266709e-02  4.49965085e-04 -9.97321607e-01]
 [-5.63259494e-02  9.97266278e-01  5.09088468e-04]]
```

13. Using this Essential matrix to get the four possible combinations of the values of R and C.

$$C_1 = U(:,3) \text{ and } R_1 = UWV^T$$
$$C_2 = -U(:,3) \text{ and } R_2 = UWV^T$$
$$C_3 = U(:,3) \text{ and } R_3 = UW^TV^T$$
$$C_4 = -U(:,3) \text{ and } R_4 = UW^TV^T$$

where, **W** = [[0, -1, 0], [1, 0, 0], [0, 0, 1]] and **U** and **V^T** are the decomposed components of **E.** (If the determinant of R comes out to be negative, we correct the camera pose by negating the values of R and C).

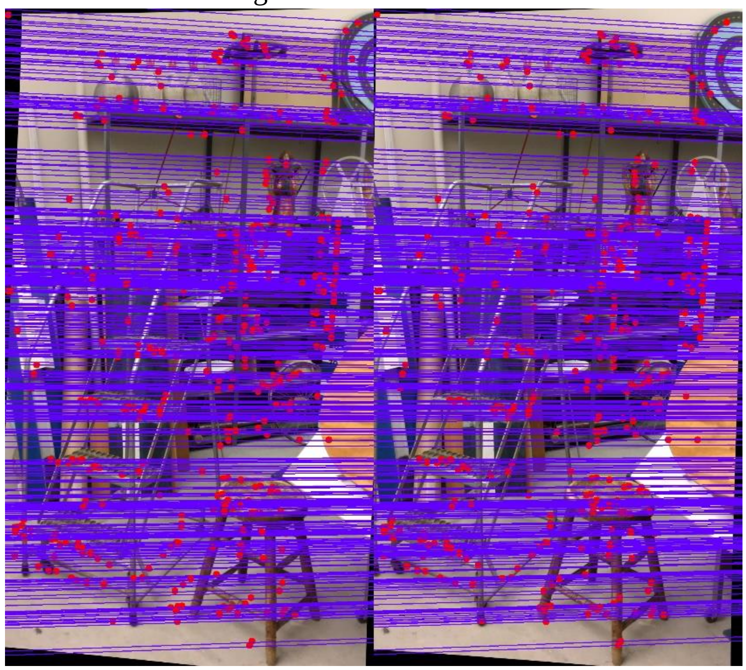
14. Using the Cheirality condition to decide the correct combination of R and C which states that the values corresponding to maximum number of positive values of z are correct.

```
Rotation:
 [[ 9.99999393e-01  5.91693436e-05 -1.09989743e-03]
 [-5.86761523e-05  9.99999898e-01  4.48424354e-04]
 [ 1.09992385e-03 -4.48359544e-04  9.99999295e-01]]
```

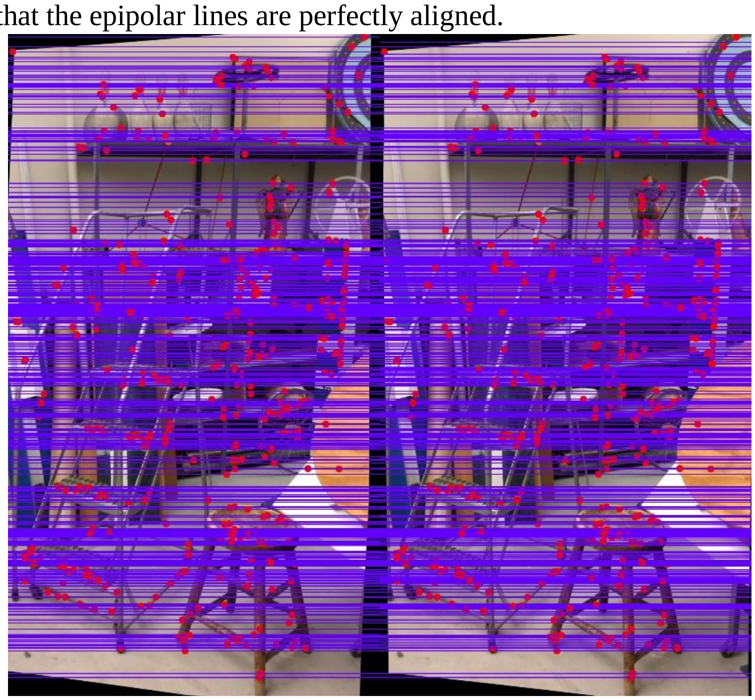
```
Camera Center:
 [0.99726971 0.05626747 0.04782362]
```

Pipeline - Rectification:

1. Draw epipolar lines on both the images.



2. Using the function **'cv2.stereoRectifyUncalibrated'** apply perspective transform on both the images to make sure that the epipolar lines are perfectly aligned.

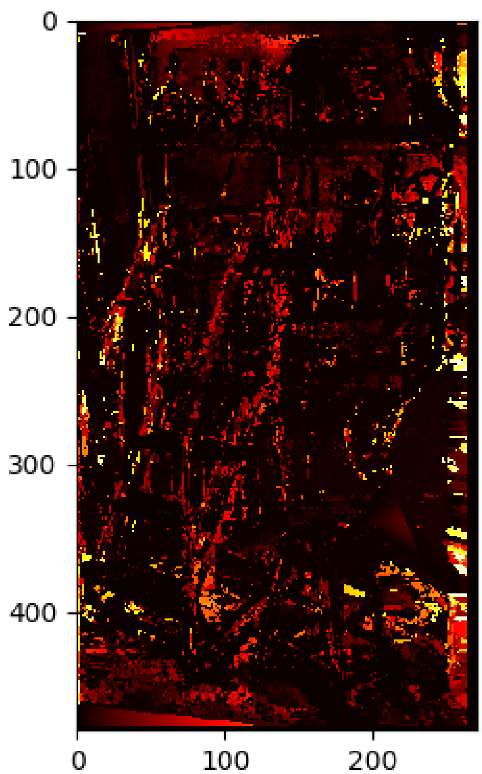


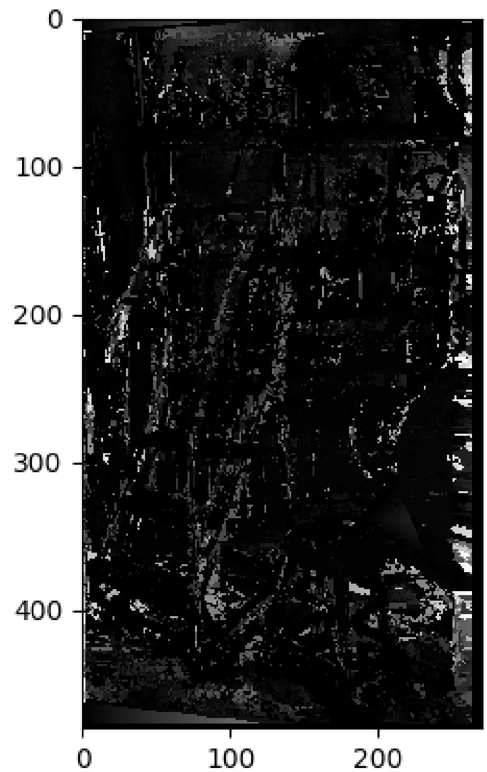
3. The function gives the homographies required to warp these respective images as output.

```
Rectification Homography for img1:
 H1
 [[-1.35715772e-03 -5.45444981e-05  1.08676723e-01]
 [ 1.13369508e-04 -1.36655166e-03 -6.84404598e-02]
 [ 3.71635861e-08  2.07186097e-09 -1.39351080e-03]]
Rectification Homography for img2:
 H2
 [[ 9.83426693e-01  5.57517703e-02 -4.45721139e+01]
 [-8.32141691e-02  9.96888142e-01  4.79230346e+01]
 [-2.77226222e-05 -1.57163241e-06  1.01647898e+00]]
```

Pipeline - Correspondence:

1. Taking 3 as the window size to stride across the image.
2. Stride the window horizontally to find the points of correspondences in image2 w.r.t image1.

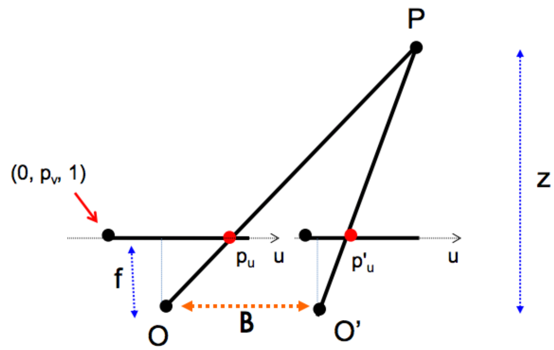




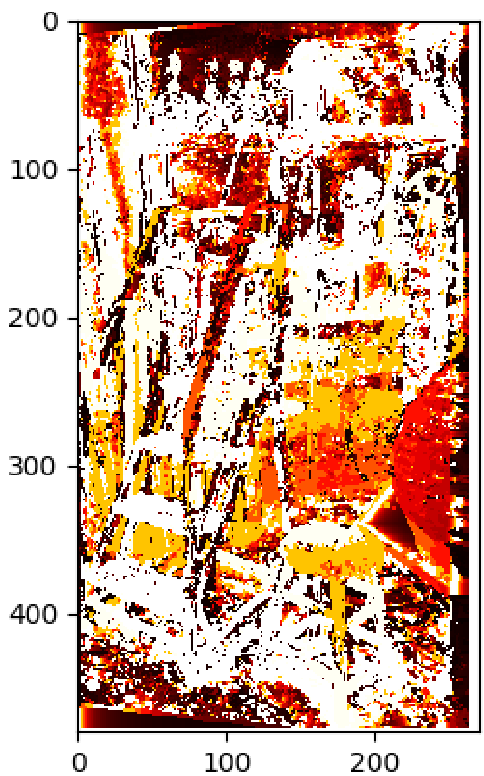
3. These are the disparity maps using grayscale and heatmap.

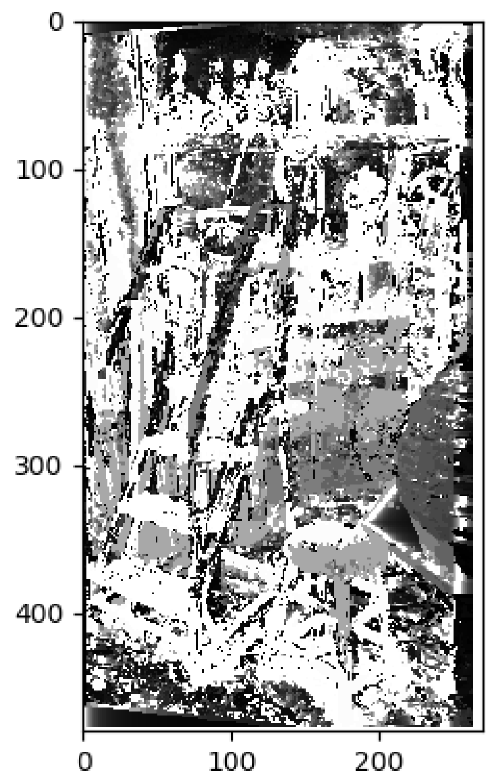
Pipeline – Depth Image:

Depth is inversely proportional to disparity.



$$\text{disparity} = p_u - p'_u \propto \frac{B \cdot f}{z} \quad \text{[Eq. 1]}$$





References:

- **https://cmsc733.github.io/2022/proj/p3/#fundmatrix**