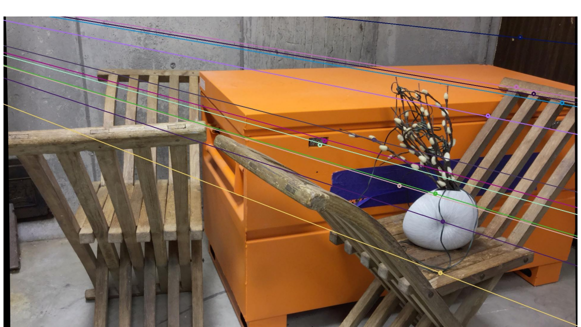
Calibration:

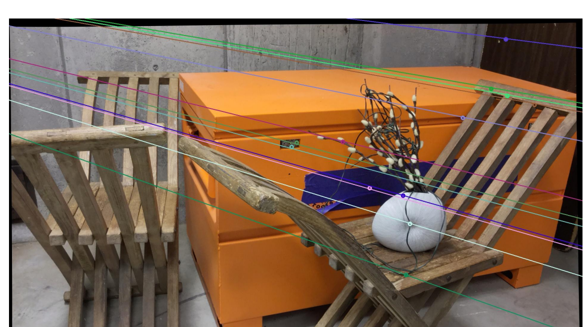
1) First, SIFT is used to detect matching points between left & right images.
2) Only a few (very similar) matches are selected so that the computational intensity of the pipeline remains tractable.
3) Then, the matching points are normalized to make the epipolar lines pass closely through the point correspondences.
4) Then the Fundamental matrix is calculated using the normalized 8-point algorithm.
   a) It is the same as the 8-point algorithm except that normalized point correspondences are used.
   b) In the 8-point algorithm, 8 point correspondences are taken to find the fundamental matrix which satisfies the equation $x_2^T F x_1 = 0$.
   c) Representing the left point by x1=(u, v) & x2=(u', v'), this leads to:

$$\begin{pmatrix} u'_1u_1 & u'_1v_1 & u'_1 & v'_1u_1 & v'_1v_1 & v'_1 & u_1 & v_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u'_nu_n & u'_nv_n & u'_n & v'_nu_n & v'_nv_n & v'_n & u_n & v_n & 1 \end{pmatrix} \begin{pmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{pmatrix} = \mathbf{0}$$

$$\Leftrightarrow \mathbf{Af} = 0$$

   d) Minimum required pairs of corresponding points to generate the Fundamental matrix is 8.
   e) A is solved by SVD to decompose it into U, S, V^T, i.e, $A = U^* S^* V^T$ & the elements of the fundamental matrix are given by the last column of V.
   f) Since, we have 8 equations for 9 unknowns, the smallest singular value of F should be 0. But, due to the noise in the correspondences, it'll be non-zero (rank of F != 2).
   g) So, to account for the noise, rank of F is forced to 2 by setting its smallest singular value to 0.
   h) So, F is decomposed by SVD into u, s, v^T & the smallest singular value is set to zero & F is recomputed as $F = u^*s'^*v^T$.
5) A modified version of RANSAC (see Note I) is applied to increase the robustness of estimation of F.
6) The epipolar lines drawn using the obtained F are shown below:





7) Then, essential matrix is estimated using E = K^T*F*K where K is the camera matrix.
    a) Again, the essential matrix is forced to have rank=2
    b) By SVD, let E = U*D*V^T & $W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
    c) Then the 4 possible camera configurations are given by:

        1. $C_1 = U(:,3)$ and $R_1 = UWV^T$
        2. $C_2 = -U(:,3)$ and $R_2 = UWV^T$
        3. $C_3 = U(:,3)$ and $R_3 = UW^TV^T$
        4. $C_4 = -U(:,3)$ and $R_4 = UW^TV^T$

    d) Then, one of the above 4 possibilities is finally selected as the translation & Rotation matrices of the camera w.r.t scene.
        i) This is done by checking the cheirality condition - reconstructed points must be in front of the camera.
        ii) But, due to the noise, not all triangulated points satisfy this condition.
        iii) So, that combination of C & R are selected which maximizes the number of points satisfying the cheirality condition.
        iv) This way C & R are obtained.

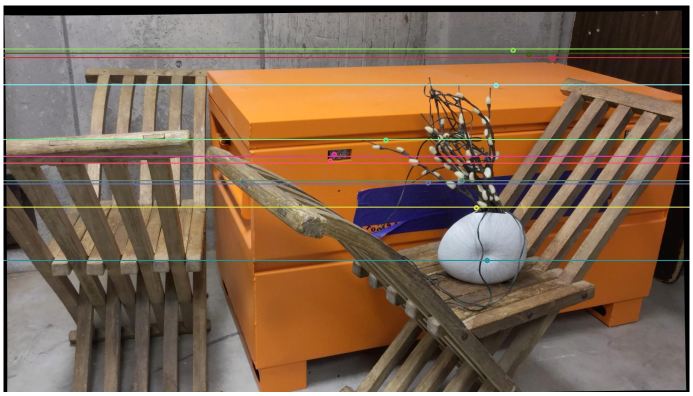
Note:

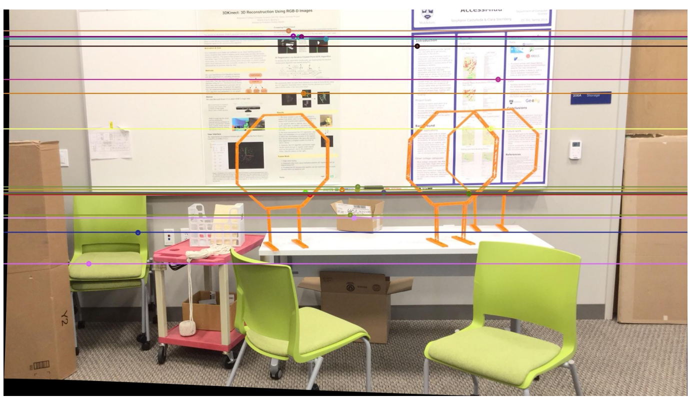
I. Using the standard ransac, subsets containing 8 pairs of point-correspondences are considered to generate the Fundamental matrix. But multiple such subsets are having the same number of inliers - leading to different Fundamental Matrix on different runs. So, I improvised RANSAC such that only 1 solution (*the best*) is outputted in the following way:
    A. Forming the subsets part is kept the same.
    B. Instead of calculating the inliers & choosing the one with the most inliers, that subset is selected which minimizes the sum of $x_2^TFx_1$ for all the pairs of corresponding points.

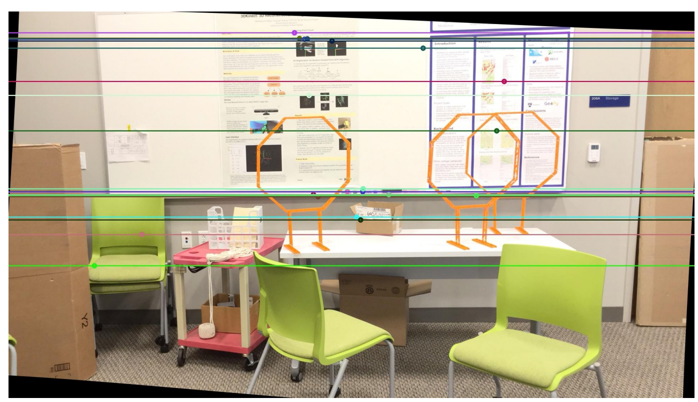
II. The fundamental matrix obtained by using the standard 8-point algorithm is generating epipolar lines which look almost parallel to each other (won't coincide to form epipoles or coincide very far from the image space). Used the normalized 8-point algorithm as described here to deal with this problem.

Rectification:

1) Then, rectification is applied to reproject image planes onto a common plane parallel to the line between camera centers.
2) OpenCV's stereoRectifyUncalibrated is used to find the rectification transform.
3) This returns the Homography matrices for the left & right images.
4) The rectified Fundamental matrix is computed & the below are the epipolar lines along with feature points:

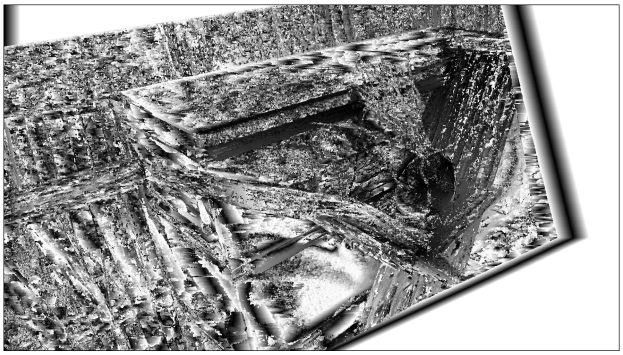


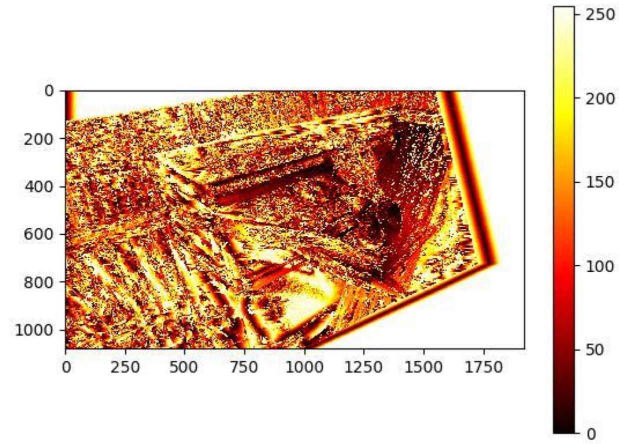


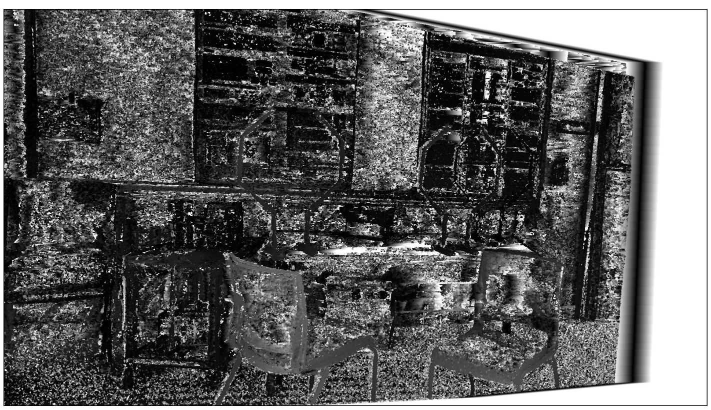


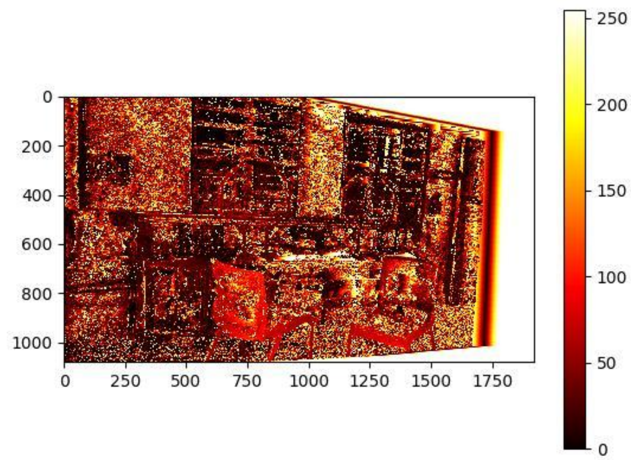
Correspondence:

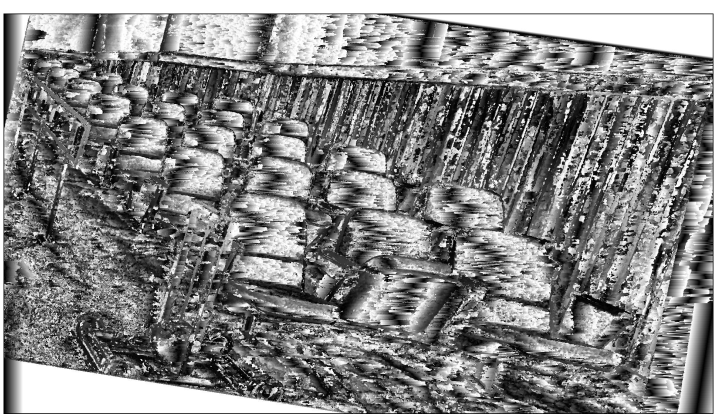
1) Then, disparity maps is calculated using the SSD (sum of squared differences) similarity.
2) For a given block on left image, similarity is checked only with a neighborhood of blocks, to reduce the computation intensity.
3) Below are the disparity maps generated.

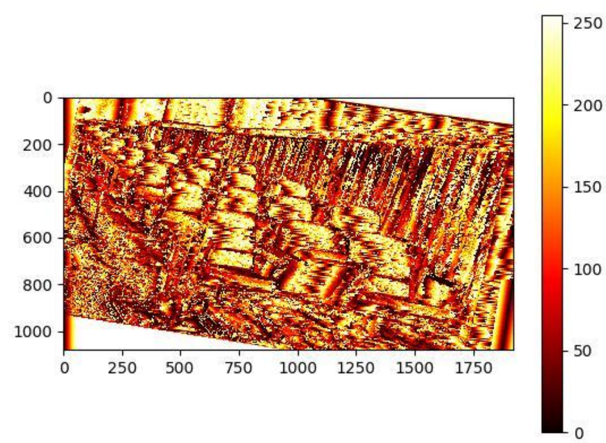












Depth Image:

1) Then, depth map is computed using disparity using: (baseline * focal_len) / (disparity)
2) Those depths which fall above the median depth are set to the median depth.