Transformations and Homographies

Discussion #5

Written by:
Brent*, Konpat*

Topics

This worksheet covers 2D transformations, focusing on the homography transformation and how to derive the linear system needed to solve for homography parameters.

1 Matching Transformations (again)

Below are matrix structures for common 2D transformations and their visual effects on a square.

Matrix Structures:

$$\text{A: } \begin{bmatrix} s\cos\theta & -s\sin\theta & t_x \\ s\sin\theta & s\cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{B: } \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix}$$

$$\text{C: } \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{D: } \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\text{E: } \begin{bmatrix} \cos\theta & -\sin\theta & t_x \\ \sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

Visual Diagrams:

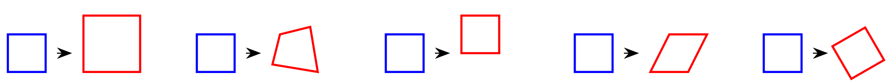


Diagram 1 Diagram 2 Diagram 3 Diagram 4 Diagram 5

Match each transformation type below to: (i) a matrix structure, (ii) visual diagram, and (iii) a degree of freedom (DoF) count.

(a) Translation:

Matrix: C Diagram: 3 DoF: 2

(b) Rigid:

Matrix: E Diagram: 5 DoF: 3

(c) Similarity:

Matrix: A Diagram: 1 DoF: 4

(d) Affine:

Matrix: D Diagram: 4 DoF: 6

(e) Projective/Homography:

Matrix: B Diagram: 2 DoF: 8

2 Deriving the Homography Linear System

Given a point $[x, y, 1]^\top$ in the source plane and point $[u, v, 1]^\top$ in the target plane, recall that 2D homographies follow

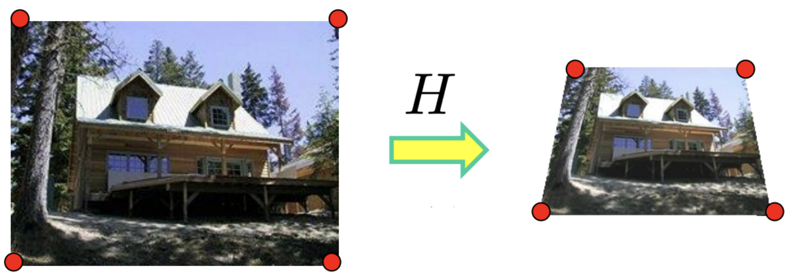
$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = H \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \tag{1}$$

where $\lambda \in \mathbb{R}$ is a scalar factor and H is a 3×3 homography matrix:

$$H = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & 1 \end{bmatrix}. \tag{2}$$

Homographies are defined up to scale, so we can fix $h_9 = 1$ (if we assume $h_9 \neq 0$).

Computing H. In Project 3, we will solve for H from a set of point correspondences:



This can be done by setting up an ordinary least squares (OLS) problem that can be solved with functions like `np.linalg.solve()` or `np.linalg.lstsq()`:

$$A\mathbf{h} = \mathbf{b}, \tag{3}$$

where $\mathbf{h} = [h_1, h_2, h_3, h_4, h_5, h_6, h_7, h_8]^\top$ flattens the H matrix. Equivalently,

$$\begin{bmatrix} \mathbf{a}_1^\top \\ \mathbf{a}_2^\top \\ \vdots \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ \vdots \\ h_8 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \end{bmatrix}. \tag{4}$$

Goal. We can solve for H by instantiating $\mathbf{a}_i^\top$ and b_i for each row i. Let's give this a try!

Problem 2.1: Single Correspondence

We'll start by setting up a system of equations. Consider a single correspondence pair: source point $[x, y, 1]^\top$ maps to target point $[u, v, 1]^\top$:

$$\begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{5}$$

(a) We can start by algebraically eliminating the λ term. Write expressions for u and v in terms of $h_1, \ldots, h_8$, and x, y. (hint: division)

$$u = (h_1 x + h_2 y + h_3)/(h_7 x + h_8 y + 1)$$
$$v = (h_4 x + h_5 y + h_6)/(h_7 x + h_8 y + 1)$$

(b) Convert your two equations from Part **(a)** into linear equations in the form: $(...)h_1 + (...)h_2 + ... + (...)h_8 = (...)$. Why is this useful?

$$h_1 x + h_2 y + h_3 - u h_7 x - u h_8 y = u$$
$$h_4 x + h_5 y + h_6 - v h_7 x - v h_8 y = v$$

This is useful because it matches the form of rows of $A\mathbf{h} = \mathbf{b}$!

(c) What $\mathbf{a}_1^\top$, $\mathbf{a}_2^\top$, b_1, and b_2 values does this correspondence pair produce?

$$\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -ux & -uy \\ 0 & 0 & 0 & x & y & 1 & -vx & -vy \end{bmatrix} \mathbf{h} = \begin{bmatrix} u \\ v \end{bmatrix}$$

The $\mathbf{a}^\top$ vectors are the rows of the wide matrix, the b values are the scalars in the vector on the right side of the equality.

Problem 2.2: Multiple Correspondences

(a) Suppose we have multiple correspondence pairs: $(x_1, y_1) \to (u_1, v_1)$, $(x_2, y_2) \to (u_2, v_2)$, $(x_3, y_3) \to (u_3, v_3)$, $(x_4, y_4) \to (u_4, v_4)$. Fill in the blanks in this matrix system for 4 correspondence pairs:

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -u_1 x_1 & -u_1 y_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -v_1 x_1 & -v_1 y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -u_2 x_2 & -u_2 y_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -v_2 x_2 & -v_2 y_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -u_3 x_3 & -u_3 y_3 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -v_3 x_3 & -v_3 y_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -u_4 x_4 & -u_4 y_4 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -v_4 x_4 & -v_4 y_4 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \end{bmatrix} = \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ u_3 \\ v_3 \\ u_4 \\ v_4 \end{bmatrix}$$

(b) How many point correspondences do we need at minimum to solve for the homography H? Why?

We need a minimum of 4 point correspondences.
Reasoning: We have 8 unknowns, so we need at least 8 equations to have a unique solution. Since each correspondence gives 2 equations, we need $8 \div 2 = 4$ correspondences.

(c) Can we still solve the system if we have more than the minimum number of correspondences? What happens?

Yes! If we have more than 4 correspondences, we get an overdetermined system (more equations than unknowns). We can still solve this using least squares methods to find the best-fit homography that minimizes the total error.

(d) How many point correspondences are needed to solve for each of the following 2D transformations?

Translation:	1
Rigid:	2
Similarity:	2
Affine:	3

General rule: Each correspondence gives 2 equations (one for x, one for y). To solve for a transformation with n degrees of freedom, we need at least $\lceil n/2 \rceil$ correspondences.