Computing a Homography from 2 sets of 4 points

Sebastien Kramm

June 15, 2021

For a given 2D point $\mathbf{p} = (s\ u, s\ v, s)$, the homography $\mathbf{H}$ (homogeneous 3x3 matrix) will project it to the point $\mathbf{p}' = (s\ u', s\ v', s)$, according to the relation $\mathbf{p}' = \mathbf{H} \cdot \mathbf{p}$:

$$\begin{bmatrix} s\ u' \\ s\ v' \\ s \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \cdot \begin{bmatrix} s\ u \\ s\ v \\ s \end{bmatrix}$$

This expands to:

$$u' = \frac{s\ u'}{s} = \frac{h_{11}u + h_{12}v + h_{13}}{h_{31}u + h_{32}v + h_{33}} \qquad v' = \frac{s\ v'}{s} = \frac{h_{21}u + h_{22}v + h_{23}}{h_{31}u + h_{32}v + h_{33}}$$

As $\mathbf{H}$ is homogeneous, we can fix $h_{33} = 1$, thus this can be written as:

$$\begin{cases} h_{31}uu' + h_{32}vu' + u' = h_{11}u + h_{12}v + h_{13} \\ h_{31}uv' + h_{32}vv' + v' = h_{21}u + h_{22}v + h_{23} \end{cases}$$

Rewritten as:

$$\begin{cases} h_{11}u + h_{12}v + h_{13} - h_{31}uu' - h_{32}vu' = u' \\ h_{21}u + h_{22}v + h_{23} - h_{31}uv' - h_{32}vv' = v' \end{cases} \tag{1}$$

We have 8 unknowns h_{ij}, that we set up in a vector $\mathbf{X}$:

$$\mathbf{X}^T = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{21} & h_{22} & h_{23} & h_{31} & h_{32} \end{bmatrix}$$

Thus (1) can be written as (2):

$$\mathbf{A} \cdot \mathbf{X} = \mathbf{B} \tag{2}$$

with:

$$\mathbf{A} = \begin{bmatrix} u & v & 1 & 0 & 0 & 0 & -uu' & -vu' \\ 0 & 0 & 0 & u & v & 1 & -uv' & -vv' \end{bmatrix} \qquad \text{and} \qquad \mathbf{B}^T = \begin{bmatrix} u' & v' \end{bmatrix}$$

Now, if we have 2 sets of 4 (non-colinear) points p_1, p_2, p_3, p_4 and p'_1, p'_2, p'_3, p'_4, such that $\mathbf{p'}_i = \mathbf{H} \cdot \mathbf{p}_i$, we will have 8 equations similar to (1) that we can use to fill the matrix $\mathbf{A}(8 \times 8)$:

$$\mathbf{A} = \begin{bmatrix} u_1 & v_1 & 1 & 0 & 0 & 0 & -u_1\, u'_1 & -v_1\, u'_1 \\ 0 & 0 & 0 & u_1 & v_1 & 1 & -u_1\, v'_1 & -v_1\, v'_1 \\ u_2 & v_2 & 1 & 0 & 0 & 0 & -u_2\, u'_2 & -v_2\, u'_2 \\ 0 & 0 & 0 & u_2 & v_2 & 1 & -u_2\, v'_2 & -v_2\, v'_2 \\ u_3 & v_3 & 1 & 0 & 0 & 0 & -u_3\, u'_3 & -v_3\, u'_3 \\ 0 & 0 & 0 & u_3 & v_3 & 1 & -u_3\, v'_3 & -v_3\, v'_3 \\ u_4 & v_4 & 1 & 0 & 0 & 0 & -u_4\, u'_4 & -v_4\, u'_4 \\ 0 & 0 & 0 & u_4 & v_4 & 1 & -u_4\, v'_4 & -v_4\, v'_4 \end{bmatrix}$$

and:

$$\mathbf{B}^T = \begin{bmatrix} u'_1 & v'_1 & u'_2 & v'_2 & u'_3 & v'_3 & u'_4 & v'_4 \end{bmatrix}$$

The exact solution will come from solving (2) (or computing $\mathbf{A}^{-1}$ and using with (3)). This will provide an exact solution for all the h_{ij} values. Any linear algebra library should be able to compute this.

$$\mathbf{X} = \mathbf{A}^{-1} \cdot \mathbf{B} \tag{3}$$

If we have more than 4 sets of points, then it will be necessary to use a robust estimator such as RANSAC [1], as a classical solving using Least Squares is much too sensitive to outliers.

[1] `https://en.wikipedia.org/wiki/Random_sample_consensus`