Abhishek Bamotra
Andrew ID: abamotra

Homework 2

Problem 1

Q 1.1

Following proves the existence of homography H that satisfies the equation.

Let's assume P_1 and P_2 as two 3 X 4 camera projection matrices corresponding to the two cameras and a plane π.

Also assume, there are two points x_1 and x_2 in a homogeneous coordinates, which means they have an additional dimension.

$$x_1 \equiv \begin{pmatrix} d_{11} & d_{12} & d_{13} & d_{14} \\ d_{21} & d_{22} & d_{23} & d_{24} \\ d_{31} & d_{32} & d_{33} & d_{34} \end{pmatrix} \begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \\ 1 \end{bmatrix} \tag{1}$$

For this given setup, x = [X, Y, Z]^T, the equation for point x_1 can be written in terms of x_2 as equation (1). Similarly, we can write the equation for x_2 in terms of x_1 by taking the inverse of the matrix. As the matrix is not a square matrix, we can't take inverse of it. To handle this, we can assume Z=0 and consider only X-Y plane. If we do that, we get new equation as follows:

$$x_1 \equiv \begin{pmatrix} d_{11} & d_{12} & d_{14} \\ d_{21} & d_{22} & d_{24} \\ d_{31} & d_{32} & d_{34} \end{pmatrix} \begin{bmatrix} X_2 \\ Y_2 \\ 1 \end{bmatrix} \tag{2}$$

Now, from the above equation, we can represent x_1 and x_2 by inverting the matrix. Hence, we get:

$$x_1 = Ax_2 \equiv Hx_2$$

The above equation proves the existence of **H**. Also, this equation is true upto a scaling factor.

Q 1.2

1. h has 8 degrees of freedom as one element of h is responsible for scaling.

2. As h has 8 unknowns, we need 8 points to solve for h. Therefore, we need 4 point pairs to solve h.

3. Let's assume two points as x_1 and x_2 as $[u_1, v_1, 1]^T$ and $[u_2, v_2, 1]^T$. We have,

$$\begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} \equiv \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} \tag{3}$$

To remove $\equiv$, we can have a scaling factor, s, with x_1.

$$s \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{bmatrix} u_2 \\ v_2 \\ 1 \end{bmatrix} \tag{4}$$

Equation for u_1 and v_1 using equation (3) and (4), we get:

$$u_1 = \frac{h_{11}u_2 + h_{12}v_2 + h_{13}}{h_{31}u_2 + h_{32}v_2 + h_{33}}$$

$$v_1 = \frac{h_{21}u_2 + h_{22}v_2 + h_{23}}{h_{31}u_2 + h_{32}v_2 + h_{33}}$$

Re-arranging the above equation:

$$u_1(h_{31}u_2 + h_{32}v_2 + h_{33}) - h_{11}u_2 - h_{12}v_2 - h_{13} = 0$$

$$v_1(h_{31}u_2 + h_{32}v_2 + h_{33}) - h_{21}u_2 - h_{22}v_2 - h_{23} = 0$$

Re-writing the equation in matrix form yields:

$$\begin{pmatrix} -u_2 & -v_2 & -1 & 0 & 0 & 0 & u_1u_2 & u_1v_2 & u_1 \\ 0 & 0 & 0 & -u_2 & -v_2 & 1 & v_1u_2 & v_1v_2 & v_1 \end{pmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = 0 \tag{5}$$

Similarly, we can find for N points and stack them together.

4.

The trival solution for $\mathbf{h}$ is $[0, 0, 0, 0, 0, 0, 0, 0, 0]^T$.

Matrix $\mathbf{A}$ has 8 degree of freedom because of which it is not full rank.

As the degree of freedom is 8, therefore, the null space is 1 (1 - D.O.F). The dimension of null space determines the number of zero eigenvalues corresponding to which we get a solution for our A matrix. This eigenvector corresponding to the zero eigenvalue is the solution for the equation $Ah = 0$.

Q 1.3

To prove there exists a homography **H** that satisfies $x_1 \equiv Hx_2$, given two cameras separated by a pure rotation. That is, for camera 1, $x_1 = K_1 \begin{bmatrix} I & 0 \end{bmatrix} X$ and for camera 2, $x_2 = K_2 \begin{bmatrix} R & 0 \end{bmatrix} X$. We know, X is a point in 3D space, therefore,

$$X = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

We can remove zero column and corresponding row from camera 1 equation. We are left with:

$$x_1 = K_1 \begin{bmatrix} I \end{bmatrix} X = K_1 X$$

We can also remove the zero column and corresponding row from camera 2 equation. We are left with:

$$x_2 = K_2 \begin{bmatrix} R \end{bmatrix} X = K_2 R X$$

Using the above two equation, we can write an equation as:

$$x_2 = K_2 R X = K_2 R K_1^{-1} x_1)$$

Therefore, by comparing the equation with:

$$x_2 \equiv H x_1$$

We get,

$$H = K_2 R K_1^{-1}$$

Therefore, we can say there exists a homography H that satisfies $x_1 \equiv Hx_2$.

Q 1.4

From the conclusion in section 1.3, we know $H = K_2RK_1^{-1}$, So, $H^2 = (K_2RK_1^{-1})^2 = (K_2RK_1^{-1})(K_2RK_1^{-1})$ as K is constant. Therefore, we can write, $H^2 = KRRK^{-1}$. We have rotation matrix as:

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

Therefore,

$$H^2 = KRRK^{-1} = K \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} K^{-1}$$

$$= K \begin{bmatrix} cos^2\theta - sin^2\theta & -2sin\theta cos\theta & 0 \\ 2sin\theta cos\theta & -cos^2\theta + sin^2\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} K^{-1}$$

$$= K \begin{bmatrix} cos2\theta & -sin2\theta & 0 \\ sin2\theta & cos2\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} K^{-1}$$

Q 1.5

Planar homography is limited to map any arbitrary scene to another viewpoint as planar homography is poor in handling images with repeated patterns.

Q 1.6

To prove this let's take an example. Let's consider a line in 3D space with coordinates as $[0, 0, 1]^T$, $[1, 1, 1]^T$ and $[5, 5, 1]^T$. So, we can use a perception projection matrix as:

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Now, we can multiply the matrix P with our line in 3D space and we get:

$$x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 5 \\ 0 & 1 & 5 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 5 \\ 0 & 1 & 5 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

From this, we can see our line is projected to XY plane and the line is still preserved. Therefore, we can state that perspective projection preserves lines (a line in 3D is projected to a line in 2D).

Problem 2

Q 2.1.1

FAST and Harris corner detectors are both used to detect corners based on change in intensity. Harris corner detector uses a sliding window which means to move a region/window over the scene. Which results in computing derivatives with the neighbouring pixels. Whereas, FAST detector samples a pixel, and considers a circle of 16 pixels around that. A threshold is set to define what is defined as a change in intensity based on which pixel intensities of four pixels on the axis are checked. Based on the intensity of chosen pixel and the pixels on the circle is above/below the threshold, the decision of a corner is made. Because of the sampling, FAST detector has lower computation cost i.e it is faster.

Q 2.1.2

The BRIEF descriptor are different from the filterbanks we have read in lecture. BRIEF helps in finding binary strings directly without finding any descriptors. It generates a binary string and uses hamming distance to find distance between the two strings. Whereas, filter banks are just responses of the images and are mostly helpful for clustering.

Q 2.1.3

Hamming distance is a distance metric using for comparing two binary strings. Hamming distance is calculated by adding the number of bit locations with different bit values in two strings. Higher the location number, higher is the distance.

BRIEF finds binary strings directly and chooses the lowest two distances and takes the ratio and depending on ratio it defines a good/bad match.

Hamming distance is faster than other conventional distance measures because it operates on binary string and does binary operations and calculations. Hamming distance works on XOR.

Q 2.1.4 The resulting matching image is shown in figure below.

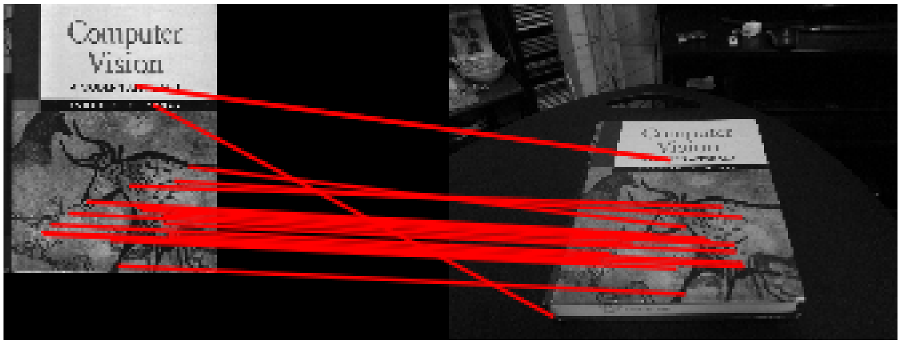
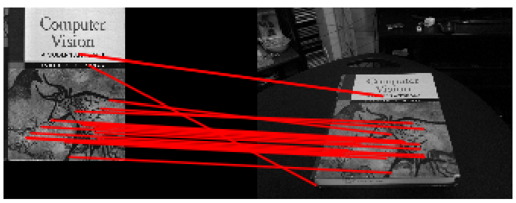


Figure 1: Matching plot for briefMatch()

Q 2.1.5

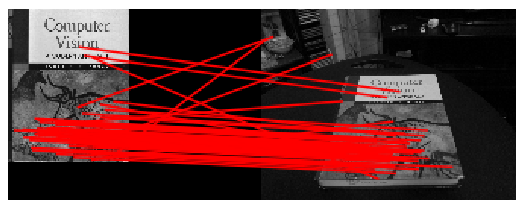
Run no.	Sigma	Ratio
a	0.15	0.7
b	0.15	0.9
c	0.15	0.2
d	0.10	0.7
e	0.2	0.7



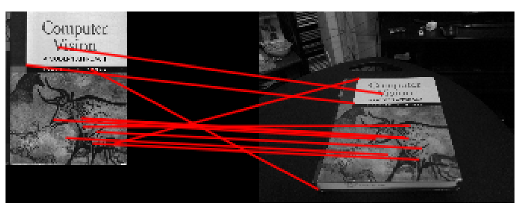
(a)

(b)

(c)



(d)



(e)

Figure 2: Results from tuning of the parameters sigma and ratio

From the tuning data, we can say that:

1. Change in Ratio has direct impact on the number of matches. If we increase the ratio, i.e in above observation to 0.9, the matching points increases. If we drop ratio to as low as 0.2, we loose matching points and in this case, we were not able to find any matches at all.

2. Change in Sigma has an inverse impact on the number of matches. If we decrease the sigma, i.e. in the above observation to 0.10, the matching points increases. If we increase the sigma i.e. 0.2, we see number of match points are dropping.

Q 2.1.6

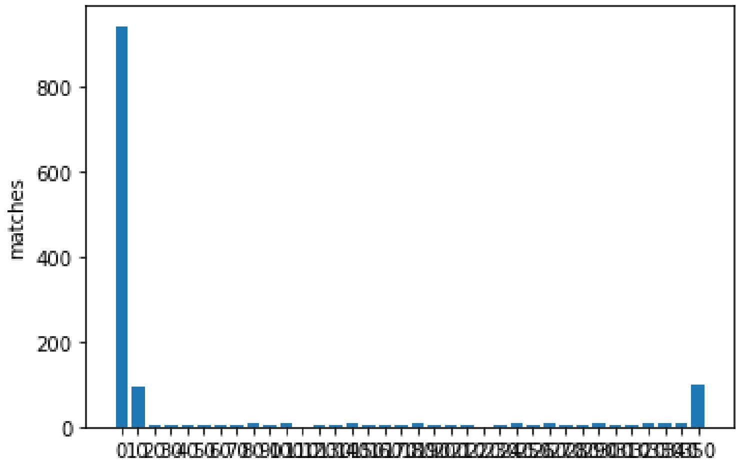
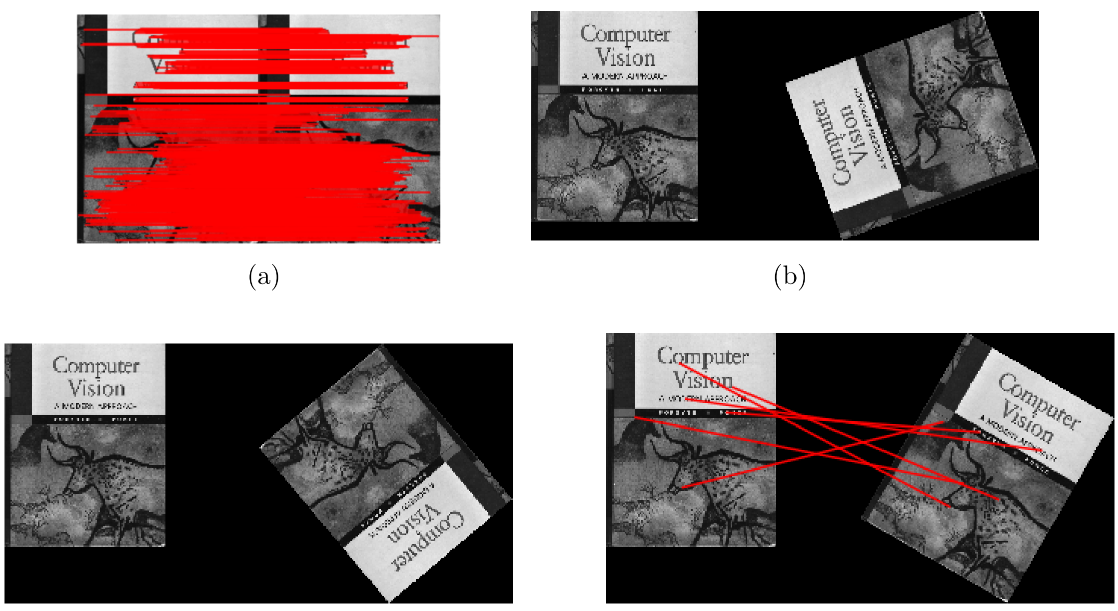


Figure 3: Histogram showing number of matches at every 10 degree rotation



(a) (b) (c) (d)

Figure 4: Matching points at rotation 0, 110, 220, 330

BRIEF descriptor is a rotation variant descriptor because of which it doesn't work well on rotated images and the same thing can be seen in the results in figure (3) and (4). It works so bad that after a certain rotation, it is not even able to find one match between the images.

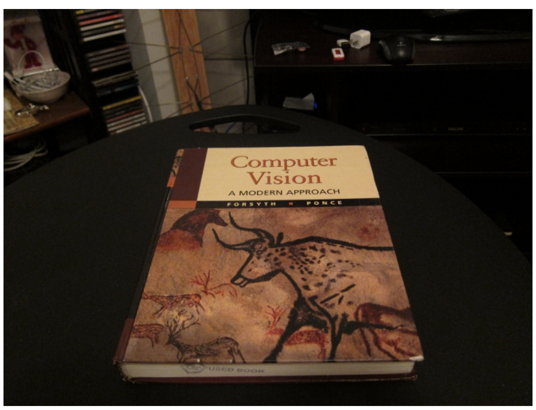
Q 2.2.4

```
4. At this point you should notice that although the image is being
warped to the correct location, it is not filling up the same space
as the book. Why do you think this is happening? How would you
modify hp cover.jpg to fix this issue?
```

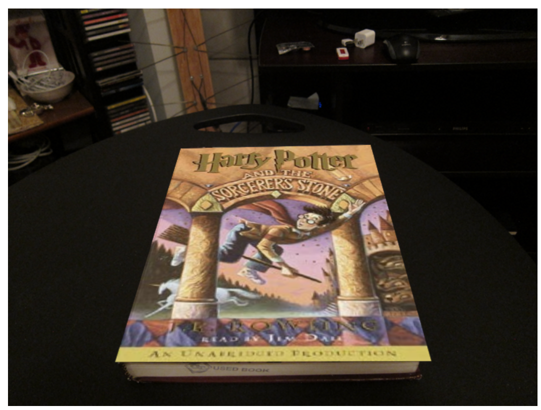
The dimension of hp_cover is **200 x 295**, whereas dimension of cv_cover is **350 x 440** which is much bigger than the hp_cover. The difference in dimension causes this issue.

This issue can be easily resolved by changing the dimension of the hp_cover to that of cv_cover. This will prevent the issue and result in image warping to be at the same location and will fill the entire space of the book.

The result of the HarryPotterize.py is shown in figure 5.



(a) Text Book

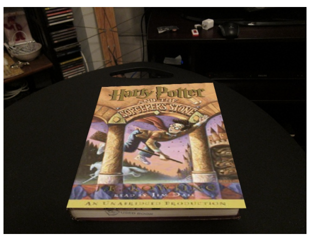


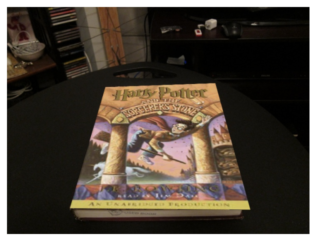
(b) Harry Potterized Text Book

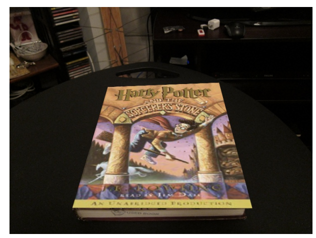
Figure 5: Result of HarryPotterize script

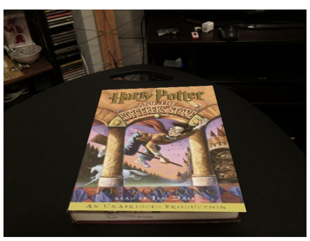
Q 2.2.5

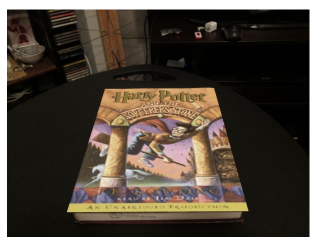
Run no.	max iterations	inliers tol	matches	inliers
a	1000	2.0	130	55
b	5000	2.0	130	56
c	400	2.0	130	55
d	100	2.0	130	52
e	500	1.0	130	34
f	500	1.5	130	50
g	500	2.0	130	55
h	500	2.5	130	58

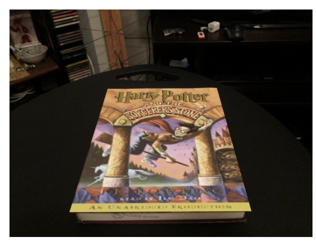

(a)

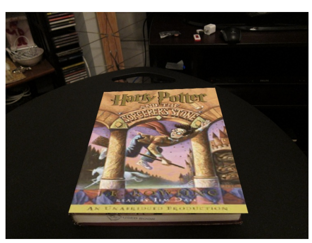

(b)


(c)


(d)


(e)


(f)


(g)

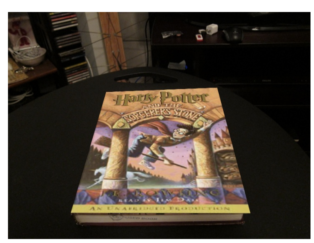

(h)

Figure 6: Hyper parameter tuning results for Ransac

From the study, we can say that as I increase the inliers tolerance, I'm able to get more inliers as seen in run 'h', but can't be increase too high as it will give bad homography matrix and drops when we lower the inliers tolerance i.e. run 'e'. Whereas for maximum iterations, I didn't see any change in the number of inliers. So, for me optiminal number was 800 iterations with 2.0 as inlier tolerance.

Q 3.1 Video is attached in the folder.