09/17/2014 Ting Zhang zhan1013@purdue.edu

ECE 661: Homework 3

Fall 2014

Solutions

1. **Two-step Projective and Affine Distortion Removal**
<u>Answer:</u>

<u>1.1 First step: remove projective distortion.</u>

To remove projective distortion, the vanishing line need to be project to infinity.

To calculate the vanishing line in an image, we first find two sets of lines in the image, which are parallel in the world coordinates. Compute the join point of each set of lines, which is known as the ideal point. If the four points are represented as x_1, x_2, x_3 and x_4, then the two sets of lines can be calculated following

$$l_i = x_p \times x_q \, . \tag{1}$$

The ideal points can be calculated following

$$P = l_1 \times l_2 \, , \tag{2}$$

$$Q = l_3 \times l_4 \, , \tag{3}$$

assuming $l_1' // l_2'$ and $l_3' // l_4'$ in world coordinates.

Then, the vanishing line can be calculated by compute the line that goes through both ideal points.

$$VL = P \times Q \, . \tag{4}$$

If a vanish line is represented as $VL = \begin{pmatrix} l_1 & l_2 & l_3 \end{pmatrix}^T$, then the Homography that maps this vanish line to infinity can be represented as

$$H_p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{bmatrix} . \tag{5}$$

Apply H_p to the original image, we can get the image free of projective distortion.
The vanishing lines for two images of the same scene do not need to be identical. In most cases, they should be different because different photo-taking angles can give different distortions. Therefore, the projective distortion in one image cannot be corrected by using

the vanishing line from another image of the same scene. Different vanishing lines lead to different Homography according to equation (5).

1.2 Second step: remove affine distortion.

After the projective distortion is removed from the original image, the image contains affine distortion. To remove affine distortion, we use two sets of orthogonal lines.

Assume we have two lines $l = \left(l_1 \quad l_2 \quad l_3\right)^T$ and $m = \left(m_1 \quad m_2 \quad m_3\right)^T$, that are orthogonal to each other in world coordinates.

Since we have

$$\cos(\theta) = \frac{l_1 m_1 + l_2 m_2}{\sqrt{\left(l_1^2 + l_2^2\right)\left(m_1^2 + m_2^2\right)}}, \tag{6}$$

and according to the dual representation of degenerate conic, equation (6) can be rewritten as

$$\cos(\theta) = \frac{l^T C_\infty^* m}{\sqrt{\left(l^T C_\infty^* l\right)\left(m^T C_\infty^* m\right)}}, \tag{7}$$

where

$$C_\infty^* = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Since $C'^* = HC^*H^T$, the numerator of equation (7) can be written as

$$\cos(\theta)\Big|_{numerator} = l'^T C_\infty^{*\prime} m'. \tag{8}$$

Since *l'* and *m'* are orthogonal to each other, then equation (8) equals to 0 because $\cos(90°) = 0$. Then equation (8) can be written as

$$l^T H_a C_\infty^* H_a^T m = \left(l_1 \quad l_2 \quad l_3\right) \begin{bmatrix} AA^T & \vec{0} \\ \vec{0}^T & 0 \end{bmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = 0, \tag{9}$$

where

$$H_a = \begin{bmatrix} A & \vec{0} \\ \vec{0}^T & 1 \end{bmatrix}. \tag{10}$$

Define $S = AA^T$ (11) , equation (9) can be written as

$$\begin{pmatrix} l_1 & l_2 \end{pmatrix} \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} \begin{pmatrix} m_1 \\ m_2 \end{pmatrix} = 0.$$

Since S is symmetric, equation (9) can be formulated as

$$s_{11} l_1 m_1 + s_{12} \left(l_1 m_2 + l_2 m_1 \right) + s_{22} l_2 m_2 = 0. \tag{12}$$

To write equation (12) in a matrix representation, we have

$$\begin{pmatrix} l_1 m_1 & l_1 m_2 + l_2 m_1 \end{pmatrix} \begin{pmatrix} s_{11} \\ s_{12} \end{pmatrix} = -m_2 l_2 \tag{13}$$

Therefore, we need two sets of orthogonal lines to get the values of S. According to equation (11), A can be computed by the SVD decomposition, following

$$A = VDV^T, \tag{14}$$

$$S = VD^2V^T. \tag{15}$$

2. One-step Projective and Affine Distortion Removal

<u>Answer:</u>

Assume we have two lines $l = \begin{pmatrix} l_1 & l_2 & l_3 \end{pmatrix}^T$ and $m = \begin{pmatrix} m_1 & m_2 & m_3 \end{pmatrix}^T$, that are orthogonal to each other in world coordinates, equation (8) equals 0. Also, according to the general form of $C_\infty^{*\prime}$,

$$C_\infty^{*\prime} = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix}, \tag{16}$$

equation (8) can be written as

$$\begin{pmatrix} l_1 & l_2 & l_3 \end{pmatrix} \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix} \begin{pmatrix} m_1 \\ m_2 \\ m_3 \end{pmatrix} = 0. \tag{17}$$

To get the values in $C_\infty^{*\prime}$, we can rewrite equation (17) as

$$\begin{pmatrix} l_1m_1 & (l_1m_2+l_2m_1)/2 & l_2m_2 & (l_1m_3+l_3m_1)/2 & (l_2m_3+l_3m_2)2 & l_3m_3 \end{pmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = 0 .$$

After we solve this equation, the values of A can be computed through a SVD decomposition according to

$$C_\infty^{*\prime} = \begin{bmatrix} AA^T & Av \\ v^T A^T & v^T v \end{bmatrix}. \quad (18)$$

Since H is defined as

$$H = \begin{bmatrix} A & \vec{0} \\ v^T & 1 \end{bmatrix}, \quad (19)$$

the value of v need to be computed as well according to equation (18).
By applying the H defined in equation (19), both projective and affine distortion can be removed from the original image.

3. Comparison between Two-step and One-step Methods

Answer:

In overall performance, I think the two-step method is more robust than the one-step method.

First, the most crucial point is that it's proven that the one-step method needs more attempts to select the five sets of orthogonal lines that are suitable for distortion removal. Failing to do so can lead to unexpected and unsuccessful results. However, in this aspect, the two-step method only needs four points and two sets of orthogonal lines in total. Also, the correctness of results is not so dependent on the selection of those points and lines. Sometimes, the two sets of orthogonal lines can also be consisted of the four points selected. Then only 6 points are needed as inputs in this case.

Second, when considering implementation, although the two-step methods generated more lines of code than the one-step method, it is more straightforward, especially the first step. It is easy to implement and debug, which decreases the possibility of errors. However, when it comes to the one-step method, it needs two SVD decompositions and complicated matrix operations, which is possible to cause more errors. This is true when I was programming.

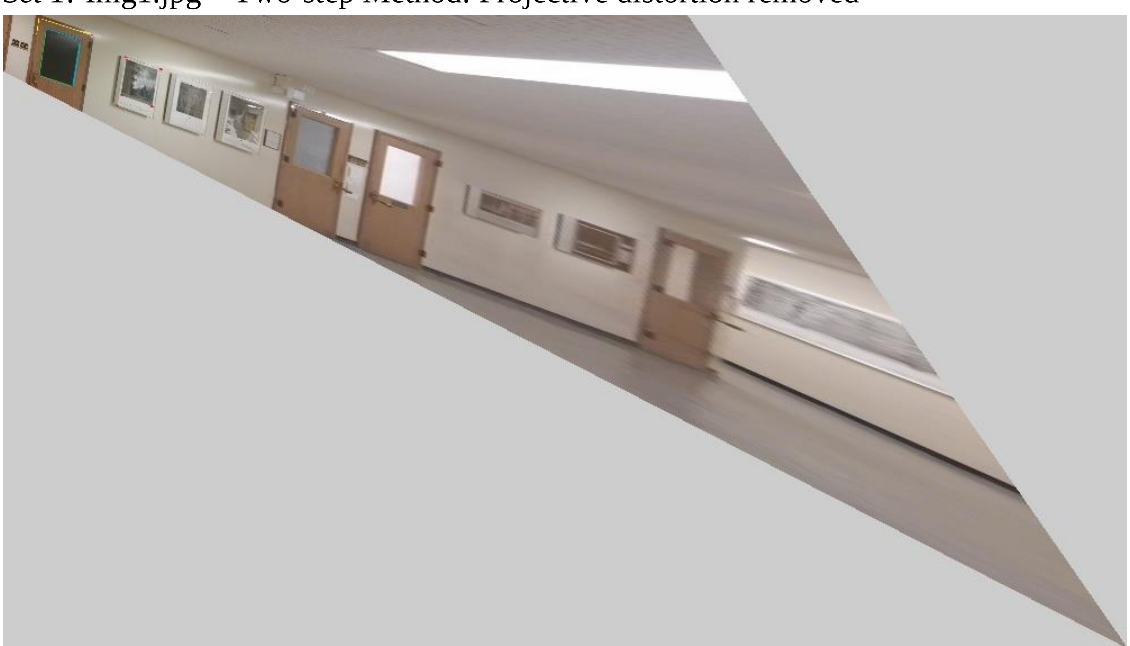
As a conclusion, the two-step method is more robust than the one-step method in both accuracy and effectiveness aspect.

Appendix A. Original and Result Photos

The two sets of orthogonal lines used in two-step method is shown as blue and yellow. The rectangle shown in blue and green is the one used to compute vanishing line.

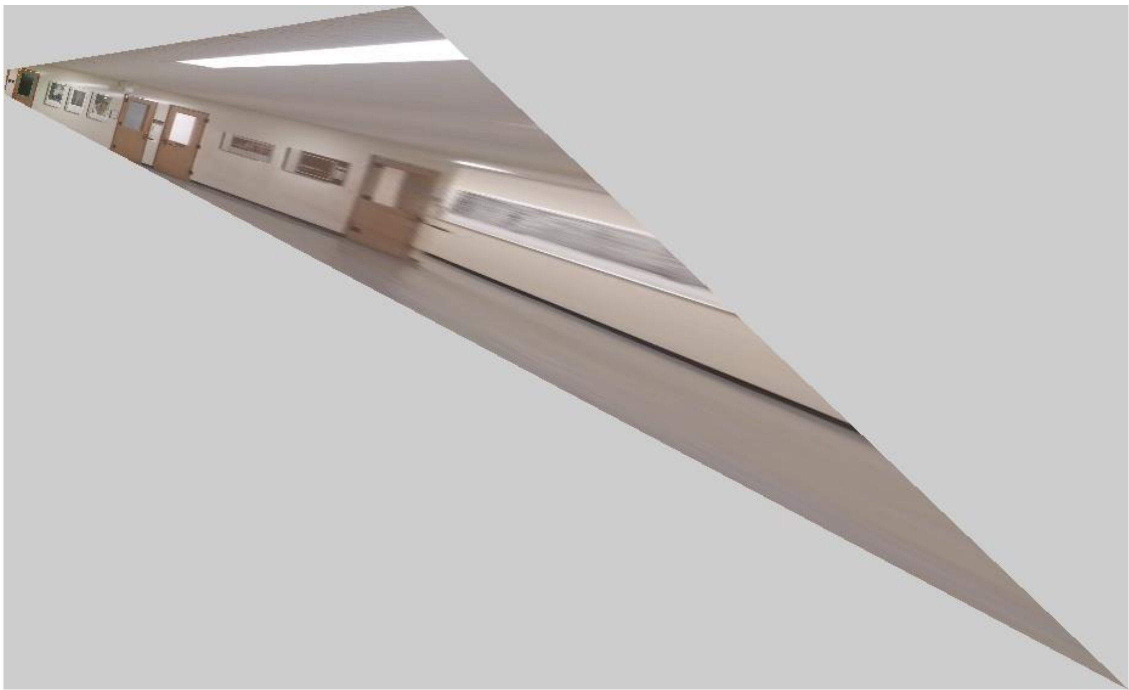
- *Set 1:* Img1.jpg – Original

- *Set 1:* Img1.jpg – Two-step Method: Projective distortion removed



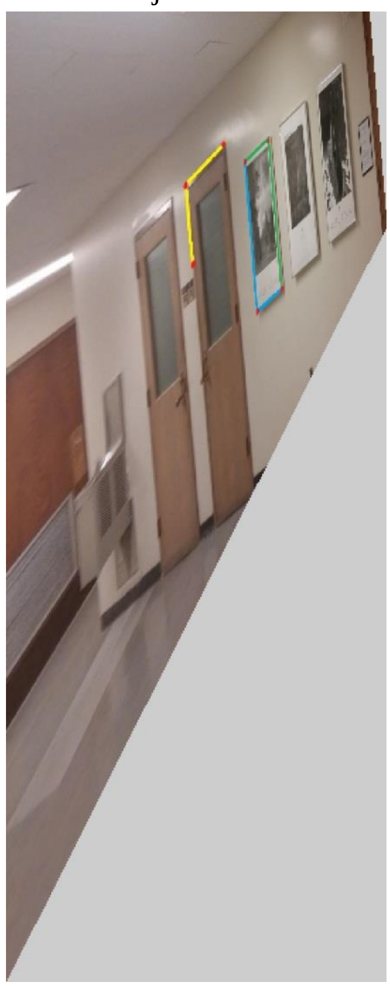
- *Set 1:* Img1.jpg – Two-step Method: Affine distortion removed

- *Set 1:* Img1.jpg – One-step Method: Projective and affine distortion removed



- *Set 1:* Img2.jpg – Original

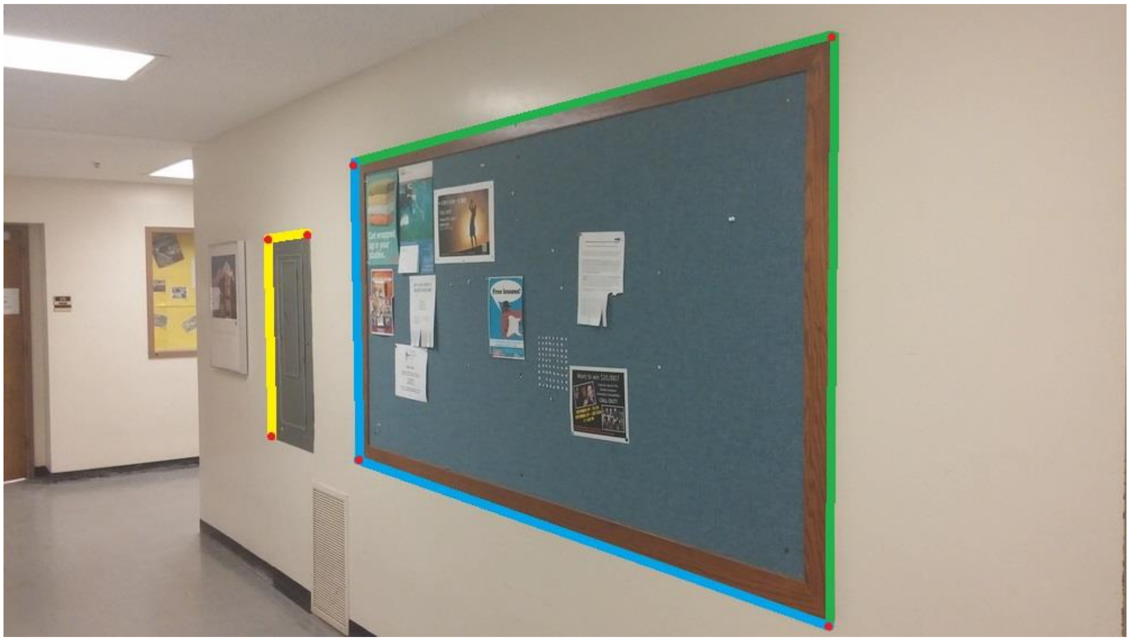
- *Set 1:* Img2.jpg – Two-step Method: Projective distortion removed



- *Set 1:* Img2.jpg – Two-step Method: Affine distortion removed

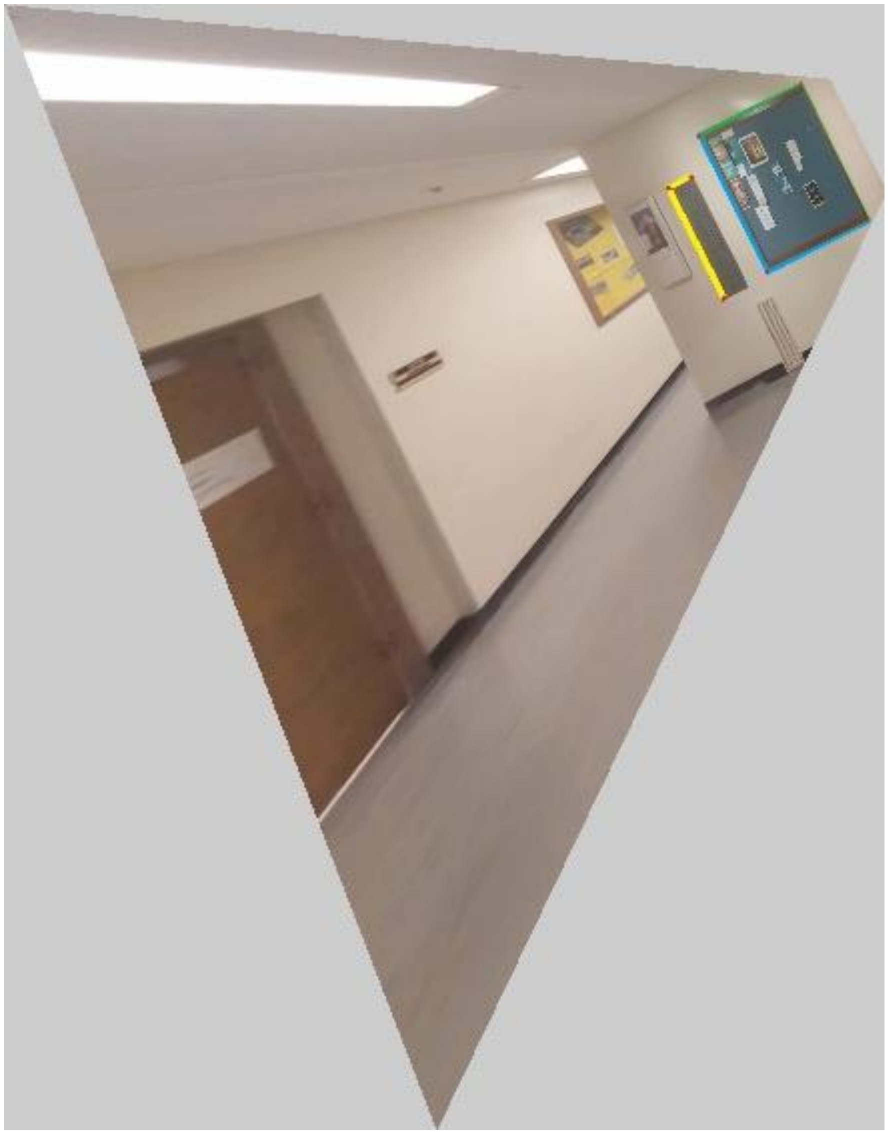
- *Set 1:* Img2.jpg – One-step Method: Projective and affine distortion removed

- *Set 2:* Img1.jpg – Original

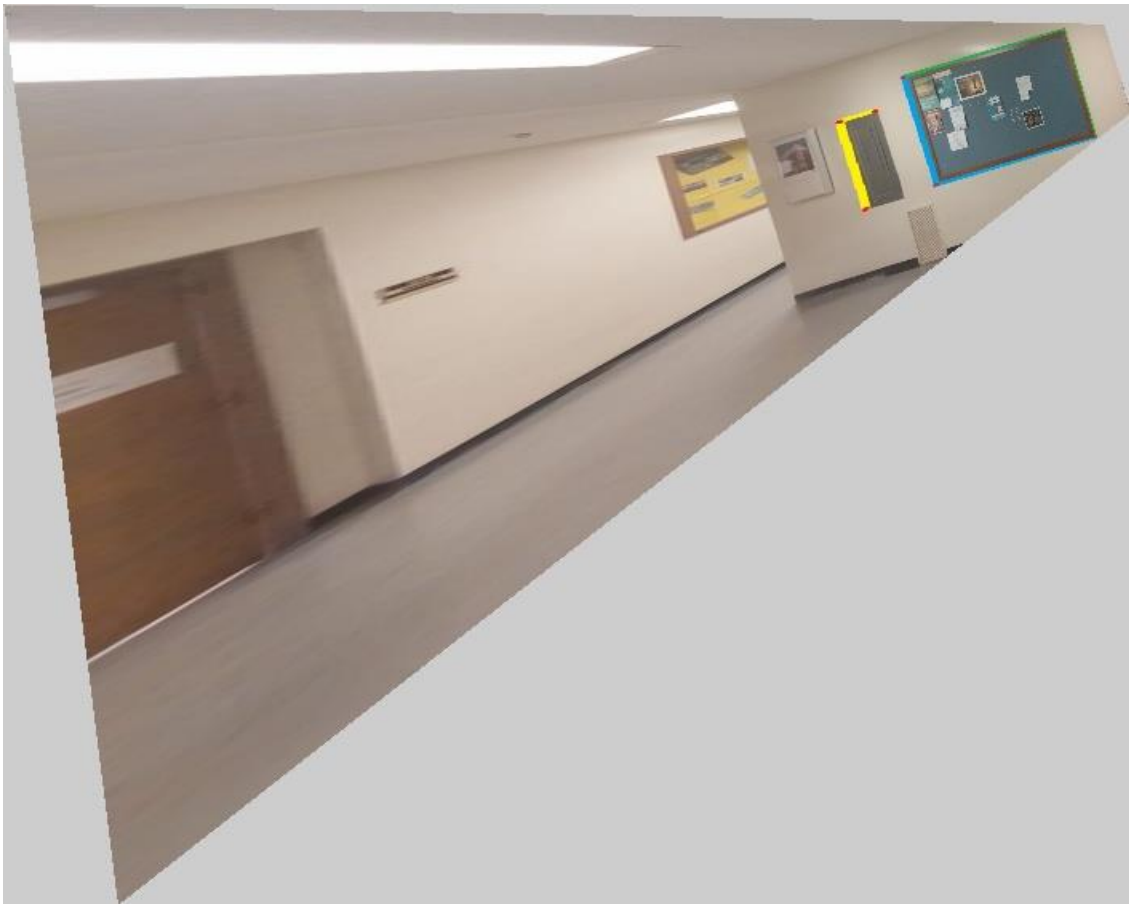


- *Set 2:* Img1.jpg – Two-step Method: Projective distortion removed

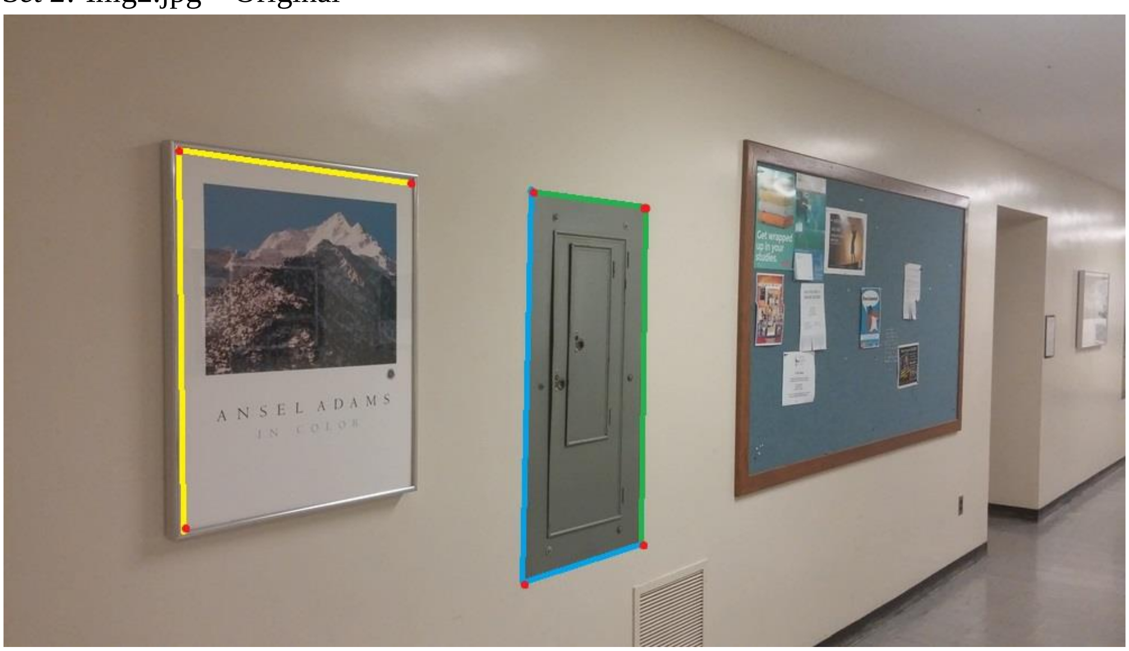
- *Set 2:* Img1.jpg – Two-step Method: Affine distortion removed



- *Set 2:* Img1.jpg – One-step Method: Projective and affine distortion removed

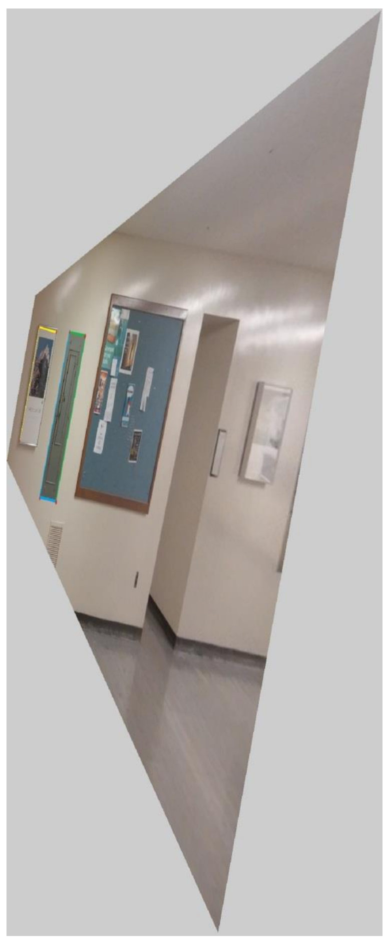


- *Set 2:* Img2.jpg – Original



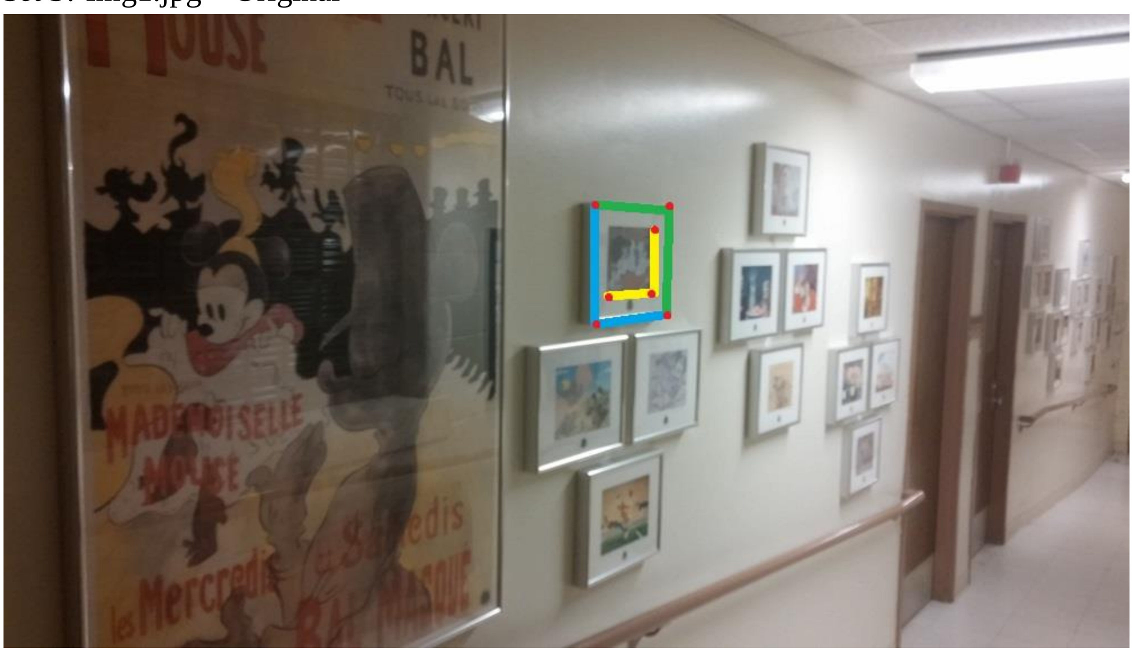
- *Set 2:* Img2.jpg – Two-step Method: Projective distortion removed

- *Set 2:* Img2.jpg – Two-step Method: Affine distortion removed



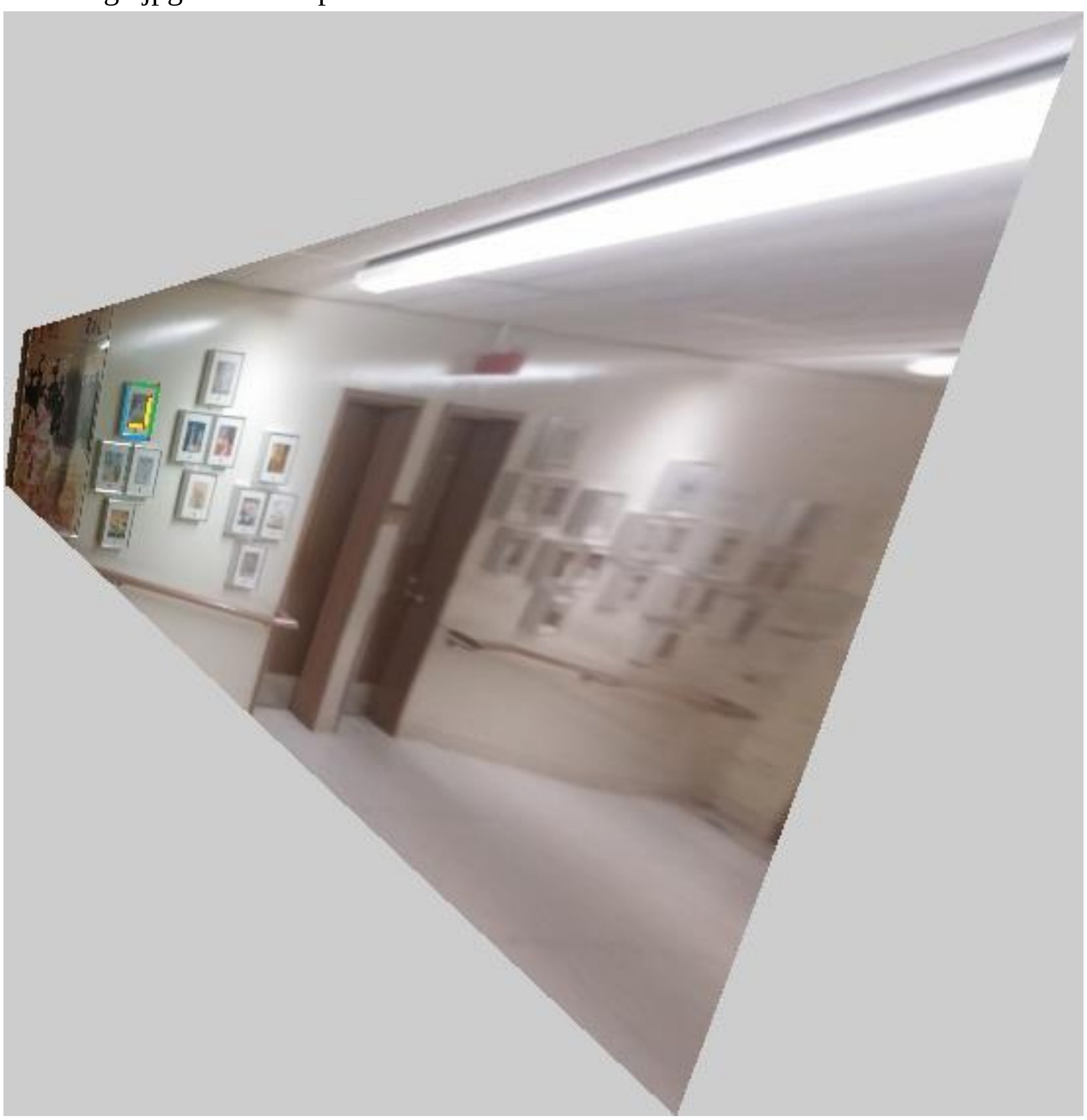
- *Set 2:* Img2.jpg – One-step Method: Projective and affine distortion removed

- *Set 3:* Img1.jpg – Original



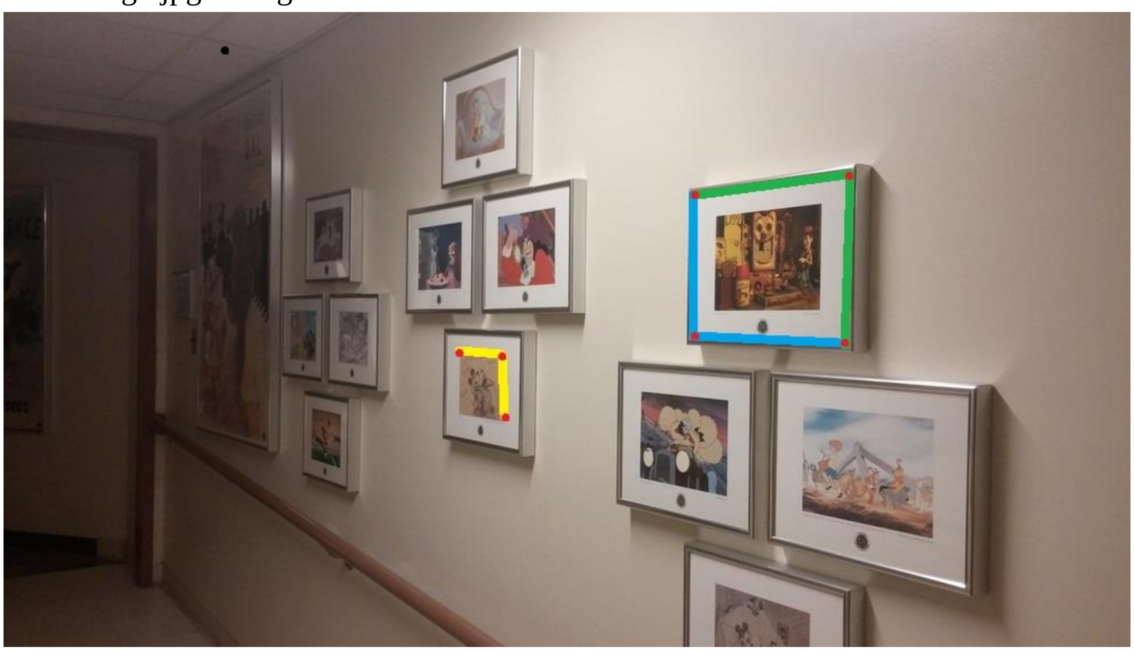
- *Set 3:* Img1.jpg – Two-step Method: Projective distortion removed

- *Set 3:* Img1.jpg – Two-step Method: Affine distortion removed

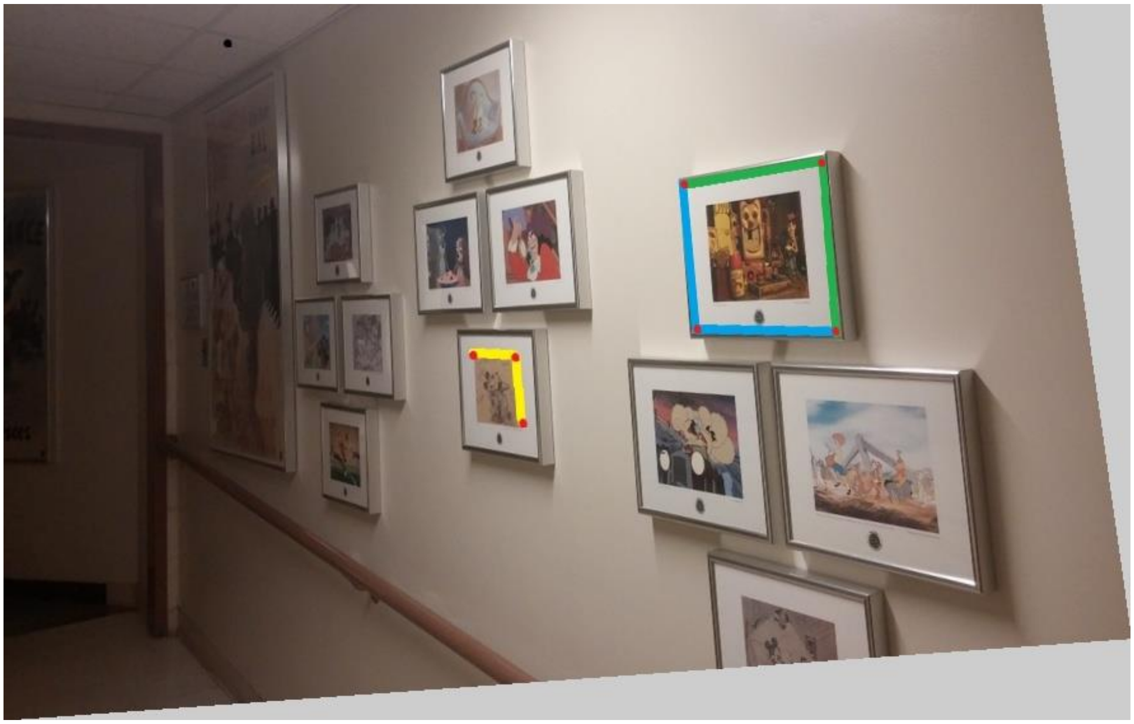


- *Set 3:* Img1.jpg – One-step Method: Projective and affine distortion removed

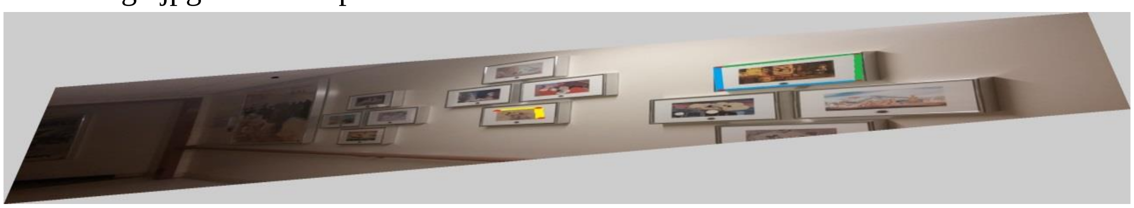
- *Set 3:* Img2.jpg – Original



- *Set 3:* Img2.jpg – Two-step Method: Projective distortion removed



- *Set 3:* Img2.jpg – Two-step Method: Affine distortion removed

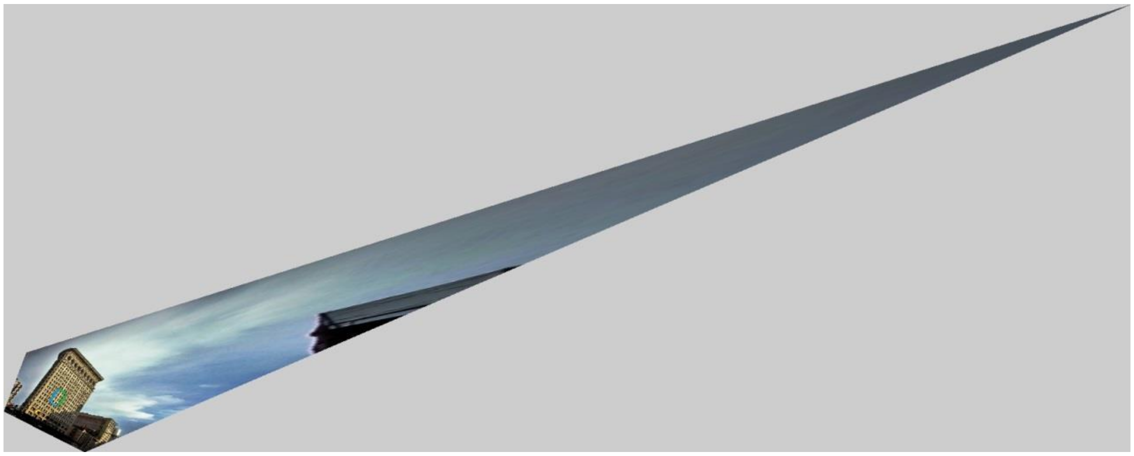


- *Set 3:* Img2.jpg – One-step Method: Projective and affine distortion removed

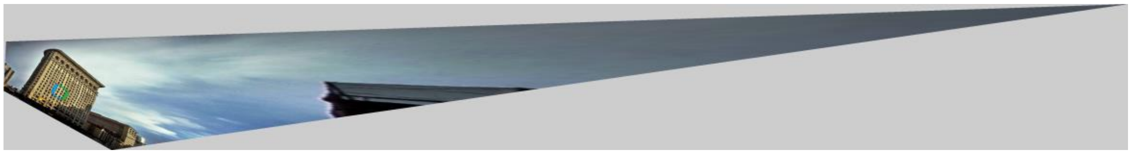
- *Set 4:* Img1.jpg – Original

- *Set 4:* Img1.jpg – Two-step Method: Projective distortion removed

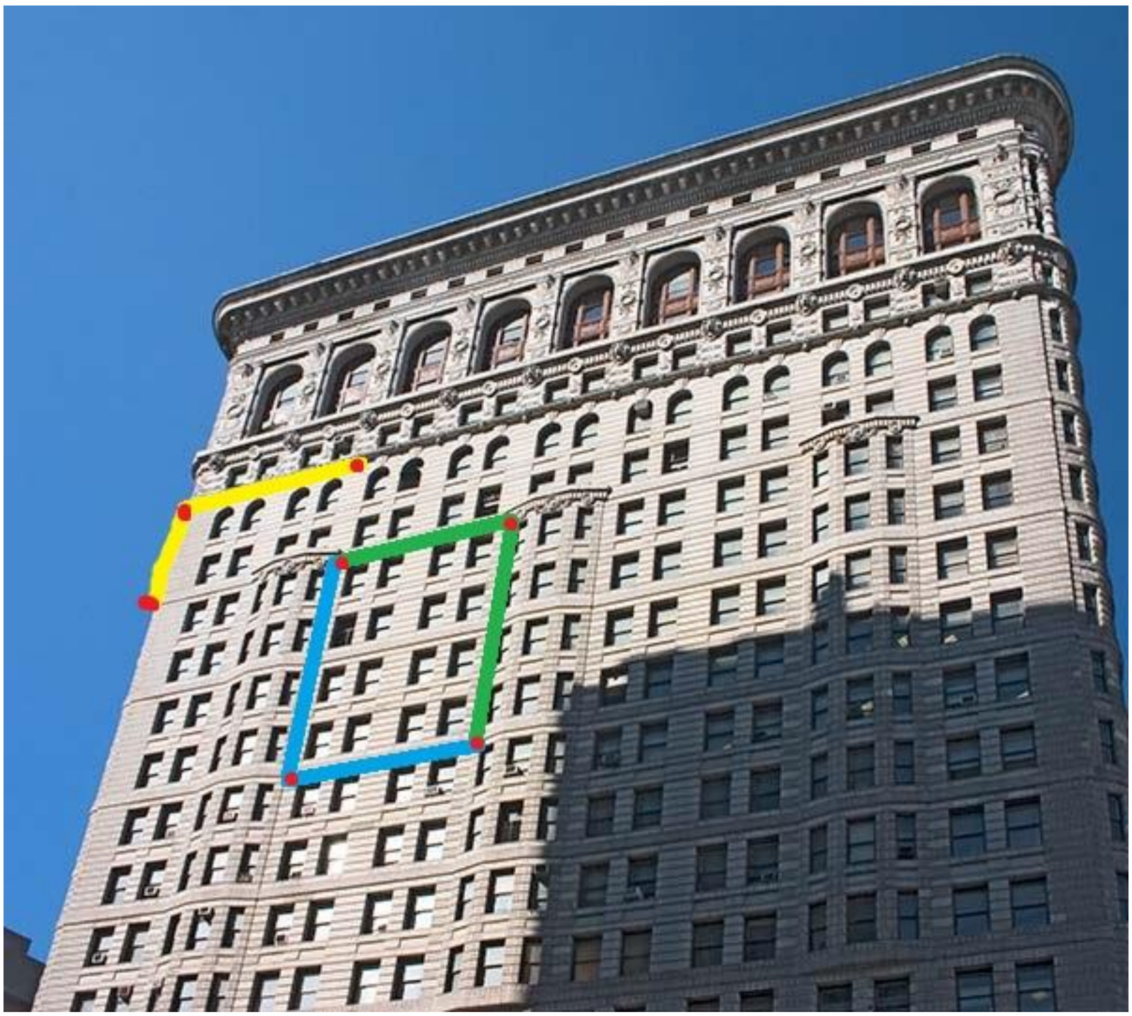
- *Set 4:* Img1.jpg – Two-step Method: Affine distortion removed



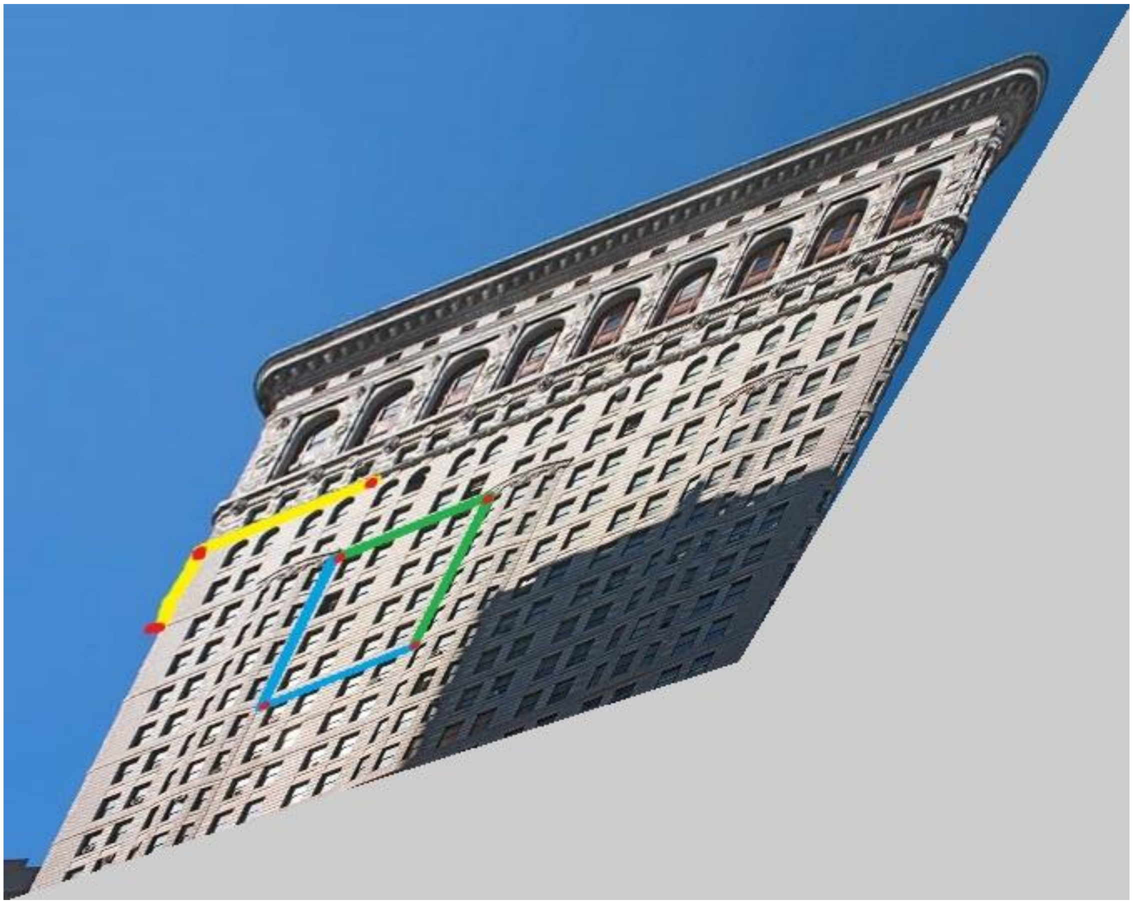
- *Set 4:* Img1.jpg – One-step Method: Projective and affine distortion removed



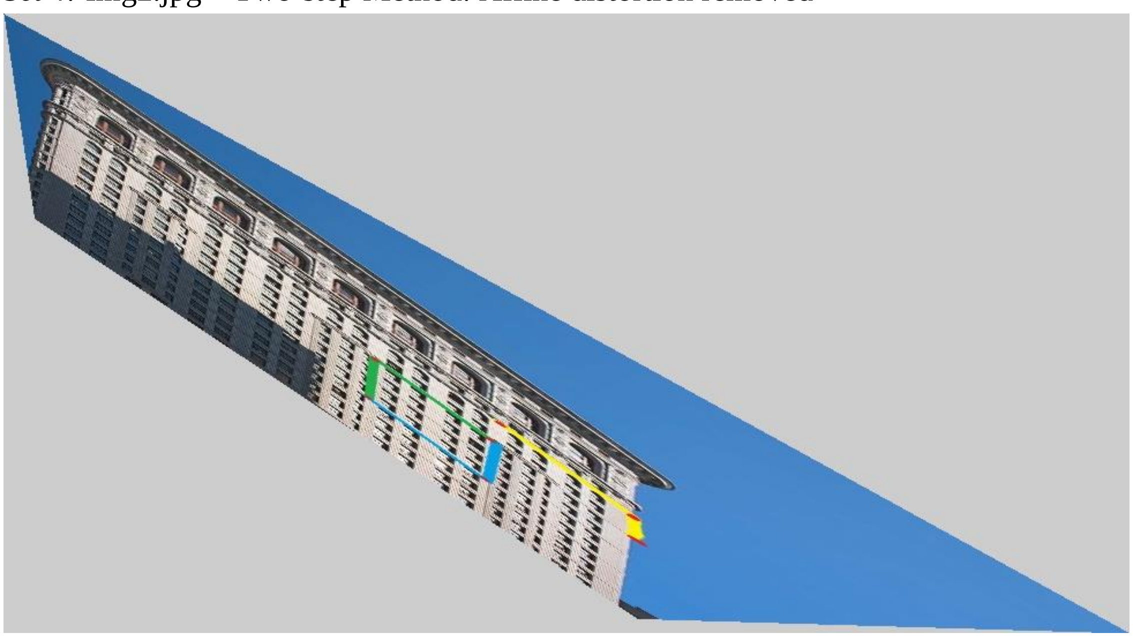
- *Set 4:* Img2.jpg – Original



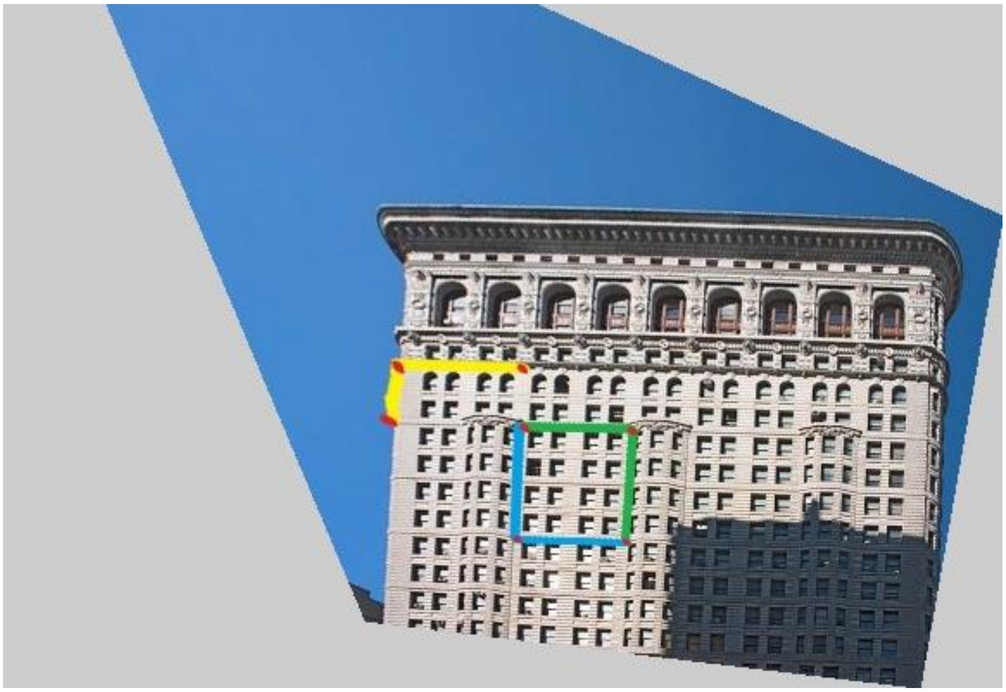
- *Set 4:* Img2.jpg – Two-step Method: Projective distortion removed



- *Set 4:* Img2.jpg – Two-step Method: Affine distortion removed



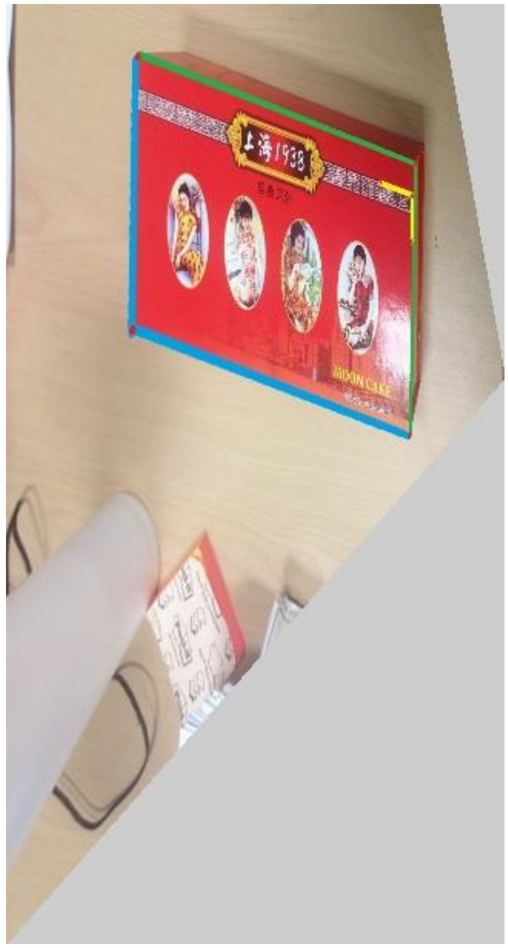
- *Set 4:* Img2.jpg – One-step Method: Projective and affine distortion removed



<u>*Additional photos taken by myself*</u>

- Img1.jpg – Original

- Img1.jpg – Two-step Method: Projective distortion removed



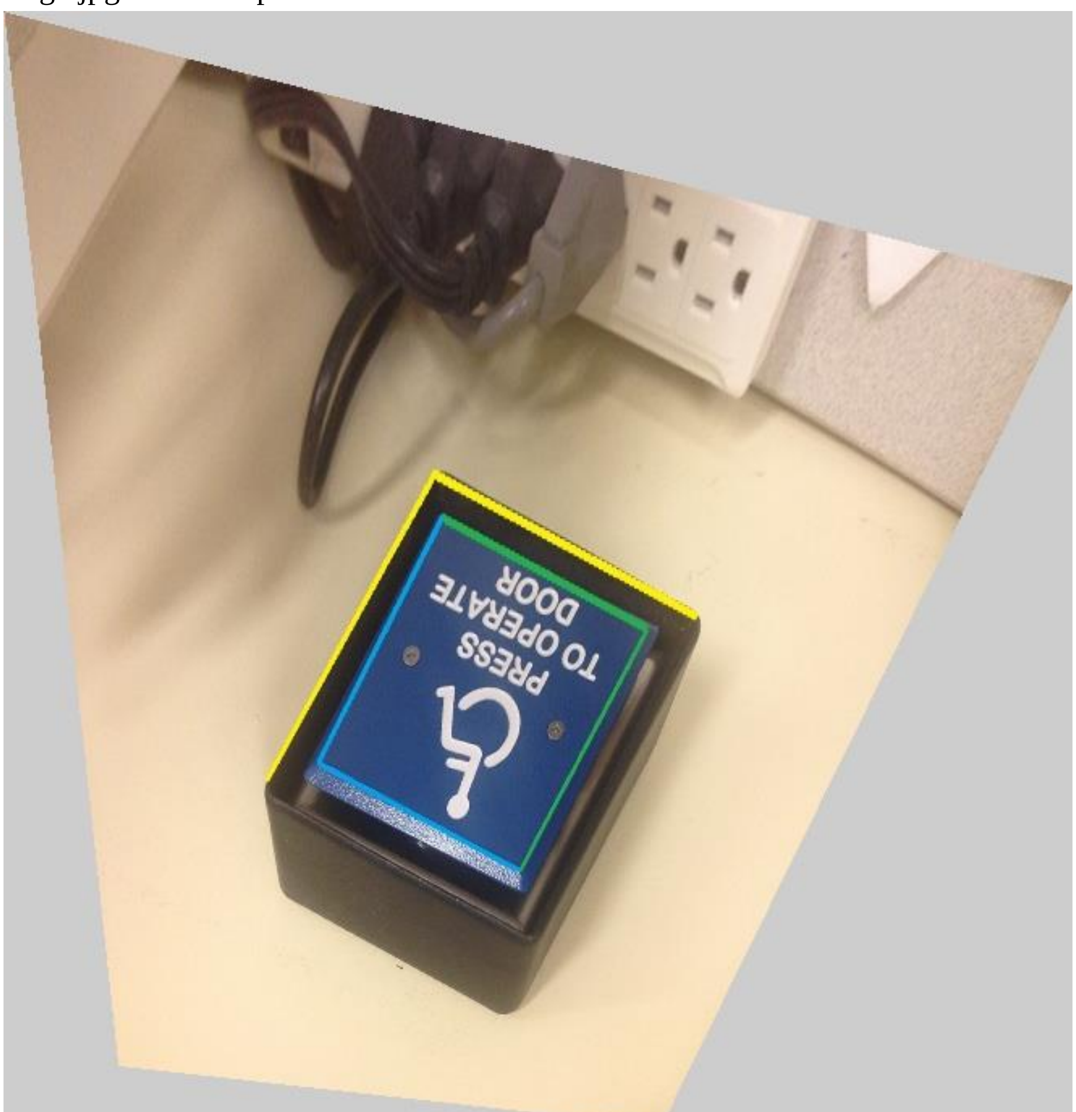
- Img1.jpg – Two-step Method: Affine distortion removed

- Img1.jpg – One-step Method: Projective and affine distortion removed

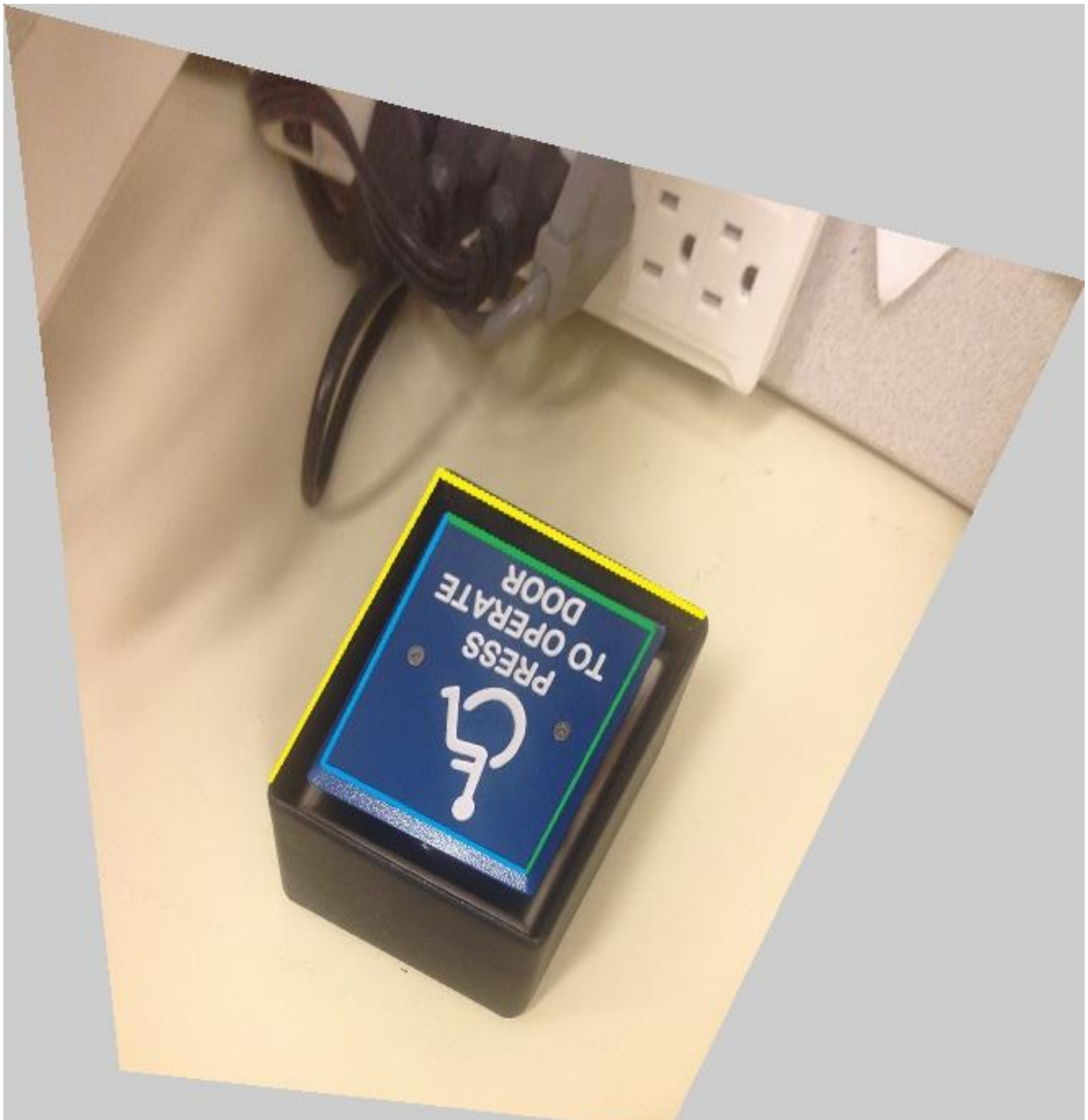
- Img2.jpg – Original

- Img2.jpg – Two-step Method: Projective distortion removed

- Img2.jpg – Two-step Method: Affine distortion removed



- Img2.jpg – One-step Method: Projective and affine distortion removed



Appendix B. Source Code

Please see hw3main.cpp