LECTURE 5: IMAGE STITCHING

Algorithm to stitch 2 images:

1. Extract the keypoints from the two images.
2. Based on the keypoints (i.e. the descriptors), find all the matches.
3. From the matches, compute the best homography matrix using RANSAC.
4. Stitch the two images, by transforming one image to the other using the computed homography matrix.

---

[1] Keypoint Matching

Goal: To determine the corresponding keypoints from two images that match best.

Algorithm: $$\text{Best}_i = \min_j \sqrt{\sum_k \| \vartheta_i[k] - \vartheta_j[k] \|^2}$$

where:

i = the index of a keypoint on image 1.

j = the index of a keypoint on image 2.

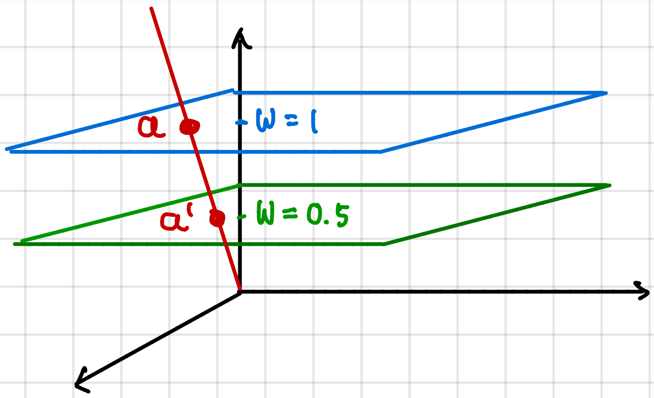
$\bar{\vartheta}_i$ = the descriptor of keypoint i

k = the index of the descriptor

[2] Homogeneous Coordinates

Real number space

In $\mathbb{R}^2$ (or a 2D space), the homogeneous coordinates are expressed as: $\begin{bmatrix} x \\ y \\ w \end{bmatrix}$



$a = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$; $a' = \begin{bmatrix} x' \\ y' \\ 0.5 \end{bmatrix}$

In the inhomogenous coord., a & a' are exactly the same point:

$$a = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ 0.5 \end{bmatrix} = \begin{bmatrix} x'/0.5 \\ y'/0.5 \\ 1 \end{bmatrix}$$

Q: Why use homogeneous coordinates?

Theoretical reason:

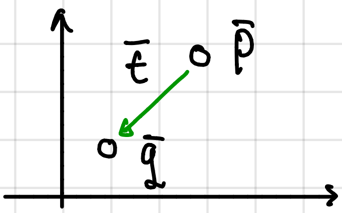
One of the main benefits is the ability to represent a point at infinity: $\begin{bmatrix} x \\ y \\ 0 \end{bmatrix}$

Practical reason:

Homogeneous coordinates allow us to express the translation operation in a matrix form.

Inhomogeneous coordinates

Translation in a matrix form is impossible:



$$\bar{p} - \bar{t} = \bar{q}$$

$$\begin{bmatrix} p_x \\ p_y \end{bmatrix} - \begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} q_x \\ q_y \end{bmatrix}$$

$\Rightarrow$ How to do this in a matrix form: $\bar{q} = T\,\bar{p}$, without using the homogeneous coord.?

Homogeneous coord.:

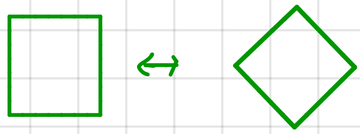
$$\bar{q} = T\,\bar{p}$$

$$\begin{bmatrix} q_x \\ q_y \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -t_x \\ 0 & 1 & 0 & -t_y \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix}$$

[3] Hierarchy of Transformations in 2D ($\mathbb{R}^2$)

1. Euclidean group : Rotation & Translation :

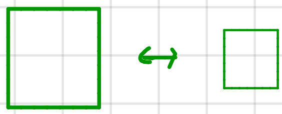
$$\begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$



Invariant properties : length & area (will be the same when we apply these translations)

2. Similarity group : Scaling :

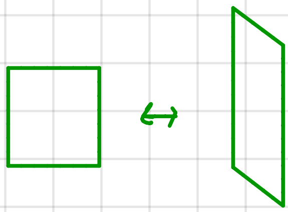
$$\begin{bmatrix} s\,r_{11} & s\,r_{12} & t_x \\ s\,r_{21} & s\,r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$



Invariant properties : ratio of lengths ; angle

3. Affine group : Shearing :

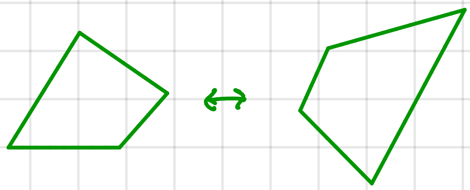
$$\begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$



Invariant properties : parallelism, ratio of lengths on midpoints, etc.

4. Projective group :

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$$

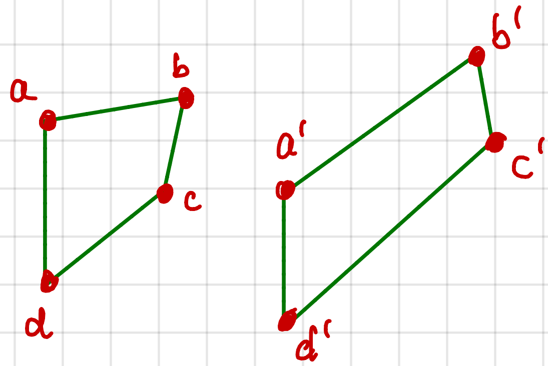


Invariant properties : concurrency, collinearity, etc

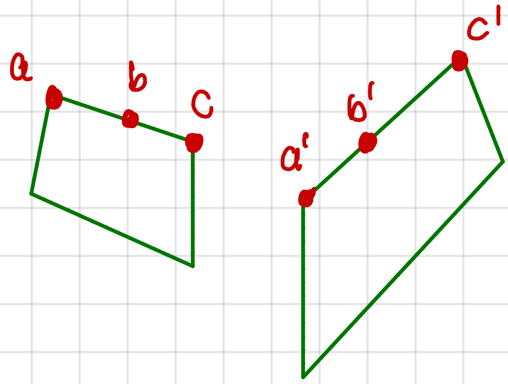
intersecting lines still intersect.

points on a line still points on a line

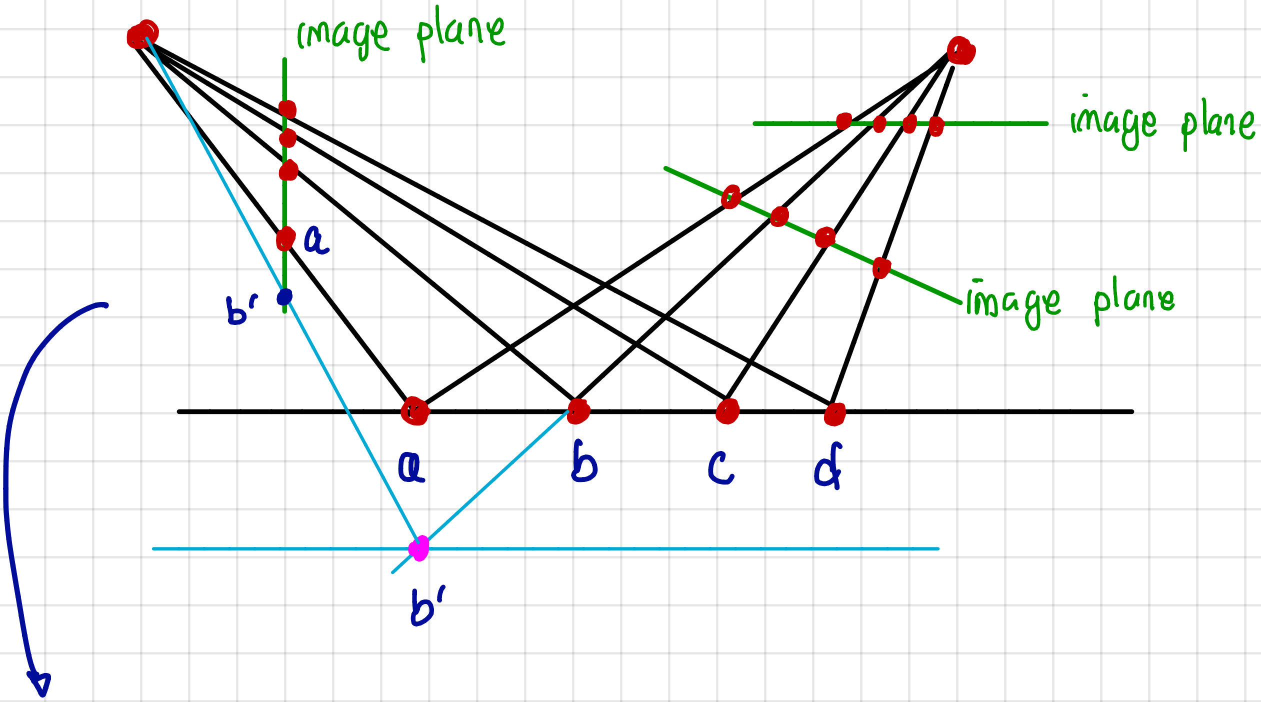
Concurrency preservation:



Collinearity preservation:



1D:

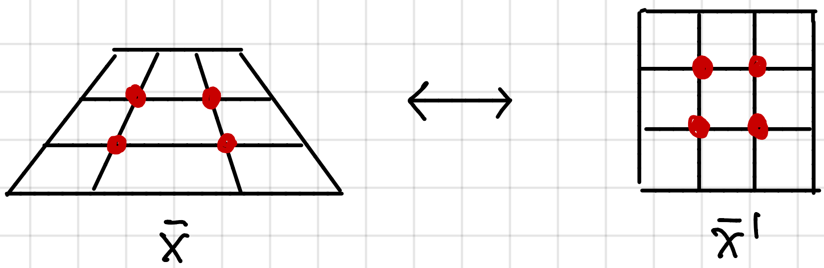


For b', collinearity is violated!

Hence, collinearity can be preserved if the actual points lie on a planar surface.

[4] Homography

In short, homography is a matrix that transforms:



Any two images of the same planar surface in space are related by a homography (assuming a pinhole camera model):

$$\boxed{\bar{x}' = \mathbb{H}\,\bar{x}}$$

How to compute $\mathbb{H}$, given pairs of $\bar{x}$ and $\bar{x}'$?

$$\underset{3\times 1}{\bar{x}'} = \underset{3\times 3}{\mathbb{H}}\ \underset{3\times 1}{\bar{x}}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

To convert matrix $\mathbb{H}$ to a vector:

$$\Leftrightarrow \quad x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}} \quad ; \quad y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}$$

$$\Leftrightarrow \quad \begin{aligned} h_{11}x + h_{12}y + h_{13} - h_{31}xx' - h_{32}yx' - h_{33}x' &= 0 \\ h_{21}x + h_{22}y + h_{23} - h_{31}xy' - h_{32}yy' - h_{33}y' &= 0 \end{aligned}$$

$$\Leftrightarrow \quad \begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -x'x & -x'y & -x' \\ 0 & 0 & 0 & x & y & 1 & -y'x & -y'y & -y' \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ \vdots \\ h_{33} \end{bmatrix} = 0$$

$$\underbrace{\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -x'x & -x'y & -x' \\ 0 & 0 & 0 & x & y & 1 & -y'x & -y'y & -y' \end{bmatrix}}_{2\times 9} \underbrace{\begin{bmatrix} h_{11} \\ h_{12} \\ \vdots \\ h_{33} \end{bmatrix}}_{9\times 1} = 0 \rightarrow \text{Homogeneous linear system}$$

Assuming we have more than 4 pairs of matched keypoints:

$$\underbrace{A}_{(n\times 2)\times 9} \underbrace{\bar{h}}_{9\times 1} = 0 \quad ; \quad n \geq 4 \text{ (overdetermined system)}$$

To avoid a trivial solution, where $\bar{h} = 0$, we need to put a constraint on $\bar{h}$, such as $|\bar{h}| = 1$:

$$\boxed{\begin{array}{c} A\bar{h} = 0 \\ \text{s.t. } |\bar{h}| = 1 \end{array}} \quad \text{or:} \quad \boxed{\begin{array}{l} \min |A\bar{h}| \\ \text{s.t. } |\bar{h}| = 1 \end{array}}$$

$$\Downarrow$$

Solution for getting $\bar{h}$ given A: use SVD

$A = UDV^T \rightarrow \bar{h}$ is the last row of V^T.

(see page #13 for more detailed discussion).

Notes:

1. Considering: $|\bar{h}| = 1$, the 9 unknowns becomes 8, since we can obtain h_{33} given $h_{11} \dots h_{32}$.

2. If the number of unknowns in $\bar{h}$ is 8, then we only need 4 pairs to recover $\bar{h}$. And, the system becomes a closed form (instead of overdetermined)

$\Downarrow$

A closed-form system is solvable directly, even without SVD.

$\Downarrow$

Q: How to do this?

A:

$$\underbrace{\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -x'x & -x'y & -x' \\ 0 & 0 & 0 & x & y & 1 & -y'x & -y'y & -y' \end{bmatrix}}_{2\times 9} \underbrace{\begin{bmatrix} h_{11}/h_{33} \\ h_{12}/h_{33} \\ \vdots \\ h_{32}/h_{33} \\ 1 \end{bmatrix}}_{9\times 1} = 0$$

if $\tilde{h}_{ij} = h_{ij}/h_{33}$

then:

$$x\tilde{h}_{11} + y\tilde{h}_{12} + \tilde{h}_{13} - x'x\tilde{h}_{31} - x'y\tilde{h}_{32} - x' = 0$$
$$x\tilde{h}_{21} + y\tilde{h}_{22} + \tilde{h}_{23} - y'x\tilde{h}_{31} - y'y\tilde{h}_{32} - y' = 0$$

$$\Leftrightarrow \underbrace{\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -x'x & -x'y \\ 0 & 0 & 0 & x & y & 1 & -y'x & -y'y \end{bmatrix}}_{2\times 8} \underbrace{\begin{bmatrix} \tilde{h}_{11} \\ \tilde{h}_{12} \\ \vdots \\ \tilde{h}_{32} \end{bmatrix}}_{8\times 1} = \underbrace{\begin{bmatrix} x' \\ y' \end{bmatrix}}_{2\times 1}$$

if we have 4 pairs:

$$\boxed{\underset{8\times 8}{A}\ \underset{8\times 1}{\tilde{h}} = \underset{8\times 1}{\bar{b}}} \rightarrow \tilde{h} = A^{-1}\bar{b}$$

3. $$\boxed{\underset{8\times 1}{\tilde{h}} = \underset{8\times 8}{A^{-1}}\ \underset{8\times 1}{\bar{b}}}$$; $$\bar{b} = \begin{bmatrix} x_1' \\ y_1' \\ x_2' \\ y_2' \\ x_3' \\ y_3' \\ x_4' \\ y_4' \end{bmatrix}$$

$$\tilde{h} = \begin{bmatrix} h_{11}/h_{33} \\ h_{21}/h_{33} \\ \vdots \\ h_{32}/h_{33} \end{bmatrix}$$

Q: How can we recover $\bar{h}$ from $\tilde{h}$?

A: $|\bar{h}| = 1 \quad \rightarrow \quad \sum_{i,j} h_{ij}^2 = 1$

$$\Leftrightarrow \quad h_{11}^2 + h_{12}^2 + \ldots + h_{33}^2 = 1$$

$$\Leftrightarrow \quad \frac{h_{11}^2}{h_{33}^2} + \frac{h_{12}^2}{h_{33}^2} + \ldots + 1 = \frac{1}{h_{33}^2}$$

$$\Leftrightarrow \quad \underbrace{\tilde{h}_{11}^2 + \tilde{h}_{12}^2 + \ldots + \tilde{h}_{32}^2}_{\text{known!}} + 1 = \frac{1}{h_{33}^2}$$ unknown

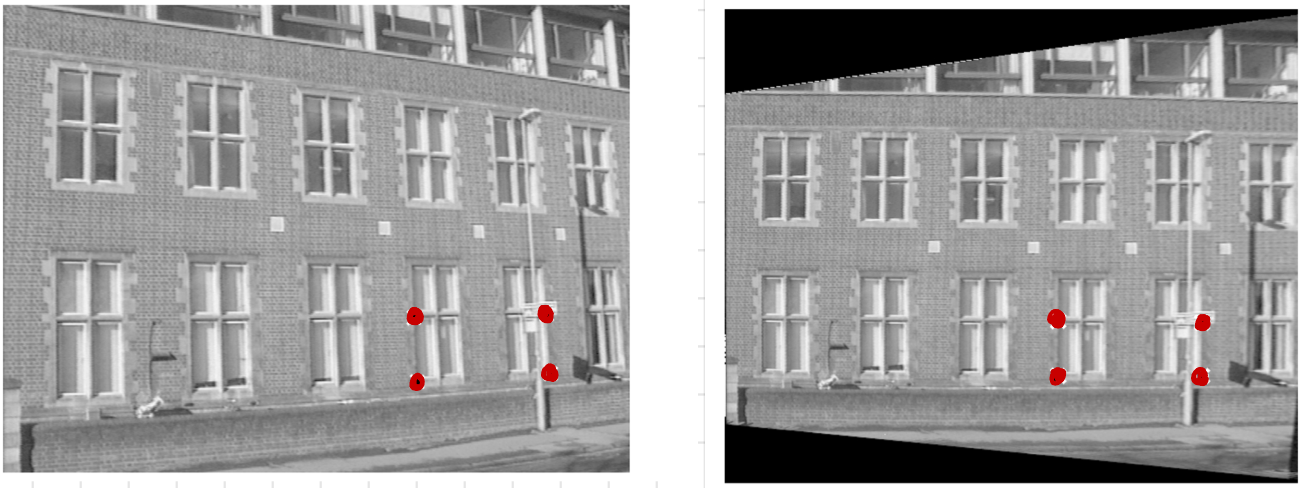
$$\Leftrightarrow \quad h_{33}^2 = \frac{1}{\sum_{i,j} \tilde{h}_{ij}^2 + 1}$$

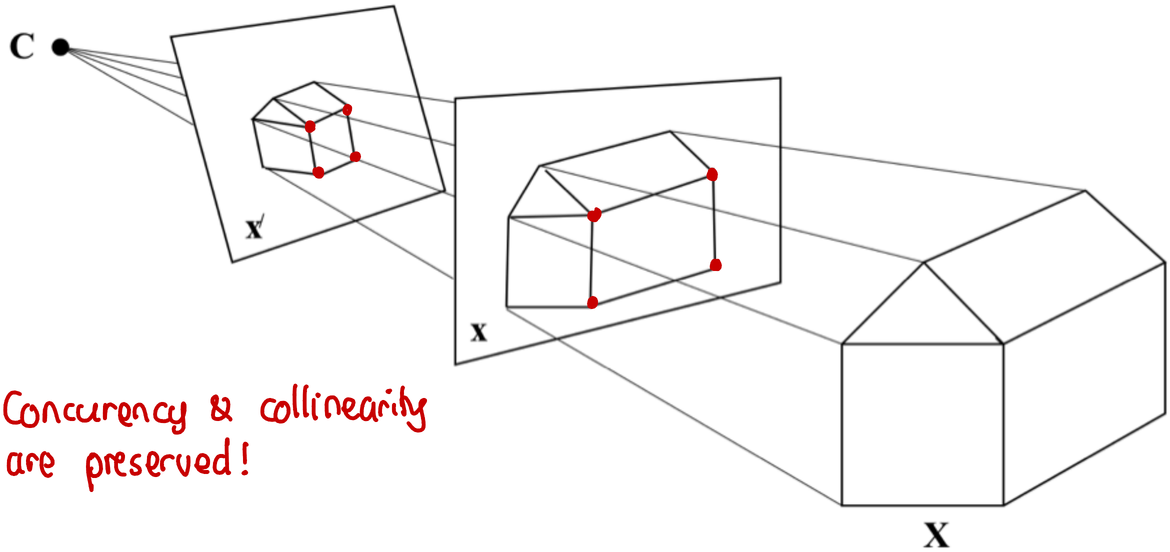
Hence: $$h_{ij} = \tilde{h}_{ij}\, h_{33} \quad ; \quad \forall i j$$

$\Downarrow$

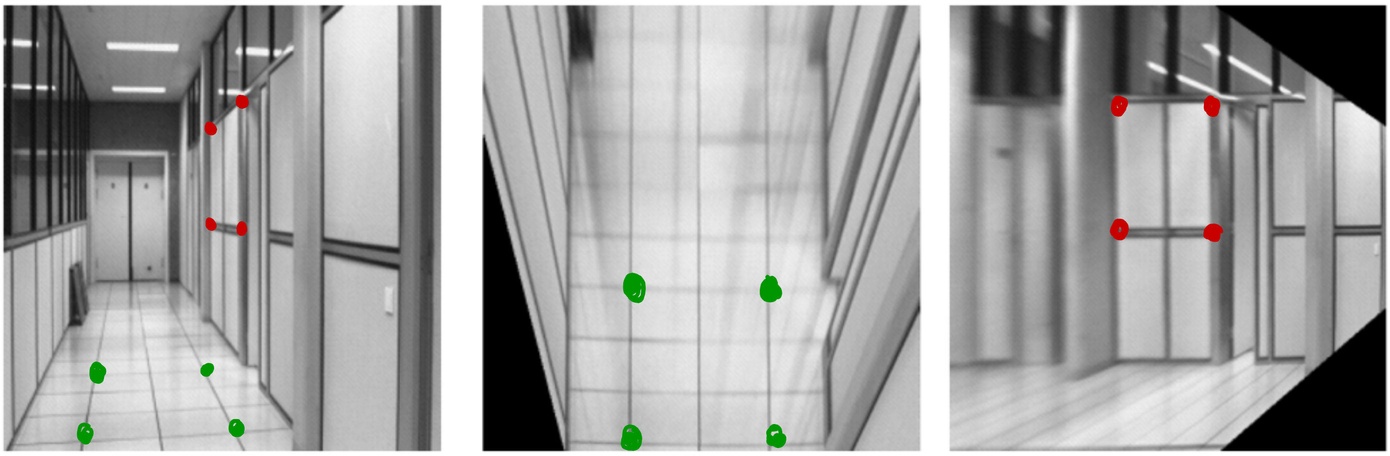
Note that, an overdetermined system (where $n > 4$) is more robust than a closed-form system, where noise can affect the values in A.

Example:





Concurrency & collinearity are preserved!



[5] RANSAC: Random Sample Consensus

Algorithm (General):

1. Select a minimum subset of points randomly
2. Fit the function (e.g. a line) using the points
3. Count the inliers (i.e. the number of other points outside the subset that agree with this solution)
4. Repeat the above process for T iterations
5. Choose the function's parameters that have the most inliers.

Algorithm (RANSAC + Homography):

but not that large

1. Select n pairs of keypoints randomly ; $n \geq 5$
2. Compute the homography matrix from the n pairs
3. Compute inliers, where the distance, $d(x'_{in}, \mathbb{H}x_{in}) < \varepsilon$
4. Repeat the above process for T iterations & keep the largest set of inliers
5. Recompute $\mathbb{H}$ on all of the inliers using least squares.

$\Downarrow$

Either using SVD or Least Squares.

Least squares:

$$\mathbb{H}^* = \underset{\{\mathbb{H}\}}{\operatorname{argmin}} \sum_n \left(x_n' - \mathbb{H}\, x_n\right)^2$$

$\downarrow$ 8 unknown parameters

[solvable using gradient descent → Levenberg-Marquardt]

$$\mathbb{H}^* = \underset{\{\mathbb{H}\}}{\operatorname{argmin}} \; E(\mathbb{H})$$

where: $$E(\mathbb{H}) = \sum_n \left(\begin{bmatrix} x_n' \\ y_n' \\ 1 \end{bmatrix} - \mathbb{H} \begin{bmatrix} x_n \\ y_n \\ 1 \end{bmatrix} \right)^2$$

$$= \sum_n \begin{pmatrix} x_n' - h_{11} x_n - h_{12} y_n - h_{13} \\ y_n' - h_{21} x_n - h_{22} y_n - h_{23} \\ 1 - h_{31} x_n - h_{32} y_n - h_{33} \end{pmatrix}^2$$

Minimization:

$$\frac{\partial E(\mathbb{H})}{\partial h_{11}} = \sum_n \frac{\partial}{\partial h_{11}} \left(x_n' - h_{11} x_n - h_{12} y_n - h_{13}\right)^2$$

$$= -2 \sum_n \left(x_n' - h_{11} x_n - h_{12} y_n - h_{13}\right) x_n = 0$$

h_{11} depends on unknowns: h_{12} & h_{13} → Hence it can't be solved directly.

Gradient Descent:

$$h_{11}^{new} = h_{11}^{old} - \eta \left. \frac{\partial E(\mathbb{H})}{\partial h_{11}} \right|_{h_{11} = h_{11}^{old}}$$

Q: How to get the initialization of this?

A: From the result of RANSAC.

[6] Stitching Images

[•] Stitching 2 images :

1. Using the homography $\mathbb{H}_{21}$: $\bar{x}_1 = \mathbb{H}\bar{x}_2$, transform image 2 to image 1.
2. Since the transformation changes the size of image 1, prepare an empty array that is large enough to handle image 1 + the transformed image 2.
3. This array becomes your canvas.

[•] Adding image 3 to the canvas

1. Compute the homography between the canvas & image 3, $\mathbb{H}_{C3}$, which is defined as :

$$x_C = \mathbb{H}_{C3}\, x_3$$

2. Enlarge the canvas so that it can handle image 3.
3. Transform image 3 to the canvas.

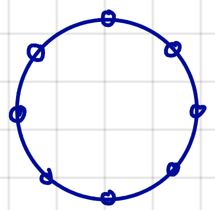
[7] Homogeneous Systems & SVD

$$\underset{m \times n}{A} = \underset{m \times m}{U} \; \underset{m \times n}{D} \; \underset{n \times n}{V^T}$$

- U: orthogonal matrix (rotation)
- D: diagonal matrix (stretching)
- V^T: orthogonal matrix (rotation)

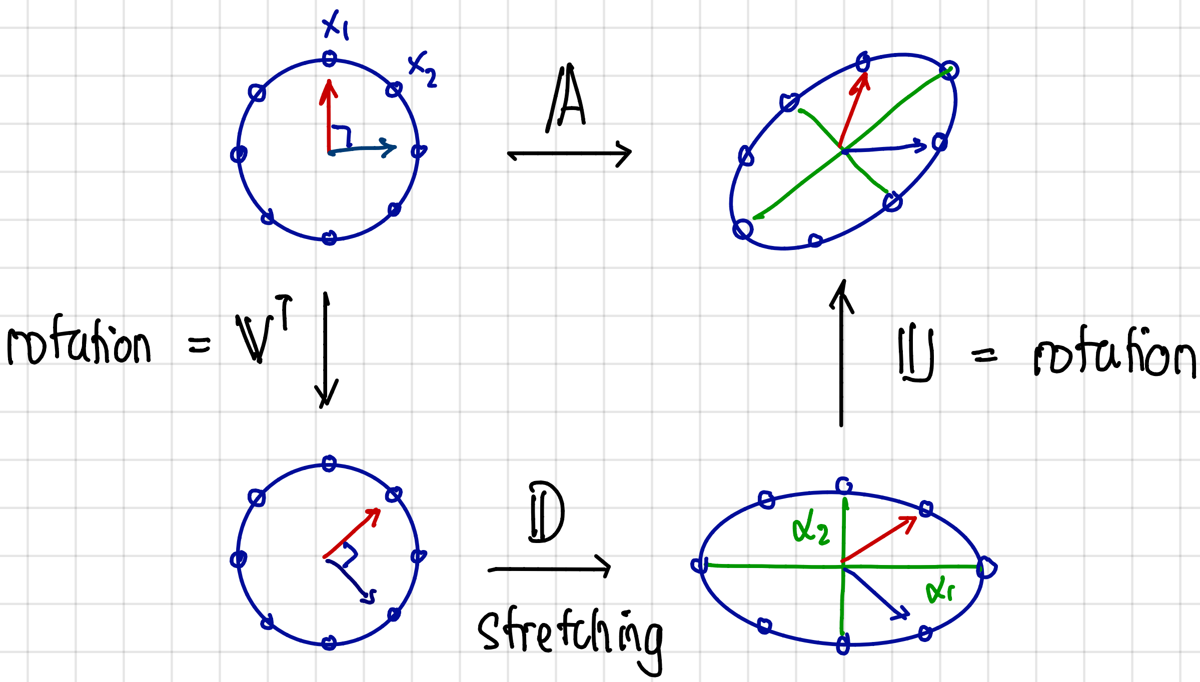
[•] SVD illustration :

Assume we have a collection of points in matrix, X.



$= X$

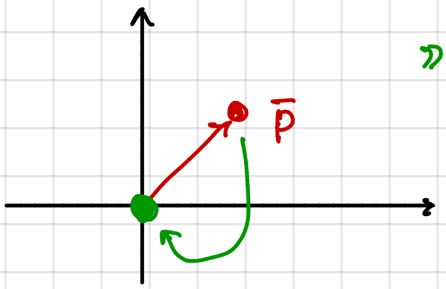
If we do : AX means X is transformed by matrix A



[•] SVD & Homogeneous Systems

We want to solve: $A\bar{p} = 0$

$\bar{p} \in \mathbb{R}^2$: where A is known & $|\bar{p}| = 1$



» What is $\bar{p}$ so that when we transform $\bar{p}$ using A, the outcome is zero (= the origin).

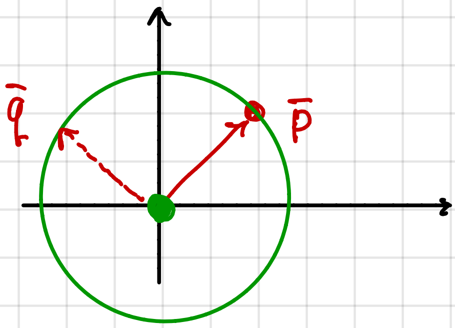
Using SVD:

$$A\bar{p} = UDV^T\bar{p} = 0$$

$$UDV^T\bar{p} = 0$$

let: $\bar{q} = V^T\bar{p}$

↳ a rotation matrix, which doesn't make $\bar{q} = 0$.



$\underbrace{U}D\bar{q} = 0$ → Assuming $\mathbb{R}^2$:

This is another rotation that doesn't make the transformation to zero.

$$D\bar{q} = \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix} \begin{bmatrix} q_x \\ q_y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$\alpha_1 \geq \alpha_2$

we want to find q_x & q_y so that the equation holds.

Due to the constraint : $|\bar{p}| = 1$
it also implies : $|\bar{q}| = 1$.
Hence, we can define $\bar{q} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, which assuming $\alpha_2 = 0$:

$$\begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 = 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

In some cases, α_2 is indeed zero ; however, in most cases $\alpha_2 > 0$.
Although, usually : $\alpha_1 > \alpha_2$.

If $\alpha_2 > 0$, then it's impossible to make $\mathbb{D}\bar{q} = 0$. For this reason :

$$\min |\mathbb{A}\bar{p}|$$
$$\text{s.t. } |\bar{p}| = 1$$

and, $\bar{q} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$ will enable us to get the minimum, but not zero.

Implying $\bar{p}$ is the last column of $\mathbb{V}$.

If $\bar{q} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$ $\rightarrow$

$$\bar{q} = \mathbb{V}^T \bar{p}$$
$$\bar{p} = \mathbb{V}\bar{q} = \begin{bmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} V_{12} \\ V_{22} \end{bmatrix}$$

[•] Proof that $\bar{p}$ = the last column of $\mathbb{V}$, minimizing $|\mathbb{A}\bar{p}|$

$$\min |\mathbb{A}\bar{p}| = \min |\mathbb{U}\mathbb{D}\mathbb{V}^T\bar{p}| \rightarrow \text{Orthogonal matrix's property: } |\mathbb{U}\bar{z}| = |\bar{z}|$$
$$= \min |\mathbb{D}\mathbb{V}^T\bar{p}| \; ; \; \bar{q} = \mathbb{V}^T\bar{p}$$
$$\left.\begin{aligned} &= \min |\mathbb{D}\bar{q}| \\ &\text{s.t. } |q| = 1 \end{aligned}\right\} \quad \bar{q} = [0, 0, \ldots, 0]^T \rightarrow \text{we want this kind of } \bar{q}.$$

corresponds to smallest element of $\mathbb{D}$

Since: $\bar{p} = \mathbb{V}\bar{q}$, thus : $\bar{p}$ is simply the last column of $\mathbb{V}$.

[•] Example 1 :

1. $A\bar{p} = 0$ and $A = \begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix}$ → Q: what is $\bar{p} = ?$

SVD : $A = \begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 2/\sqrt{5} & 1/\sqrt{5} \\ -1/\sqrt{5} & 2/\sqrt{5} \end{bmatrix}}_{U} \underbrace{\begin{bmatrix} \sqrt{10} & 0 \\ 0 & 0 \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}}_{V^T}$

$\alpha_2 = 0$ → what does it imply ?

The stretching makes the second axis disappear !

Hence : $\bar{p} = \begin{bmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix}$

This value makes $A\bar{p} = 0$:

$$A\bar{p} = \begin{bmatrix} 2 & 2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Example 2 :

$\min |A\bar{p}|$ for $A = \begin{bmatrix} 2 & 3 \\ 4 & 2 \\ 1 & 2 \end{bmatrix}$

SVD : $\underset{3\times2}{A} = \begin{bmatrix} 2 & 3 \\ 4 & 2 \\ 1 & 2 \end{bmatrix} = \underbrace{\overset{3\times3}{\begin{bmatrix} -0.6 & 0.5 & -0.6 \\ -0.7 & -0.7 & 0.1 \\ -0.3 & 0.5 & 0.8 \end{bmatrix}}}_{U} \underbrace{\overset{3\times2}{\begin{bmatrix} 5.9 & 0 \\ 0 & 1.7 \\ 0 & 0 \end{bmatrix}}}_{D} \underbrace{\overset{2\times2}{\begin{bmatrix} -0.7 & -0.6 \\ -0.6 & 0.7 \end{bmatrix}}}_{V^T}$

The smallest eigenvalue & its corresponding eigenvector is $\begin{bmatrix} -0.6 \\ 0.7 \end{bmatrix}$

Hence : $\bar{p} = \begin{bmatrix} -0.6 \\ 0.7 \end{bmatrix}$

$$|A\bar{p}| = \left| \begin{bmatrix} 2 & 3 \\ 4 & 2 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} -0.6 \\ 0.7 \end{bmatrix} \right| = 1.69$$

↳ it's not zero, yet it's the minimum value

[•] SVD: What is it ?

$A = UDV^T$ → $A^TA = (UDV^T)^T (UDV^T)$

$= V D^T \underline{U^T U} D V^T = V D^T D V^T$

Orthogonal matrices: $U^TU = I$

$D^TD = D^2$ = the eigenvalues of A^TA

V's columns are the eigenvectors of A^TA

(Using the same derivation (AA^T), U's columns are the eigenvectors of AA^T