16822 - Geometry based vision - Homework 1

Michael Jaison Gnanasekar
Andrew Id: mgnanase
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
mgnanase@andrew.cmu.edu

October 5, 2016

1 Problem 1

1.1 H for mirror reflection image

1.1.1

Let a point in world as P and its reflection around L as P'. Also, let R be the rotation that defines the reflection around the line L. ie. $P' = RP$

$$\begin{aligned} p &= K[I|0]P \\ p' &= K[I|0]P' \\ &= K[I|0]RP \\ &= K[I|0]RK^{-1}p \\ p' &= KRK^{-1}p \end{aligned}$$

Hence the homography exist as $H = KRK^{-1}$.

1.1.2

$$\begin{aligned} H^2 &= (KRK^{-1})(KRK^{-1}) \\ &= I \end{aligned}$$

Since R is orthogonal matrix, $H^2 = I$.

1.2 Fun with pencils of line

1.2.1

Slope of the line: $\frac{-u}{v}$

1.2.2

Any lines in the pencil of lines can be represented as a linear combination of two lines. Since we are dealing with projective 1-D space.

1.2.3

Dimension of H is 2x2. Since the projective space is 1D.

1.2.4

$$H = \begin{pmatrix} h_1 & h_2 & 0 \\ h_3 & h_4 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

By homography,

$$l' = H \begin{pmatrix} u \\ v \\ w \end{pmatrix}$$
$$l' = \begin{pmatrix} h_1 & h_2 & 0 \\ h_3 & h_4 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix}$$
$$= \begin{pmatrix} h_1 u + h_2 v \\ h_3 u + h_4 v \\ w \end{pmatrix}$$

The slope is given by $\frac{-u}{v}$.

$$s' = -\frac{h_1 u + h_2 v}{h_3 u + h_4 v}$$
$$= -\frac{h_1 \frac{u}{v} + h_2}{h_3 \frac{u}{v} + h_4}$$

Hence, it can be expressed in terms of slope s.

$$s' = \frac{h_1 s + h_2}{h_3 s + h_4}$$

1.2.5

The determinant of the matrix H should not be zero. The homography should be invertible. So, $ad - bc \neq 0$

1.3 Rodrigues's rotation formula (and some motion)

1.3.1 Matrix notation

$$R = I + sin(\theta)[w]_x + (1 - cos(\theta))[w]_x^2$$
$$[w]_x = W = \begin{pmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{pmatrix}$$

1.3.2

When θ is small, $sin(\theta)$ tends to θ, $cos(\theta)$ tends to $1 - \frac{\theta^2}{2}$. Hence, Rp is:

$$
\begin{aligned}
Rp &= Ip + \theta Wp + \frac{\theta^2}{2}W^2p \\
&= Ip + [\omega]_x p + \frac{[\omega]_x^2}{2}p \\
&= Ip + \Omega p + \frac{\Omega^2}{2}p
\end{aligned}
$$

1.3.3

Essestial matrix is given by,

$$
\begin{aligned}
E &= [t]_x R \\
&= [t]_x (I + \Omega + \frac{\Omega^2}{2})
\end{aligned}
$$

1.3.4

Epipole in one image is t. *i.e.* $e' = t$.

Epipole in other image is given by, $e = KR^T t$. Since $K = I$,

$$
\begin{aligned}
e &= R^T t \\
&= (I + \Omega + \frac{\Omega^2}{2})^T t \\
&= I^T t + \Omega^T t + \frac{(\Omega^2)^T}{2} t
\end{aligned}
$$

(since Ω is skew symmetric, $\Omega^T = -\Omega$)

$$
e = It - \Omega t + \frac{(\Omega^2)^T}{2} t
$$

Also, since the translation vector t is parallel to rotation axis w, $w \times t = 0$. Hence,

$$
e = t
$$

1.3.5

Projection of $[X\ Y\ Z\ 1]^T$ in Camera-1 is $[u\ v\ 1]^T$. Similarly, projection in Camera-2 is given by $[u'\ v'\ 1]^T$.

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = K[I|0] \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$Since\ K = I$$

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} X/Z \\ Y/Z \\ 1 \end{pmatrix}$$

Similarly,

$$\begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} = [R|t] \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

From the equation 9.7 from [**?**] Multiple View Geometry in Computer Vision book by Zisserman, the mapping from an image point p to an image point p' is given by

$$\begin{aligned}
p' &= K'RK^{-1}p + K't/Z \\
(since\ K = K' = I) \\
p' &= Rp + t/Z \\
\begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} &= (I + \Omega + \frac{\Omega^2}{2}) \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} + t/Z \\
\begin{pmatrix} u + du \\ v + dv \\ 1 \end{pmatrix} &= \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} + (\Omega + \frac{\Omega^2}{2}) \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} + t/Z \\
\begin{pmatrix} du \\ dv \\ 1 \end{pmatrix} &= (\Omega + \frac{\Omega^2}{2}) \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} + t/Z
\end{aligned}$$

From above equation, du and dv are expressed as a function of u, v, Ω, t, and Z.

1.3.6

Scene plane homography is defined as

$$H = R - \frac{tn^T}{d}$$

Since the point P is on a plane (n, d), the point in camera-2 p' is defined as

$$\begin{aligned}
p' &= Hp \\
\begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} &= H \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \\
&= (R - \frac{tn^T}{d}) \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \\
&= (I + \Omega + \frac{\Omega^2}{2} - \frac{tn^T}{d}) \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \\
\begin{pmatrix} u + du \\ v + dv \\ 1 \end{pmatrix} &= \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} + (\Omega + \frac{\Omega^2}{2} - \frac{tn^T}{d}) \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \\
\begin{pmatrix} du \\ dv \\ 1 \end{pmatrix} &= (\Omega + \frac{\Omega^2}{2} - \frac{tn^T}{d}) \begin{pmatrix} u \\ v \\ 1 \end{pmatrix}
\end{aligned}$$

From the above equation, it is shown that du and dv does not depend on the depth Z.

1.3.7

The expression of du and dv is given by,

$$\begin{pmatrix} du \\ dv \\ 1 \end{pmatrix} = (\Omega + \frac{\Omega^2}{2} - \frac{tn^T}{d}) \begin{pmatrix} u \\ v \\ 1 \end{pmatrix}$$

1.3.8

Scene plane homography is defined as

$$\begin{aligned}
H &= R - \frac{tn^T}{d} \\
&= I + \Omega + \frac{\Omega^2}{2} - \frac{tn^T}{d}
\end{aligned}$$

1.3.9

Since the robot is moving on ground plane, Ω and t is orthogonal to n.
To find the point that is invariant to H, we need to find the point for which H is I.
Assuming that H is I,

$$\begin{aligned}
I &= I + \Omega + \frac{\Omega^2}{2} - \frac{tn^T}{d} \\
\Omega + \frac{\Omega^2}{2} - \frac{tn^T}{d} &= 0
\end{aligned}$$

From the above equation, it is clear that we are interested in the null space of the above matrix.
The rank of last term $\frac{tn^T}{d}$ is 1, and t is in the null space of tn^T.
And since we have proved $e = t$, the point that is invariant to H is e. The epipole.

2 Problem 2

Image rectification is split into two sub problems.

- Affine Rectification
- Euclidean Rectification

2.1 Affine Rectification

Algorithm.

- In the input image, two pairs of parallel lines are annotated.
- Find the intersection of parallel lines at point at ∞ (p_∞).
- Find the line at ∞ (L_∞) connecting two points at ∞ as shown in Figure 1. $L_\infty = [l_1 \ l_2 \ l_3]^T$.
- Create the Homography as

$$H = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ l_1 & l_2 & l_3 \end{pmatrix}$$

- Apply warping to the image by the homography H.

2.2 Euclidean Rectification

Algorithm:

- Annotate two perpendicular lines in the affine transformed image from the previous section as shown in 1
- Form the below equation, we should solve for the unknowns.

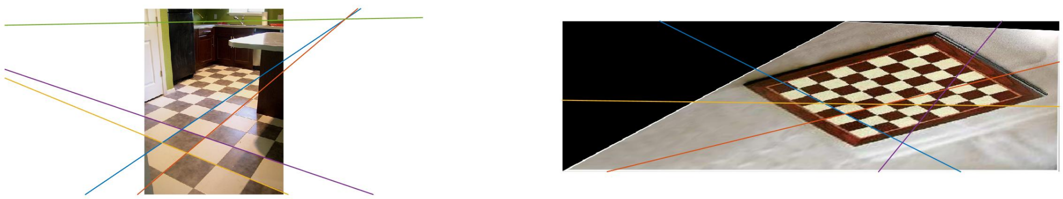
$$l_1^T H^{-1} C^* H^{-T} l_2 = 0$$

- As shown in the class, it can be solved by combining $L = AA^T$ and solving L.
- L can be solved by $L = UDU^T$, where $U^{-1} = U^T$. Because L is symmetric.
- Now, the original matrix A can be obtained by $A = U\sqrt{D}U^T$.
- From this, the homography matrix H can be framed as

$$H = \begin{pmatrix} AA^T & 0 \\ 0 & 0 \end{pmatrix}$$

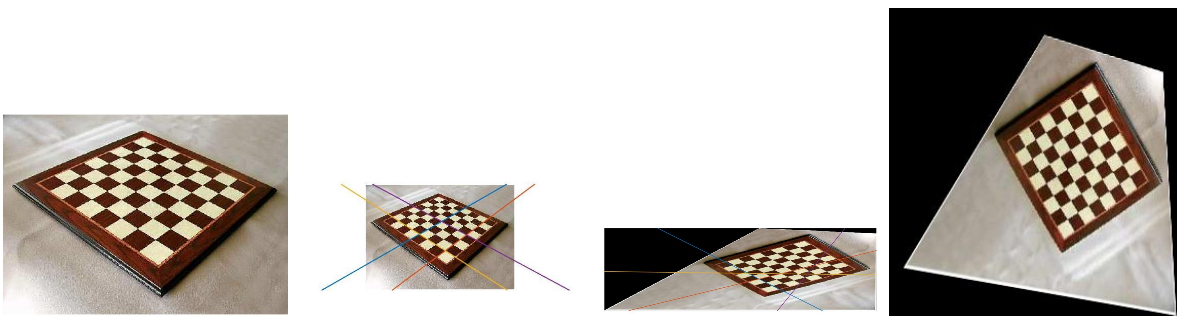
2.3 Problems Encountered

- Since the transformation depends on the parallel and perpendicular lines annotation, even a small error in the annotation gave a wrong transformation.
- While choosing the perpendicular lines, if the two pairs of perpendicular lines are in the same direction, the system did not solve.
- Numerical issues were handled smoothly if the annotation on image is precise.



(a) Visualization of Line at ∞ (b) Annotated perpendicular lines

Figure 1: An example of images in the pipeline

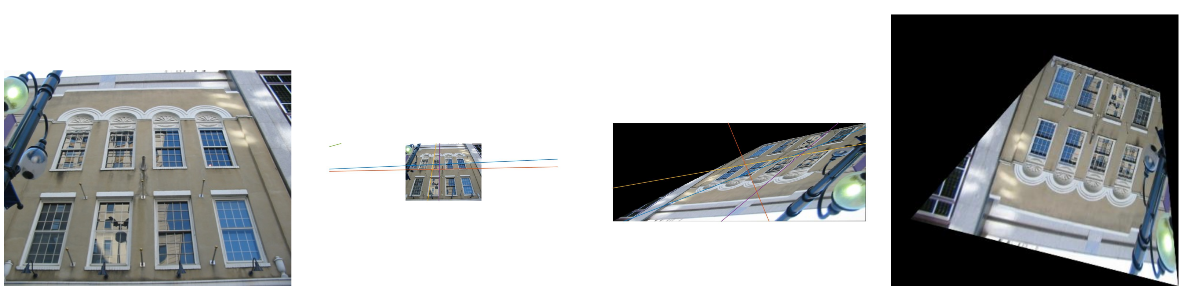


(a) chess1.jpg

2.4 Angles

2.4.1 Figure: chess1.jpg

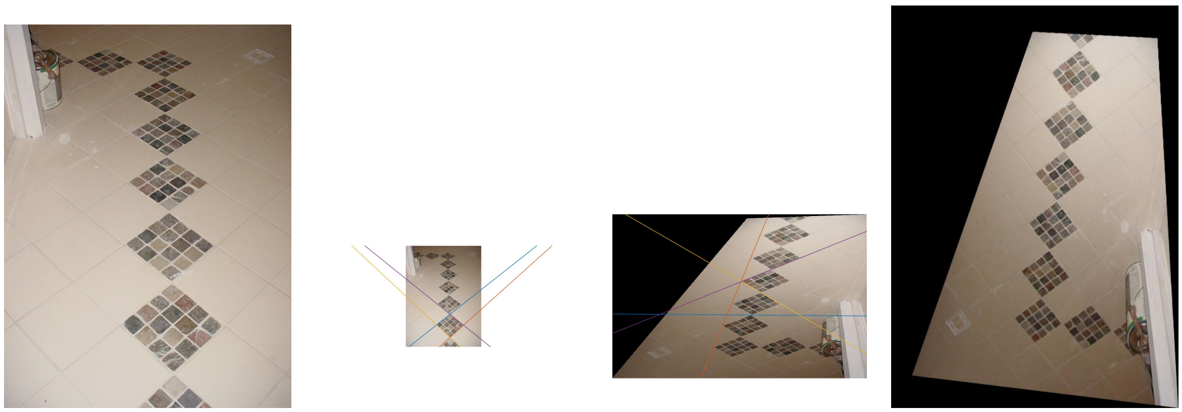
Before	After
0.7638856	0
-0.6249080	-1.1259619E-16
-0.5341788	1.7515324E-01



(b) facade.jpg

2.4.2 Figure: facade.jpg

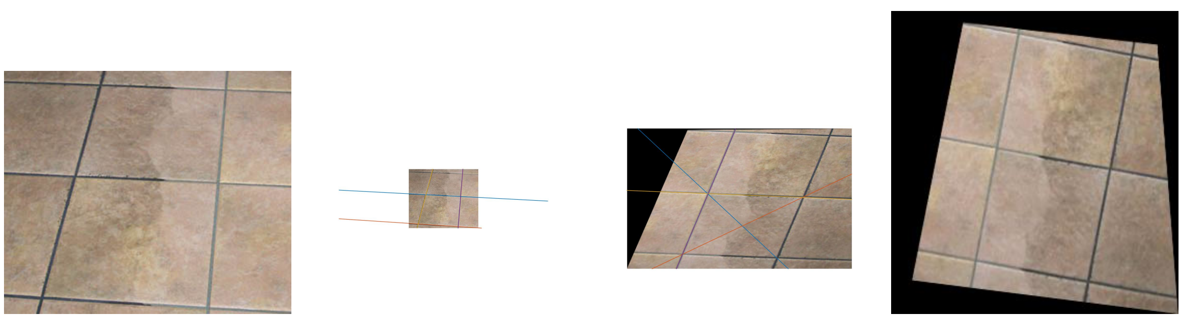
Before	After
0.0246178	-4.0250391E-16
0.8654985	-1.9446465E-16
-0.2569672	-1.1198973E-01



(c) tiles3.jpg

2.4.3 Figure: tiles3.jpg

Before	After
-0.3762770	7.8206839E-17
0.6189759	1.0455350E-16
0.6045568	-3.5030400E-02



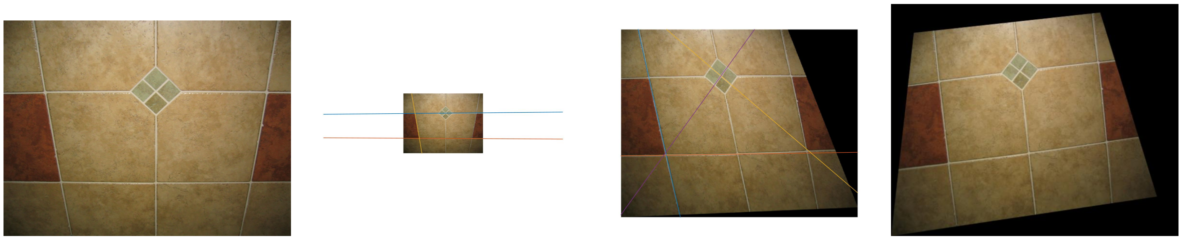
(d) tiles4.jpg

2.4.4 Figure: tiles4.jpg

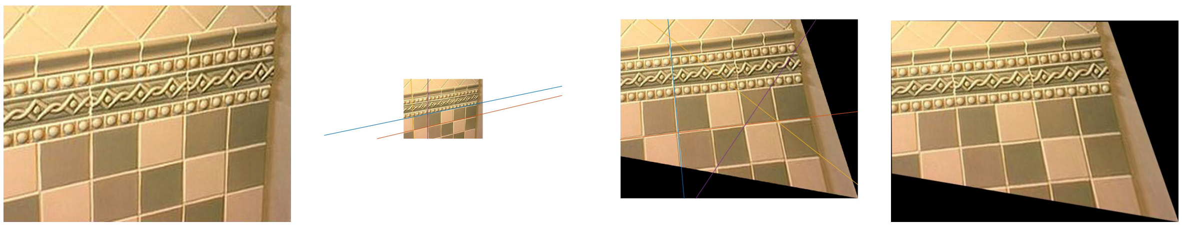
Before	After
0.3704599	2.7226264E-16
0.3541860	2.3781163E-16
-0.3501436	1.5342336E-03

2.4.5 Figure: tiles5.jpg

Before	After
0.2050029	-1.0660826E-16
-0.0816590	0.0000000E+00
0.2108932	5.4974449E-03



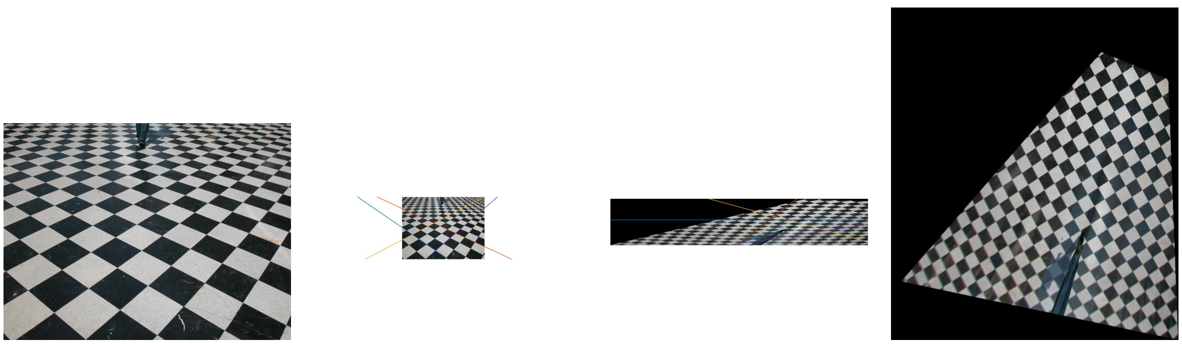
(e) tiles5.jpg



(f) tiles6.jpg

2.4.6 Figure: tiles6.jpg

Before	After
-0.0258799	-1.5895978E-16
-0.0676860	-6.9942619E-16
0.0794271	8.8854489E-03



(g) checker1.jpg

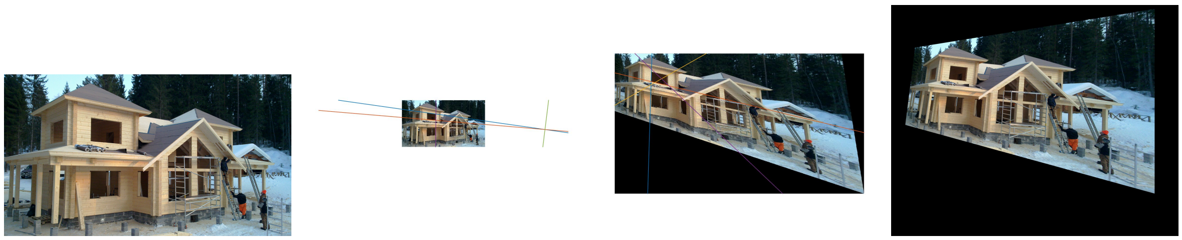
2.4.7 Figure: checker1.jpg

Before	After
-0.7961622	4.1091966E-17
0.9169904	-2.7865405E-16
-0.7672571	1.2124164E-01

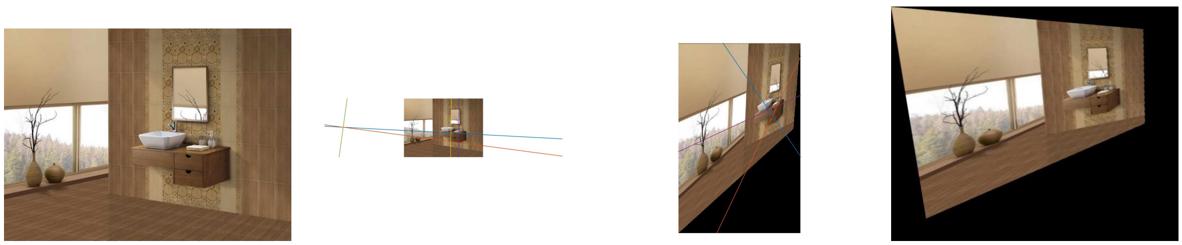
2.5 Own Images

2.5.1 Figure: web-house.jpg

Before	After
-0.2038131	-3.6498809E-17
-0.2921571	-4.2676765E-16
-0.1917144	2.7471455E-02



(h) web-house.jpg



(i) web2.jpg

2.5.2 Figure: web2.jpg

Before	After
-0.5004252	7.8058757E-17
0.4175002	0.0000000E+00
0.4084644	8.2416027E-02

(j) web-house2.jpg

2.5.3 Figure: web-house2.jpg

Before	After
0.3003940	-1.2550527E-16
-0.4394226	-8.6467521E-17
0.3325585	8.2952982E-02