ECE661: Computer Vision (Fall 2014)

Shaobo Fang: s-fang@purdue

November 27, 2014

Contents

1 Overview

The task of this assignment is to reconstruct the 3D scene based on two views based on canonical camera model. To begin with, we will first manually select at least 8 correspondences. The initial fundamental matrix will be estimated based on the 8 pairs of correspondences. Then as always, in order to increase the accuracy of fundamental matrix we will perform `Levenberg Marquardt` non linear optimization.

Image rectification will be performed on both of the views and the correspondences based on SIFT will be established using the rectified images. Similar as in previous homework, NCC will be used to establish the best correspondences.

Then, after sufficient interest points were obtained we will estimate the fundamental matrix again and form the P' matrix and 3D reconstruction will be performed accordingly based on the larger size of the interest points detected in the rectified images.

We will plot the reconstructed 3D stem points using `stem3` function in `Matlab`. If the matrix P' is good then we can easily see the reconstructed straight lines, which will be marked in the 3D plots in the results section.

As we are more interested in the edges in the images, we will be using `Canny Edge Operator` to extract the edges and then perform the SIFT interest points extraction on the Canny edges.

2 Image Rectification

Before the interest points were extracted from the images, image rectification will be performed first. According to the lecture notes, we are in a `chicken and egg situation`. The image rectification could be done by manually selection of at least 8 points correspondences. Based on the correspondences, denote as x and x', we can formulate the equations below:

$$\vec{x}^T F \vec{x}' = 0$$

Where $F = \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix}$ is the fundamental matrix and define: $\vec{x} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$, $\vec{x}' = \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix}$.

Now, similar as it was done in previous homework, given the correspondences we will again transform the matrix calculation into linear functions. Thus, for each of the correspondence, we will have:

$$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0$$

Assume we have a total number of n correspondences, then:

$$A^{n\times 9}\vec{f} = 0$$

Where i^{th} row of matrix A is:

$$A^{i,1:9} = [x'x \ x'y \ x' \ y'x \ y'y \ y' \ x \ y \ 1]$$

and

$$\vec{f} = [f_{11} \ f_{12} \ f_{13} \ f_{21} \ f_{22} \ f_{23} \ f_{31} \ f_{32} \ f_{33}]^T$$

Now, in order to achieve get optimized results, the number of equations is roughly five times the number of unknowns when using any least squared algorithm for solving a system of homogeneous or inhomogeneous equations.

However, at the first stage of image rectification, we can use the correspondences that we manually picked. The fundamental matrix F will be refined using non-linear `Levenberg - Marquardt` method. Besides, the rank of constrain also need to be enforced by the following command:

$$[U \ D \ V] = svd(F)$$

$$D(3,3) = 0 \ (set\ minimum\ singular\ value\ to\ `0')$$

$$F = U * D * V'$$

In order to improve the performance, we can introduce the normalization matrix T_1 and T_2. T_1 and T_2 could be obtained accordingly (book section 4.4.4):

1. Detect Interest Points (at least 8)

2. Find the centroid of the interest points, and move it to the origin
3. Estimate the Similarity transform so that the average between each interest point with regard to the origin is $\sqrt{2}$
4. Do the same thing for both of the images.

After we have obtained the refined fundamental matrix F, the epipoles are simply the left and right null vector of the fundamental matrix. **As we already refine the order of Fundamental matrix to be 2, it will guarantee to yield non-zero left and right null vector**. Denote as:

$$\vec{e} \; and \; \vec{e}'$$

Now, in order to find homography H', we first need to transform the center of the image to the origin $(0,0)$, denote the transformation is done by matrix T. After obtaining T (Euclidean Transform), we then will estimate the rotation matrix R. While our goal is to transform $\vec{e}'$ to the form of $\vec{\hat{e}}' = \begin{bmatrix} f \\ 0 \\ 1 \end{bmatrix}$

Assume the rotation matrix is:

$$R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Where:

$$\theta = \arctan(\frac{-\vec{e}'(2)}{\vec{e}'(1)})$$

Then finally we have;

$$G = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\frac{1}{f} & 0 & 1 \end{bmatrix}$$

Accordingly,

$$H' = GRT$$

Or, with normalization (similarity transform):

$$H' = T_2 GRT$$

After we have estimated H', we need to estimate H. Assume that we have:

$$\vec{\hat{x}} = H\vec{x}$$

$$\vec{\hat{x}}' = H'\vec{x}'$$

$$H = \arg\min_{H} ||\vec{\hat{x}} - \vec{\hat{x}}'||$$

$$M = P'P^{\dagger}$$

Where

$$P = [I|\vec{0}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$P' = [[\vec{e'}]_X F|\vec{e'}] = \begin{bmatrix} 0 & -e'_3 & e'_2 & e'_1 \\ e'_3 & 0 & -e'_1 & e'_2 \\ -e'_2 & e'_1 & 0 & e'_3 \end{bmatrix}$$

Then assume H_A is:

$$H_A = \begin{bmatrix} a & b & c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Using least squared to estimate parameters a, b and c:

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \arg\min_{a,b,c} \sum ||a\hat{x} + b\hat{y} + c - \hat{x}'||$$

Finally,

$$H = H_A H_0,\ where\ H_0 = H_2 M$$

If images rectification has good performance, than all the epipolar lines would be transformed to horizontal lines (see results section).

3 Interest Points Detection

SIFT operator was used to extract interest points from both of the images. Now the challenge is SIFT is not an edge detector, thus the interest points extracted are not necessarily lying on the edges. Although the interest points not lying on the edges will not impact establishment of correspondences, it will not give a `visually strong` case for 3D reconstruction. The points reconstructed in 3D will be scattered all over the space.

In order to have the the interest points lying all over the images, we will use SIFT to extract features from `Canny Edge` of the image. This improve the results a bit however still not all edges could have been detected.

After interest points were extracted base don Canny Edge, we will then establish correspondences using `Normalized Cross Correlation`. The following two criteria has to be met in order to establish a correspondence:

1. NCC score larger or equal to 0.8
2. Maximum NCC score for a specific interest is larger or equal to 1.1*{Second Largest} NCC score

4 3D Reconstruction

The 3D reconstruction will be performed based on the assumption of the canonical camera models. Then

$$P = \begin{bmatrix} \vec{p}_1 \\ \vec{p}_2 \\ \vec{p}_3 \end{bmatrix}$$

$$P' = \begin{bmatrix} \vec{p}'_1 \\ \vec{p}'_2 \\ \vec{p}'_3 \end{bmatrix}$$

X is a 3D coordinate, and the A matrix in the form AX is :

$$A = \begin{bmatrix} x\vec{p}_3 - \vec{p}_1 \\ y\vec{p}_3 - \vec{p}_2 \\ x'\vec{p}'_3 - \vec{p}'_1 \\ y'\vec{p}'_3 - \vec{p}'_2 \end{bmatrix}$$

Then, 3D coordinate X could be solved applying $SVD(A)$.

Once more, using `Levernberg Marquardt` to refine the result.

5 Results

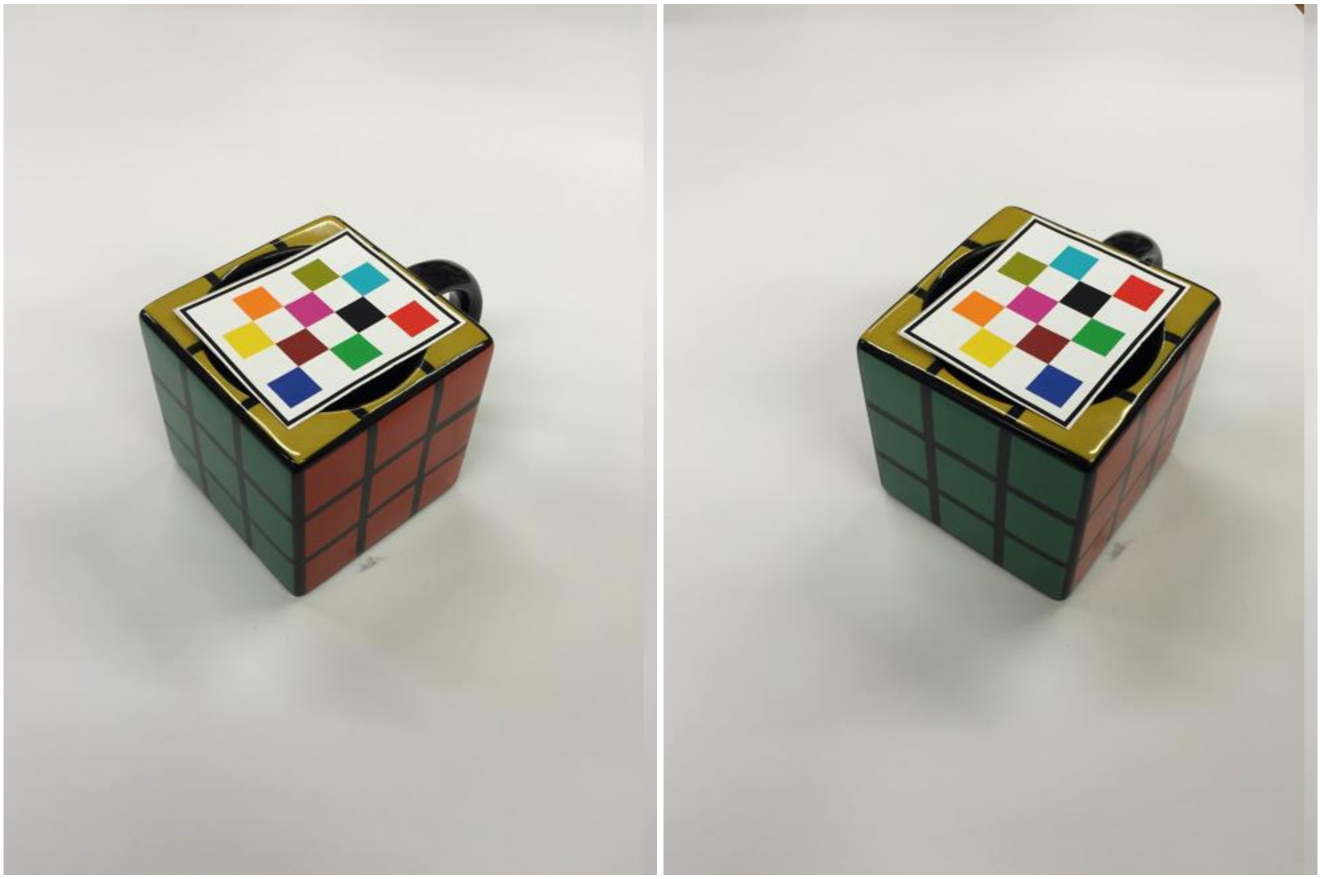


Fig 1. Original Image `pic1.jpg` and `pic2.jpg`

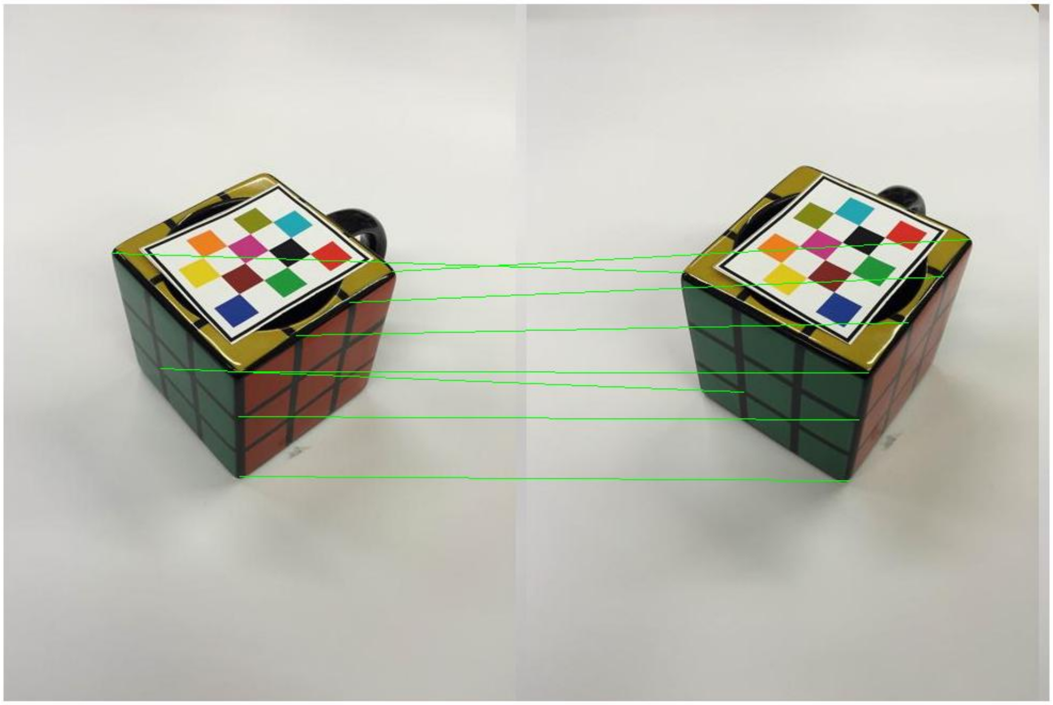
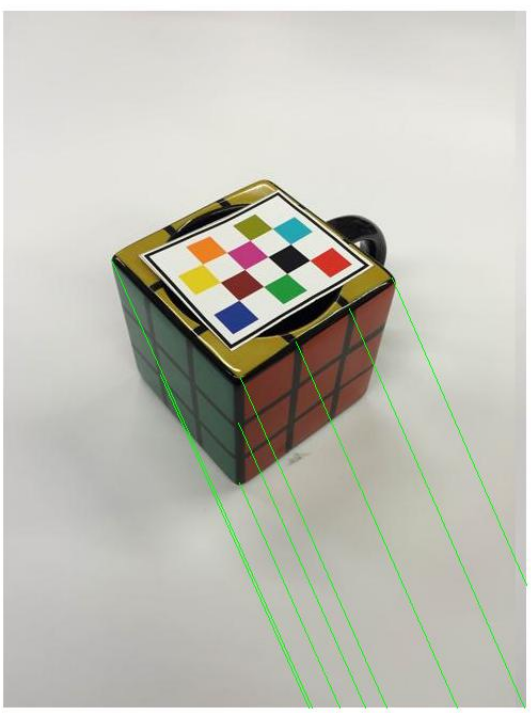


Fig 2. Manually Selection of 8 Correspondences



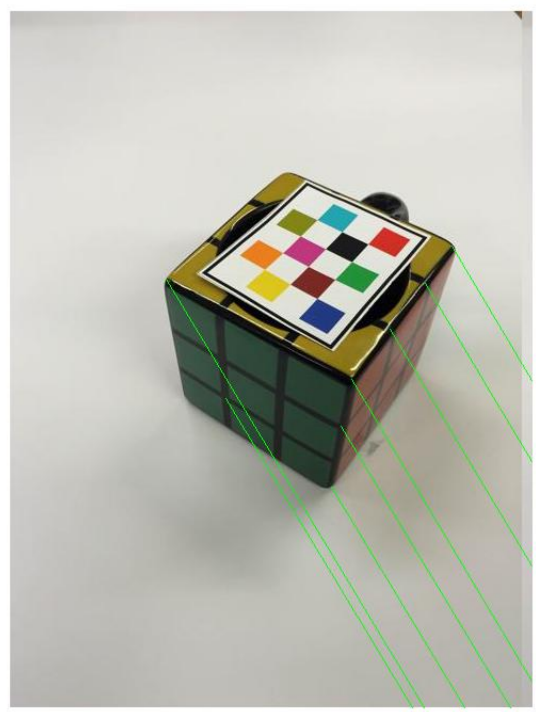


Fig 3. The epipolar lines (connecting epipole and interest points)

Now, after image rectification, the epipolar lines should be horizontal.

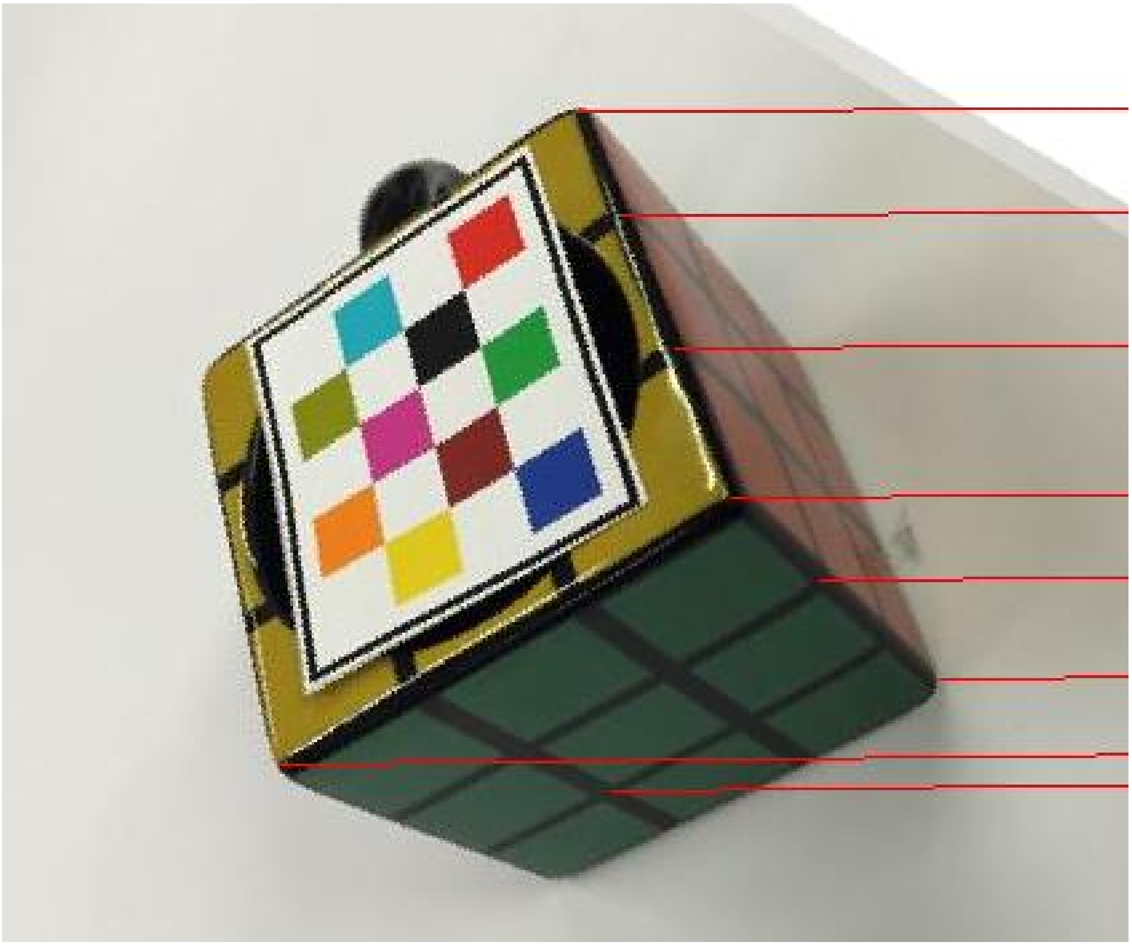


Fig 4. The transformed epipolar lines for `pic1.jpg`

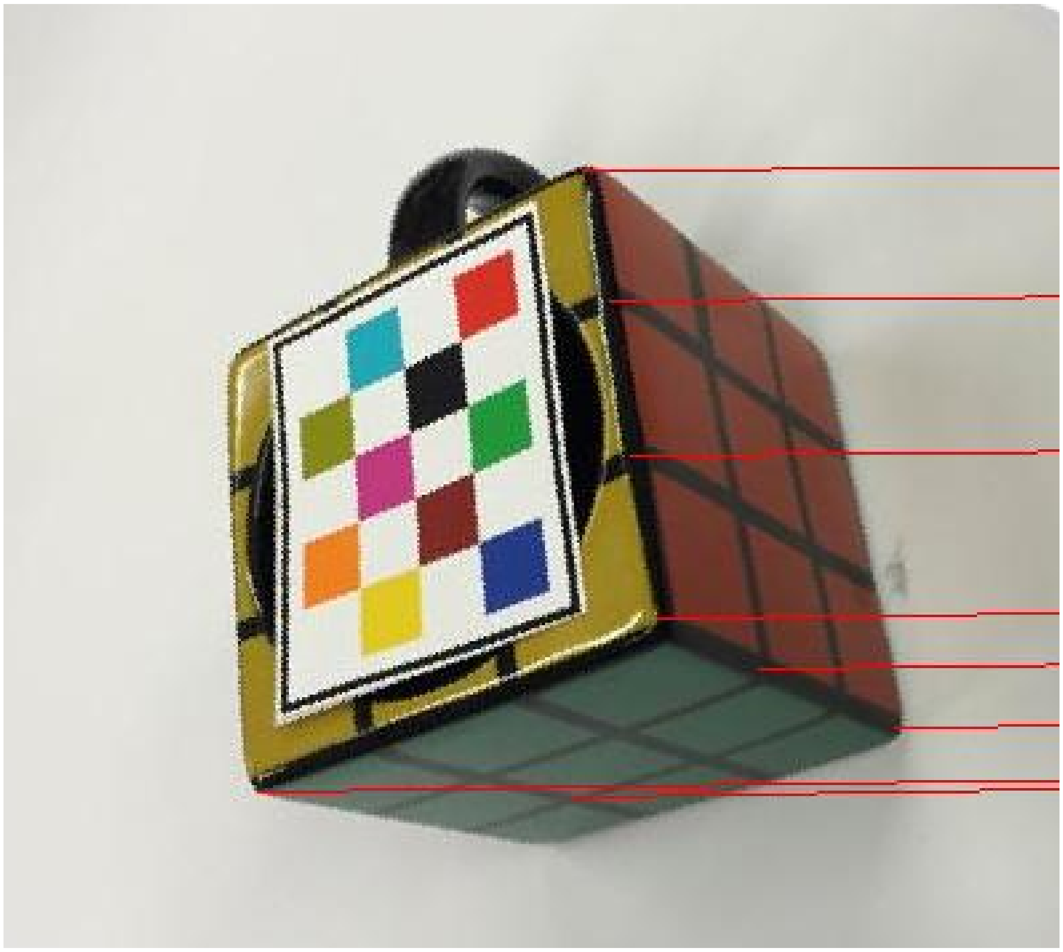


Fig 5. The transformed epipolar lines for `pic2.jpg`

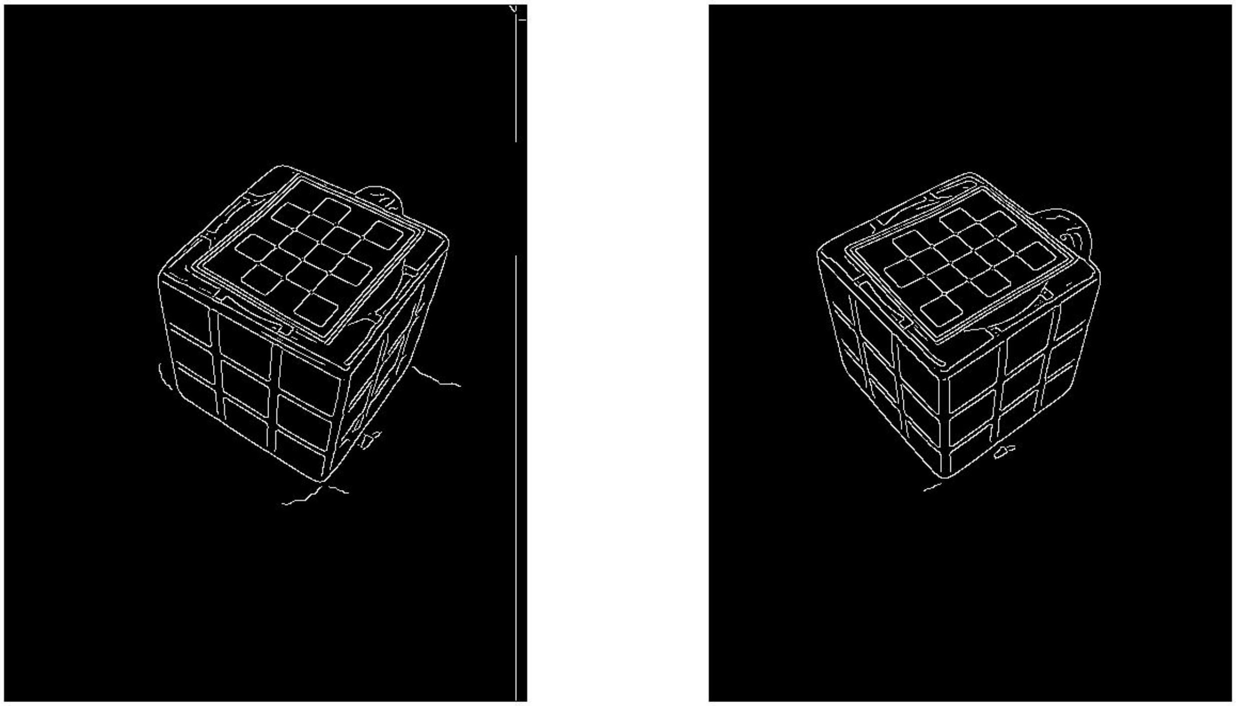


Fig 6. The Canny Edge for the both images

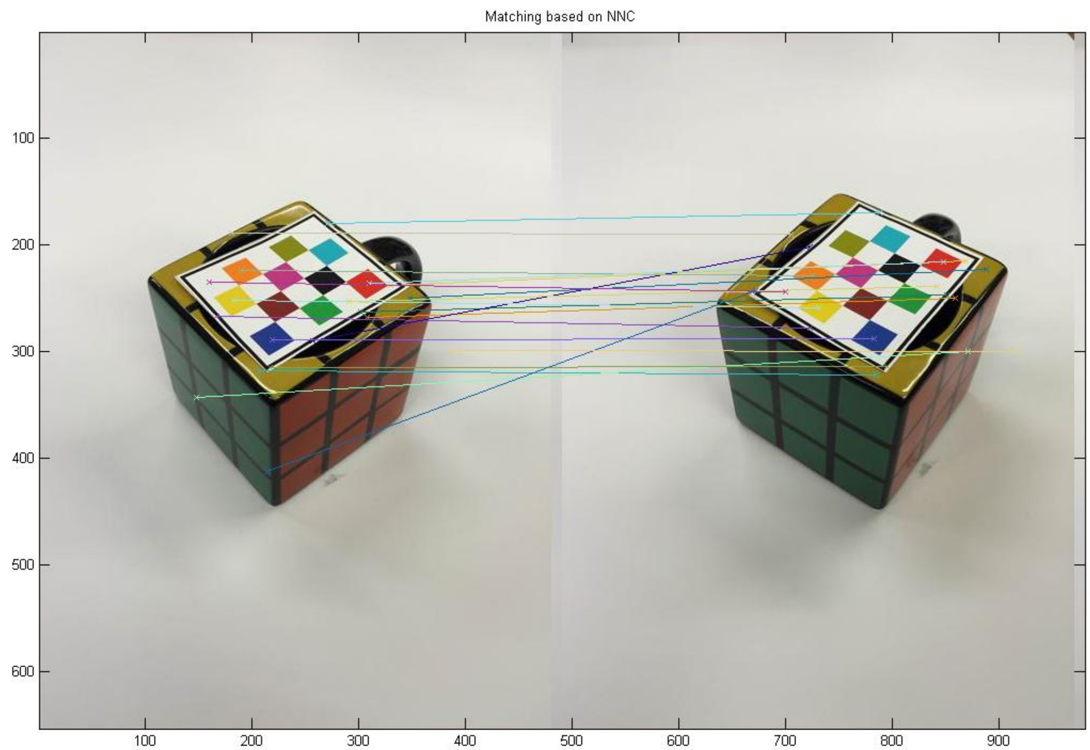


Fig 7. SIFT Correspondences using NCC based on Canny Edge.

Now the problem is that although most of the correspondences are correct and they are mostly lying on the edges, the correspondences do not have a strong pattern regarding **lines**. `(Because SIFT is not a line detector)` Thus, in order to have a better 3D representation of lines, we will manually select 40 correspondences as a last resort (very unfortunately).

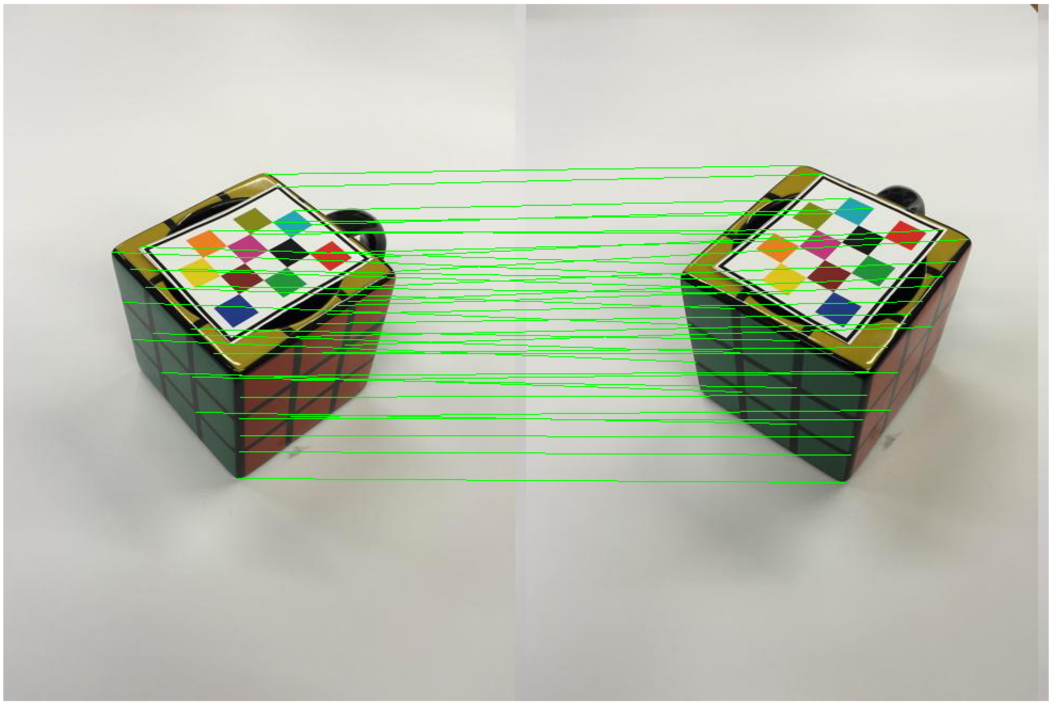


Fig 8. Correspondences On Edges

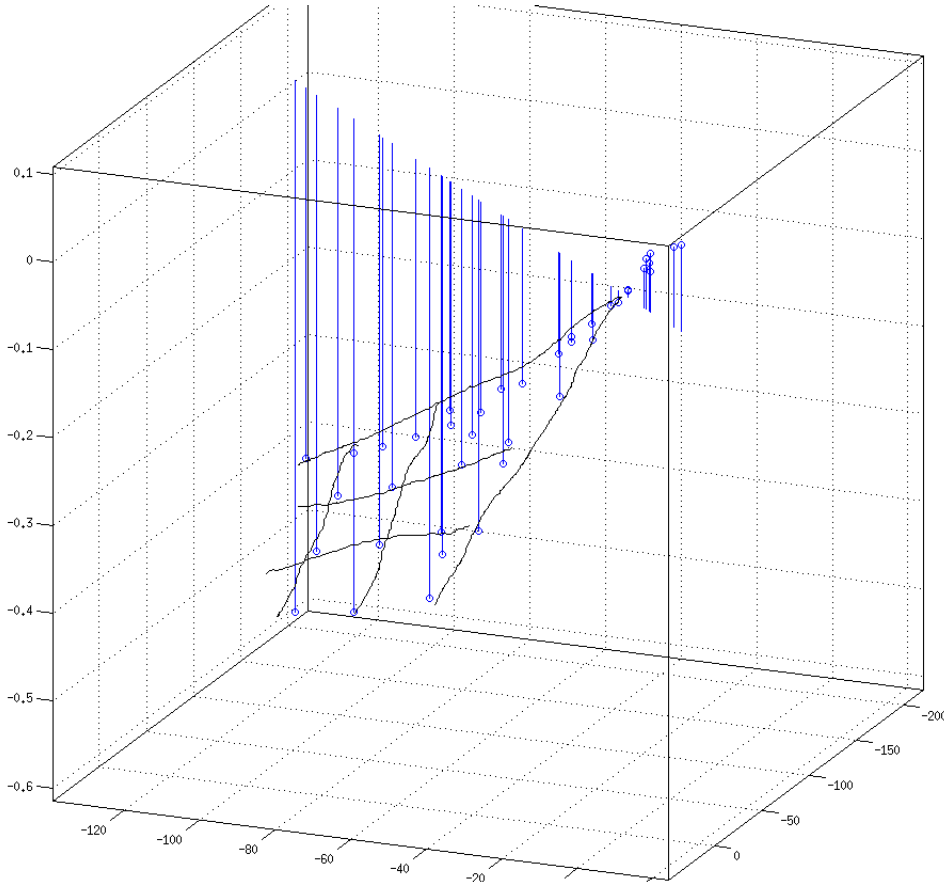


Fig 9. The manually demonstrated `lines`

6 Appendix: Code

Please refer to the zip file.