GZ720J: Homework 1 - Planar Homographies

Name - Jiaxin Chen

September 19, 2016

1 Planar Homographies: Theory (40pts)

Question 1.1 (10pts)

Prove that there exists an $\boldsymbol{H}$ that the satisfies homography equation.
The easiest way to show this is to assume that the points are on the 3D space. Then the points in the plane are of the form $[\boldsymbol{X}\,\boldsymbol{Y}\,\boldsymbol{Z}\,\mathbf{1}]^{\boldsymbol{T}}$

Therefore, the original equations for $\boldsymbol{p}\mathbf{1}$ and $\boldsymbol{p}\mathbf{2}$:

$$p_1 \equiv \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}_1 \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{1}$$

$$p_2 \equiv \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}_2 \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2}$$

Because matrix $\boldsymbol{M}$ is a 3×4 matrix, which cannot be invertible. We can take the plane into only XY plane by setting Z=0.

Hence, take p_1 as the example, we can get:

$$p_1 \equiv \begin{bmatrix} m_{11} & m_{12} & 0 & m_{14} \\ m_{21} & m_{22} & 0 & m_{24} \\ m_{31} & m_{32} & 0 & m_{34} \end{bmatrix}_1 \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} \tag{3}$$

which can be simplified as:

$$p_1 \equiv \begin{bmatrix} m_{11} & m_{12} & m_{14} \\ m_{21} & m_{22} & m_{24} \\ m_{31} & m_{32} & m_{34} \end{bmatrix}_1 \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{4}$$

And the new matrix $\boldsymbol{M}$ is a 3×3 matrix now, which can be invertible.

Therefore, we can obtain:

$$p_2 \equiv M_2 P \equiv M_2 M_1^{-1} P \equiv H p_1 \tag{5}$$

So there exists an $\boldsymbol{H}$ where

$$H \equiv M_2 M_1^{-1} \tag{6}$$

When does this fail?

If M is a singular Matrix, it doesn't have the inverse matrix, this equation will fail.

Question 1.2 (10pts)

Prove that there exists an $\boldsymbol{H}$ that satisfies homography equation given two cameras separated by a pure rotation.

$$p_1 = K_1[I\,0]P \tag{7}$$

$$p_2 = K_2[R\,0]P \tag{8}$$

HINT: Now you are trying to find that the H exists. It will be a good idea to try to manipulate these expressions in the direction of some expression

$$p_1 = K_1[I\,0]P = K_1[I\,0]\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = K_1\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{9}$$

So the point P can be represented as

$$P = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} K_1^{-1}p_1 \\ 1 \end{bmatrix} \tag{10}$$

Therefore, the point p_2 is

$$p_2 = K_2[R\,0]P = K_2[R\,0]\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = K_2[R\,0]\begin{bmatrix} K_1^{-1}p_1 \\ 1 \end{bmatrix} = K_2RK_1^{-1}p_1 \tag{11}$$

So

$$H = K_2RK_1^{-1} \tag{12}$$

Question 1.3 (5pts)

From Section **Question 1.2**

$$H = KRK^{-1} \tag{13}$$

Because $\boldsymbol{H}$ is the homography that maps the view form one camera orientation to the view at a second orientation, there 3 situations for $\boldsymbol{R}$ when it fixes into xOy plane, yOz plane and zOx plane respectively.

1) When the rotation $\boldsymbol{R}$ is on xOy plane:

$$R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{14}$$

We can compute

$$
\begin{aligned}
H^2 &= KR_z(\theta)K^{-1}KR_z(\theta)K^{-1} \\
&= KR_z(\theta)R_z(\theta)K^{-1} \\
&= K \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} K^{-1} \\
&= K \begin{bmatrix} \cos^2\theta - \sin^2\theta & -2\sin\theta\cos\theta & 0 \\ 2\sin\theta\cos\theta & cos^2\theta - \sin^2\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} K^{-1} \\
&= K \begin{bmatrix} \cos(2\theta) & -\sin(2\theta) & 0 \\ \sin(2\theta) & \cos(2\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} K^{-1} \\
&= KR_z(2\theta)K^{-1}
\end{aligned} \tag{15}
$$

2) When the rotation $\boldsymbol{R}$ is on yOz plane:

$$
R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \tag{16}
$$

We can compute

$$
\begin{aligned}
H^2 &= KR_x(\theta)K^{-1}KR_x(\theta)K^{-1} \\
&= KR_x(\theta)R_x(\theta)K^{-1} \\
&= K \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} K^{-1} \\
&= K \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos^2\theta - \sin^2\theta & -2\sin\theta\cos\theta \\ 0 & 2\sin\theta\cos\theta & cos^2\theta - \sin^2\theta \end{bmatrix} K^{-1} \\
&= K \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(2\theta) & -\sin(2\theta) \\ 0 & \sin(2\theta) & \cos(2\theta) \end{bmatrix} K^{-1} \\
&= KR_x(2\theta)K^{-1}
\end{aligned} \tag{17}
$$

3) When the rotation $\boldsymbol{R}$ is on zOx plane:

$$
R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \tag{18}
$$

We can compute

$$
\begin{aligned}
H^2 &= KR^y(\theta)K^{-1}KR^y(\theta)K^{-1} \\
&= KR_y(\theta)R_y(\theta)K^{-1} \\
&= K \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} K^{-1} \\
&= K \begin{bmatrix} \cos^2\theta - \sin^2\theta & 0 & 2\sin\theta\cos\theta \\ 0 & 1 & 0 \\ -2\sin\theta\cos\theta & 0 & cos^2\theta - \sin^2\theta \end{bmatrix} K^{-1} \\
&= K \begin{bmatrix} \cos(2\theta) & \sin(2\theta) & 0 \\ 0 & 1 & 0 \\ -\sin(2\theta) & \cos(2\theta) & 0 \end{bmatrix} K^{-1} \\
&= KR_y(2\theta)K^{-1}
\end{aligned} \tag{19}
$$

To sum up of the 3 situations above, we can conclude

$$H^2 = KR(2\theta)K^{-1} \tag{20}$$

Therefore, $\boldsymbol{H^2}$ is the homography corresponding to a rotation of 2θ.

Question 1.4 (5pts)

Why is the planar homography not completely sufficient to map any arbitrary scene image to another viewpoint?

Between the subregions of the two images, there exists different homographies that can correspond the viewpoint on the subregions to the same planar.

Hence, it is not completely sufficient for the planar homography to map any arbitrary scene image to another viewpoint.

Question 1.5 (10pts)

We have a set of points $\boldsymbol{p_1^i}$ in an image taken by camera $\boldsymbol{C_1}$ and points $\boldsymbol{p_2^i}$ in an image taken by $\boldsymbol{C_2}$. Suppose we know there exists an unknown homography $\boldsymbol{H}$ such that

$$\boldsymbol{p_1^i \equiv H p_2^i} \tag{21}$$

Assume the points are homogeneous coordinates in the form $\boldsymbol{p_j^i = (x_j^i, y_j^i, 1)^T}$. For a single point pair, write a matrix equation of the form

$$\boldsymbol{Ah = 0} \tag{22}$$

Where $\boldsymbol{h}$ is a vector of the elements of $\boldsymbol{H}$ and $\boldsymbol{A}$ is a matrix composed of the point coordinates.

1) We already know that the homography $\boldsymbol{H}$ is a 3×3 matrix, and $\boldsymbol{h}$ is a vector of the elements of $\boldsymbol{H}$.

   Therefore, there are 9 elements in $\boldsymbol{h}$.

2) We need 4 point pairs to solve this system.

   Because if we have 4 point pairs $(\boldsymbol{p_1^i, p_2^i})$, i = 1, 2, 3, 4, we can obtain a system of 8 homogenouse equations with 9 unknowns $\boldsymbol{h_{ij}}$. The 9 column vectors of $\boldsymbol{H}$ cannot be linearly independent, which means that there exists a linear combination of $\boldsymbol{h_{ij}}$ that sums to be the zero vector.

   Hence, we can find $\boldsymbol{H}$ by finding the null space of this system, namely the 8×9 matrix $\boldsymbol{A}$.

3) We are thinking about the relation

$$p_1^i \equiv H p_2^i \tag{23}$$

   We let the coordinates equations of $\boldsymbol{p_1^i}$ degenerate to:

$$\begin{aligned} \boldsymbol{u} &= \boldsymbol{x_1^i} \\ \boldsymbol{v} &= \boldsymbol{y_1^i} \end{aligned} \tag{24}$$

   Make them linear:

$$\begin{aligned} H_1^T p_2 - (H_3^T p_1)u = 0 \\ H_2^T p_2 - (H_3^T p_1)v = 0 \end{aligned} \tag{25}$$

   Therefore, we can write them in matrix form:

$$\begin{bmatrix} p_2^T & 0 & -up_2^T \\ 0 & p_2^T & -vp_2^T \end{bmatrix} h = 0 \tag{26}$$

   When we put equation (19) to connect with N point pairs, we can rewrite matrix:

$$Ah = 0 \tag{27}$$

   as the equation:

$$\begin{bmatrix} (p_2^1)^T & 0 & -u(p_2^1)^T \\ 0 & (p_2^1)^T & -v(p_2^1)^T \\ \vdots & \vdots & \vdots \\ (p_2^i)^T & 0 & -u(p_2^i)^T \\ 0 & (p_2^i)^T & -v(p_2^i)^T \\ \vdots & \vdots & \vdots \\ (p_2^N)^T & 0 & -u(p_2^N)^T \\ 0 & (p_2^N)^T & -v(p_2^N)^T \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = 0 \qquad (i = 1, ..., N) \tag{28}$$

   Therefore, matrix A is a 2N × 9 matrix.

   If we square euqation (20), we can obtain:

$$|Ah|^2 = (Ah)^T Ah = h^T A^T Ah = h^T (A^T A) h \tag{29}$$

   Hence, matrix $\boldsymbol{A^T A}$ is a 2N × 2N matrix.

   And according to the Rayleigh Quotient Theorem, we can find $\boldsymbol{h}$ to minimize the homogeneous linear least squares system by constrainting $\boldsymbol{||h||^2 = 1}$. And $\boldsymbol{h^T(A^T A)h}$ will reach its absolute minimum when $\boldsymbol{h}$ is an eigenvector of $\boldsymbol{A^T A}$ corresponding to the smallest eigenvalue $\boldsymbol{\lambda_{min}}$.

2 Planar Homographies: Implementation (30pts)

Question 2.1 (15pts)

The steps of my `computeH.m` code in MATLAB:

(1) Get the number of points as N and set matrix A to the required size.

(2) Append p2 with a column vector of 1's to N $\times$ 3 matrix.

(3) In the loop from 1 to N, get the coordinates of p1 as u and v. And set the corresponding coordinates value in A.

(4) Compute the SVD of matrix $\boldsymbol{A}^T\boldsymbol{A}$ by using eig and get the minimal eigenvalue.

(5) Get the minimal eigenvector corresponding to the minimal eigenvalue and reshape H to the 3×3 matrix.

Question 2.2 (15pts)

a) Create an image with my name and picture on it. And I save this picture as pnc_tomap.jpg.

b) The minumum amount of point correspondences required to compute a homography is 4. I choose the corresponding 4 vertices of both the pnc.jpg and pnc_tomap.jpg. And I used the cpselect in `create_p1p2.m` to select the matching p1, p2 and save them in Q4.2.p1p2.mat.

c) The steps of my `warp2PNCpark.m` code in MATLAB:

   (1) Transpose p1 and p2 to targeted format and compute H with input p1, p2

   (2) Compute warped image by using warpH function

   (3) Get the mask for PNCpark image, whose region won't be overlaped by the warped image as well as the corresponding mask for warped image.

   (4) Overlap the PNCpark image with the warped image by mask.

d) After I run the provided `q42checker.m`, I can obtain the result as Figure 1.

3 Panoramas (30pts)

Question 3.1 (15pts)

The steps of my `q5_1.m` code as below:

(1) Obtain p1 and p2 according to pts as well as compute H and save it

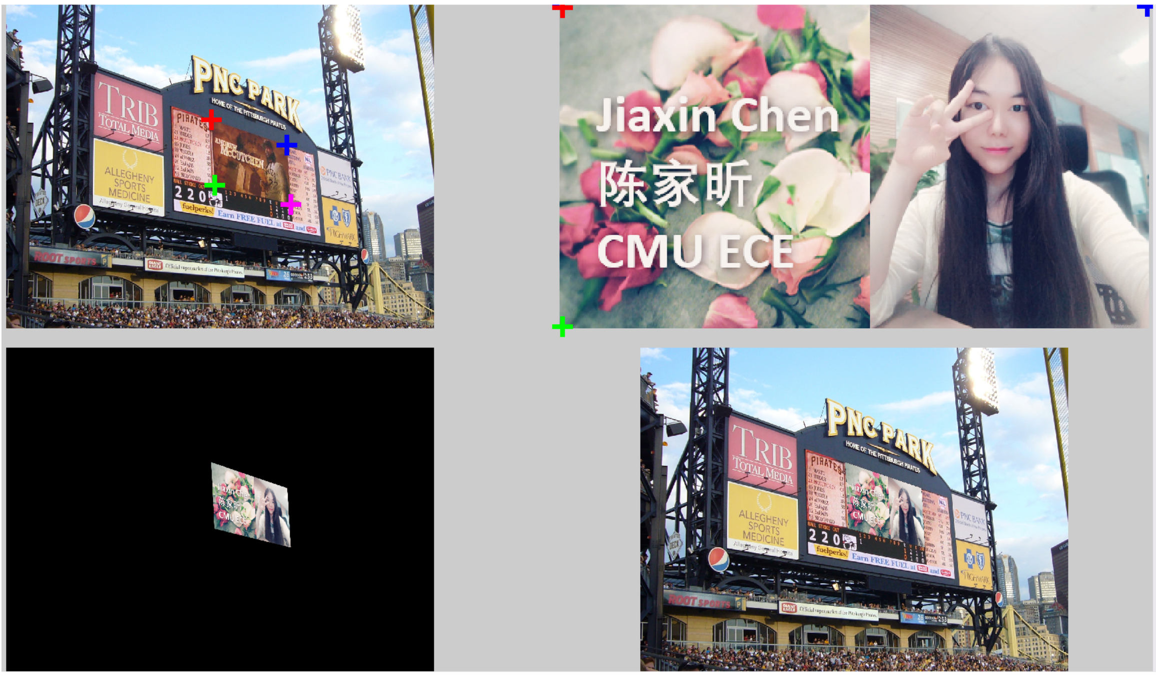
(2) Append p2 with the row of 1's and compute the corresponding p2 with homography and normalize it to calculate Root Mean Squared Error.

(3) Warp img2 and append the size of img1 corresponding to the size of warpedImg.

(4) Combine the parameters above into matrix M and warp img1 and img2 into the fixed size

(5) Get the mask of 3 channels for warpedImg, whose region won't be overlaped by the img1_append as well as the corresponding mask of 3 channels for img1_append.

Figure 1: q42checker output.



(6) Overlap the warpImg with the img1_append by mask.

The Root Mean Square Error between the corresponding points p1 and p2:

$$\begin{aligned} RMSE &= \sqrt{\frac{1}{N} \cdot \sum_{i=1}^{N} [(x_{p1}^{i} - x_{p2}^{i})^2 + (y_{p1}^{i} - y_{p2}^{i})^2]} \\ &= 0.85128 \end{aligned} \tag{30}$$

Consequently, the output of warpImg as Figure 2 and panoImg as Figure 3.

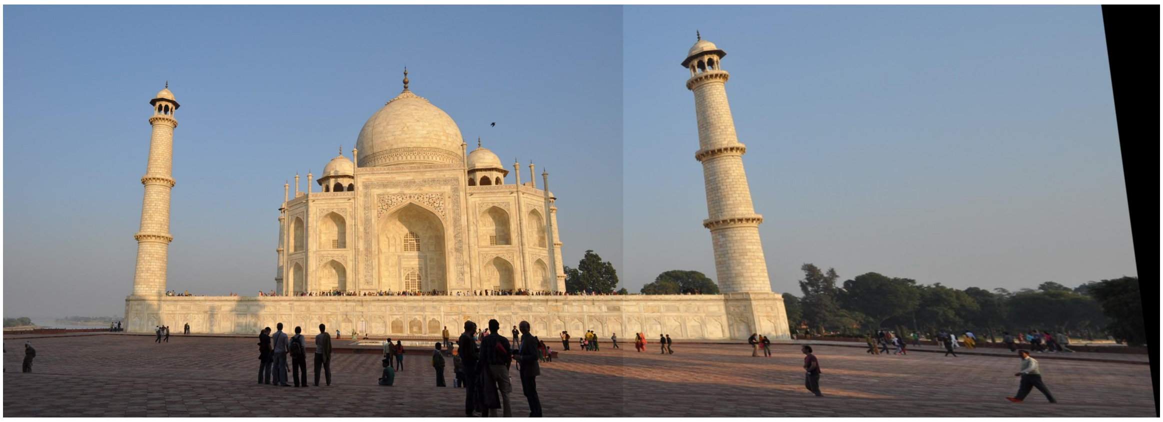
Figure 2: q5checker warpedImg output.

Question 3.2 (15pts)

In order to display all the visible pixels in the panorama, I write my `q5_2.m` code in the following steps:

(1) Obtain p1 and p2 according to pts as well as compute H and save it

Figure 3: q5checker panoImg1 output.



(2) Obtain the 4 vertices of img2 and compute the corresponding vertices of img2 with homograpy and normalize it.

(3) Compute the x and y extremnum for the initial size of the panoImg, and then convert the initial size into the fixed size to compute the scale alpha and belta

(4) Combine the parameters above into matrix M and warp img1 and img2 into the fixed size

(5) Append the size of img1 corresponding to the size of warpedImg. Get the mask of 3 channels for warpedImg2, whose region won't be overlaped by the warpedImg1_append, as well as the corresponding mask of 3 channels for warpedImg1_append

(6) Overlap the warpImg2 with the warpedImg1_append by mask.

Therefore, we can obtain the panoImg of all the visible pixels as Figure 4.

Figure 4: q5checker panoImg2 output.