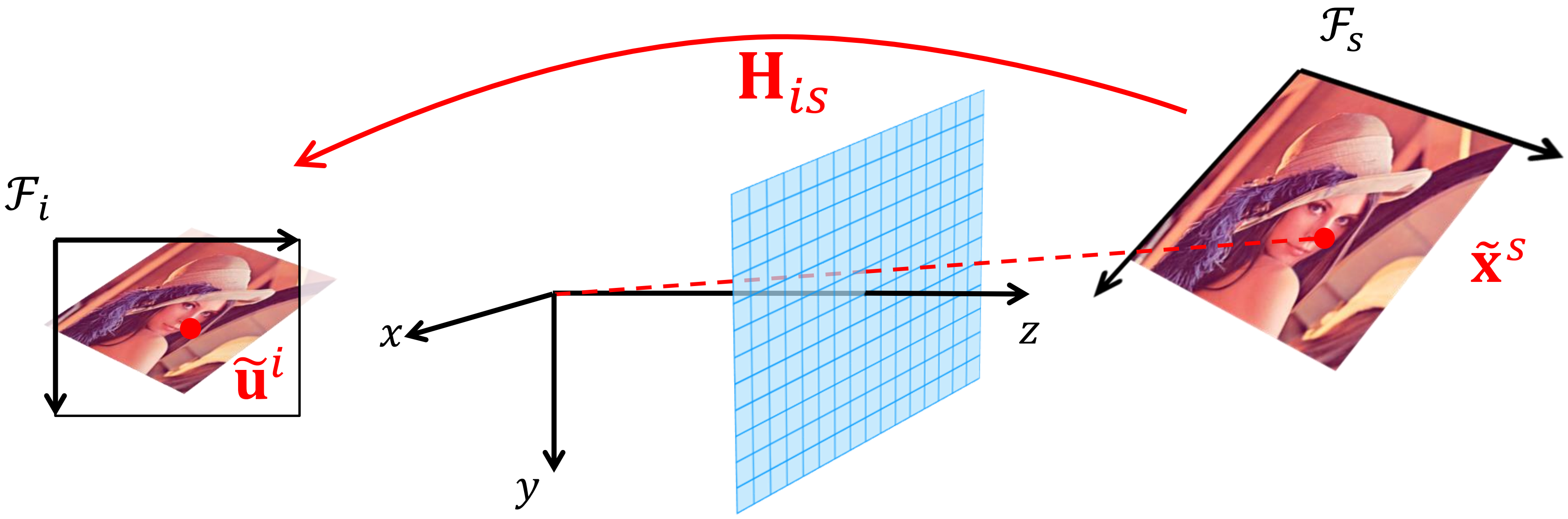
UiO : Department of Technology Systems
University of Oslo

Estimating homographies from feature correspondences

Thomas Opsahl

2023

Homographies and perspective imaging



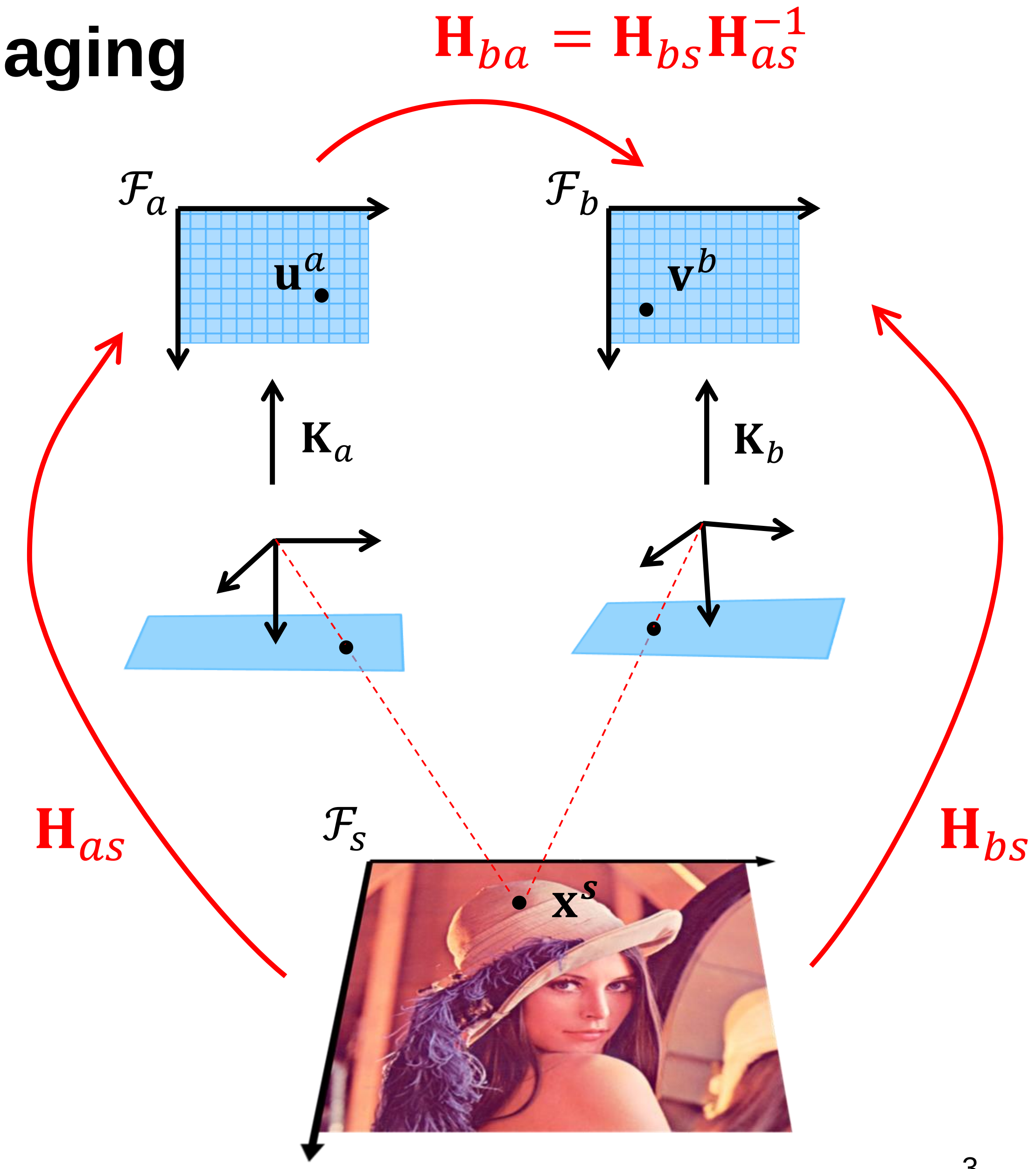
The perspective imaging of a planar surface corresponds (exactly!) to a homography

$$\mathbf{H}_{is}\tilde{\mathbf{x}}^s = \tilde{\mathbf{u}}^i$$

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

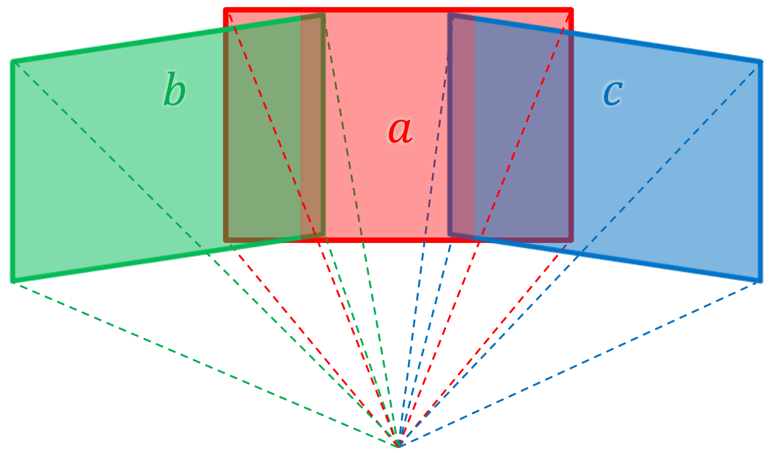
Homographies and perspective imaging

Two overlapping perspective images of a planar scene are “perfectly” related by a homography

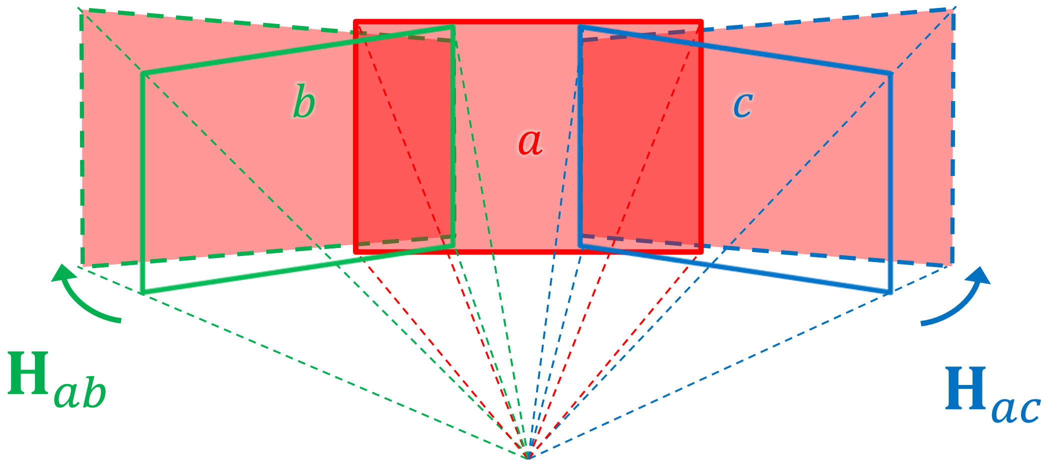


Homographies and perspective imaging

Two overlapping perspective images of a planar scene are “perfectly” related by a homography



Two overlapping images from a camera rotating about its projective center (pinhole) are also “perfectly” related by a homography



Homographies and perspective imaging

Hence it is often reasonable to assume that two overlapping images are related by a homography

- Planar or almost planar scene, i.e. when the distance to the scene is relatively much larger than the 3D structures in the scene

- Purely rotating camera or when the distance to the scene is relatively much larger than the camera translation between images

Homographies and perspective imaging

Knowing the homography between two images allows us to map one image into the other image's coordinate system

$$H_{ba} \colon \mathrm{Img}_a \to \mathrm{Img}_b$$

$$H_{ab} = H_{ba}^{-1} \colon \mathrm{Img}_b \to \mathrm{Img}_a$$

Coregistering images like this allows us to compare and combine information in overlapping images

Coregistration is therefore an important first step for applications like change detection and image mosaicing

Homography estimation
Understanding the problem

We know that

- A homography is a projective transformation that we can represent by a homogeneous 3x3 matrix

$$\mathbf{H}_{ba}\tilde{\mathbf{u}}^a = \tilde{\mathbf{u}}^b$$

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

- Point correspondences $\mathbf{u}_i^a \leftrightarrow \mathbf{u}_i^b$ can be established automatically, but some wrong correspondences are to be expected

Homography estimation
Understanding the problem

Unknown

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

Known

This equality involves a homogeneous matrix and two homogenous points/vectors

Since the points are known, we can fix their numerical representation to be the standard representation (last coordinate equal to 1)

The scale ambiguity of the matrix, is however something that we have to take into account when we try to solve for its unknown elements

Homography estimation
Understanding the problem

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

Unknown

Known

Direct linear transform

The direct linear transformation (DLT) is an algorithm for solving a set of variables from a set of equalities like this (equal up to scale)

By rewriting the equality into a set of proper linear homogeneous equations, we represent the problem in a way that is naturally scale ambiguous

This will allow us to determine the variables using standard methods from linear algebra

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} h_1 u^a + h_2 v^a + h_3 \\ h_4 u^a + h_5 v^a + h_6 \\ h_7 u^a + h_8 v^a + h_9 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

This equality corresponds to a system of three equations

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\begin{array}{c} \text{I} \\ \text{II} \\ \text{III} \end{array} \quad \begin{Bmatrix} h_1 u^a + h_2 v^a + h_3 = u^b \\ h_4 u^a + h_5 v^a + h_6 = v^b \\ h_7 u^a + h_8 v^a + h_9 = 1 \end{Bmatrix}$$

This equality corresponds to a system of three equations

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\begin{matrix} \text{I} \\ \text{II} \\ \text{III} \end{matrix} \quad \begin{Bmatrix} h_1 u^a + h_2 v^a + h_3 = u^b \\ h_4 u^a + h_5 v^a + h_6 = v^b \\ h_7 u^a + h_8 v^a + h_9 = 1 \end{Bmatrix}$$

We can reformulate this into three linear homogeneous equations

$$\begin{matrix} v^b \text{I} = u^b \text{II} \\ \text{I} = u^b \text{III} \\ \text{II} = v^b \text{III} \end{matrix} \quad \begin{Bmatrix} h_1 u^a v^b + h_2 v^a v^b + h_3 v^b = h_4 u^a u^b + h_5 v^a u^b + h_6 u^b \\ h_1 u^a + h_2 v^a + h_3 = h_7 u^a u^b + h_8 v^a u^b + h_9 u^b \\ h_4 u^a + h_5 v^a + h_6 = h_7 u^a v^b + h_8 v^a v^b + h_9 v^b \end{Bmatrix}$$

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\begin{matrix} \text{I} \\ \text{II} \\ \text{III} \end{matrix} \quad \begin{Bmatrix} h_1 u^a + h_2 v^a + h_3 = u^b \\ h_4 u^a + h_5 v^a + h_6 = v^b \\ h_7 u^a + h_8 v^a + h_9 = 1 \end{Bmatrix}$$

We can reformulate this into three linear homogeneous equations

$$\begin{Bmatrix} u^a v^b h_1 + v^a v^b h_2 + v^b h_3 - u^a u^b h_4 - v^a u^b h_5 - u^b h_6 = 0 \\ u^a h_1 + v^a h_2 + 1 h_3 - u^a u^b h_7 - v^a u^b h_8 - u^b h_9 = 0 \\ u^a h_4 + v^a h_5 + 1 h_6 - u^a v^b h_7 - v^a v^b h_8 - v^b h_9 = 0 \end{Bmatrix}$$

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\begin{matrix} \text{I} \\ \text{II} \\ \text{III} \end{matrix} \quad \begin{Bmatrix} h_1 u^a + h_2 v^a + h_3 = u^b \\ h_4 u^a + h_5 v^a + h_6 = v^b \\ h_7 u^a + h_8 v^a + h_9 = 1 \end{Bmatrix}$$

Rewrite to get a homogeneous matrix equation

$$\begin{Bmatrix} u^a v^b h_1 + v^a v^b h_2 + v^b h_3 - u^a u^b h_4 - v^a u^b h_5 - u^b h_6 + 0h_7 + 0h_8 + 0h_9 = 0 \\ u^a h_1 + v^a h_2 + 1h_3 + 0h_4 + 0h_5 + 0h_6 - u^a u^b h_7 - v^a u^b h_8 - u^b h_9 = 0 \\ 0h_1 + 0h_2 + 0h_3 + u^a h_4 + v^a h_5 + 1h_6 - u^a v^b h_7 - v^a v^b h_8 - v^b h_9 = 0 \end{Bmatrix}$$

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\Updownarrow$$

$$\begin{bmatrix} u^a v^b & v^a v^b & v^b & -u^a u^b & -v^a u^b & -u^b & 0 & 0 & 0 \\ u^a & v^a & 1 & 0 & 0 & 0 & -u^a u^b & -v^a u^b & -u^b \\ 0 & 0 & 0 & u^a & v^a & 1 & -u^a v^b & -v^a v^b & -v^b \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \\ h_9 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\Updownarrow$$

$$\begin{bmatrix} u^a v^b & v^a v^b & v^b & -u^a u^b & -v^a u^b & -u^b & 0 & 0 & 0 \\ u^a & v^a & 1 & 0 & 0 & 0 & -u^a u^b & -v^a u^b & -u^b \\ 0 & 0 & 0 & u^a & v^a & 1 & -u^a v^b & -v^a v^b & -v^b \end{bmatrix} \mathbf{h} = \mathbf{0}$$

The nature of homogeneous equations implies that all solutions are scale ambiguous, i.e. if $\mathbf{A}\mathbf{h}^* = \mathbf{0}$, then we also have that $\mathbf{A}(\alpha \mathbf{h}^*) = \mathbf{0}$ for any non-zero scalar α

This means that we can use standard methods in linear algebra to determine $\mathbf{h}$

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\Updownarrow$$

$$\begin{bmatrix} u^a v^b & v^a v^b & v^b & -u^a u^b & -v^a u^b & -u^b & 0 & 0 & 0 \\ u^a & v^a & 1 & 0 & 0 & 0 & -u^a u^b & -v^a u^b & -u^b \\ 0 & 0 & 0 & u^a & v^a & 1 & -u^a v^b & -v^a v^b & -v^b \end{bmatrix} \mathbf{h} = \mathbf{0}$$

At first glance, it seems like each point correspondence $\mathbf{u}_i^a \leftrightarrow \mathbf{u}_i^b$ puts three constraints on $\mathbf{h}$

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\Updownarrow$$

$$\begin{bmatrix} u^a v^b & v^a v^b & v^b & -u^a u^b & -v^a u^b & -u^b & 0 & 0 & 0 \\ u^a & v^a & 1 & 0 & 0 & 0 & -u^a u^b & -v^a u^b & -u^b \\ 0 & 0 & 0 & u^a & v^a & 1 & -u^a v^b & -v^a v^b & -v^b \end{bmatrix} \mathbf{h} = \mathbf{0}$$

At first glance, it seems like each point correspondence $\mathbf{u}_i^a \leftrightarrow \mathbf{u}_i^b$ puts three constraints on $\mathbf{h}$

They are however not independent: $v^\mathrm{b} \cdot row_2 - u^b \cdot row_3 = row_1$

Homography estimation

Direct Linear Transform

$$\begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} u^a \\ v^a \\ 1 \end{bmatrix} = \begin{bmatrix} u^b \\ v^b \\ 1 \end{bmatrix}$$

$$\Updownarrow$$

$$\begin{bmatrix} u^a v^b & v^a v^b & v^b & -u^a u^b & -v^a u^b & -u^b & 0 & 0 & 0 \\ u^a & v^a & 1 & 0 & 0 & 0 & -u^a u^b & -v^a u^b & -u^b \\ 0 & 0 & 0 & u^a & v^a & 1 & -u^a v^b & -v^a v^b & -v^b \end{bmatrix} \mathbf{h} = \mathbf{0}$$

At first glance, it seems like each point correspondence $\mathbf{u}_i^a \leftrightarrow \mathbf{u}_i^b$ puts three constraints on $\mathbf{h}$

They are however not independent: $v^{\mathrm{b}} \cdot row_2 - u^b \cdot row_3 = row_1$

Each point correspondence $\mathbf{u}_i^a \leftrightarrow \mathbf{u}_i^b$ corresponds to two constraints on $\mathbf{h}$, so we can choose to disregard one of them (does not matter which one)

Homography estimation
Direct Linear Transform

$$\begin{bmatrix} u_1^a & v_1^a & 1 & 0 & 0 & 0 & -u_1^a u_1^b & -v_1^a u_1^b & -u_1^b \\ 0 & 0 & 0 & u_1^a & v_1^a & 1 & -u_1^a v_1^b & -v_1^a v_1^b & -v_1^b \\ u_2^a & v_2^a & 1 & 0 & 0 & 0 & -u_2^a u_2^b & -v_2^a u_2^b & -u_2^b \\ 0 & 0 & 0 & u_2^a & v_2^a & 1 & -u_2^a v_2^b & -v_2^a v_2^b & -v_2^b \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_n^a & v_n^a & 1 & 0 & 0 & 0 & -u_n^a u_n^b & -v_n^a u_n^b & -u_n^b \\ 0 & 0 & 0 & u_n^a & v_n^a & 1 & -u_n^a v_n^b & -v_n^a v_n^b & -v_n^b \end{bmatrix} \mathbf{h} = \mathbf{0}$$

In a situation with several point correspondences, $\{\mathbf{u}_i^a \leftrightarrow \mathbf{u}_i^b\}_{\mathrm{i}=1,\ldots,\mathrm{n}}$, we can combine all of them into a $2n \times 9$ matrix $\mathbf{A}$

Homography estimation
Direct Linear Transform

$$\underbrace{\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_i^a & v_i^a & 1 & 0 & 0 & 0 & -u_i^a u_i^b & -v_i^a u_i^b & -u_i^b \\ 0 & 0 & 0 & u_i^a & v_i^a & 1 & -u_i^a v_i^b & -v_i^a v_i^b & -v_i^b \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{\mathbf{A}} \mathbf{h} = \mathbf{0}$$

Since we are not after the trivial solution, $\mathbf{h} = \mathbf{0}$, it is clear that $\mathbf{h}$ must lie in the null space of $\mathbf{A}$

There are several ways to determine the null space of a matrix, but we will use singular value decomposition (SVD)

Singular Value Decomposition (SVD)

SVD is a factorization of a $m \times n$ matrix $\mathbf{A}$ into a product $\mathbf{A} = \mathbf{USV}^T$

Where

$\mathbf{U}$ is a $m \times m$ orthogonal matrix

$\mathbf{S}$ is a $\mathrm{m} \times n$ rectangular diagonal matrix

$\mathbf{V}$ is a $n \times n$ orthogonal matrix

$$\mathbf{S} = \begin{bmatrix} s_1 & & & & \\ & \ddots & & & \mathbf{0} \\ & & s_n & & \\ & \mathbf{0} & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix}$$

The non-zero diagonal entries of $\mathbf{S}$ are known as the singular values of $\mathbf{A}$

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m]$$

The columns of $\mathbf{U}$ and $\mathbf{V}$ are known as left-singular vectors and right-singular vectors correspondingly

$$\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n]$$

Singular Value Decomposition (SVD)

SVD is a factorization of a $m \times n$ matrix $\mathbf{A}$ into a product $\mathbf{A} = \mathbf{USV}^T$

For homography estimation:
$n = 9$ and $m \geq 8$

Where

- $\mathbf{U}$ is a $m \times m$ orthogonal matrix
- $\mathbf{S}$ is a $\mathrm{m} \times 9$ rectangular diagonal matrix
- $\mathbf{V}$ is a 9×9 orthogonal matrix

$$\mathbf{S} = \begin{bmatrix} s_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & s_9 \end{bmatrix}$$

The non-zero diagonal entries of $\mathbf{S}$ are known as the singular values of $\mathbf{A}$

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m]$$

$$\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_9]$$

The columns of $\mathbf{U}$ and $\mathbf{V}$ are known as left-singular vectors and right-singular vectors correspondingly

Singular Value Decomposition (SVD)

The factorization $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ is not unique, but one can always choose a factorization such that the diagonal entries of $\mathbf{S}$ are in descending order

For homography estimation:
$n = 9$ and $m \geq 8$

Key result
The right singular vectors corresponding to vanishing singular values, i.e. $s_i = 0$, spans the null space of $\mathbf{A}$

$$\mathbf{S} = \begin{bmatrix} s_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & s_9 \end{bmatrix}$$

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m]$$

$$\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_9]$$

This means that, if $\mathbf{A}$ has rank 8 and the diagonal entries in $\mathbf{S}$ are in descending order, we have that $s_9 = 0$ and so

$$\mathbf{h} = \mathbf{v}_9$$

Homography estimation
Direct Linear Transform

$$\underbrace{\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_i^a & v_i^a & 1 & 0 & 0 & 0 & -u_i^a u_i^b & -v_i^a u_i^b & -u_i^b \\ 0 & 0 & 0 & u_i^a & v_i^a & 1 & -u_i^a v_i^b & -v_i^a v_i^b & -v_i^b \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{\mathbf{A}} \mathbf{h} = \mathbf{0}$$

Since each point correspondence $\mathbf{u}_i^a \leftrightarrow \mathbf{u}_i^b$ provides 2 rows in matrix $\mathbf{A}$, we need at least 4 correspondences for $\mathbf{A}$ to have rank 8

Given 4 point correspondences, we are guaranteed $\mathbf{A}$ to have rank 8 as long as no three points (in either of the two images) are co-linear

Homography estimation
Direct Linear Transform

$$\underbrace{\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_i^a & v_i^a & 1 & 0 & 0 & 0 & -u_i^a u_i^b & -v_i^a u_i^b & -u_i^b \\ 0 & 0 & 0 & u_i^a & v_i^a & 1 & -u_i^a v_i^b & -v_i^a v_i^b & -v_i^b \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{\mathbf{A}} \mathbf{h} = \mathbf{0}$$

In principle, the matrix $\mathbf{A}$ should have rank 8 even when we have more than 4 point correspondences

But, errors in the feature matching and uncertainties in the actual positioning of the feature points will typically cause $\mathbf{A}$ to have rank 9

Homography estimation
Direct Linear Transform

$$\underbrace{\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_i^a & v_i^a & 1 & 0 & 0 & 0 & -u_i^a u_i^b & -v_i^a u_i^b & -u_i^b \\ 0 & 0 & 0 & u_i^a & v_i^a & 1 & -u_i^a v_i^b & -v_i^a v_i^b & -v_i^b \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{\mathbf{A}} \mathbf{h} = \mathbf{0}$$

Analyzing matrix $\mathbf{A}$'s singular values should however reveal that the 9$^{\text{th}}$ and smallest singular value is close to 0 while the other 8 singular values are not

This indicates that $\mathbf{h} = \mathbf{v}_9$ still will be a reasonable solution to the problem

Homography estimation
Direct Linear Transform

$$\underbrace{\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_i^a & v_i^a & 1 & 0 & 0 & 0 & -u_i^a u_i^b & -v_i^a u_i^b & -u_i^b \\ 0 & 0 & 0 & u_i^a & v_i^a & 1 & -u_i^a v_i^b & -v_i^a v_i^b & -v_i^b \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{\mathbf{A}} \mathbf{h} = \mathbf{0}$$

Key result
The right-singular vector corresponding to the smallest singular value, is the optimal solution to $\mathbf{Ah} = \mathbf{0}$ in the sense that it minimizes the total least squares $||\mathbf{Ah}||$ under the constraint $||\mathbf{h}|| = 1$

Homography estimation
Direct Linear Transform

$$\underbrace{\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_i^a & v_i^a & 1 & 0 & 0 & 0 & -u_i^a u_i^b & -v_i^a u_i^b & -u_i^b \\ 0 & 0 & 0 & u_i^a & v_i^a & 1 & -u_i^a v_i^b & -v_i^a v_i^b & -v_i^b \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{\mathbf{A}} \mathbf{h} = \mathbf{0}$$

When estimating the homography from only 4 point correspondences in a non-degenerate configuration, $\mathbf{A}$ will always have rank 8 and its smallest singular value will always be 0

So by choosing $\mathbf{h} = \mathbf{v}_9$ we are guaranteed that $\|\mathbf{Ah}\| \equiv \mathbf{0}$ (at least up to numerical precision)

This means that we are guaranteed to find a perfect homography for the 4 given point correspondences!

Homography estimation
Direct Linear Transform

Algorithm – DLT

$$\underbrace{\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_i^a & v_i^a & 1 & 0 & 0 & 0 & -u_i^a u_i^b & -v_i^a u_i^b & -u_i^b \\ 0 & 0 & 0 & u_i^a & v_i^a & 1 & -u_i^a v_i^b & -v_i^a v_i^b & -v_i^b \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{\mathbf{A}} \mathbf{h} = \mathbf{0}$$

1. Build the matrix $\mathbf{A}$ from at least 4 point-correspondences $(u_i^a, v_i^a) \leftrightarrow (u_i^b, v_i^b)$
2. Obtain the SVD of $\mathbf{A}$: $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$
3. If $\mathbf{S}$ is diagonal with positive values in descending order along the main diagonal, then $\mathbf{h}$ equals the last column of $\mathbf{V}$
4. Reconstruct $\mathbf{H}_{ba}$ from $\mathbf{h}$

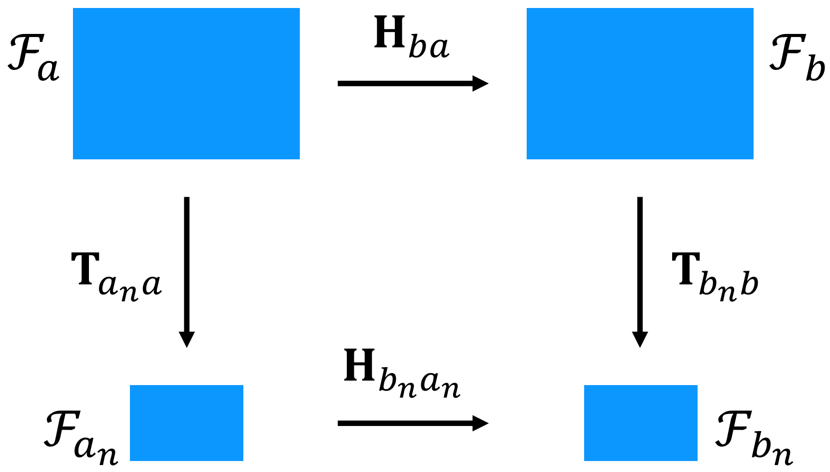
$$\mathbf{H}_{ba} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix}$$

Homography estimation
Normalized Direct Linear Transform

For more than 4 point correspondences, the basic DLT algorithm is however not recommended

The terms of matrix $\mathbf{A}$ will in general be of very different orders of magnitude $[10^0, 10^6]$
This causes errors in the point positions to have very different impact on the estimation

To alleviate this, it is custom perform the estimation on normalized image coordinates instead



$$\mathbf{H}_{ba} = \mathbf{T}_{b_n b}^{-1} \mathbf{H}_{b_n a_n} \mathbf{T}_{a_n a}$$

Estimate with basic DLT

Homography estimation
Normalized Direct Linear Transform

For more than 4 point correspondences, the basic DLT algorithm is however not recommended

The terms of matrix $\mathbf{A}$ will in general be of very different orders of magnitude $[10^0, 10^6]$
This causes errors in the point positions to have very different impact on the estimation

To alleviate this, it is custom perform the estimation on normalized image coordinates instead

The normalizing transformations can be created in several ways, e.g. based on the mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ of the two point sets $\{(u_i^a, v_i^a)\}$ and $\{(u_i^b, v_i^b)\}$

$$\mathbf{T}_{a_n a} = \begin{bmatrix} \boldsymbol{\Sigma}_a^{-\frac{1}{2}} & -\boldsymbol{\Sigma}_a^{-\frac{1}{2}}\boldsymbol{\mu}_a \\ \mathbf{0} & 1 \end{bmatrix} \qquad \mathbf{T}_{b_n b} = \begin{bmatrix} \boldsymbol{\Sigma}_b^{-\frac{1}{2}} & -\boldsymbol{\Sigma}_b^{-\frac{1}{2}}\boldsymbol{\mu}_b \\ \mathbf{0} & 1 \end{bmatrix}$$

Homography estimation
Normalized Direct Linear Transform

For more than 4 point correspondences, the basic DLT algorithm is however not recommended

The terms of matrix $\mathbf{A}$ will in general be of very different orders of magnitude $[10^0, 10^6]$
This causes errors in the point positions to have very different impact on the estimation

To alleviate this, it is custom perform the estimation on normalized image coordinates instead

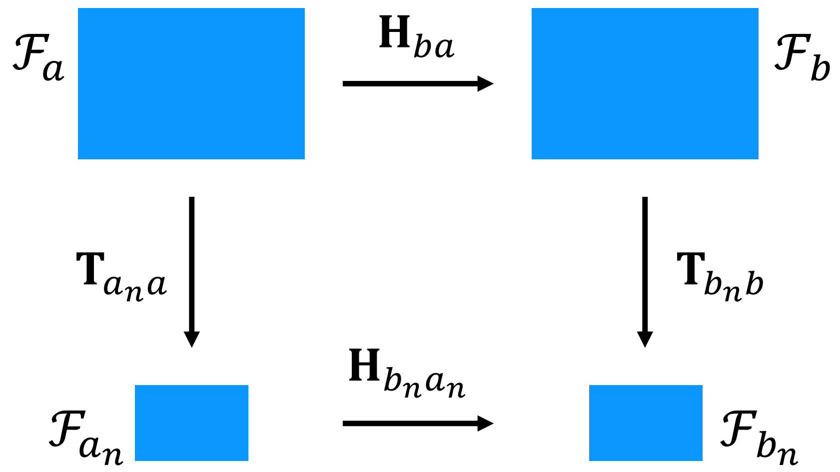
The normalizing transformations can be created in several ways, e.g. based on the mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ of the two point sets $\{(u_i^a, v_i^a)\}$ and $\{(u_i^b, v_i^b)\}$

Matrix squareroot of $\boldsymbol{\Sigma}^{-1}$

$$\mathbf{T}_{a_n a} = \begin{bmatrix} \boldsymbol{\Sigma}_a^{-\frac{1}{2}} & -\boldsymbol{\Sigma}_a^{-\frac{1}{2}}\boldsymbol{\mu}_a \\ \mathbf{0} & 1 \end{bmatrix} \qquad \mathbf{T}_{b_n b} = \begin{bmatrix} \boldsymbol{\Sigma}_b^{-\frac{1}{2}} & -\boldsymbol{\Sigma}_b^{-\frac{1}{2}}\boldsymbol{\mu}_b \\ \mathbf{0} & 1 \end{bmatrix}$$

Homography estimation
Normalized Direct Linear Transform

Algorithm – Normalized DLT



1. Normalize the pointsets $\{(u_i^a, v_i^a)\}$ and $\{(u_i^b, v_i^b)\}$ with $\mathbf{T}_{a_n a}$ and $\mathbf{T}_{b_n b}$
2. Estimate the homography $\mathbf{H}_{b_n a_n}$ from the normalized point correspondences $(u_i^{a_n}, v_i^{a_n}) \leftrightarrow (u_i^{b_n}, v_i^{b_n})$ using the DLT algorithm
3. Compute the homography $\mathbf{H}_{ba} = \mathbf{T}_{b_n b}^{-1} \mathbf{H}_{b_n a_n} \mathbf{T}_{a_n a}$

Homography estimation
Errors

How do we know if an estimated homography is good or bad?

We really want to estimate the homography in a RANSAC scheme, but how can we determine if a given point correspondence $(u_i^a, v_i^a) \leftrightarrow (u_i^b, v_i^b)$ is an inlier or an outlier?

Homography estimation
Errors

How do we know if an estimated homography is good or bad?

We really want to estimate the homography in a RANSAC scheme, but how can we determine if a given point correspondence $(u_i^a, v_i^a) \leftrightarrow (u_i^b, v_i^b)$ is an inlier or an outlier?

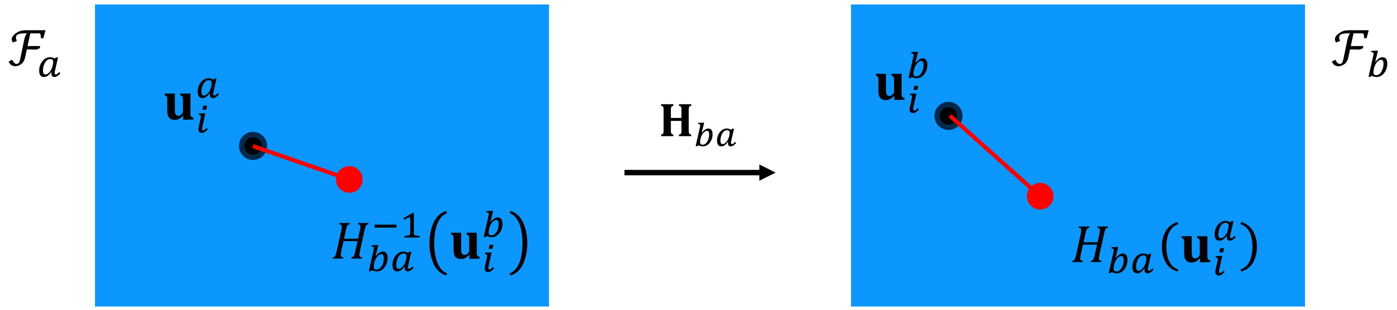
Algebraic error for a point correspondence $(u_i^a, v_i^a) \leftrightarrow (u_i^b, v_i^b)$

$$\varepsilon_i = \|\mathbf{A}_i \mathbf{h}\| \quad \text{where} \quad \mathbf{A}_i = \begin{bmatrix} u_i^a & v_i^a & 1 & 0 & 0 & 0 & -u_i^a u_i^b & -v_i^a u_i^b & -u_i^b \\ 0 & 0 & 0 & u_i^a & v_i^a & 1 & -u_i^a v_i^b & -v_i^a v_i^b & -v_i^b \end{bmatrix}$$

Total squared algebraic error for the homography

$$\varepsilon^2 = \sum_i \|\mathbf{A}_i \mathbf{h}\|^2 = \|\mathbf{A}\mathbf{h}\|^2$$

Homography estimation
Errors



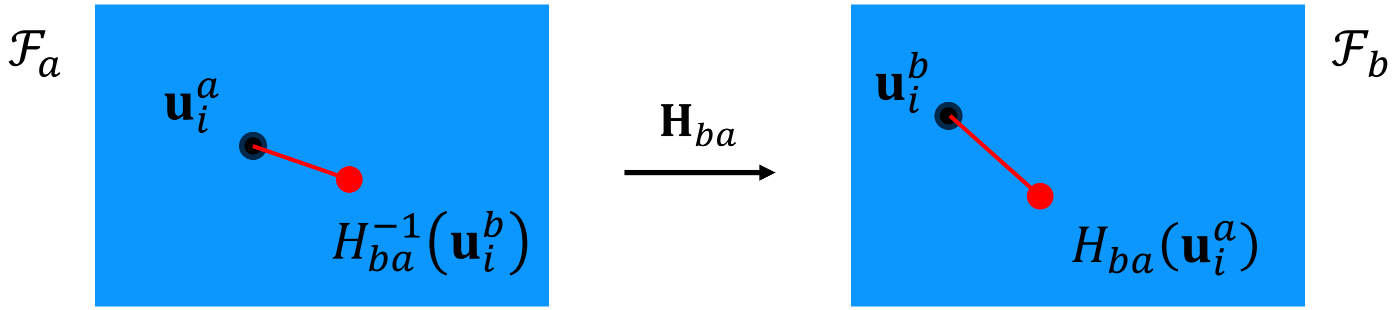
Geometric error for a point correspondence $(u_i^a, v_i^a) \leftrightarrow (u_i^b, v_i^b)$

$$\varepsilon_i = d\left(\mathbf{u}_i^a, \mathbf{H}_{ba}^{-1}(\mathbf{u}_i^b)\right) + d\left(\mathbf{u}_i^b, \mathbf{H}_{ba}(\mathbf{u}_i^a)\right)$$ where $d(\cdot, \cdot)$ is the Euclidean distance

Total squared geometric error for the homography

$$\varepsilon^2 = \sum_i \varepsilon_i^2$$

Homography estimation
Errors



Geometric error for a point correspondence $(u_i^a, v_i^a) \leftrightarrow (u_i^b, v_i^b)$

$$\varepsilon_i = d\left(\mathbf{u}_i^a, \mathbf{H}_{ba}^{-1}(\mathbf{u}_i^b)\right) + d\left(\mathbf{u}_i^b, \mathbf{H}_{ba}(\mathbf{u}_i^a)\right)$$ where $d(\cdot,\ \cdot)$ is the Euclidean distance

Total squared geometric error for the homogra

$$\varepsilon^2 = \sum_i \varepsilon_i^2$$

If we're only concerned with the error in one of the images, we can also consider a one-sided geometric error

$$\varepsilon_i = d\left(\mathbf{u}_i^a, \mathbf{H}_{ba}^{-1}(\mathbf{u}_i^b)\right)$$

$$\varepsilon_i = d\left(\mathbf{u}_i^b, \mathbf{H}_{ba}(\mathbf{u}_i^a)\right)$$

Homography estimation
Errors

Algebraic error

- $\varepsilon_i = \|\mathbf{A}_i \mathbf{h}\|$
- Not physically meaningful
- Estimating the homography with minimal algebraic error, is easy (DLT) and well suited for use in a RANSAC estimation scheme

Geometric error

- $\varepsilon_i = d\left(\mathbf{u}_i^a, \mathbf{H}_{ba}^{-1}(\mathbf{u}_i^b)\right) + d\left(\mathbf{u}_i^b, \mathbf{H}_{ba}(\mathbf{u}_i^a)\right)$
- Physically meaningful
- Estimating the homography with minimal geometric error is a non-linear least squares problem and requires iterative estimation techniques

Homography estimation
RANSAC

Algorithm – RANSAC

For a set of point-correspondences $S = \{(u_i^a, v_i^a) \leftrightarrow (u_i^b, v_i^b)\}$, perform N iterations

1. Compute $\mathbf{H}_{ba,tst}$ for 4 random correspondences with the DLT algorithm
2. Determine the set of inliers for $\mathbf{H}_{ba,tst}$

$$S_{tst} = \{(u_i^a, v_i^a) \leftrightarrow (u_i^b, v_i^b) \; s.t. \; \varepsilon_i < t\}$$

   Here ε_i is the geometric error and t is a chosen value for max acceptable error

3. If S_{tst} is the largest set of inliers so far

$$S_{IN} = S_{tst}$$

$$\mathbf{H}_{ba} = \mathbf{H}_{ba,tst}$$

$$N = \frac{log(1-p)}{log(1-\omega^n)} \quad \text{where } \omega = \frac{|S_{IN}|}{|S|}, n = 4 \text{ and } p = 0.99$$

Afterwards, estimate the homography $\mathbf{H}_{ba}$ based on only inlier-correspondences

- Minimal algebraic error – normalized DLT
- Minimal geometric error – iterative, non-linear least squares

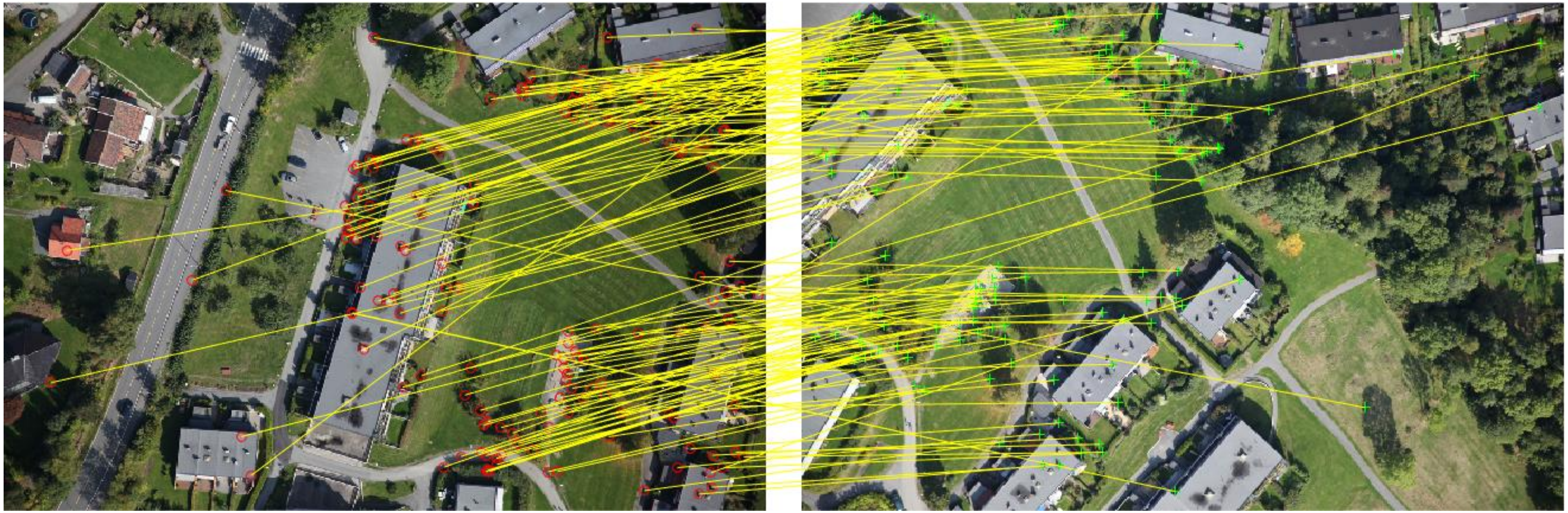
Creating an image mosaic from two images

Let us compose these two images into a larger image, an image mosaic

Creating an image mosaic from two images

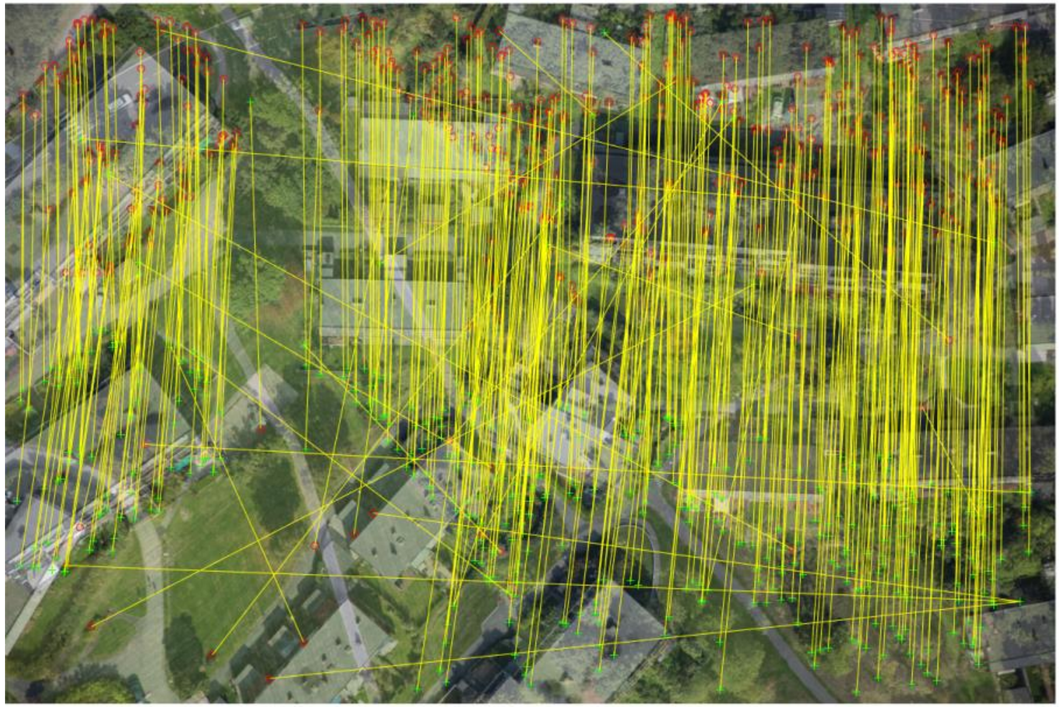
We start by finding key points and representing them by descriptors

Creating an image mosaic from two images



Establish point-correspondences by matching descriptors

Creating an image mosaic from two images

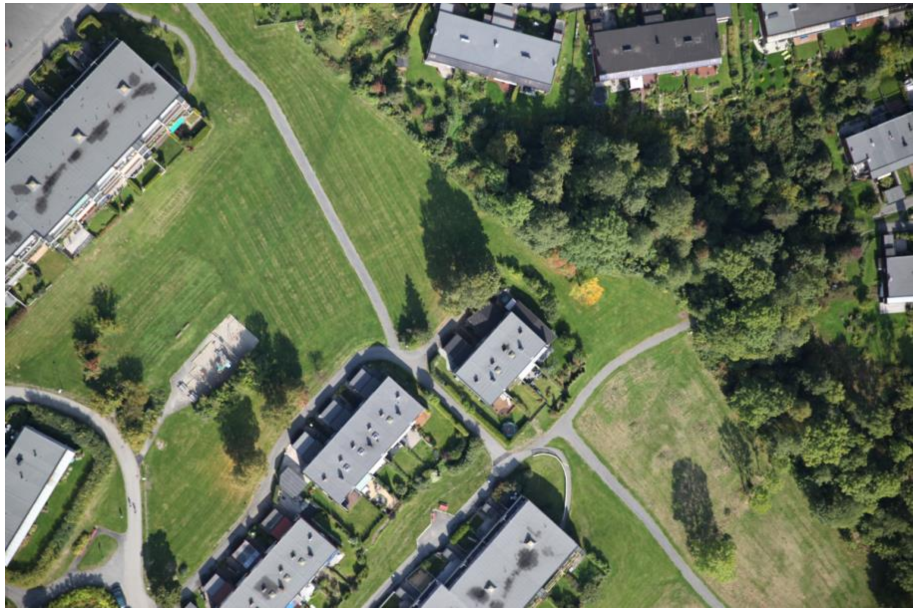


Some bad matches!

Establish point-correspondences by matching descriptors

Creating an image mosaic from two images

$\mathbf{H}_{ba}$ →

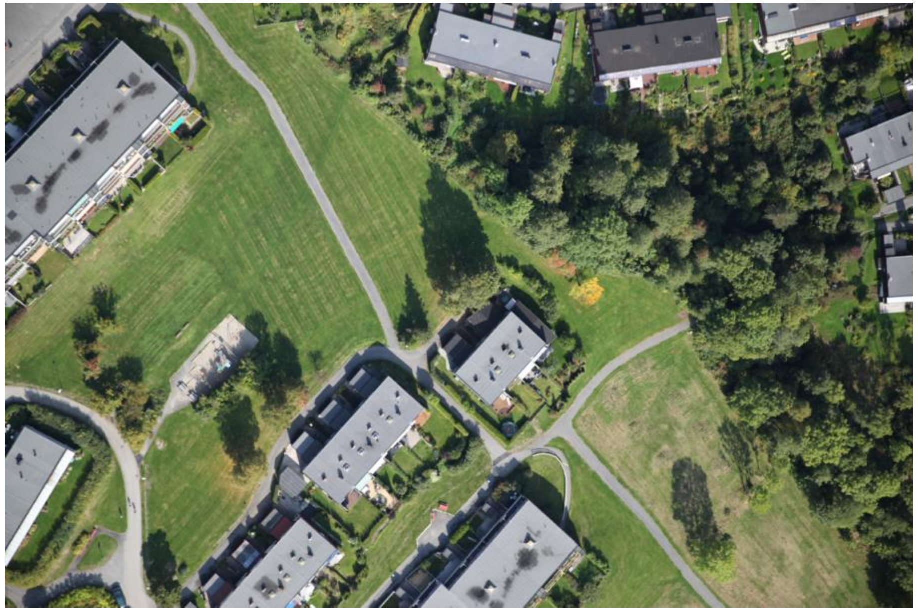


Determine the inlier set of point correspondences by estimating the homography in a RANSAC scheme

Estimate the homography $\mathbf{H}_{ba}$ based on inlier correspondences only

Creating an image mosaic from two images

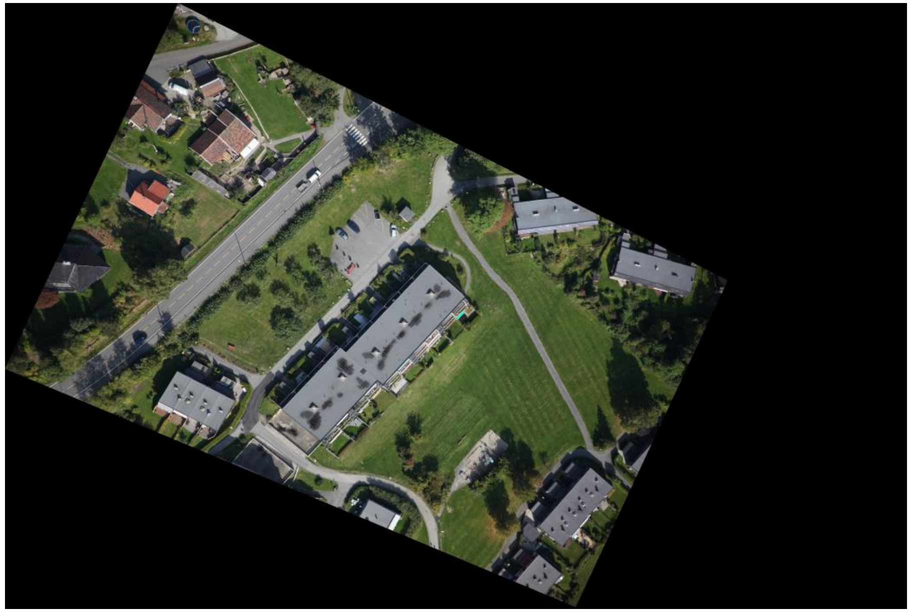
$\xrightarrow{\mathbf{H}_{ba}}$



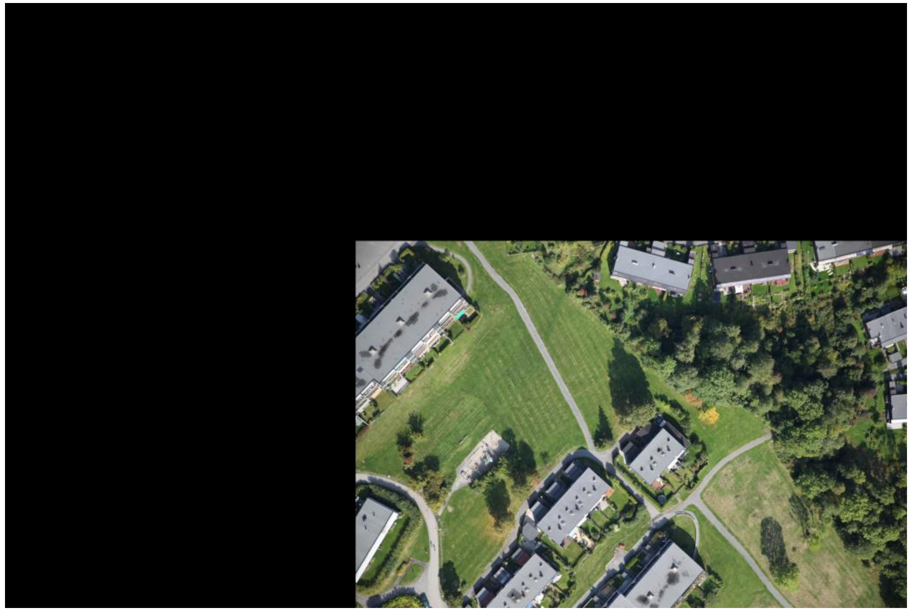
Now we could use the homography $\mathbf{H}_{ba}$ to transform (warp) the left image into the right image's coordinate system

But, if we want to see the full mosaic it is necessary to transform both images into a more suitable coordinate system

Creating an image mosaic from two images



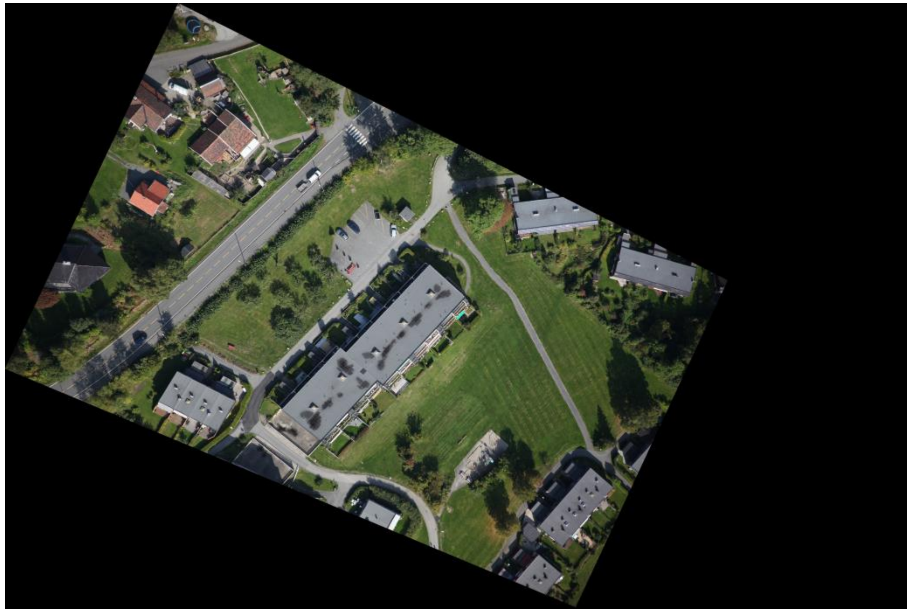
$$\xrightarrow{\mathbf{T}_{mb}\mathbf{H}_{ba}}$$



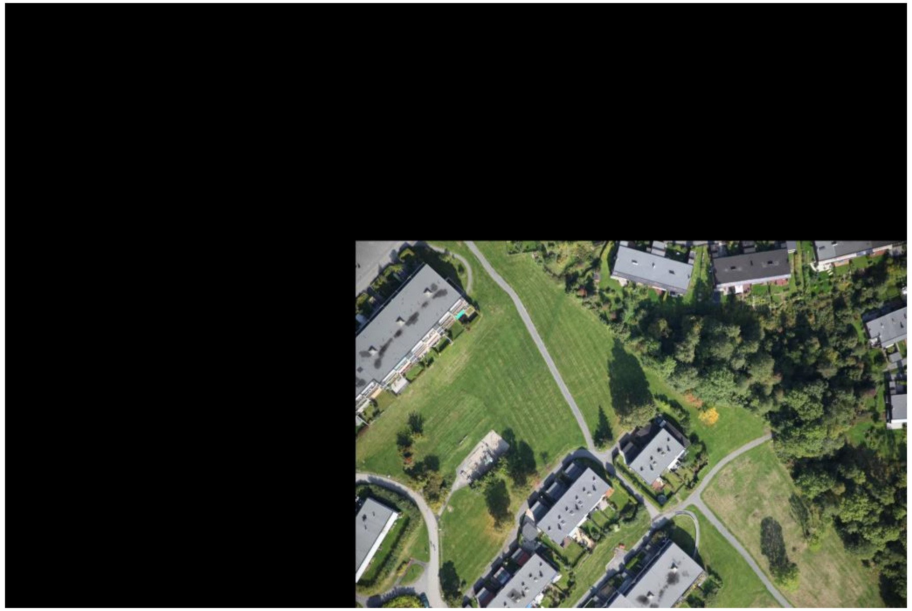
Here we've chosen to transform both images into a shifted version of the right image's coordinate system

If $\mathbf{T}_{mb}$ is the transformation from the right image to this new "mosaic image", the transformation of the left image is given by $\mathbf{T}_{mb}\mathbf{H}_{ba}$

Creating an image mosaic from two images



$$\xrightarrow{\mathbf{T}_{mb}\mathbf{H}_{ba}}$$



Once both images are represented in the same coordinates, we can choose to compose them in several different ways

Overwriting

Blending with a ramp

Blending with a ramp + histogram equalization

Summary

Often reasonable to assume that two overlapping perspective images are related by a homography

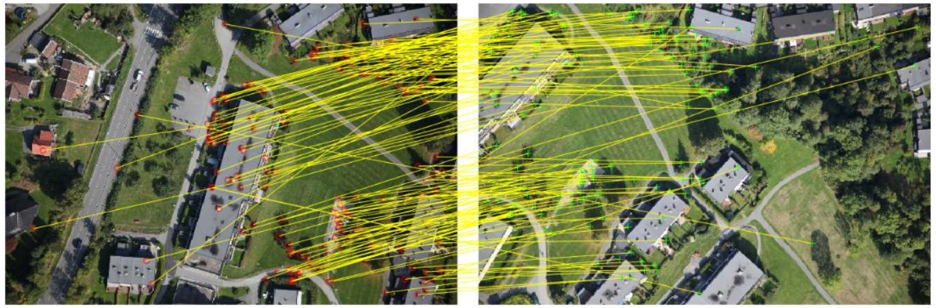
- Planar or almost planar scene
- Purely rotating or almost purely rotating camera

Homography estimation

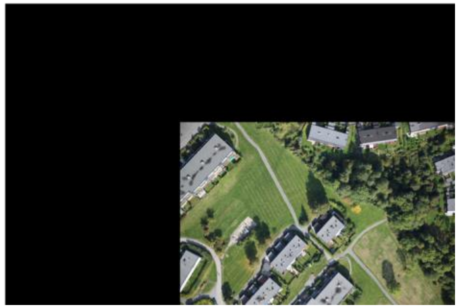
1. Establish point correspondences
2. RANSAC estimation of homography to remove bad correspondences
3. Estimate homography based on good correspondences
   - Minimal algebraic error: Normalized DLT
   - Minimal geometric error: Iterative, non-linear least squares

Image mosaic from two images

The homography can be used to transform both images into a common coordinate frame



$\xrightarrow{\mathbf{T}_{mb}\mathbf{H}_{ba}}$



Supplementary material

Recommended

- *Richard Szeliski: Computer Vision: Algorithms and Applications 2nd ed*
  - Chapter 8 “Image alignment and stitching” and in particular section 8.2 about “image stitching”