Project 3

ENPM 673

Usnik Singh Chawla

Project Description

In this project, we are going to implement the concept of Stereo Vision. We will be given 3 different datasets, each of them contains 2 images of the same scenario but taken from two different camera angles. By comparing the information about a scene from 2 vantage points, we can obtain the 3D information by examining the relative positions of objects.

Solution:

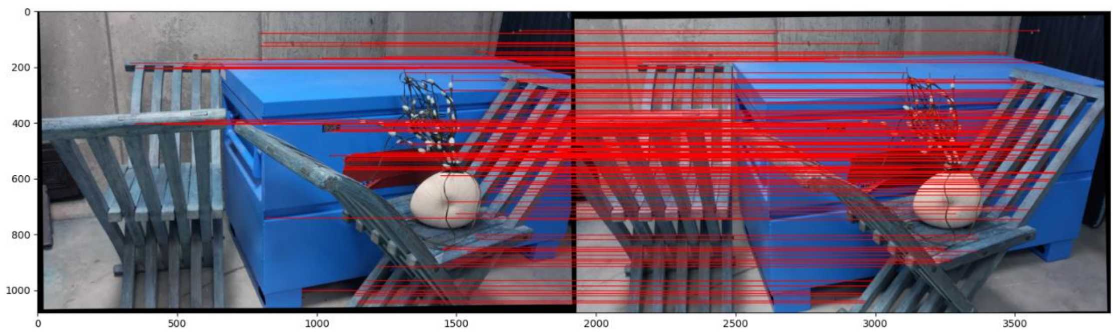
- **Calibration**

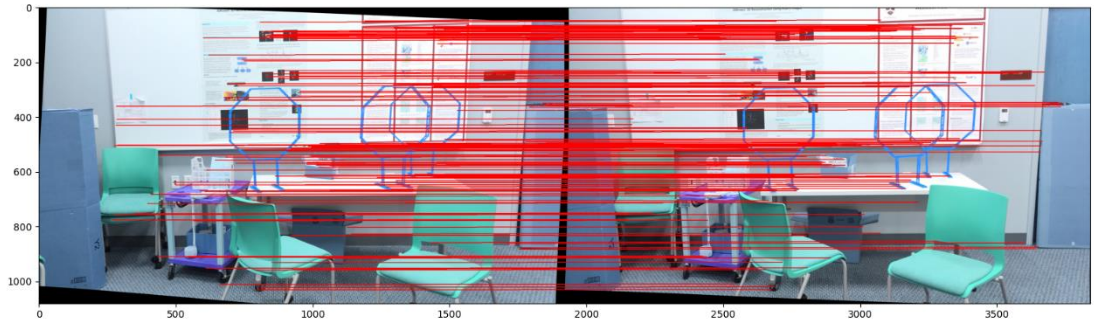
  We load both the images from the folder and convert them into grey scale images. After getting these grey scale images we use sift detector along with Brute force matcher to get matching features.

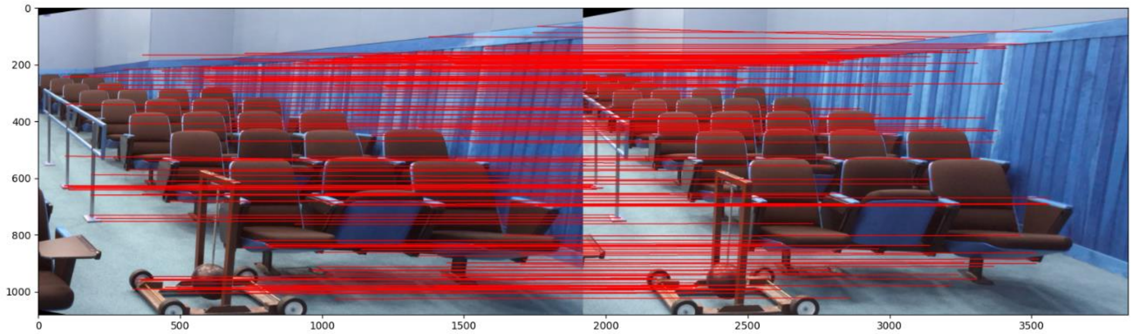
  We use these matching features to get the fundamental matrix. We use RANSAC to get the best fit of the fundamental matrix on the matching features. In this we use the vector constraint equation ($x''^T F x'$, where x'' & x^i are the pixel coordinates for the matching features in right and left image) to separate out the inliers and outliers. Here we also collect the inliers for the best fundamental matrix. These inliers are nothing but a subset of the matching features.

  After Getting the F matrix we derive the essential matrix from it by $K^T F K$ where K is the intrinsic matrix of the camera.

  After getting the E matrix we decompose it into translation and rotation matrix. This decomposition gives four possible solutions of which only one is correct. To get the correct rotation and translations values we do triangulation of the matching features and figure out for which R & T the depth values for the triangulated point is positive.







- **Rectification**

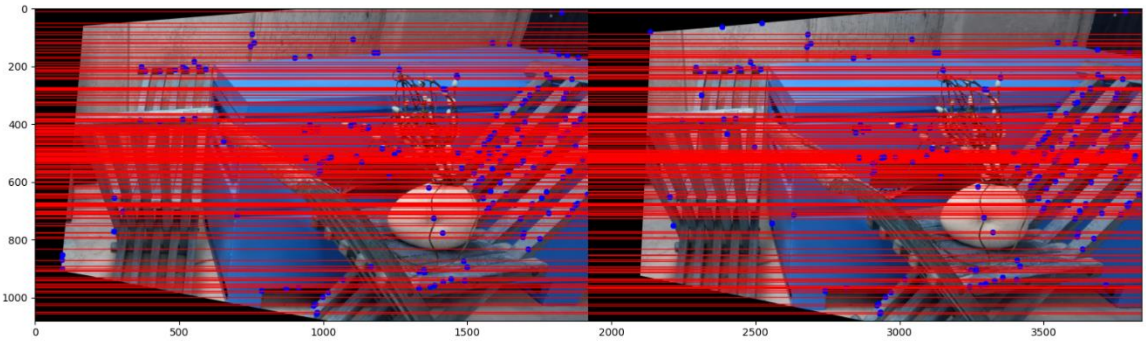
After obtaining the Fundamental matrix and the inliers associated with it, we perform stereo rectification for both the images using cv2.stereoRectifyUncalibarted(). This function gives the corresponding Homography matrices which we use to perform perspective transformation of the images.

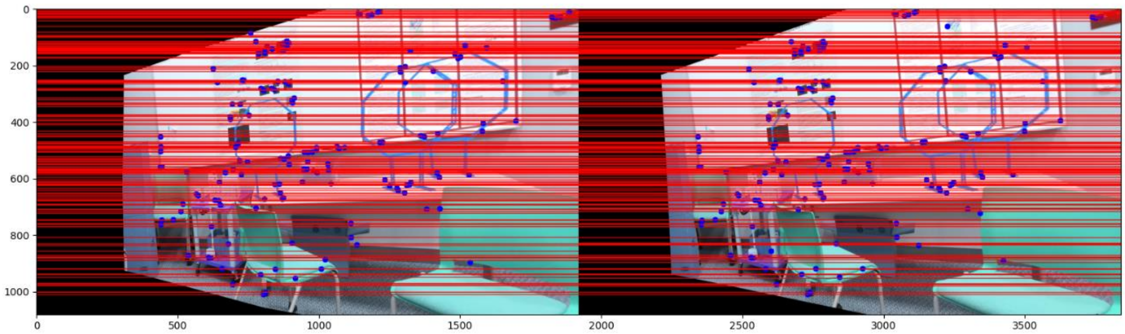
We also perform the transformation of the inliers we had obtained as these will be used to plot the epipolar lines.

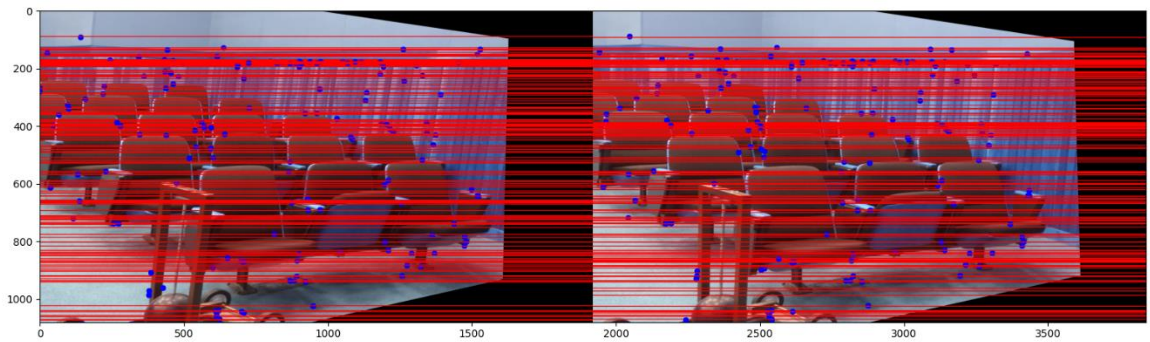
We also calculate the fundamental matrix for the rectified images by $Fnew = H_2^{-T} F H_1$.

Using the new Fundamental matrix and the transformed inliers which are nothing but the matching features in the two rectified images, we obtain the epipolar lines. The epipolar lines for the right image are given by $F_{new}.x'$ and epipolar line for the left image are given by F_{Nnew}^{T}, x''.

As we have rectified the image these epipolar lines will be parallel to each other as the epipole for both the images has gone to infinity.







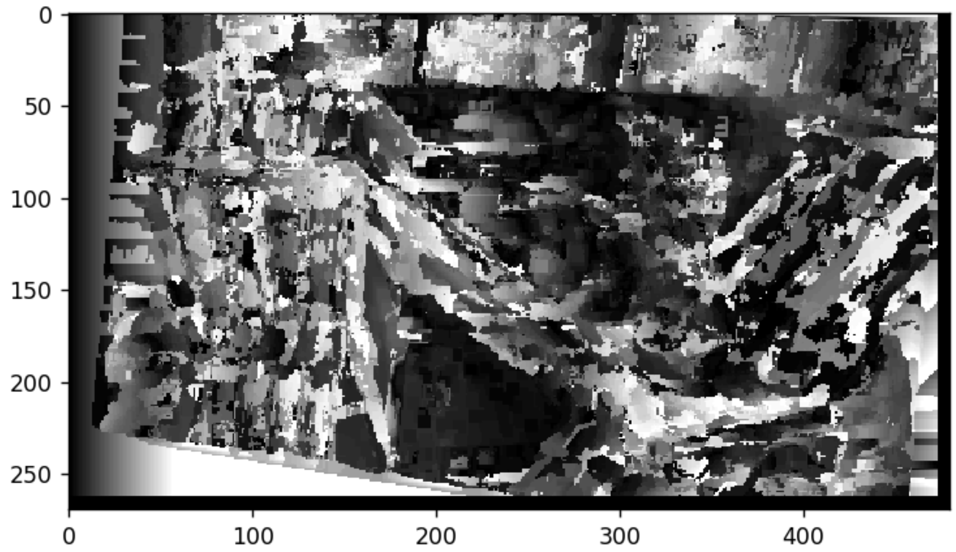
- **Correspondence**

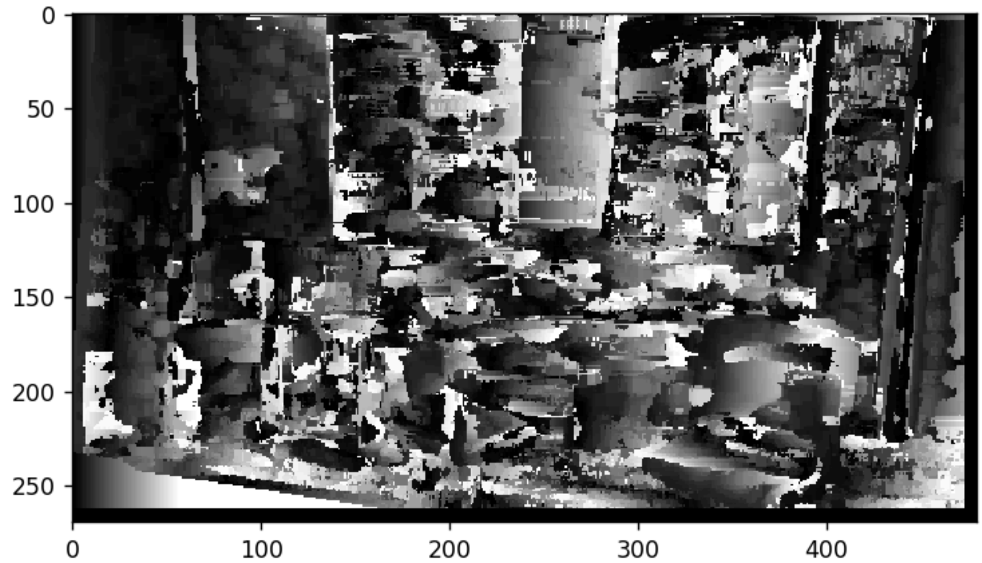
  We use the rectified images obtained in the previous step to get the disparity values for each pixel.

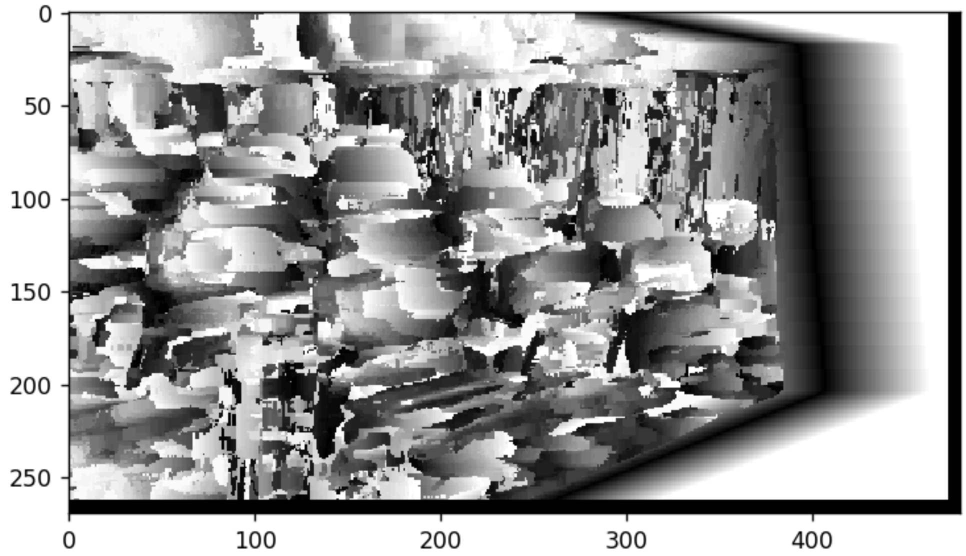
  We first convert the rectified images obtained in the previous step to grey scale images.

  After obtaining the grey scale images we compare a block of pixels (which is obtained by forming a window of pixels on every pixel) in the left image with blocks of pixels in the right image. For the code the window size has been taken to be 7 X 7 pixel. In addition to the window size, we also choose a value for *search_block_size,* which in the code has been taken as 56 pixels. With this we form 7 x 7 windows for every pixel in the range x-56 to x+56 for the right image where x is the column value for the pixel in the left image for which correspondence is being searched. After getting these windows in the right image we perform sum of absolute difference between them and the window in the left image and select the pixel for which this is minimum. After getting the matching pixel from the right image we calculate disparity between them and store it a matrix.

After getting the disparity matrix we rescale the values in it to 0-255 to plot it.



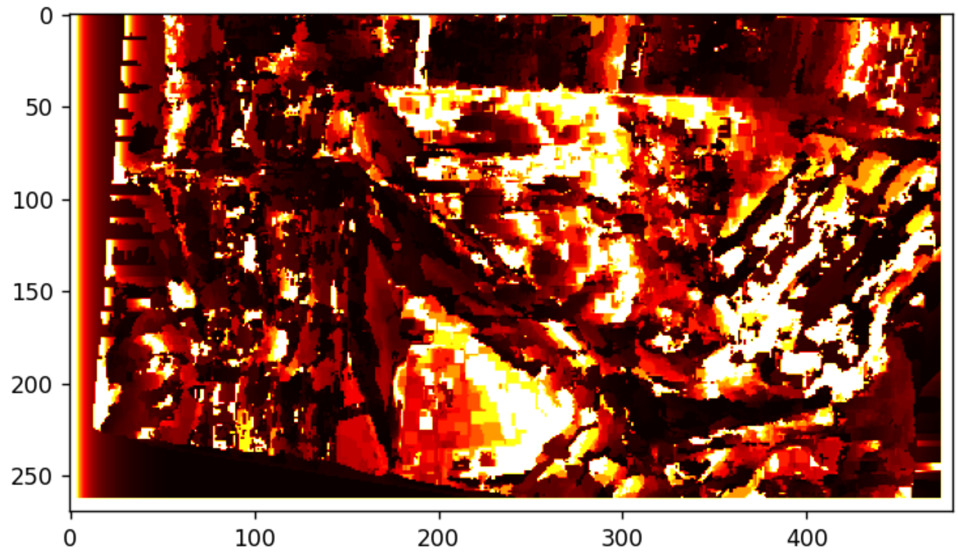


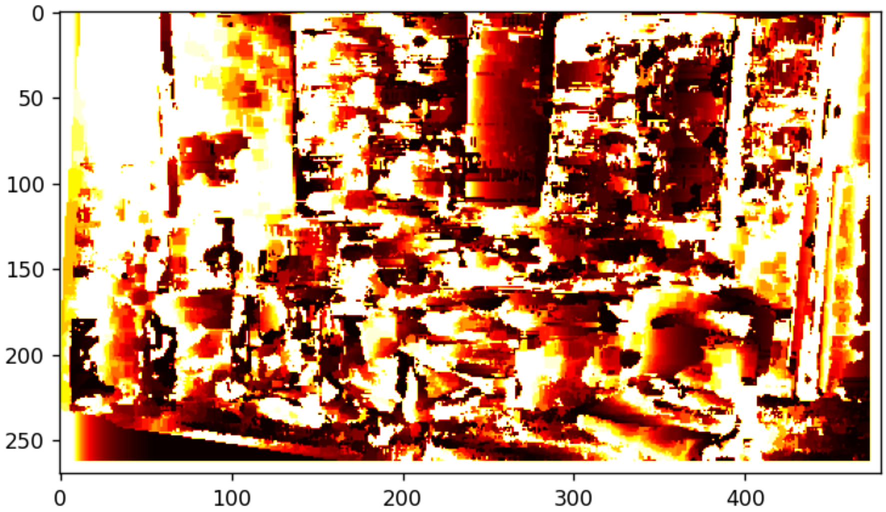


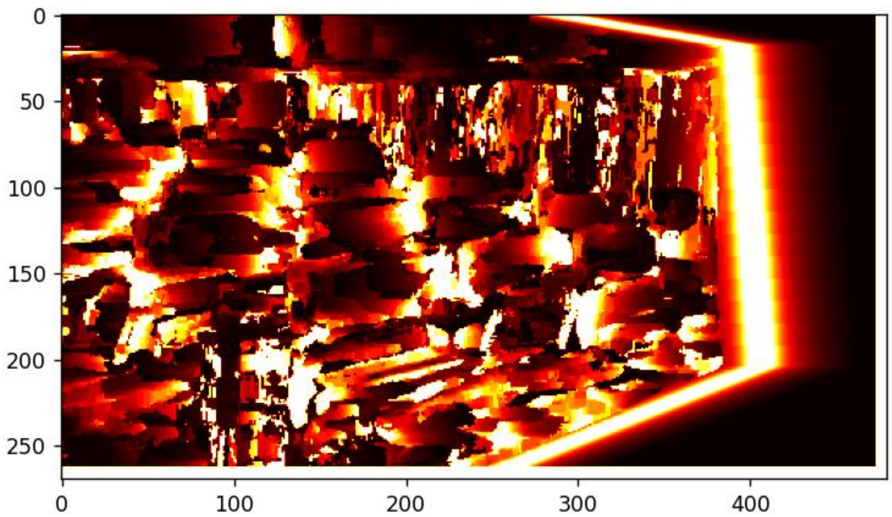
- **Compute Depth**

After getting the disparity matrix we compute the depth information for each pixel by the formula

$$Baseline * \frac{f}{Disparity}$$




0
50
100
150
200
250
0
100
200
300
400


0
50
100
150
200
250
0
100
200
300
400