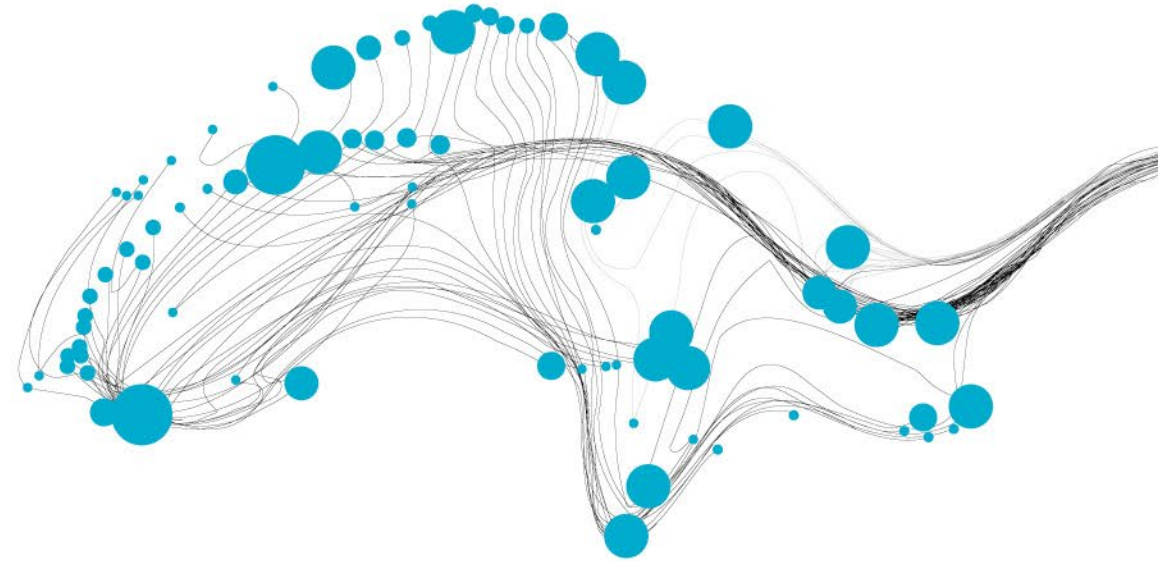


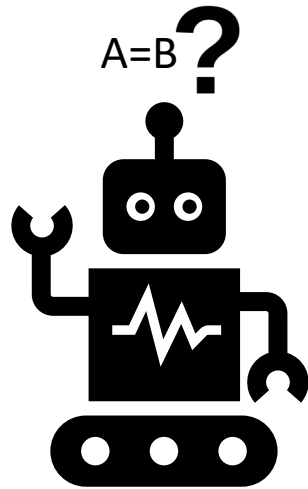
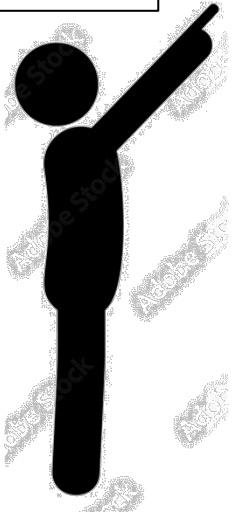
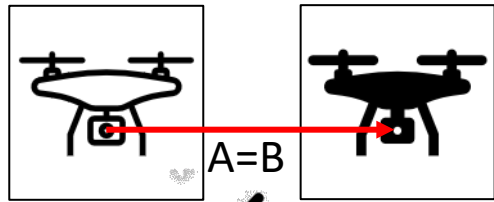
# Image Orientation

BASHAR ALSADIK  
ITC - UAV CENTER



# Introduction

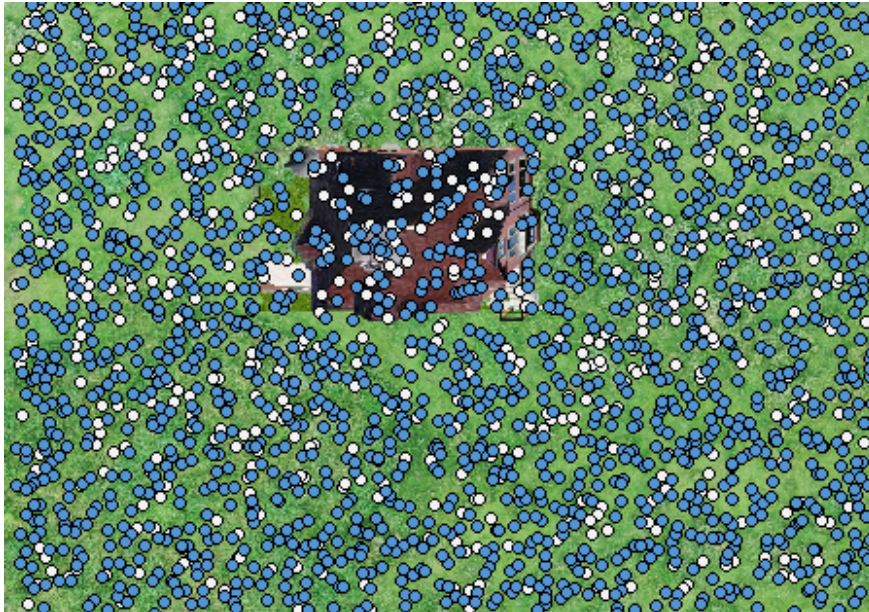
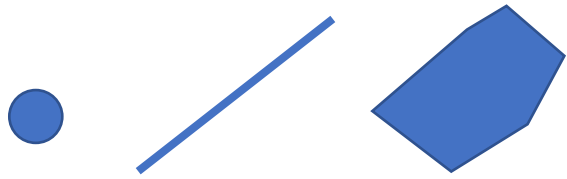
- What is image matching?
- Why we need image matching?



- Establishing Correspondence Between Image Points
- Homologous Points in Pairs of Images
- Enabling Quick Identification of Matching Features
- Support for Structure from Motion and Image Orientation

# Image Features

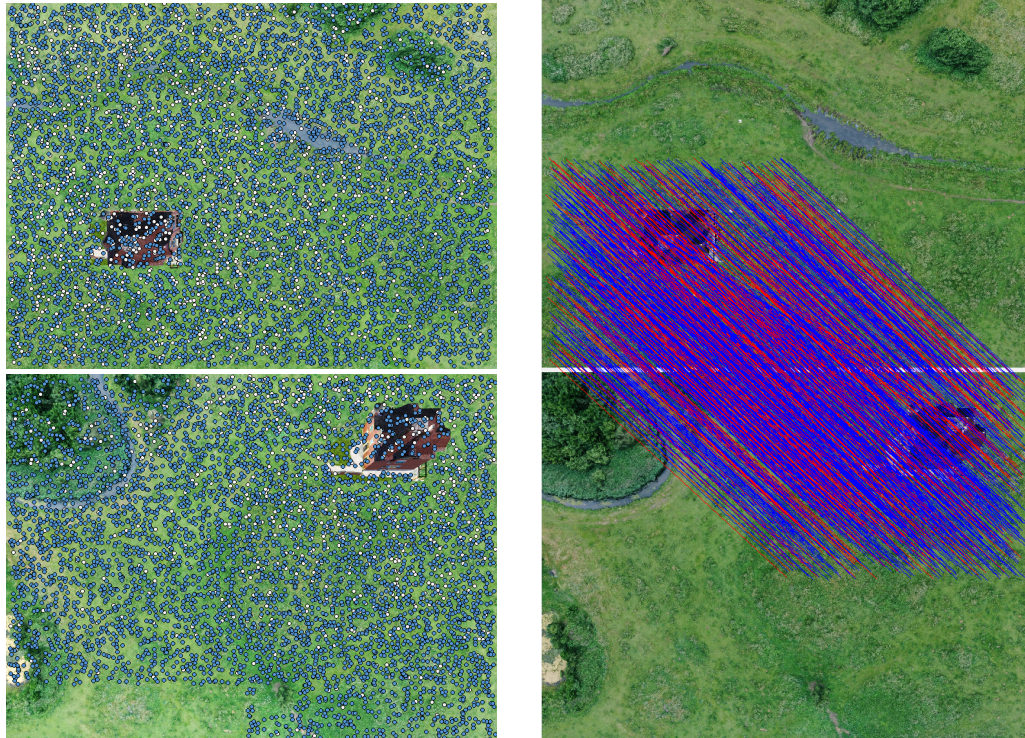
- As shown in the image below the blue points show the features detected.



Features are elements or patterns that assist identify an item in an image. Features include objects like ridges, corners, edges, and points of interest.



# From Feature To Correspondence

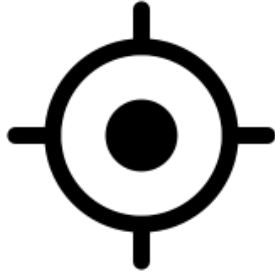


## Feature matching

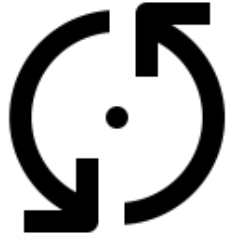
In a last step, the description of every feature in one image is used to compare it with the description of every feature in the second image. If two descriptions are similar, it is a match



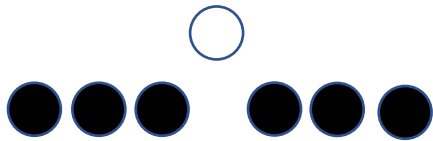
# How to Apply a Good Feature Detector?



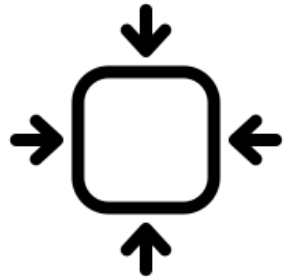
Locality



Repeatability



Distinctiveness



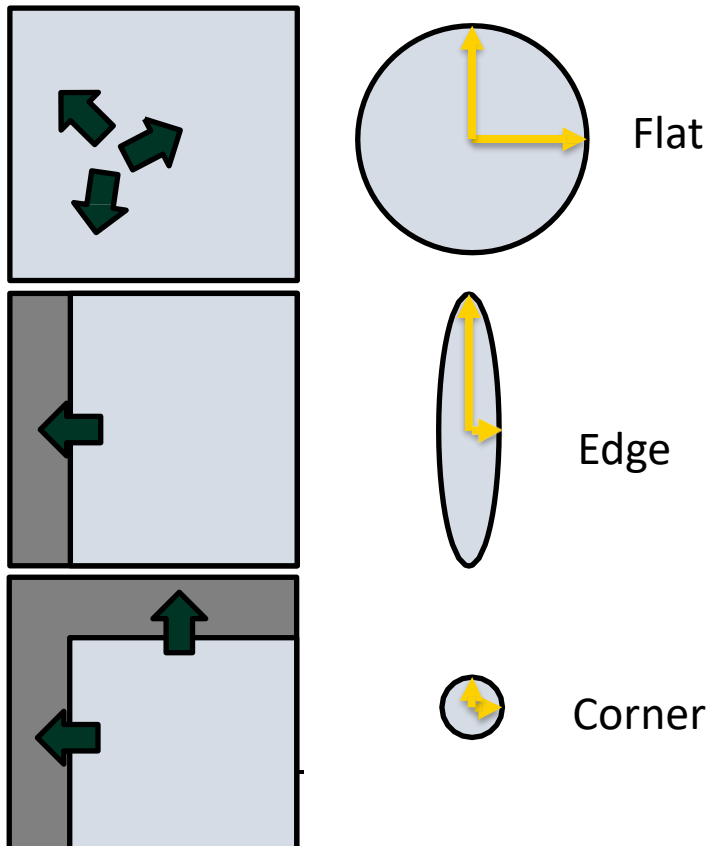
Efficiency

**Good features should have the following characteristics**

- Locality — robustness to clutter and occlusion, a feature should only occupy a small area of an image
- Compactness — efficiency, many fewer features than the number of pixels in the image
- Repeatability — The same feature can be found in several images despite geometric and photometric transformations
- Saliency or distinctiveness: The computed features should be invariant to geometric and photometric differences between the two images- this is achieved in the description stage

# Available interest point detectors

- Corners/interest points: repeatable and distinctive
- Blobs/region of interest



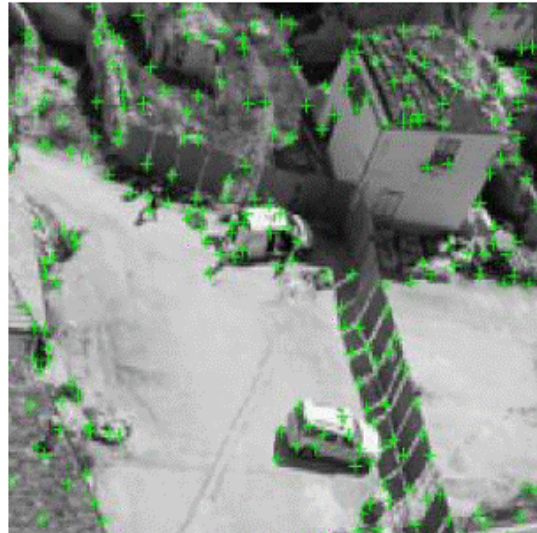
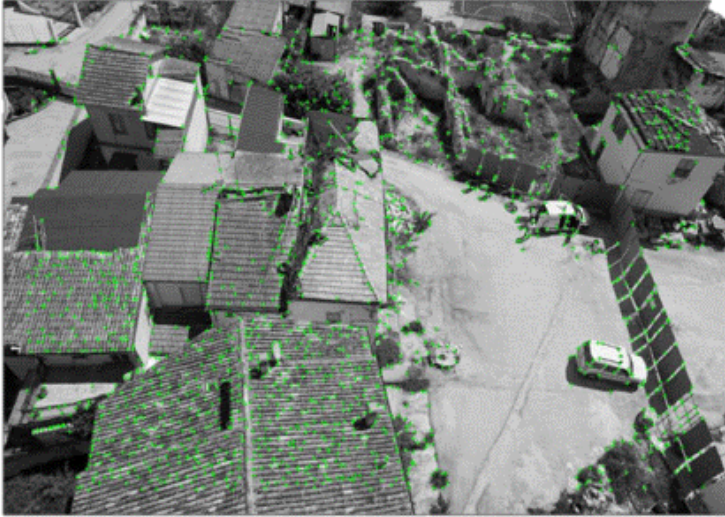
## Well-Known Corner Detection Methods

- Classic Approaches: Harris & Förstner
- Standard Technique: FAST
- Latest Development: AGAST

## Corners: Crucial and Distinct Features

- Corners with Multiple Dominant Gradient Directions
- High Recognition Value for Image Features
- Change in Intensity with Shifting Window

# Interest points and Blobs



- **Blobs** are small image regions which share similar image properties, such as brightness or colour.
- blobs represent a region with at least one extremum, positive or negative -> bright or dark
- The size of a blob is defined by the intensity in the region.
- Known detectors for blobs could be the Laplacian of Gaussian (LoG), Difference of Gaussian (DoG) & Determinant of Hessian (DoH), Maximally Stable Extremal Regions (MSER)



# FEATURE DESCRIPTION

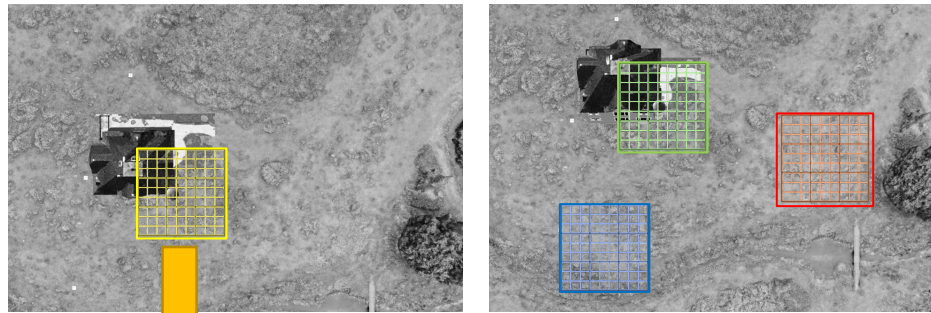
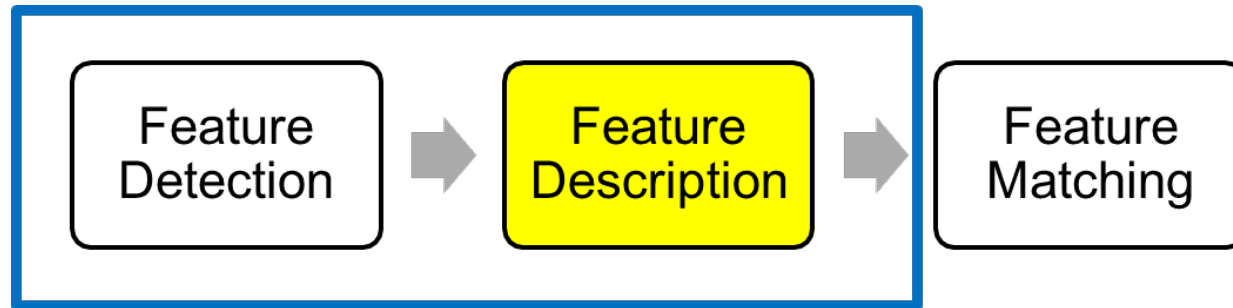
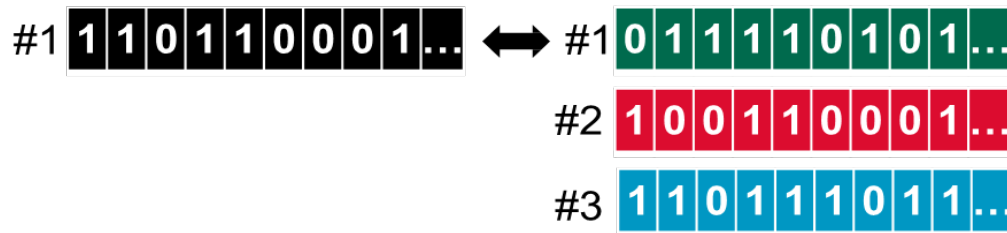


Image A

Image B



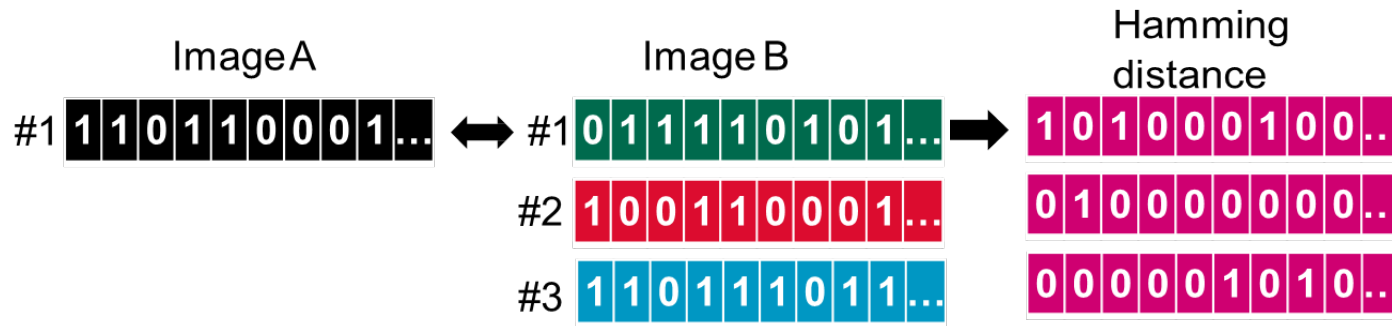
## •Feature Description in Image Matching

- Once a feature is identified, it needs to be "described" for cross-image matching.
- Descriptors capture pixel neighborhood information and convert it into a compact vector.
- Information can include image gradients or intensity comparisons.

## •Descriptor Varieties and Invariances

- Descriptors vary based on detector and method used.
- Account for invariances like scale, rotation, and lighting variations.

# FEATURE MATCHING



- Matching involves identifying similarities between images of the same scene or object.
- Descriptor distances are computed to measure similarity.
- Distances act as similarity metrics (e.g., Hamming distance).

ImageA	Image B	Desc. Dist
1	1	3
1	2	1
1	3	2

**Match!**

# FEATURE MATCHING

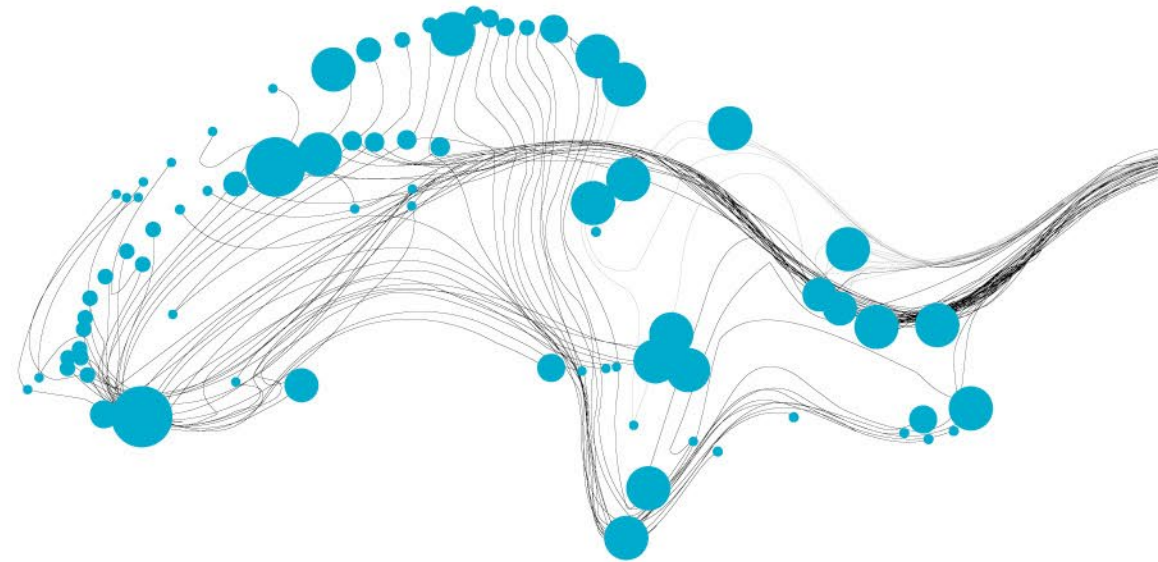
- Matching results may include incorrect matches, known as outliers.
- Outliers deviate from the mapping function that transforms image coordinates between images.



# STRUCTURE FROM MOTION (SFM)

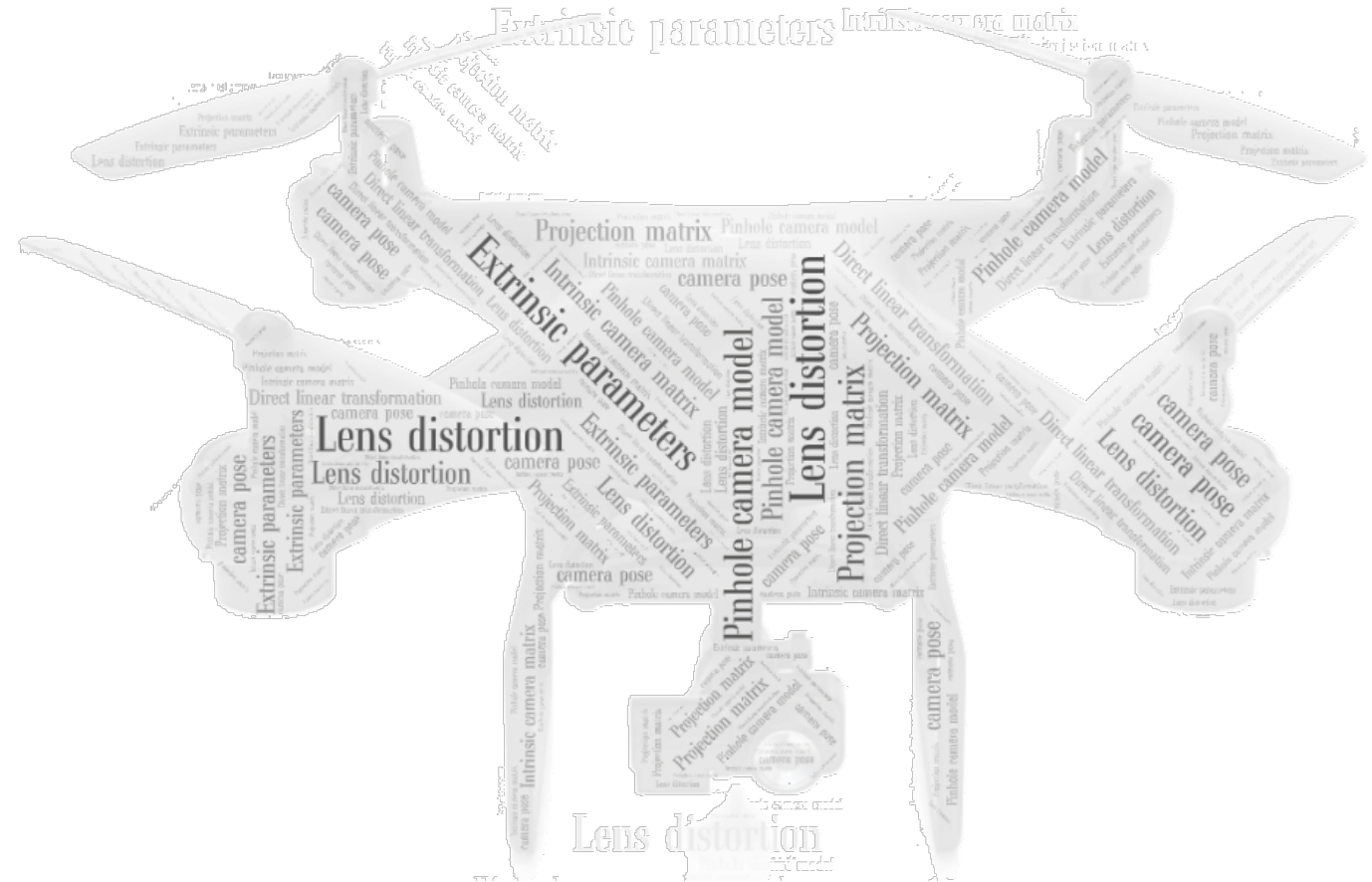
SINGLE IMAGE ORIENTATION

SLIDES BY B. ALSADIK



# CONTENTS

- Camera settings
- Pinhole camera model
- Collinearity equations
- Projection matrix
- Direct linear transformation
- Intrinsic camera matrix
- 3D Rotation
- Extrinsic parameters & camera pose



# CAMERA TYPES

- RGB: DSLR, compact, MILC
- Multispectral
- Thermal



FLIR Duo Pro R



FLIR Duo & Duo R



FLIR Vue Pro



FLIR Vue Pro R



FLIR Aerial Thermal Imaging Kits



DJI Zenmuse XT

UAV thermal cameras



(A) DJI Zenmuse X7; (B) MAPIR Survey3 (also available in multispectral option); (C) PhaseOne iXU-RS 1000; (D) Sony ILCE-QX1; (E) senseFly S.O.D.A



UAV Multispectral cameras

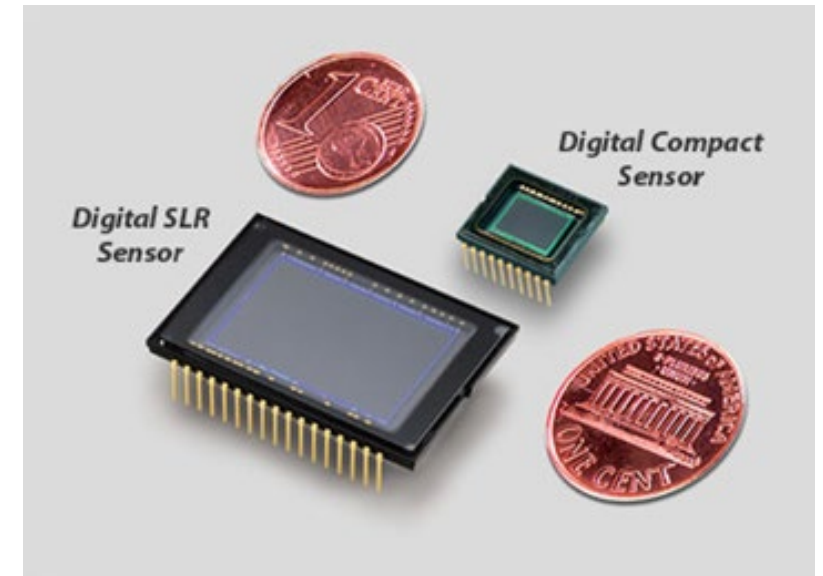


# CAMERA SENSORS

- **Sensor Size:** Larger sensors will have better light gathering ability at the same resolutions while smaller sensors will need greater exposure times to achieve the same effective outcome.

For UAV inspections it is important to have a high-resolution camera, with a larger sensor which allows to:

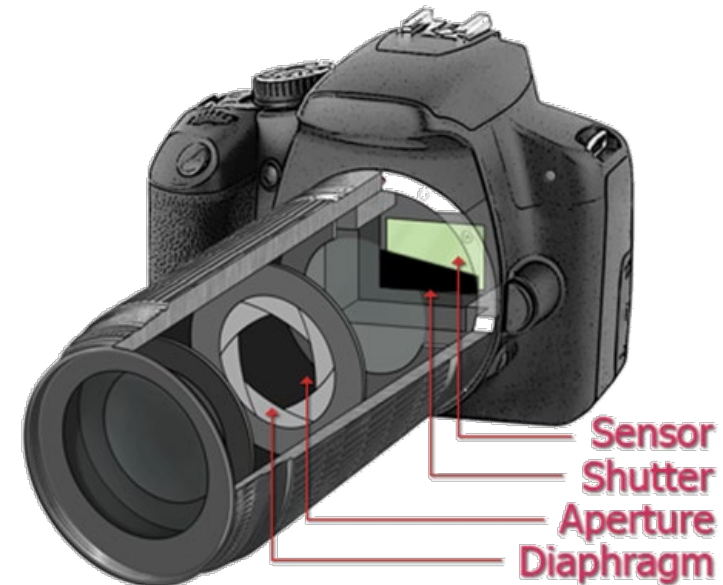
- Provide more details about any point of the inspected object.
- Increase image contrast to help getting better 3D models and point clouds.



# CAMERA SETTINGS

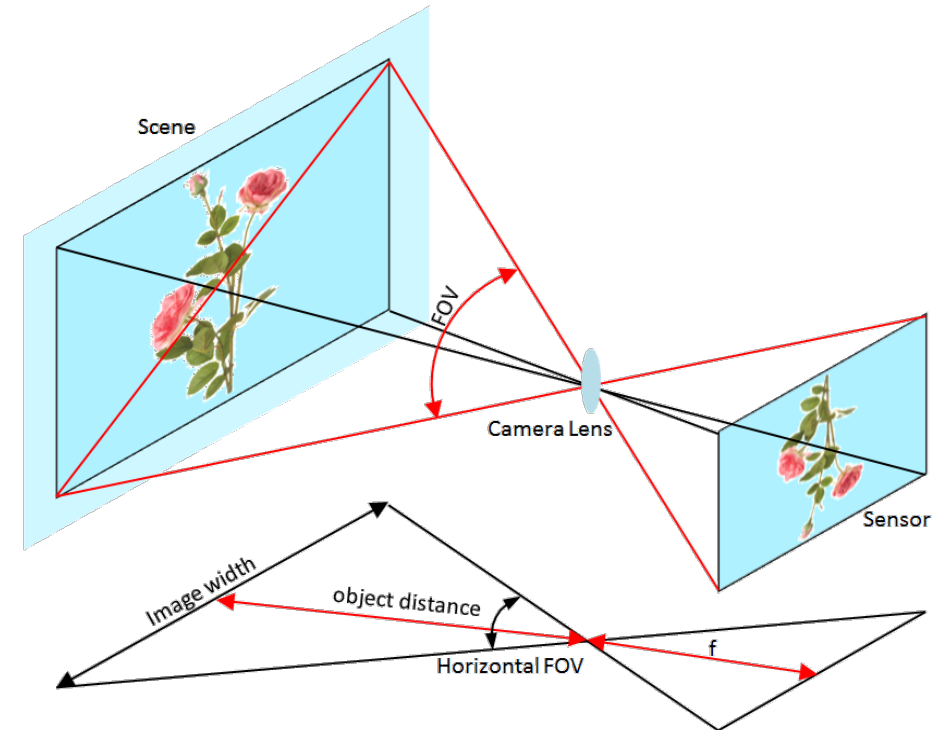
- **Depth of field:** Depth of field (DOF) refers to the areas of the photograph both in front and behind the focus point which remains 'sharp' (in focus).
- **Aperture:** Aperture refers to the size of the opening in the lens that controls the amount of light reaching the sensor. The relation between aperture and the focal length is called f-number or **f-stop**=focal length / aperture diameter Adjust the aperture value helps to match lighting conditions to effectively expose sensor and obtain the sharpest images.

<http://www.canonoutsideofauto.ca/play/>



# FIELD OF VIEW FOV

- Preferable cameras will have a quick autofocus and short focal length so when you are above 20m everything will be in the infinity focus region.
- Shorter focal length lenses provide a wider field of view FOV but offer less magnification. On the other hand, longer focal lengths offer a smaller FOV, but provides larger magnification.

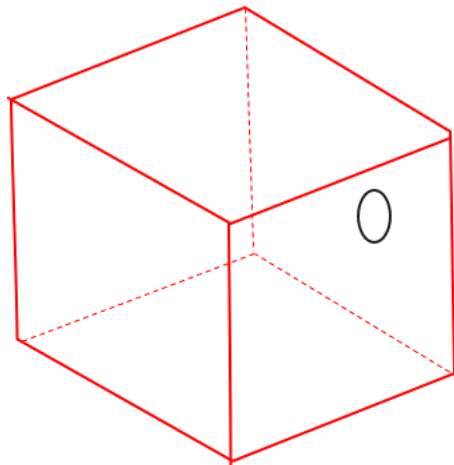


Try to experience the change of the focal length on the image FOV

<https://camerasim.com/camerasim-free-web-app/>

# PINHOLE CAMERA MODEL

The **pinhole camera model** describes the mathematical relationship between the coordinates of a 3D point and its projection onto the image plane



Please check the video describing the pinhole camera principle  
<https://www.youtube.com/watch?v=hhWVJ4SmkF0>



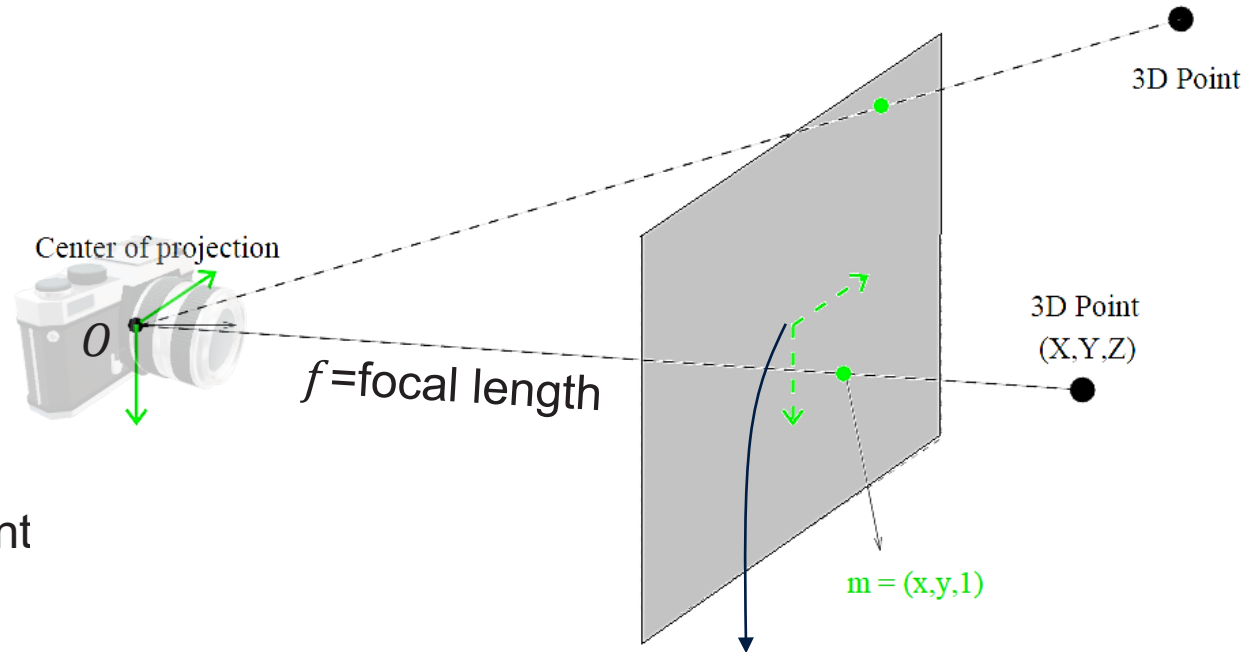
# PINHOLE CAMERA GEOMETRY

Mathematically, the central perspective projection modeled by the pinhole camera can be formulated using trigonometric proportion

$\bar{X}, \bar{Y}, \bar{Z}$  are the world coordinates.  
 $x, y$ , are the image coordinates.

From trigonometry:

$$x = f \left( \frac{\bar{X}}{\bar{Z}} \right); y = f \left( \frac{\bar{Y}}{\bar{Z}} \right)$$



- The distance from the optical center is the focal length  $f$
- $\bar{X}, \bar{Y}$ , and  $\bar{Z}$  are the world coordinates to the origin located at the camera perspective center  $O$  (lens).

$p.p = \text{principal point}$   
 $= \text{image center with coordinates } (u_0, v_0)$

# MATHEMATICAL MODELS OF THE PINHOLE CAMERA

**Collinearity equations** is based on the geometric condition that the object point  $A(X_A, Y_A, Z_A)$ , its projection on the image  $(x_a, y_a)$  and the camera lens  $(X_o, Y_o, Z_o)$  are all collinear (green dotted line)

$$x_a = x_o - f \frac{r_{11}(X_A - X_o) + r_{12}(Y_A - Y_o) + r_{13}(Z_A - Z_o)}{r_{31}(X_A - X_o) + r_{32}(Y_A - Y_o) + r_{33}(Z_A - Z_o)}$$

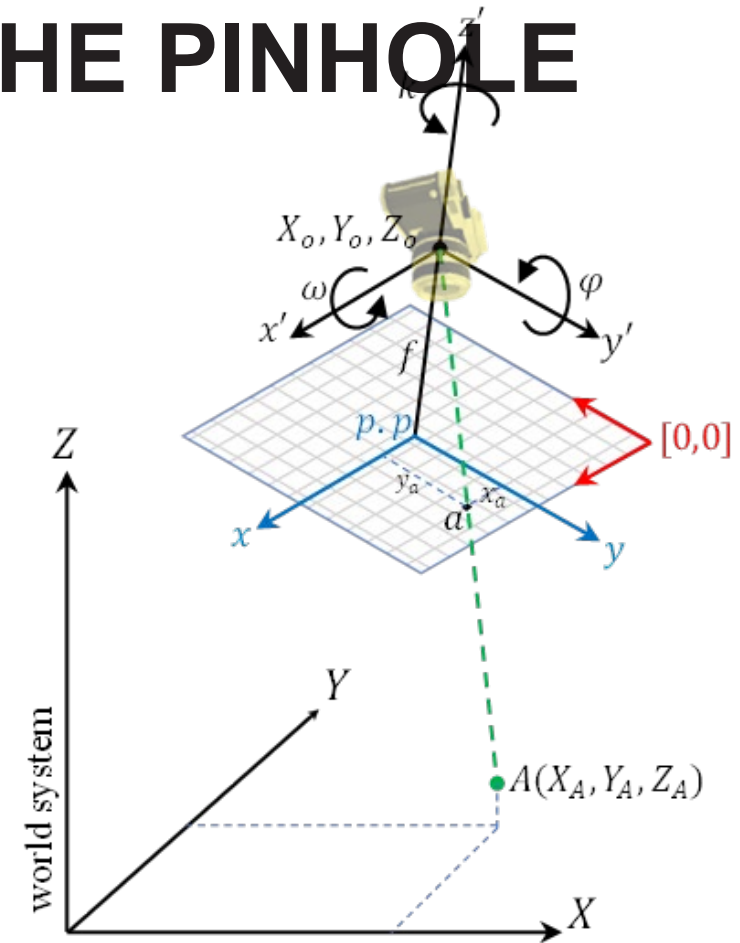
$$y_a = y_o - f \frac{r_{21}(X_A - X_o) + r_{22}(Y_A - Y_o) + r_{23}(Z_A - Z_o)}{r_{31}(X_A - X_o) + r_{32}(Y_A - Y_o) + r_{33}(Z_A - Z_o)}$$

Exterior orientation parameters

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, \quad T = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix}$$

rotation is a function of  $(\omega, \varphi, k)$  translation

$x_o, y_o, f$ : interior parameters



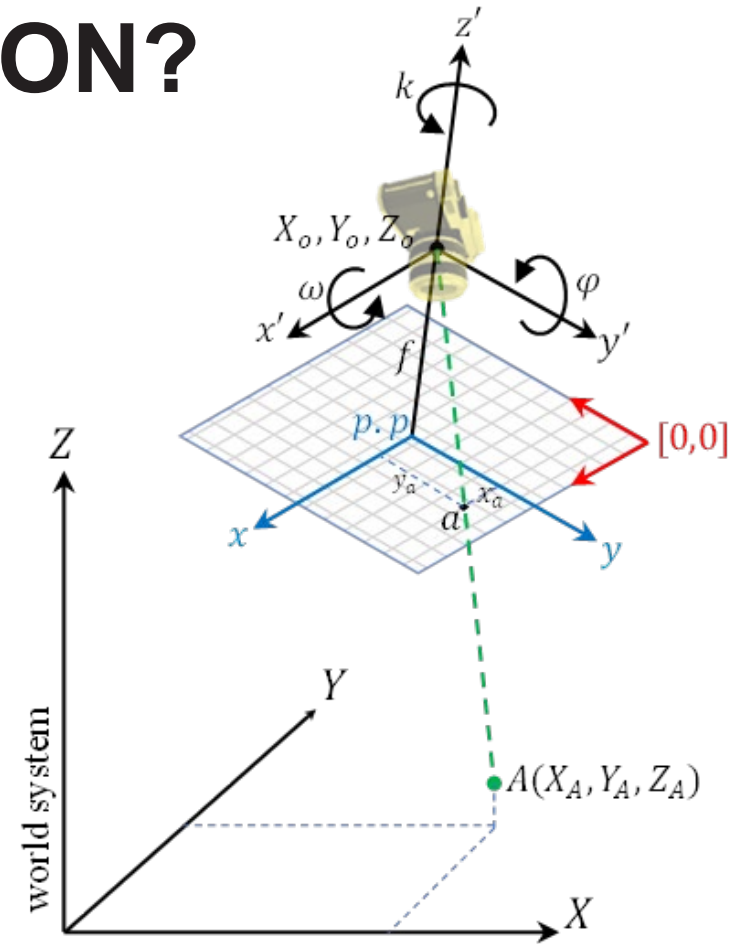
# WHAT IS EXTERIOR ORIENTATION?

- Both the camera position and angular orientation at the moment of taking the image are called the exterior orientation parameters (extrinsic).
- The extrinsic parameters represent a rigid transformation from 3-D world coordinate system to the 3-D camera's coordinate system.
- There are six extrinsic parameters 3 for rotation R and 3 for translation T.
- The determination of the extrinsic parameters will define the **camera pose**.

## Exterior orientation parameters

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, \quad T = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix}$$

*rotation is a function of  $(\omega, \varphi, k)$  translation*

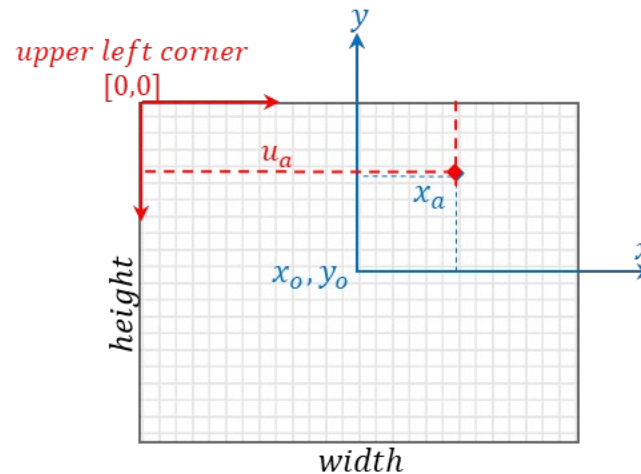


# WHAT IS INTERIOR ORIENTATION?

- The internal geometry of a camera as it existed at the moment of data capture is defined by its interior orientation
- camera parameters are mainly the focal length  $f$ , principal point  $p.p$  coordinates, and lens distortion parameters.
- The camera parameters are represented by a special matrix called the **intrinsic camera matrix K**.
- K matrix represents the transformation from image coordinates to pixel coordinates

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \text{ or } \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$x_0, y_0, f$ : interior parameters

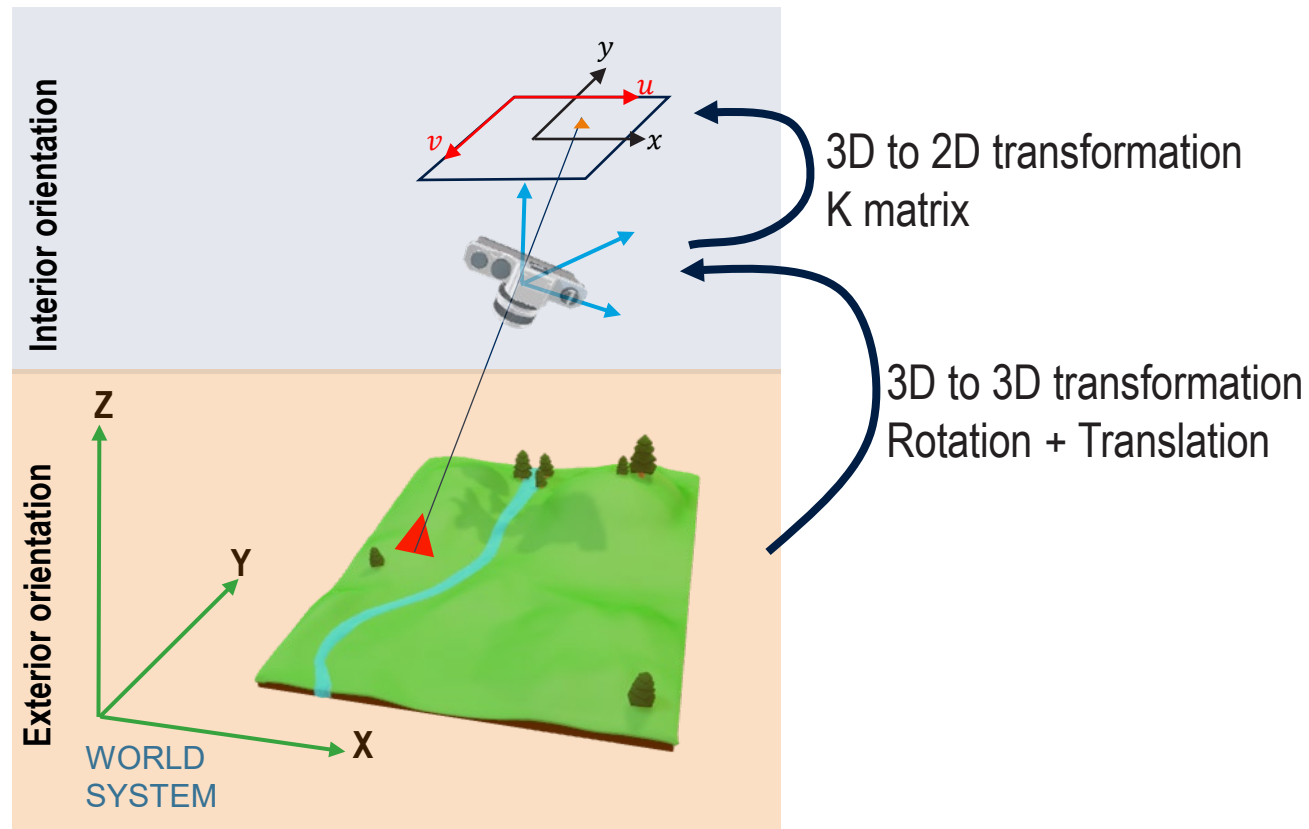


$$x_0 = \frac{width}{2}, y_0 = \frac{height}{2}$$



# EXTERIOR AND INTERIOR ORIENTATION

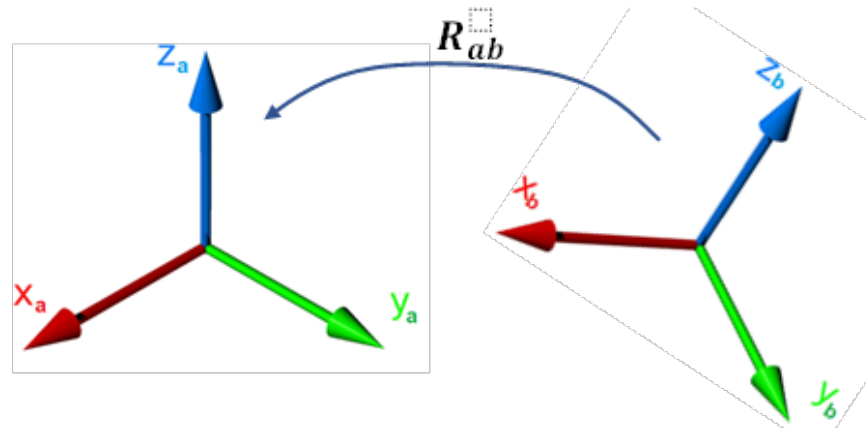
- The determination of the extrinsic parameters will define the **camera pose**.
- The determination of the intrinsic parameters will define the **camera calibration** parameters.



# ROTATION IN 3D

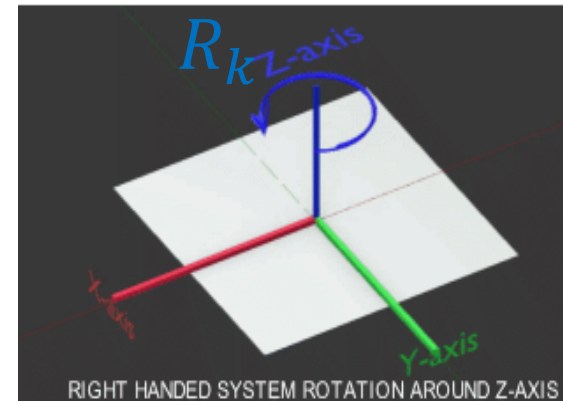
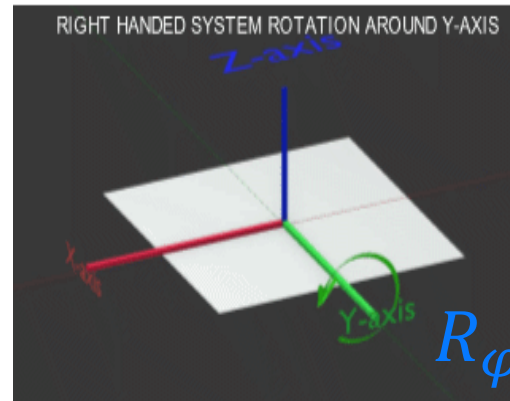
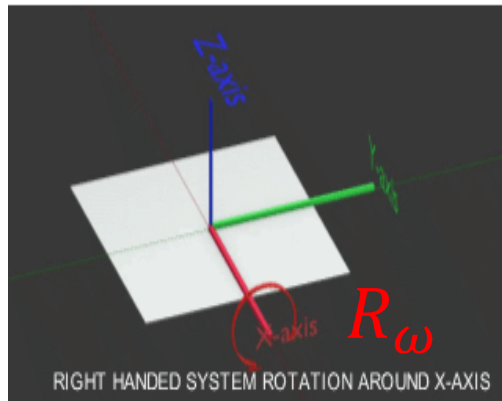
- Understanding the 3D rotation is essential in studying the topic of photogrammetry
- The rotation can be clockwise if a left-handed system is adopted or anticlockwise if a right-handed system
- The orientation of the camera coordinate system with respect to the world system can be represented by an orthonormal rotation matrix  $R$
- Point  $a$  can be transformed from  $x^a = [X_a \ Y_a \ Z_a]^t$  to point  $b$  as  $x^b = [X_b \ Y_b \ Z_b]^t$

$$x^a = R_{ab} x^b$$



# ROTATION IN 3D

Euler angles are typically denoted as omega  $\omega$ , phi  $\varphi$ , and kappa  $k$ . The  $\omega$  angle is applied around the X-axis,  $\varphi$  around the Y-axis, and  $k$  around the Z-axis



$$R_\varphi = \begin{bmatrix} \cos \varphi & 0 & -\sin \varphi \\ 0 & 1 & 0 \\ \sin \varphi & 0 & \cos \varphi \end{bmatrix}, \quad R_k = \begin{bmatrix} \cos k & \sin k & 0 \\ -\sin k & \cos k & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad R_\omega = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \omega & \sin \omega \\ 0 & -\sin \omega & \cos \omega \end{bmatrix}$$

- In the ZYX sequence, angles will be:

$$R_{ZYX} = R_k R_\varphi R_\omega = \begin{bmatrix} \cos \varphi \cos k & \cos \omega \sin k + \sin \omega \sin \varphi \cos k & \sin \omega \sin k - \cos \omega \sin \varphi \cos k \\ -\cos \varphi \sin k & \cos \omega \cos k - \sin \omega \sin \varphi \sin k & \sin \omega \cos k + \cos \omega \sin \varphi \sin k \\ \sin \varphi & -\sin \omega \cos \varphi & \cos \omega \cos \varphi \end{bmatrix}$$

# SPATIAL RESECTION – IMAGE POSE

- Spatial resection is the process of computing the six exterior parameters.

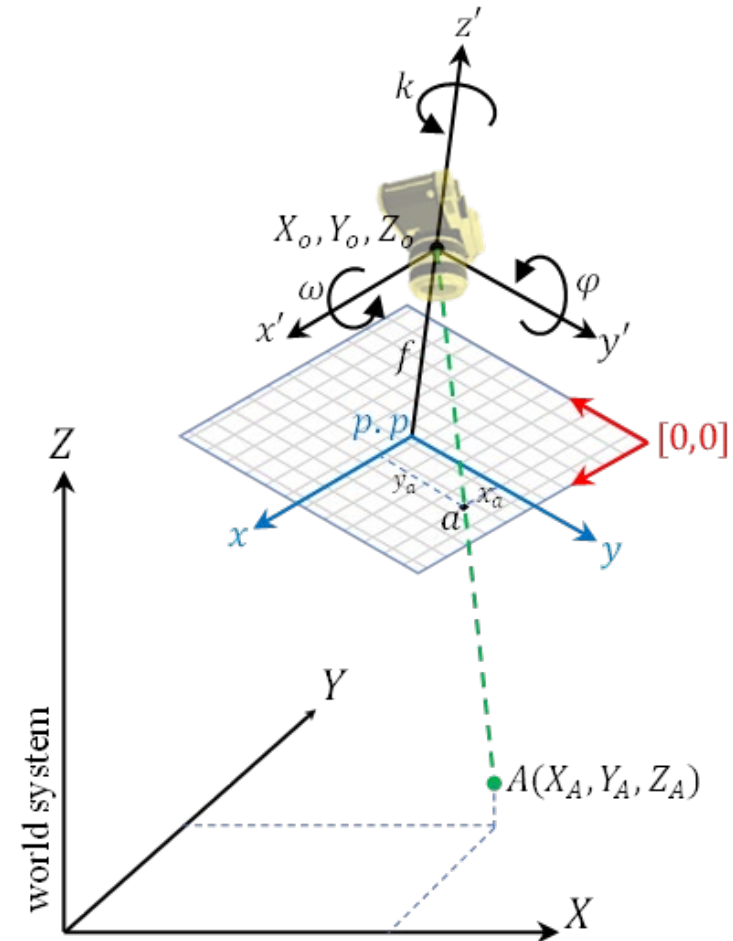
$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, \quad T = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix}$$

*rotation is a function of  $(\omega, \phi, k)$  translation*

- Spatial resection can be applied using the nonlinear collinearity equations.
- a minimum of three reference points should be measured in the single image.

$$x_a = x_o - f \frac{r_{11}(X_A - X_o) + r_{12}(Y_A - Y_o) + r_{13}(Z_A - Z_o)}{r_{31}(X_A - X_o) + r_{32}(Y_A - Y_o) + r_{33}(Z_A - Z_o)}$$

$$y_a = y_o - f \frac{r_{21}(X_A - X_o) + r_{22}(Y_A - Y_o) + r_{23}(Z_A - Z_o)}{r_{31}(X_A - X_o) + r_{32}(Y_A - Y_o) + r_{33}(Z_A - Z_o)}$$





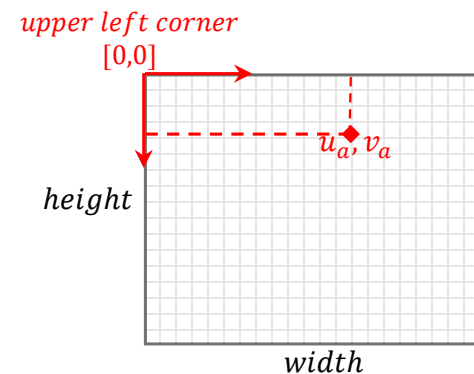
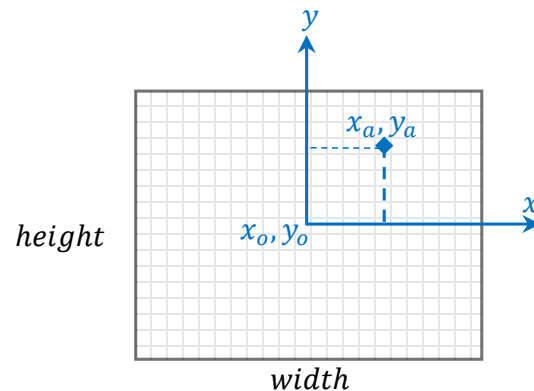
# PROJECTION MATRIX

- Projection matrix specifies how a pinhole camera applies a projective mapping of 3D points in the world to 2D points in an image.
- We can represent the ideal pinhole camera shown previously using homogeneous coordinates by:

$$\left. \begin{array}{l} x = f \frac{\bar{X}}{\bar{Z}} \\ y = f \frac{\bar{Y}}{\bar{Z}} \end{array} \right\} \text{in homogenous coordinates} \rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \bar{X} \\ \bar{Y} \\ \bar{Z} \\ 1 \end{bmatrix}$$

$x, y$  are the coordinates related to the center of the image at the principal point and we need to translate the image coordinates to the pixel coordinates origin at the top left corner.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & x_0 & 0 \\ 0 & f & y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \bar{X} \\ \bar{Y} \\ \bar{Z} \\ 1 \end{bmatrix}$$



# PROJECTION MATRIX

- $\bar{X}\bar{Y}\bar{Z}$  coordinates are the world's three-dimensional coordinates of points that have been rotated by  $R$  and translated by  $t$  into the camera frame

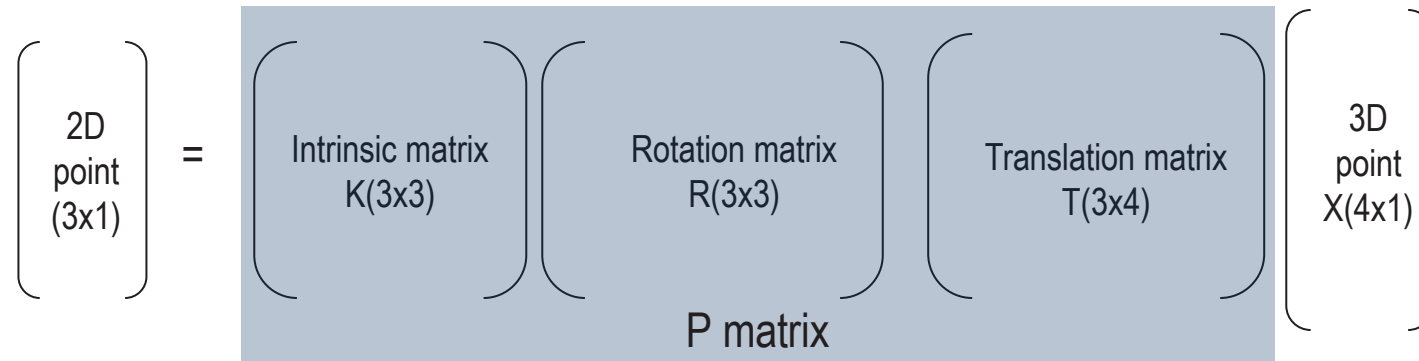
$$\underbrace{\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{3 \times 1} = \underbrace{\begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix}}_{3 \times 3} \underbrace{\begin{bmatrix} R_{3 \times 3} & -R_{3 \times 3} t_{3 \times 1} \end{bmatrix}}_{3 \times 4} \underbrace{\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}}_{4 \times 1}$$



$$\begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \rightarrow \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}$$

# PROJECTION MATRIX

Projection matrix specifies how a pinhole camera applies a projective mapping of 3D points in the world to 2D points in an image.



$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

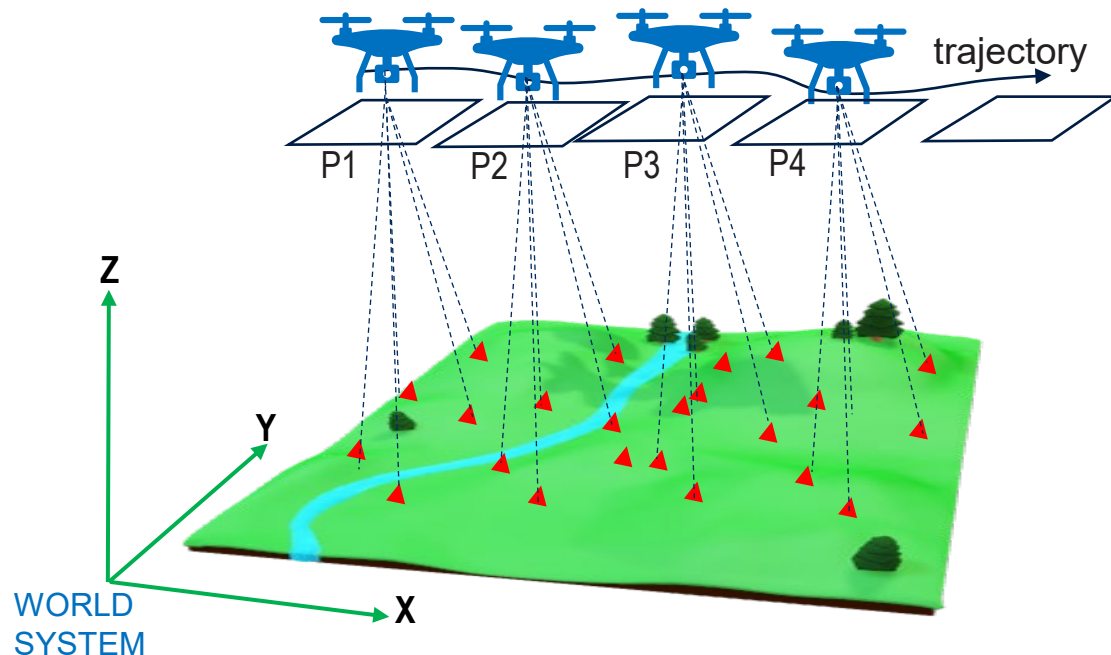
P is a 3x4 Matrix=12 elements  
11 degree of freedom:  
5 from intrinsic matrix K  
3 from rotation R  
3 from translation T

Camera pose can be solved when finding the projection matrix P.

# WHY COMPUTING CAMERA POSE?

- Each camera projection matrix along a trajectory defines the UAV motion.
- In SfM and visual odometry, we need to compute the camera motion.

Camera motion = multiple image/camera pose ( $P_i$ ) along its trajectory.



When camera motion (multiple poses) is defined, the structure of the scene is reconstructed from the images.

Then mapping of the scene can be applied.

More details will be given in the SfM lecture.

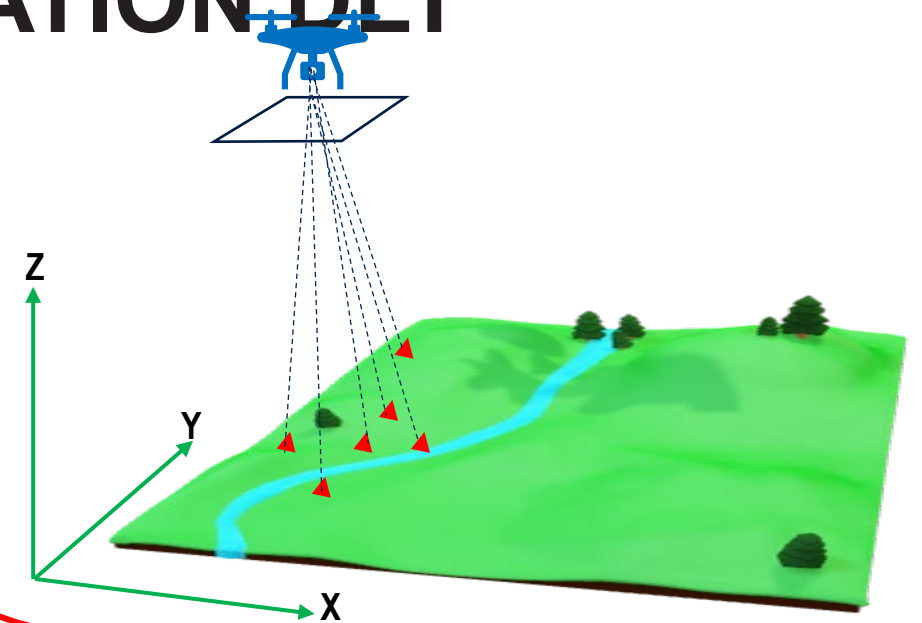


# DIRECT LINEAR TRANSFORMATION DLT

- There are different techniques to solve the projection matrix either linear or nonlinear methods.
- DLT equations for one observed point can be formulated as follows:

$$\begin{pmatrix} X & Y & Z & 1 & 0 & 0 & 0 & 0 & -xX & -xY & -xZ & -x \\ 0 & 0 & 0 & 0 & X & Y & Z & 1 & -yX & -yY & -yZ & -y \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_{12} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

- With six measured image points, twelve equations can be formed in the DLT which are the minimum number of points to determine the parameters of the projection matrix.
- The DLT solution of the form  $Ax = 0$  using singular value decomposition **SVD** to solve such an equation system by decomposing matrix  $A$ .



$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

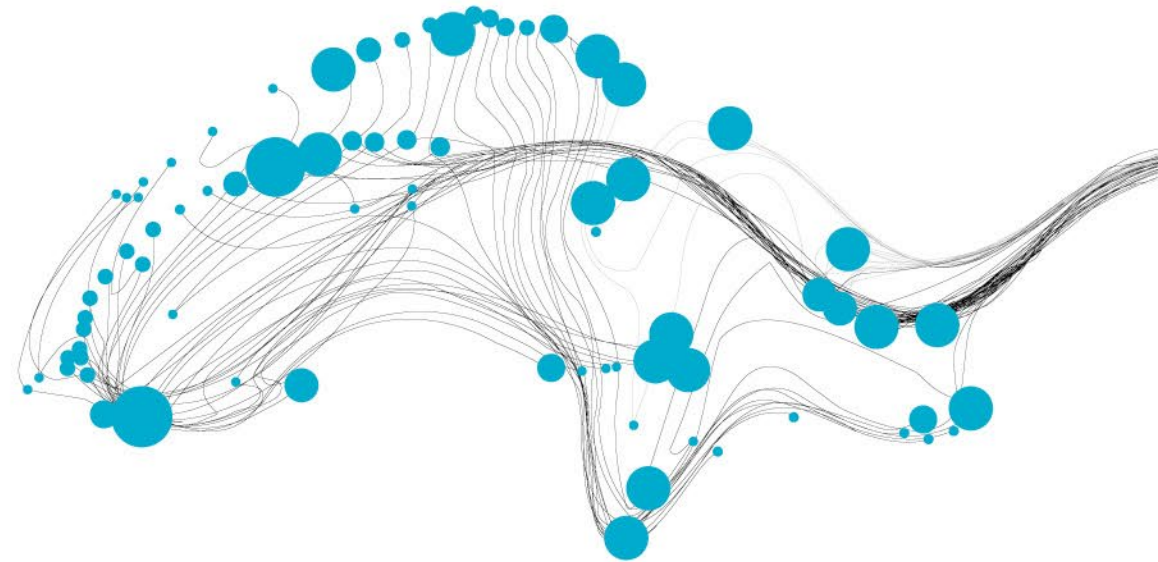
# WHAT HAVE WE LEARNED?

- Some useful camera settings and its relation to have good images.
- The geometry of a single image , 3D to 2D.
- The rotation in 3D space and some rotational variants.
- Deamination of camera pose using linear and nonlinear methods.

# STRUCTURE FROM MOTION (SFM)

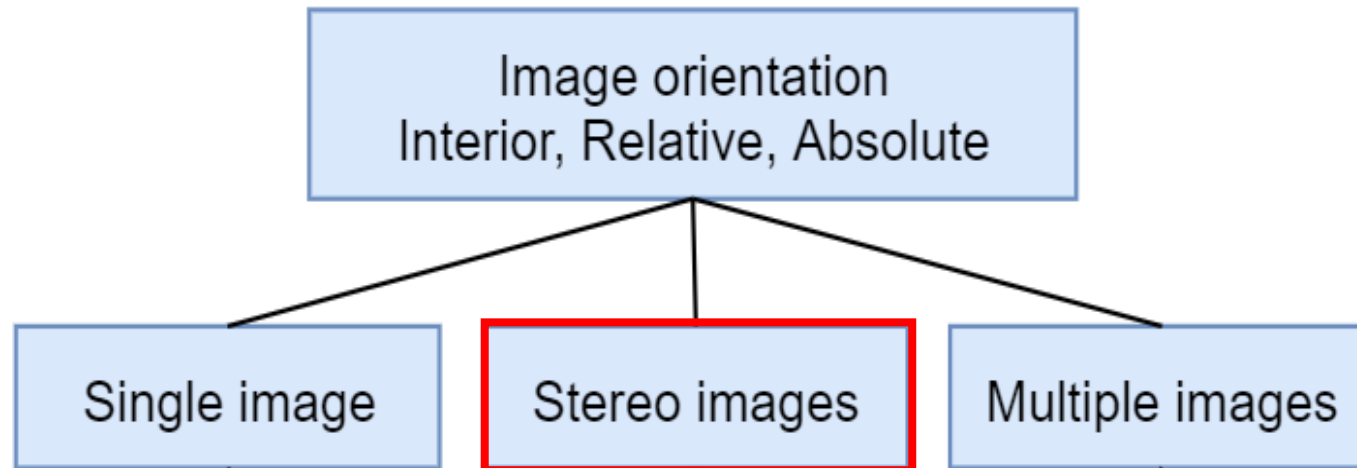
STEREO PAIR IMAGE ORIENTATION LECTURE

SLIDES BY B. ALSADIK



# WHAT WILL WE LEARN IN THIS LECTURE?

- Explain the necessity of relative orientation.
- Describe briefly how the relative orientation can be solved.
- Explain the necessity of image triangulation.
- Describe briefly the mathematical models for image triangulation.



# IMAGE ORIENTATION PROBLEM

It means:

- Reconstruct the geometric relation between the stereo images at the moment of capture.
- Reconstruct the geometric relation between the captured stereo images and the scene.

Then:

- We have a relative orientation of one local image with respect to another local image
- We have an absolute orientation between one local image system and the world coordinates system.

# RELATIVE ORIENTATION

Relative orientation is the process of camera motion estimation in right image relative to left image.

Q/ How to solve the RO problem?

A/ Estimate the fundamental and essential matrices.

Q/ What is the geometric constraint to solve the RO?

A/ The epipolar geometry constraint



UNIVERSITY  
OF TWENTE.

Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)



# RELATIVE ORIENTATION PROBLEM

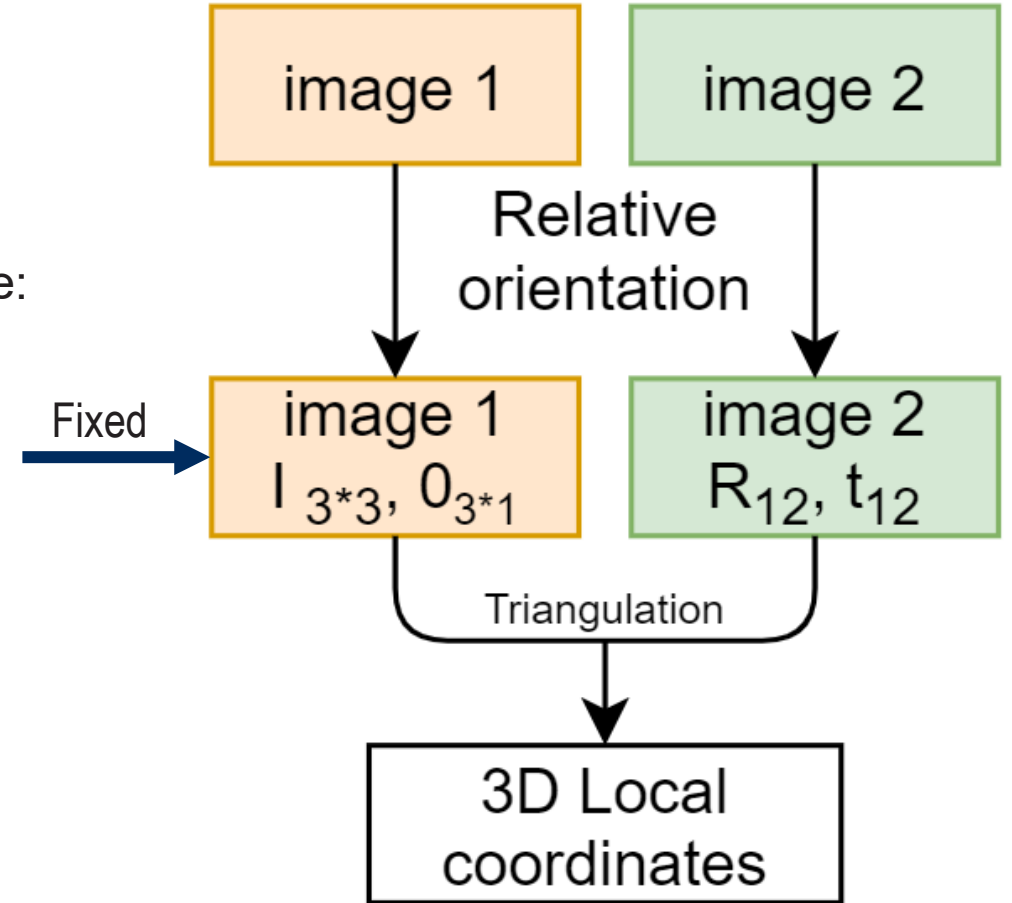
Relative orientation  $\rightarrow$  Camera Motion (R,T)

Structure XYZ  $\leftarrow$  Triangulation

So, for two stereo or multi stereo images we can estimate:  
**camera motion (R,T)** + **scene structure (XYZ)**

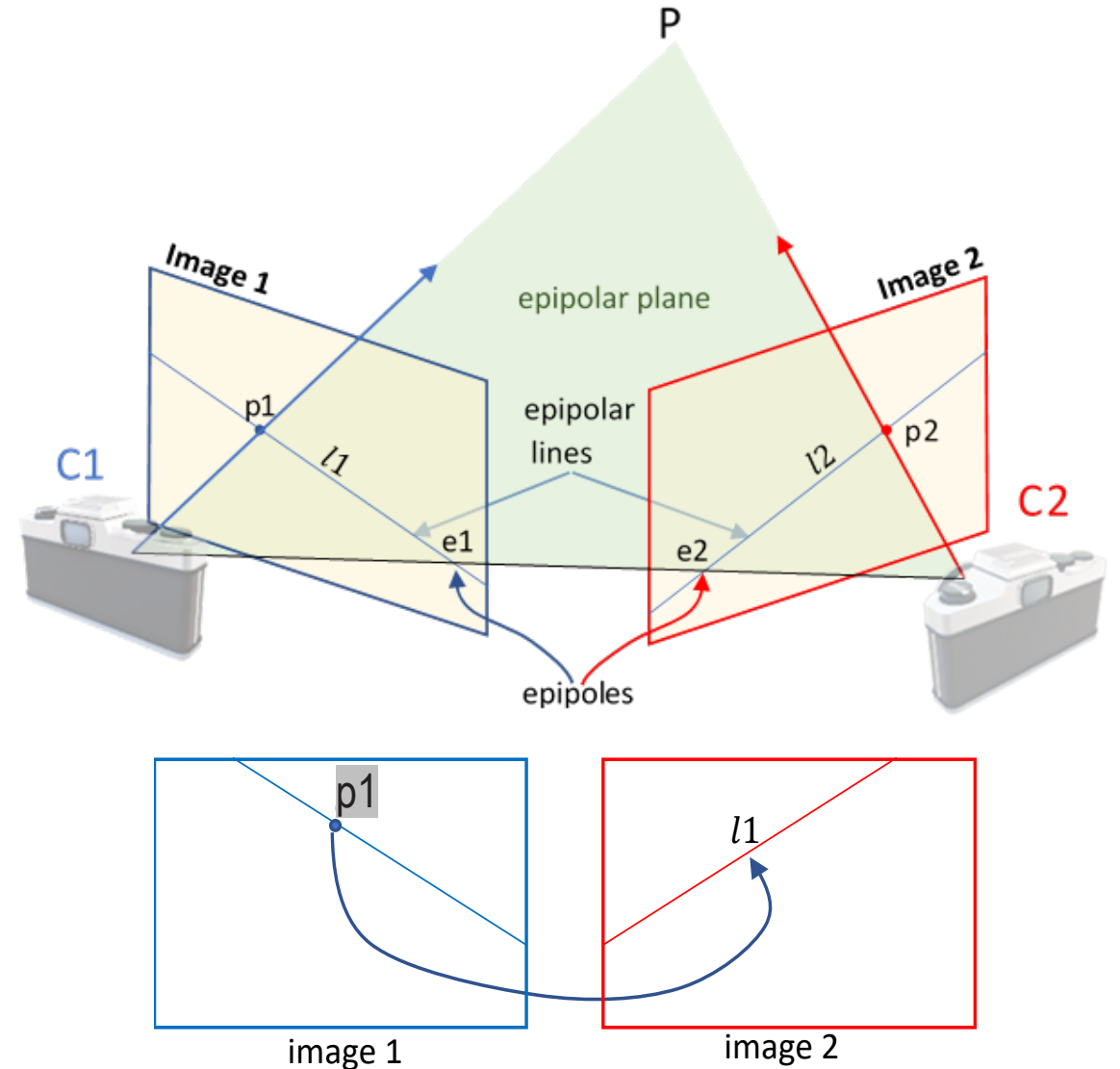
Q/ the scene XYZ after RO is local or absolute?

To understand the RO problem in more details, lets explain the epipolar geometry.



# EPIPOLAR GEOMETRY

- The center of each camera  $C_1, C_2$  and the world point  $P$  defines a plane in space – the **epipolar plane**
- Two special points:  $e_1$  and  $e_2$  (the **epipoles**): projection of one camera into the other.
- Epipolar lines  $l_1$  pass through  $e_1$  and  $l_2$  pass in  $e_2$ .
- Given a point in one image we know that its conjugate is constrained to lie along a line in the other image.
- By definition the conjugate point  $p_2$  must lie on that line  $l_2$  and vice versa.



How to relate?

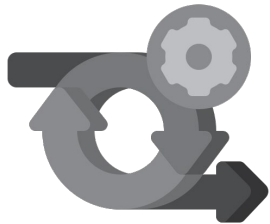


Relative orientation

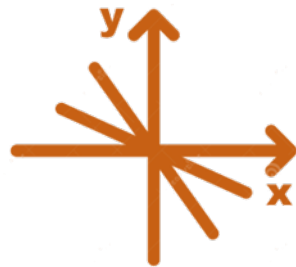
# METHODS FOR SOLVING RO

Relative orientation problem can be solved:

- Iteratively using coplanarity or collinearity equations.
- Directly using Fundamental and Essential matrices.

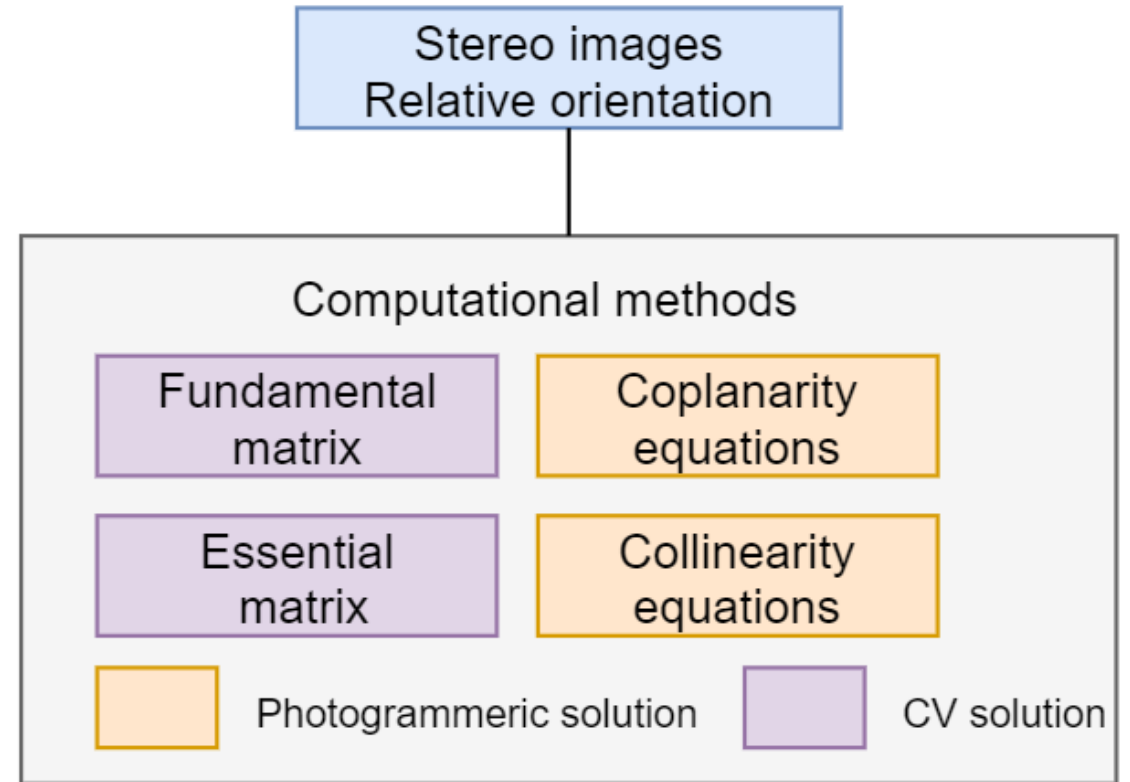


Nonlinear



Linear

- Requires good initialization
- Iterative
- Needs a stopping criterion
- No need for initialization
- Direct solution
- May result in more solutions



Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)

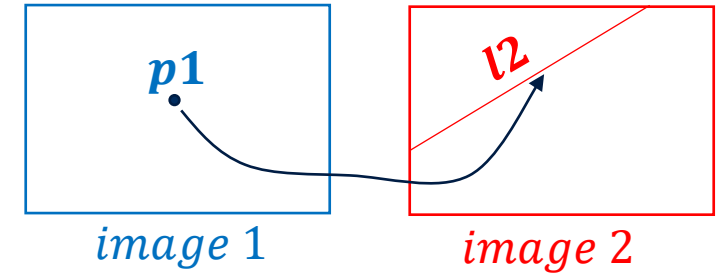
# THE FUNDAMENTAL MATRIX F

- This *epipolar geometry* of two stereo images is described by a very special 3x3 matrix called the *fundamental matrix F*
- In photogrammetry, it expresses the relative orientation between the two stereo frames.
- The fundamental matrix is a function of the camera parameters and the relative camera pose between the views.
- $F$  is a powerful tool in 3D reconstruction and multi-view object/scene matching.
- It is a singular matrix with a rank of two.
- $F e_1 = 0$  ,  $F^t e_2 = 0$

# PROPERTIES OF THE FUNDAMENTAL MATRIX

- $F$  is used to map homologues points in image 1 to image 2

$$p1 = \begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \text{in homogenous coord.} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$
$$p2 = \begin{bmatrix} x' \\ y' \end{bmatrix} \rightarrow \text{in homogenous coord.} = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix}$$



- The epipolar line  $l2$  of point  $p1$  in image 2 is:  $Fp1$
- The epipolar line  $l1$  of point  $p2$  in image 1 is:  $F^t p2$
- Epipolar constraint on corresponding points  $p2^T F p1 = 0$

$F$  explains the epipolar constraint between two conjugate points as  $p2^T F p1 = 0$  or

$$p2^T F p1 = [x' \quad y' \quad 1] \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0$$

# CALCULATIONS OF THE F MATRIX

The  $F$  matrix can be computed in two ways:

- 1- If the intrinsic  $K$  and extrinsic parameters  $R, T$  are known.
- 2- If we know enough corresponding (matching) points between the two images.

## The first option

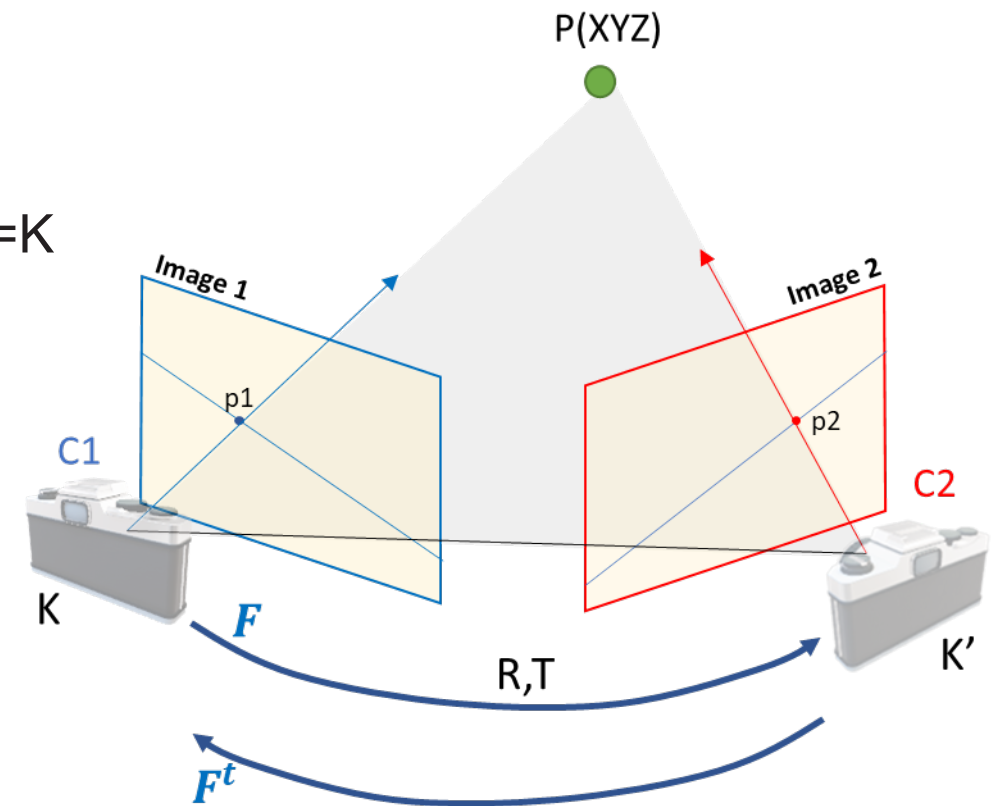
Given:  $K, R, T$  assuming the same camera is used  $K'=K$

$$F = K'^{-T} [t]_x R K^{-1}$$

Where

$$T = T1 - T2 = \begin{bmatrix} t1 \\ t2 \\ t3 \end{bmatrix}$$

$$[t]_x = \begin{bmatrix} 0 & -t3 & t2 \\ t3 & 0 & -t1 \\ -t2 & t1 & 0 \end{bmatrix}$$



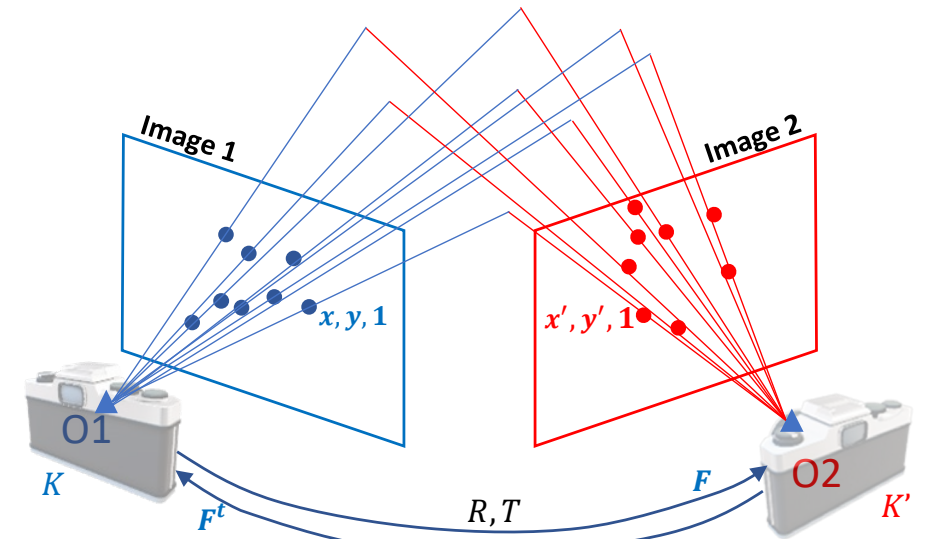
Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)



# CALCULATIONS OF THE F MATRIX

- If we don't know  $K$ ,  $R$ , and  $T$ , can we estimate  $F$  for two images?
- Yes, given enough correspondences (SIFT, SURF,..) and then apply the 8-points algorithm.
- **8-points algorithm** – linear solution. Enforce the matrix to have rank=2. It can have problems especially with noisy points.

$$AF = \begin{bmatrix} x'_1x_1 & x'_1y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_nx_n & x'_ny_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{bmatrix} = 0$$



Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)

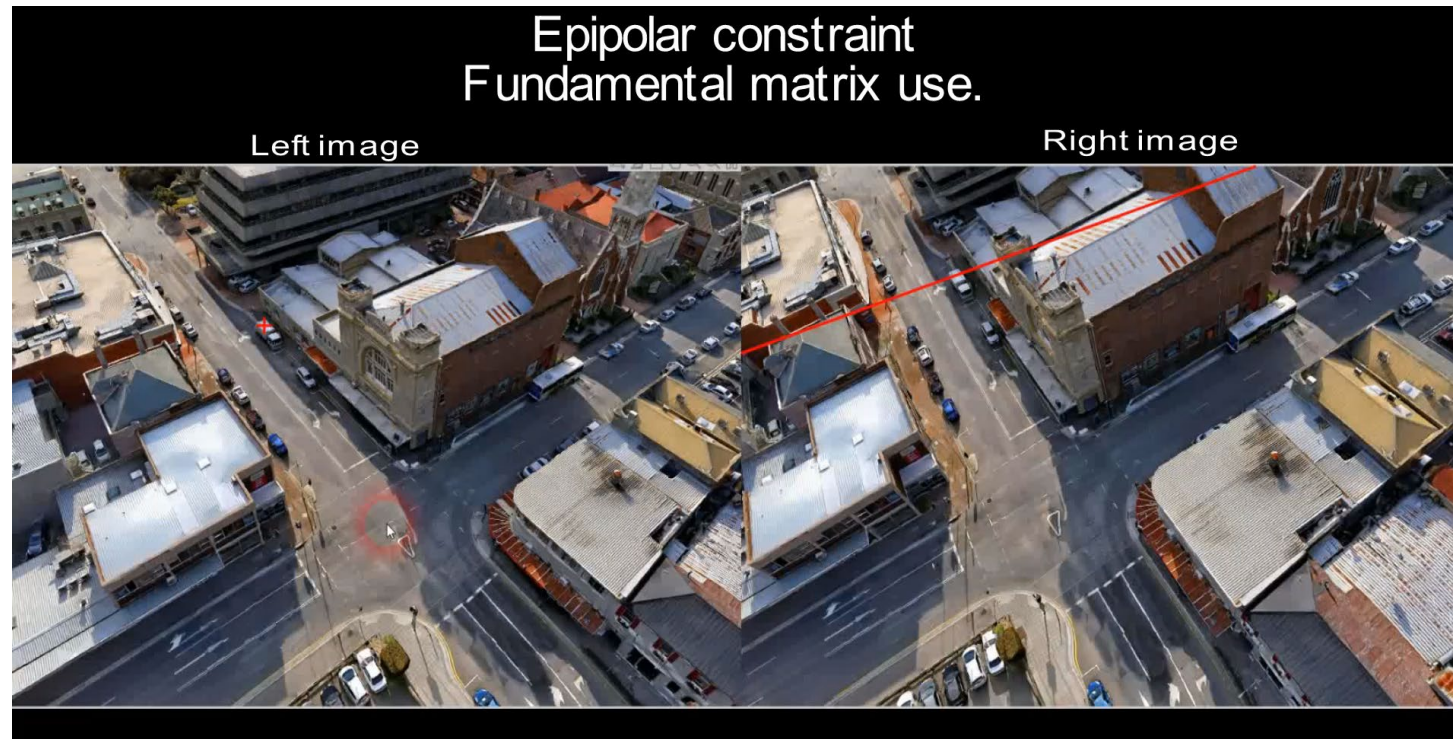
We need RANSAC to remove matching outliers

$n$  is the number of corresponding points

System is solved using least squares by the singular value decomposition  $SVD$  and selecting the last eigenvector of  $V$ .

# EPIPOLAR GEOMETRY

If RO is solved, then we can map corresponding conjugate points in left image to right image where the epipolar line of point  $p_1$  is  $Fp_1$  and that epipolar line of point  $p_2$  is  $F^t p_2$



Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)

# RANSAC: RANDOM SAMPLE CONSENSUS

- An algorithm for robust fitting of models in the presence of many data outliers.
- **Idea:** instead of starting with the full data and trying to work down to a smaller subset of good data, start small and work to larger.

## Example of LINE FITTING

- Given a hypothesized line
  - Count the number of points that “agree” with the line
  - “Agree” = within a small distance of the line  
i.e., the **inliers** to that line
- 
- For all possible lines, select the one with the largest number of inliers

# RANSAC

- Unlike least-squares, no simple closed-form solution
- Hypothesize-and-test
- Try out many lines, keep the best one

Procedure in summary:

1. Randomly choose  $s$  samples

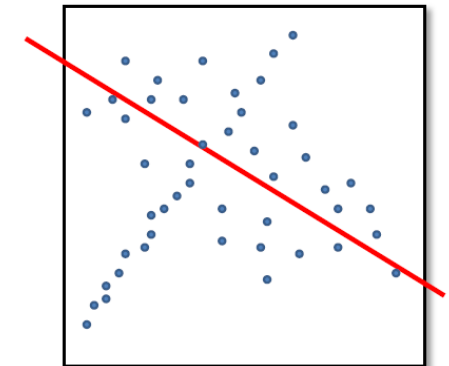
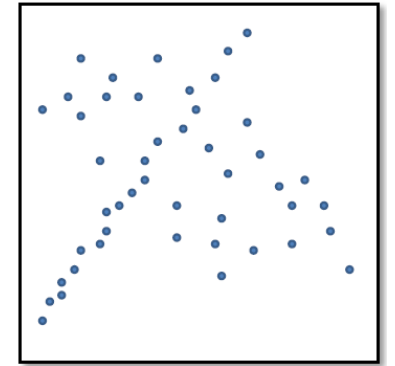
Typically,  $s = \text{minimum sample size that lets you fit a model}$

2. Fit a model (e.g., line) to those samples

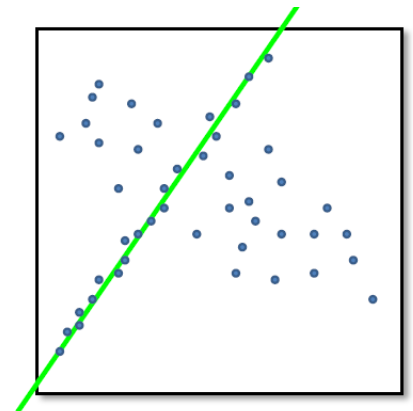
3. Count the number of inliers that approximately fit the model

4. Repeat  $N$  times

5. Choose the model that has the largest set of inliers



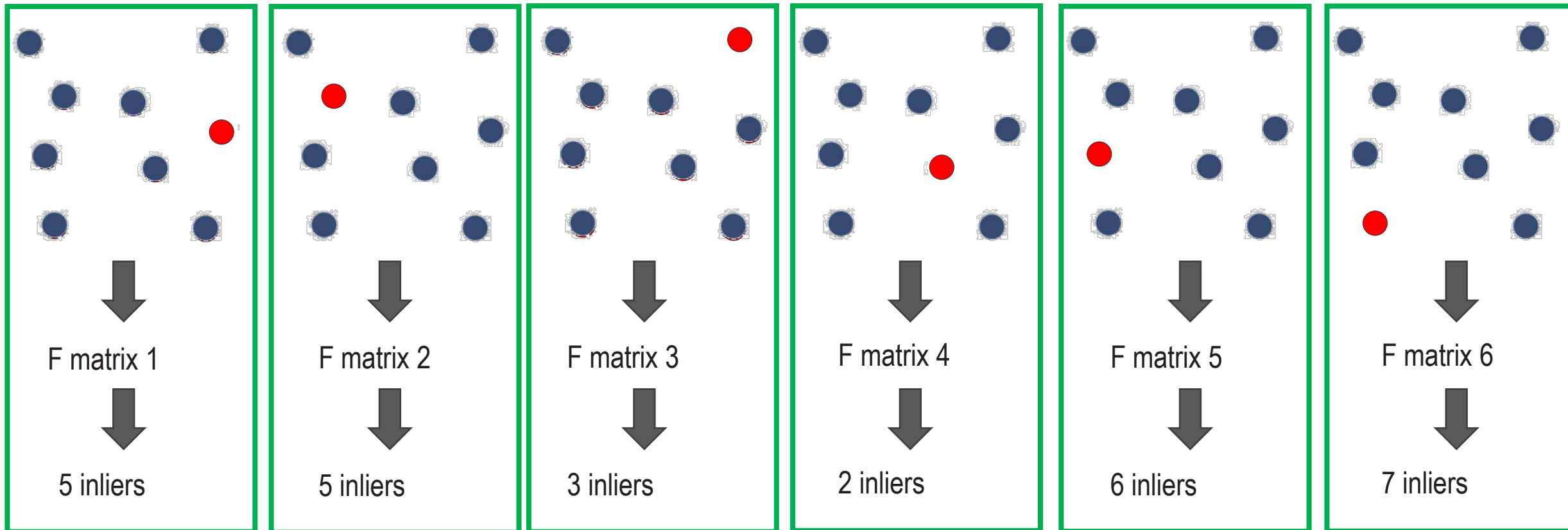
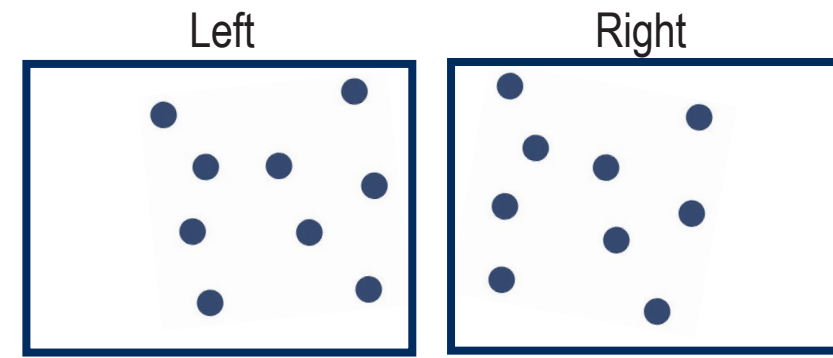
Inliers: 3



Inliers: 20

# RANSAC + F MATRIX

● excluded matching point

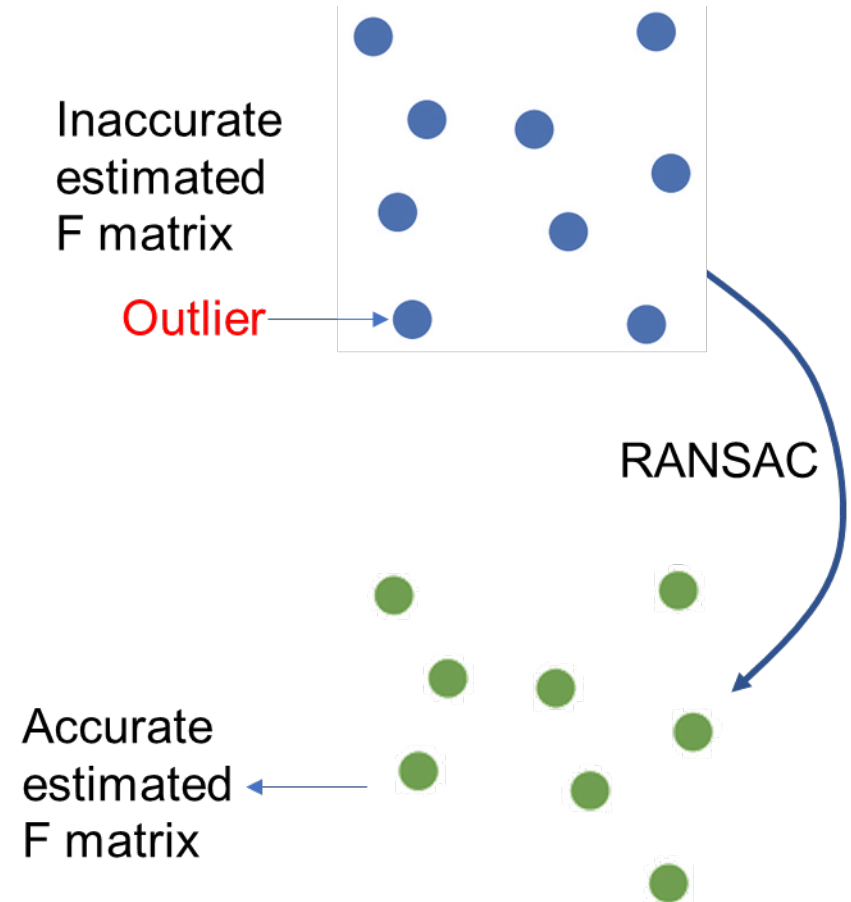


Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)

Vote for the best f matrix with min. outliers

# RANSAC + F MATRIX

- Step 1. Extract features (they are  $> 8$  points)
- Step 2. Compute a set of potential matches
- Step 3. do
  - Step 3.1 select minimal sample (i.e., 8 matches)
  - Step 3.2 compute solution(s) for F
  - Step 3.3 determine inliers
    - stop when satisfy a threshold
- Step 4. Compute F based on all inliers





# ESSENTIAL MATRIX E

Using the  $F$  matrix without knowing the intrinsic parameters  $K$  leads to uncalibrated SfM solution (not preferred because of ambiguity).

If the intrinsic parameter are known ( $K$  matrix), Then we can derive the **Essential matrix**  $E$

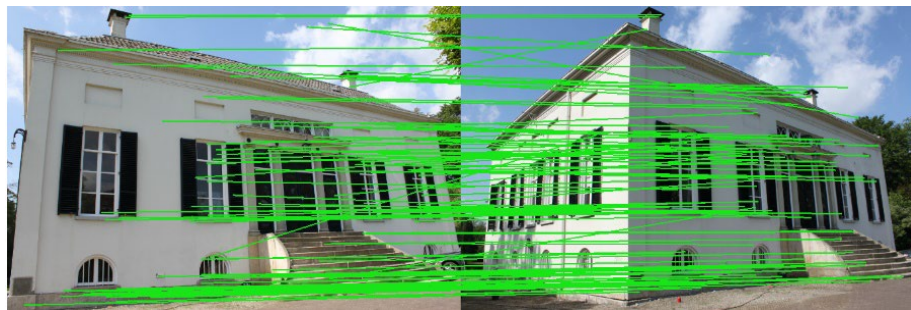
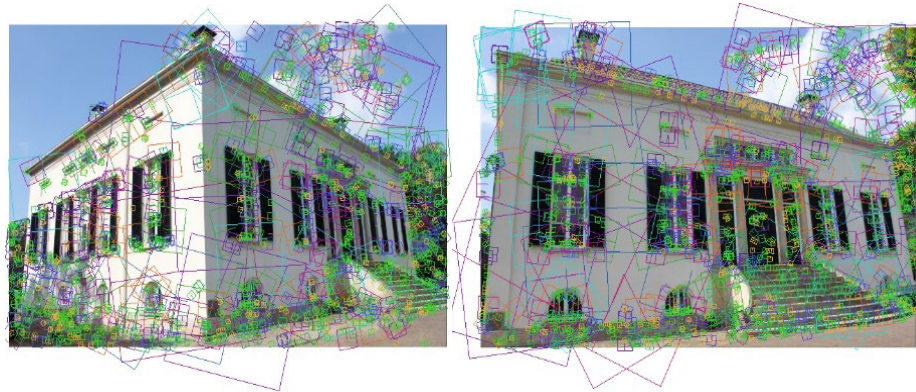
$$E = K^t F K = [t]_x R$$

- essential matrix is singular, has a rank of two.
- essential matrix has only 5 D.O.F and is completely defined by 3 rotational and 2 translational parameters (**relative orientation**)

After relative orientation,  $T$  and  $R$  are translation and rotation for the 2<sup>nd</sup> camera, then  $P1 = [I|0]$  and  $P2 = [R|T]$  ( $P$ : projection matrix)

- Four solutions are found and in the decomposition of  $E$  and the selection of the most proper matrix should be applied

# SUMMARY: TWO STEREO IMAGES SFM



Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)



Intrinsic  $K_1$

Intrinsic  $K_2$

Keypoints

Keypoints

Matching

$x_1, x_2$ ,  
compute

Fundamental matrix

compute

Essential matrix

Rotation  $R$   
Translation  $T$

(motion) structure

$P_1, P_2$

Triangulation

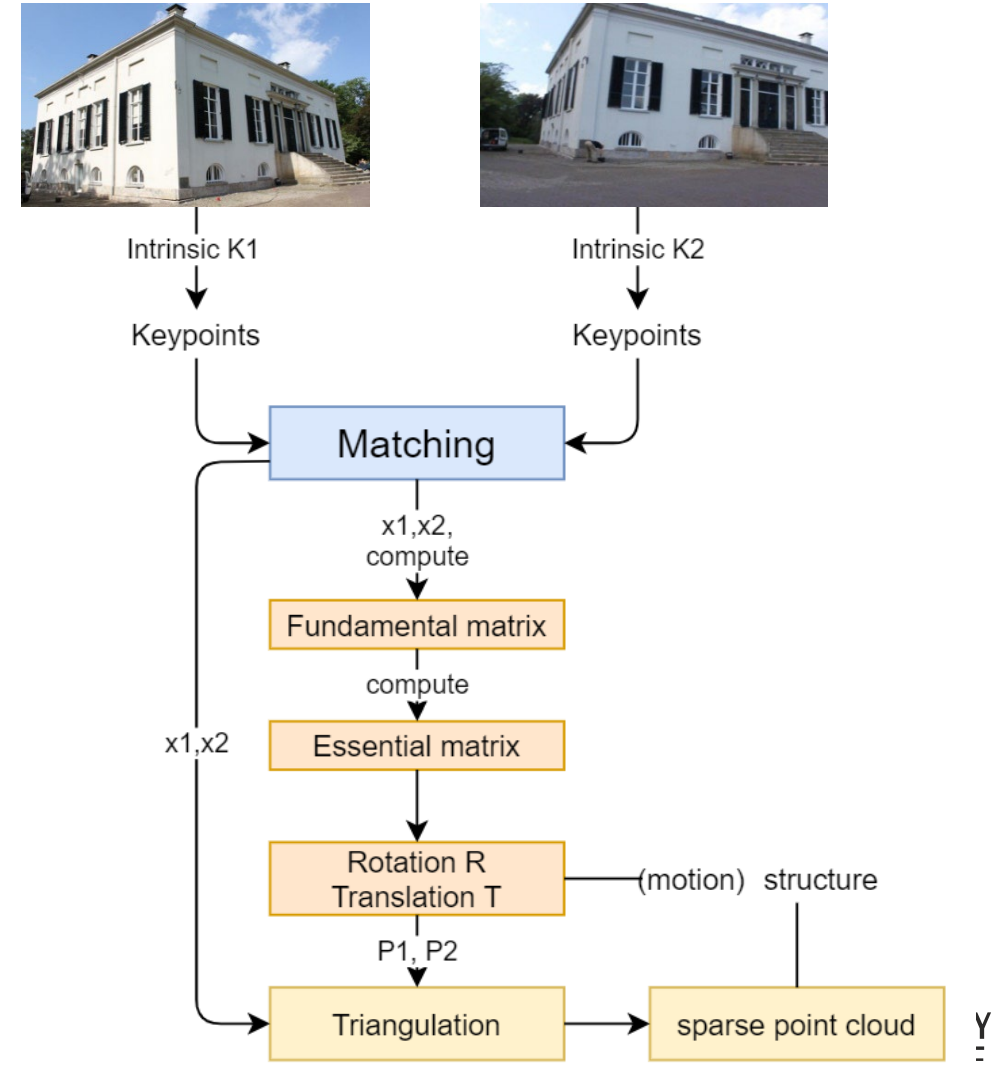
sparse point cloud

$x_1, x_2$

# WHAT HAVE WE LEARNT SO FAR?

- SfM for a stereo pair of images  
Q/ How to compute the sparse point cloud?  
A/ by image intersection or Triangulation

The sparse point cloud, is it georeferenced or still local?



# WHY TRIANGULATION?

Image triangulation » Spatial points XYZ

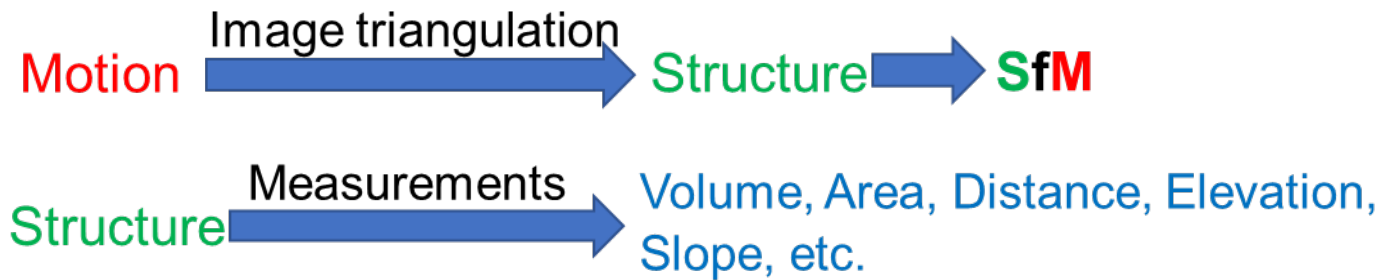
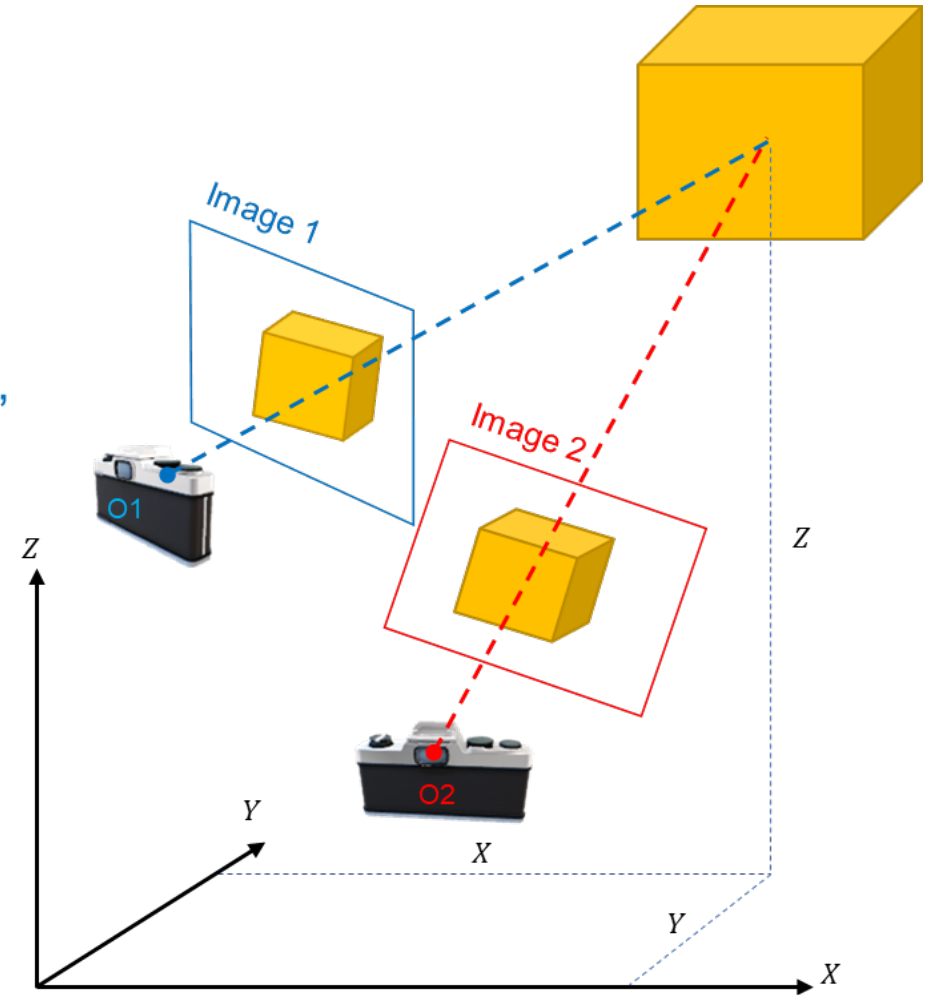


Image triangulation is a method used to determine the 3D position (XYZ) of a spatial point viewed by two or more images (Multiview). Used in 3D reconstruction and SfM. Also called 'spatial image Intersection'



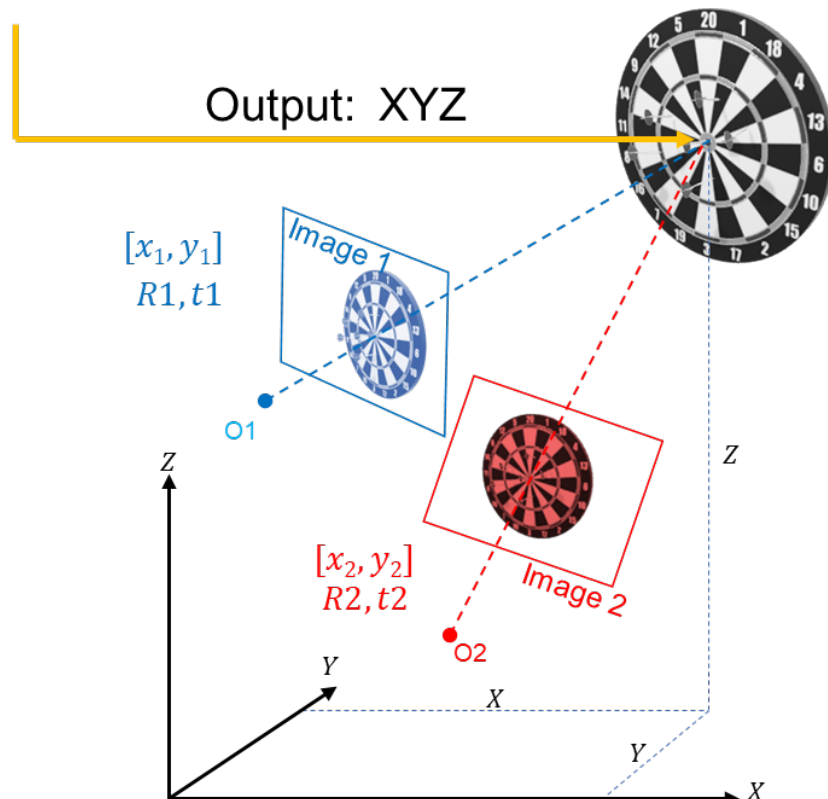
Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)

UNIVERSITY  
OF TWENTE.

# TRIANGULATION PROBLEM

Input

- Known camera parameters  $K$
- Measurements: 2D coordinates of the point in every viewing image  $[x_1, y_1]$   $[x_2, y_2]$
- Known (Pre-computed) orientation of stereo images  $R_1, t_1$  and  $R_2, t_2$



Q/ How to get the  $R_1, t_1$  and  $R_2, t_2$ ?  
A/ By applying relative orientation using F matrix and E matrix.

# TRIANGULATION PROBLEM

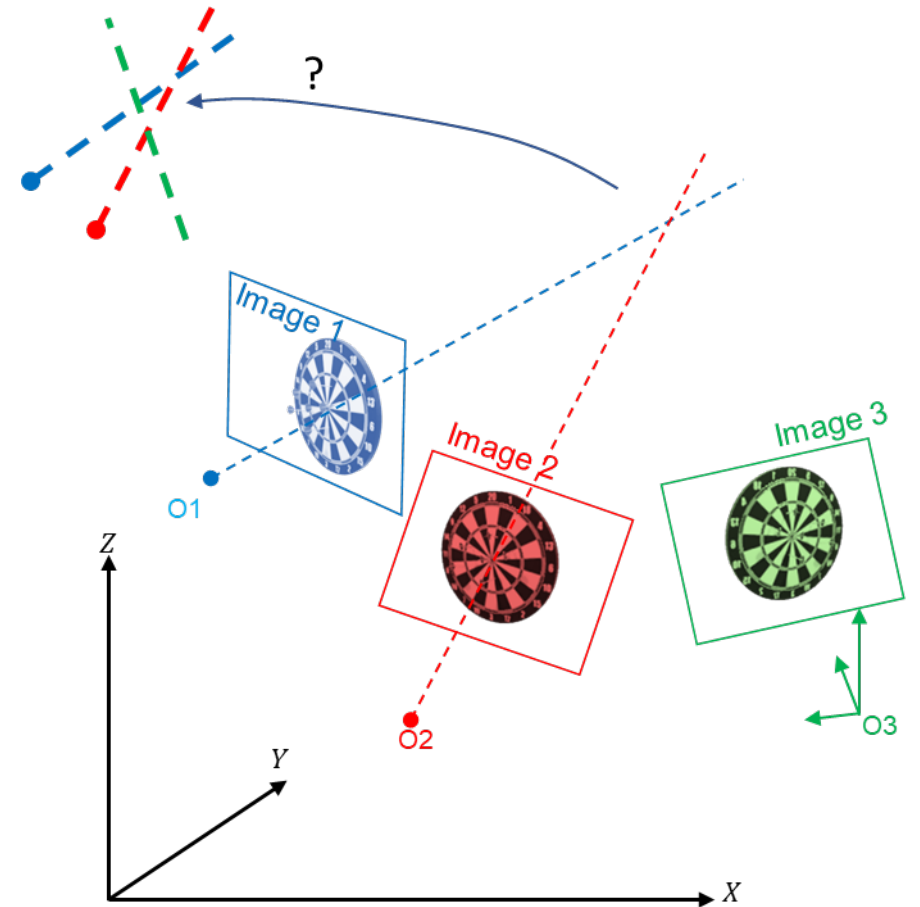
Multiple intersected lines in 3D space do not meet at one location in practice due to measurements noise.

Then determining the best 3D point of intersection becomes hard.

Several methods are found for the triangulation problem solution. One solution can be applied using

Linear least-squares method (DLT-based) using the projection matrix of each image  $P_1$  and  $P_2$ .

$$\begin{bmatrix} x_1 P_1^{3T} - P_1^{1T} \\ y_1 P_1^{3T} - P_1^{2T} \\ x_2 P_2^{3T} - P_2^{1T} \\ y_2 P_2^{3T} - P_2^{2T} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = 0$$



Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)



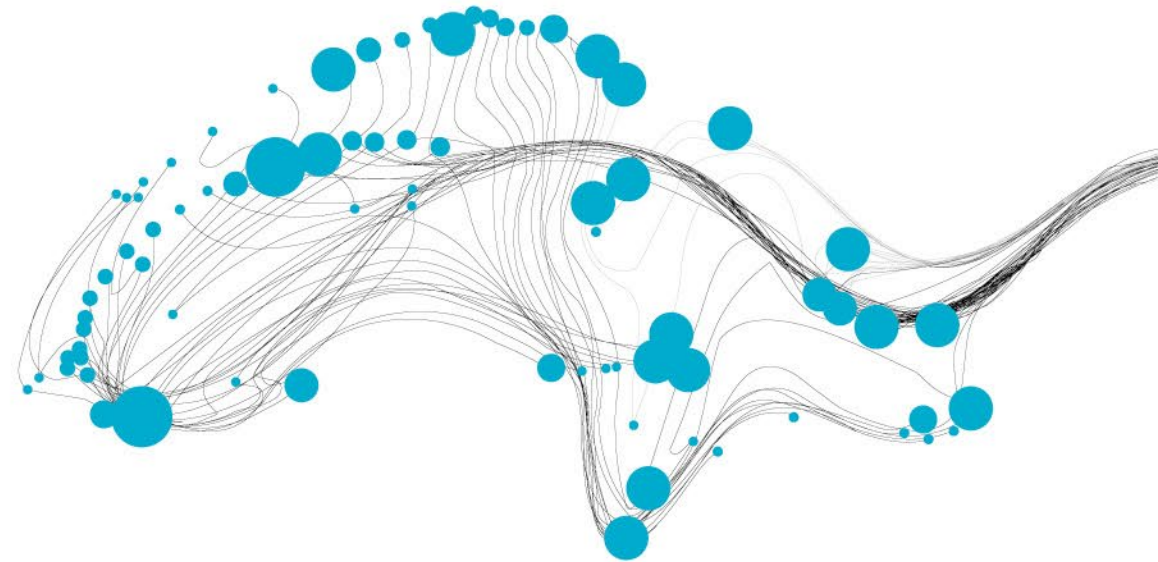
# WHAT HAVE WE LEARNED?

- Explain the stereo image orientation problem.
- Define the epipolar geometry.
- Define the RO method.
- Explain the image triangulation problem.

# STRUCTURE FROM MOTION (SFM)

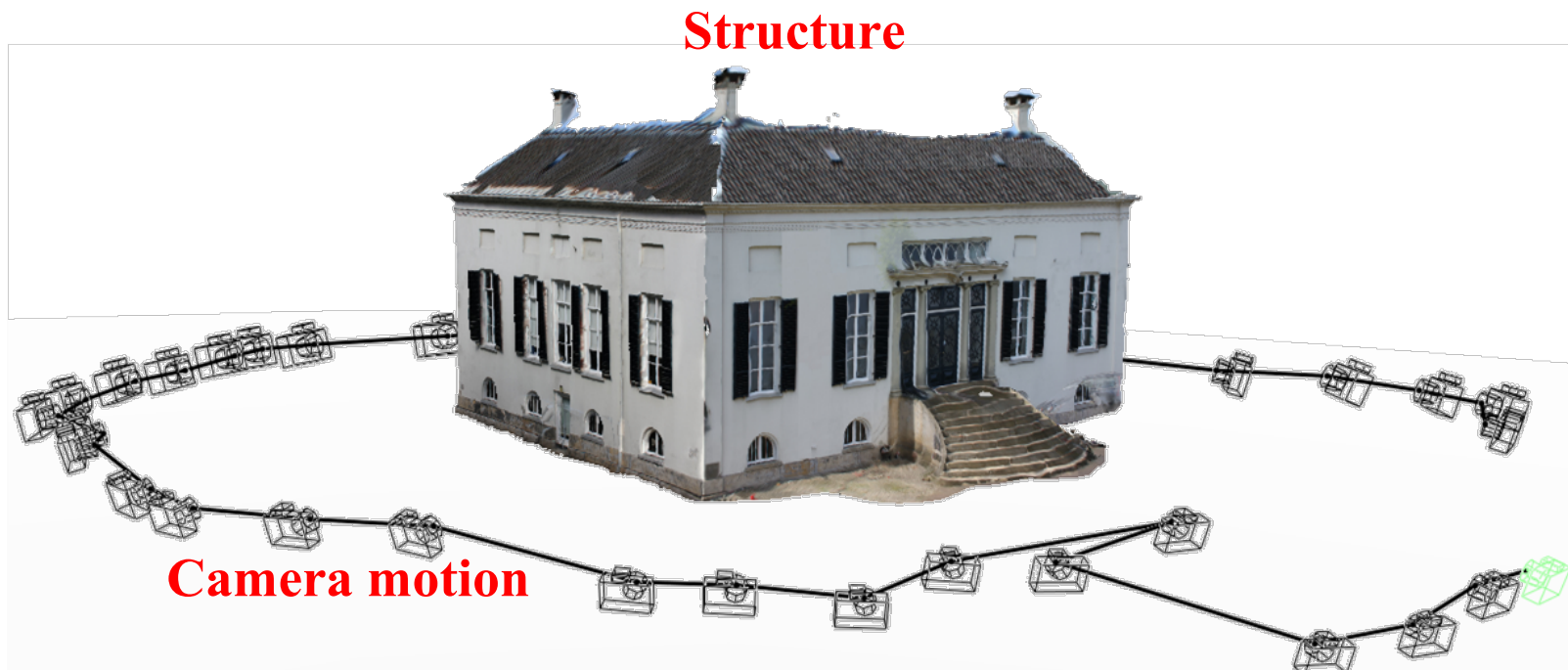
MULTI VIEW IMAGE ORIENTATION LECTURE

SLIDES BY B. ALSADIK



# WHAT IS THE STRUCTURE FROM MOTION SFM?

- SfM is the process of estimating the 3D structure of a rigid scene from a set of overlapped 2D images (min. 2)
- Motion is represented by Rotation  $\mathbf{R}$  and Translation  $\mathbf{T}$  while the structure is represented by a point cloud  $\mathbf{XYZ}$ .

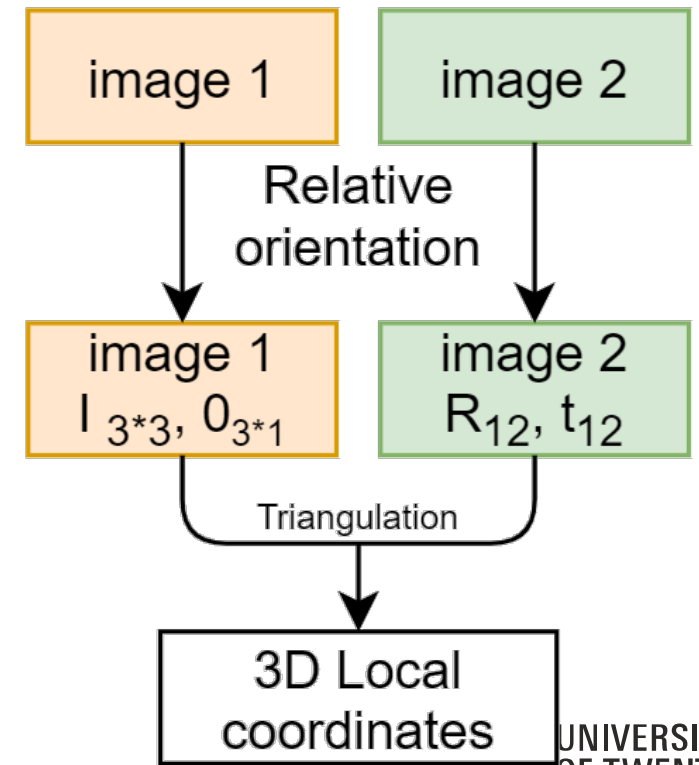
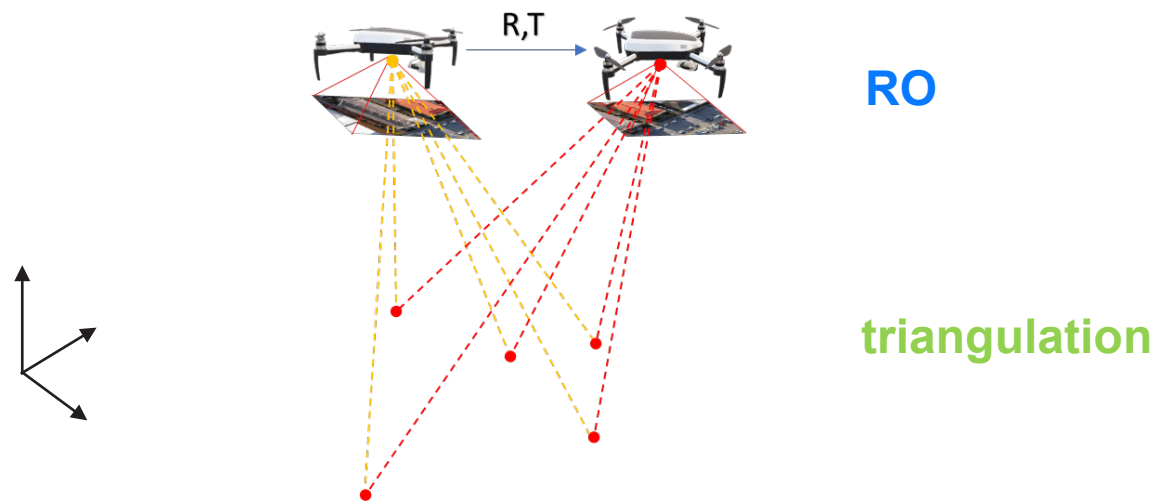


# HOW TO APPLY SFM FOR A STEREO PAIR?

Find the corresponding image points by feature matching.

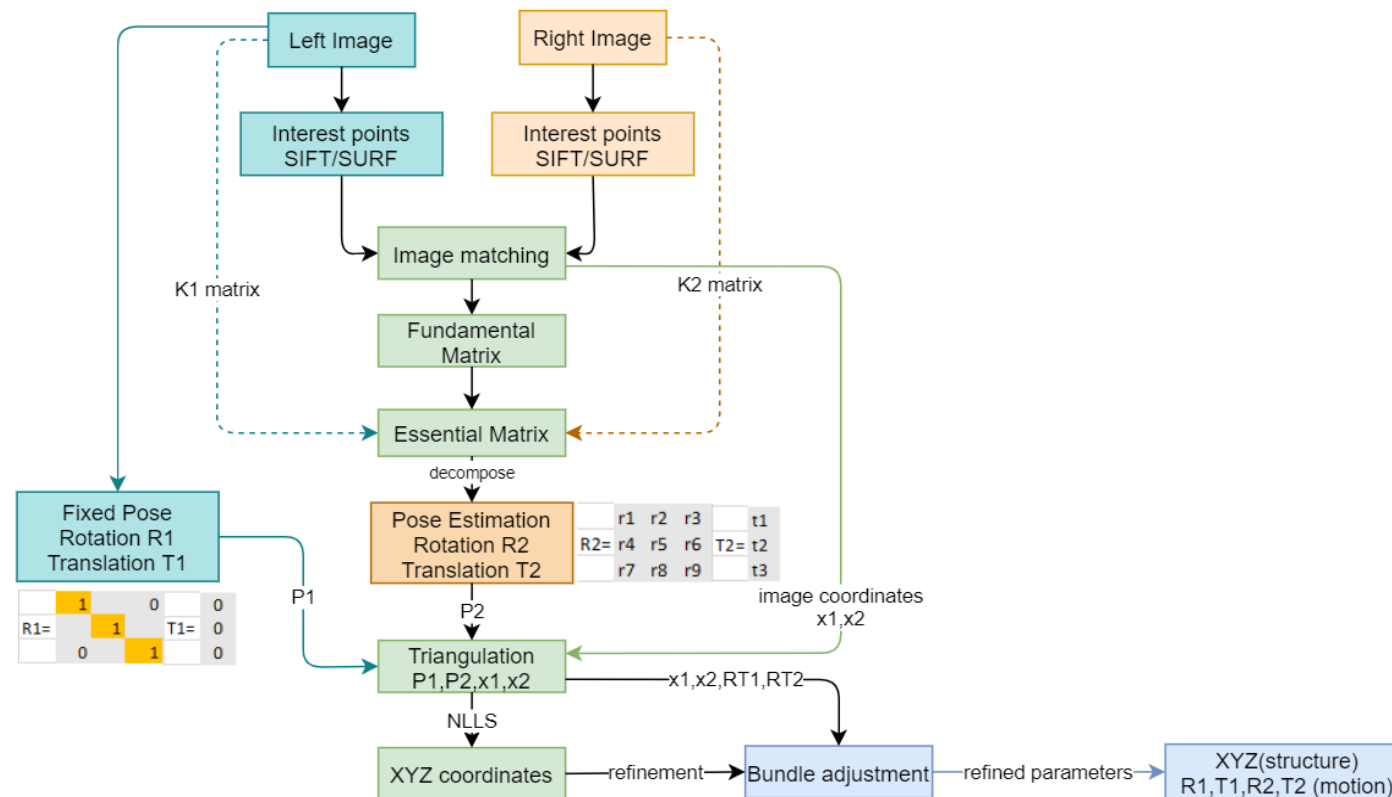


Apply the relative orientation & triangulation.



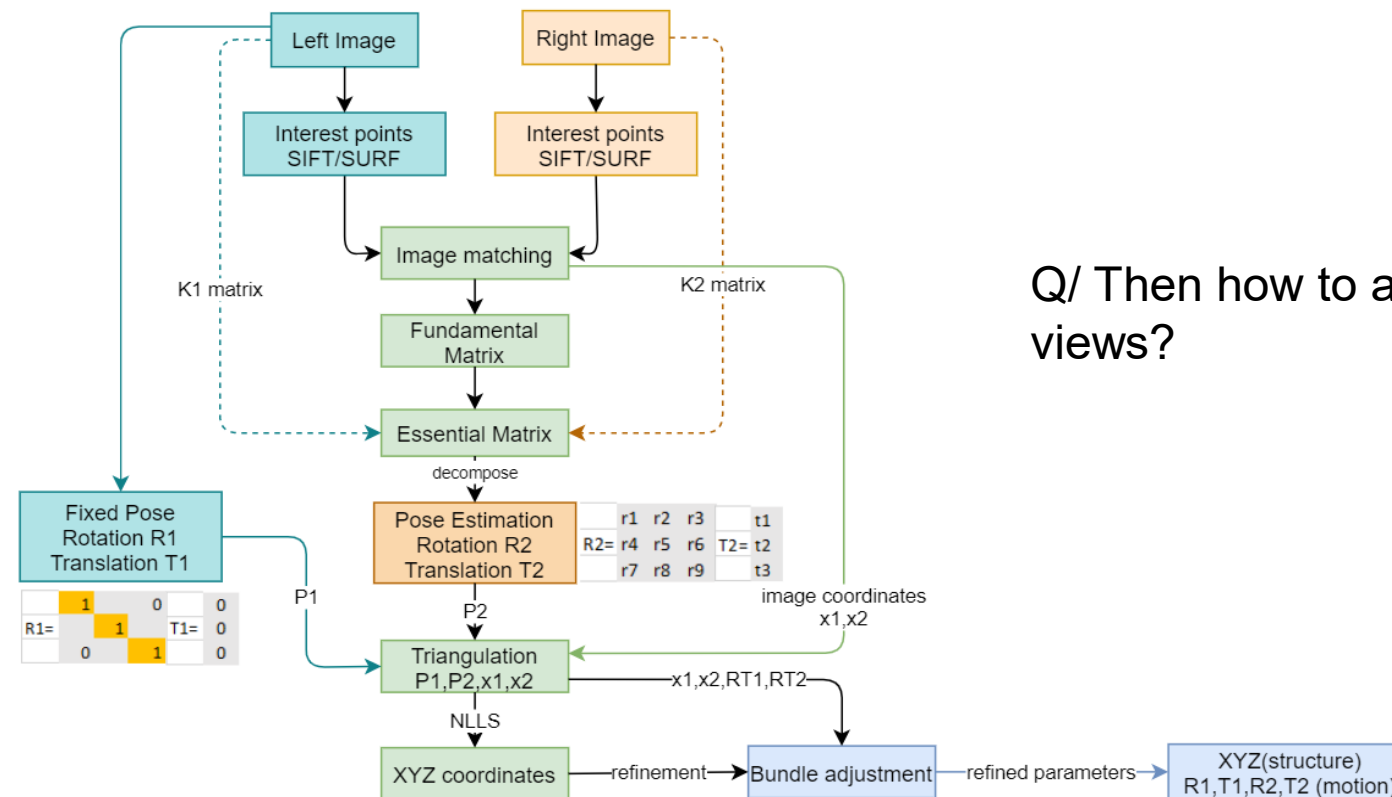
# SFM FOR A STEREO PAIR

1. Find the corresponding image points by feature matching technique.
2. Apply the relative orientation as described in the previous lecture.
3. Find the fundamental matrix and the essential matrix and solve for the rotation and translation parameters.



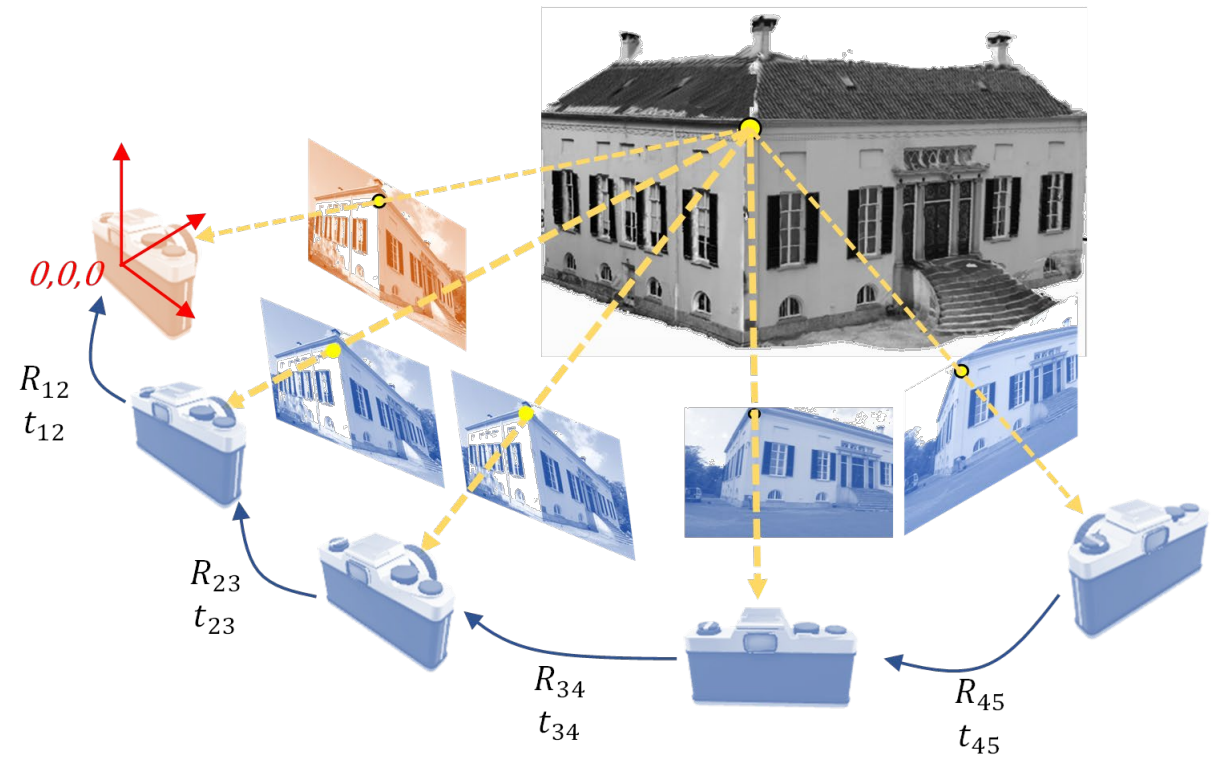
# SFM FOR A STEREO PAIR

1. Find the corresponding image points by feature matching technique.
2. Apply the relative orientation as described in the previous lecture.
3. Find the fundamental matrix and the essential matrix and solve for the rotation and translation parameters.
4. Compute the local 3D coordinates of the matched point features (tie points) using the triangulation technique. Then we will have a sparse point cloud referenced locally to the first image.
5. Refine the computed rotation, translation, and the 3D local coordinates of the tie points by the *bundle adjustment* method.



# MULTIVIEW SfM

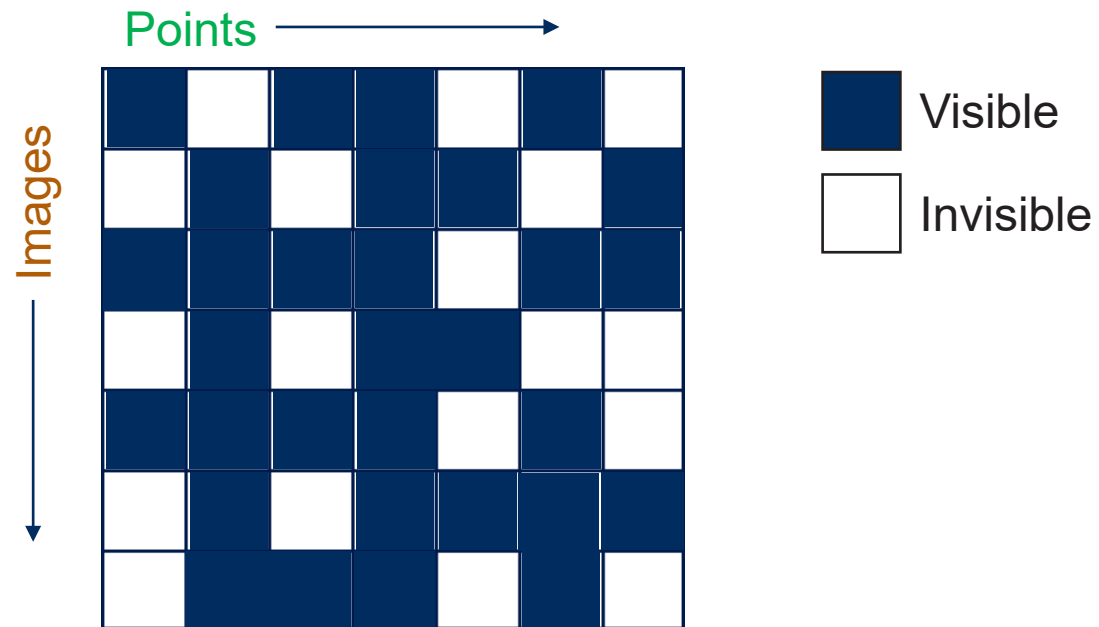
- The SfM approach described for a stereo pair can be expanded to include multiple images which is also called Multiview SfM.
- Those multiple images can be sequential (ordered) which reduces the image matching from **full pairwise** to **guided search**.
- The Multiview SfM starts from two images by finding the pose of camera 2 relative to camera 1. Then find the pose of camera 3 relative to camera 2 and so on to extend this approach to the multiple view scenario which is also known as **Incremental SfM**.
- All camera poses are usually computed relative to camera 1 so that they are all in the same local coordinate system.





# TRACKS

- SfM from Multiview images necessitates point correspondences across numerous images, which are referred to as **tracks**
- Feature tracks for a point mean to build a tensor track that specifies the point matches within images
- Tracks can be sorted in a **visibility matrix** (**m images** × **n features**)

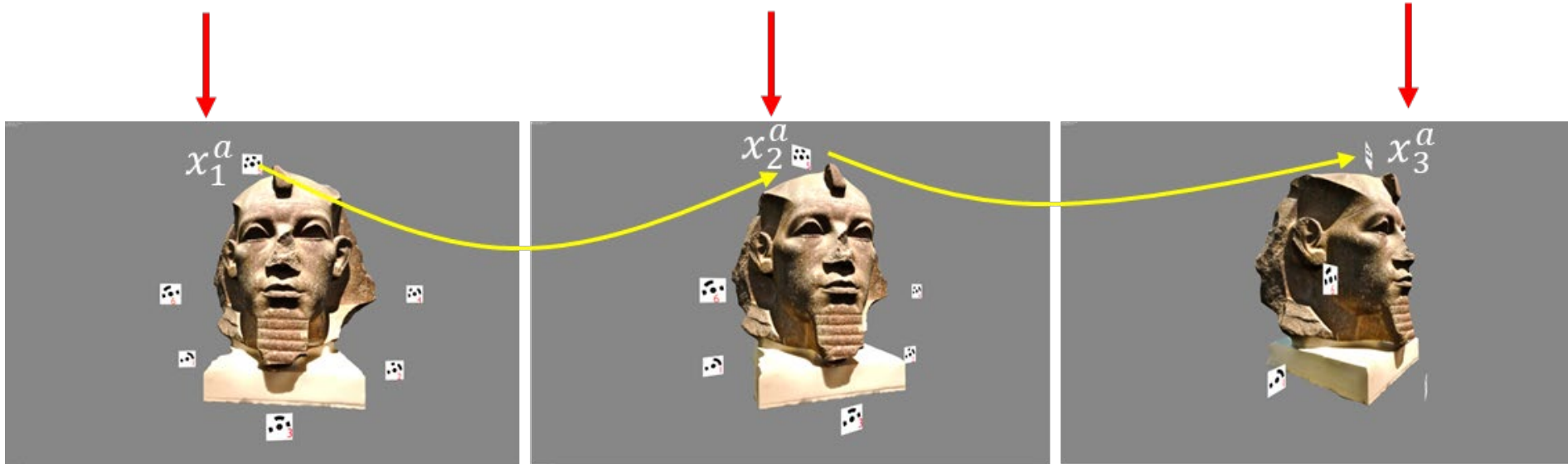


# TRACKS

Feature tracks for a point mean to build a tensor track that specifies the point matches within images

point  $a$  is visible in the three images 1, 2, and 3  $x_1^a \leftrightarrow x_2^a \leftrightarrow x_3^a$  so track  $(1, a, :) = x_1^a$ , track  $(2, a, :) = x_2^a$ , and track  $(3, a, :) = x_3^a$

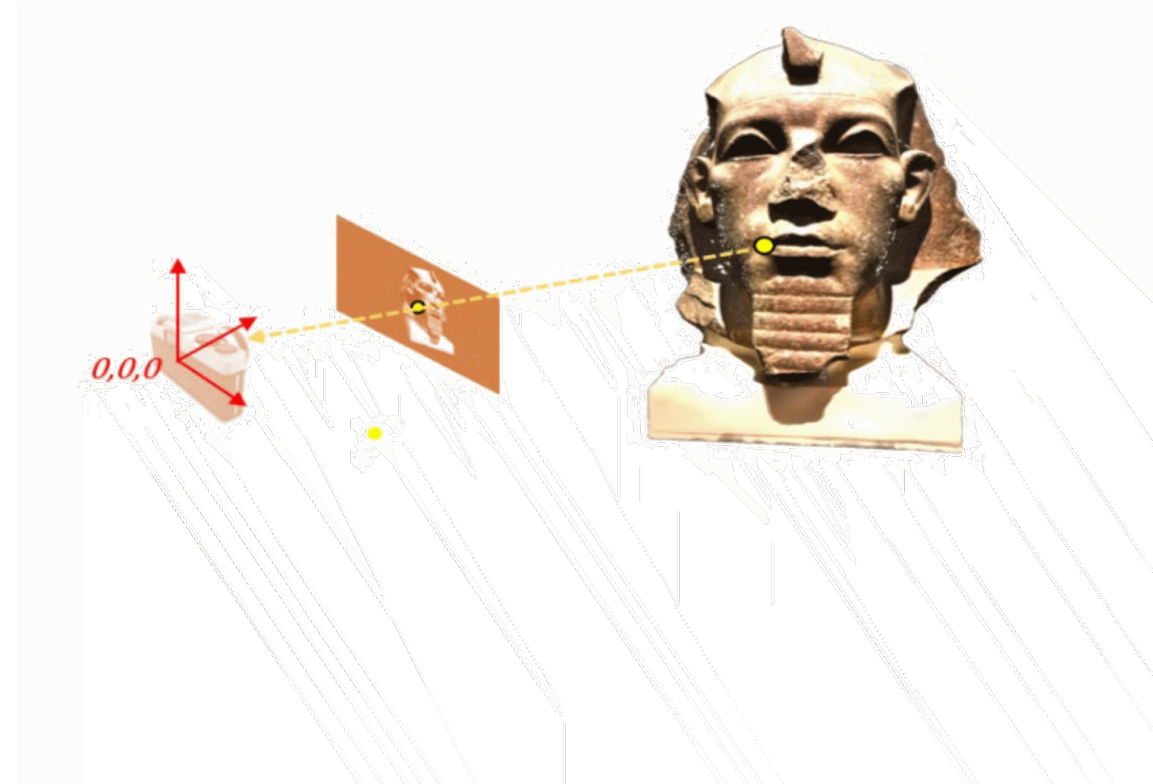
To build the whole track, we need to combine all pairwise matches of the points.



# INCREMENTAL SfM

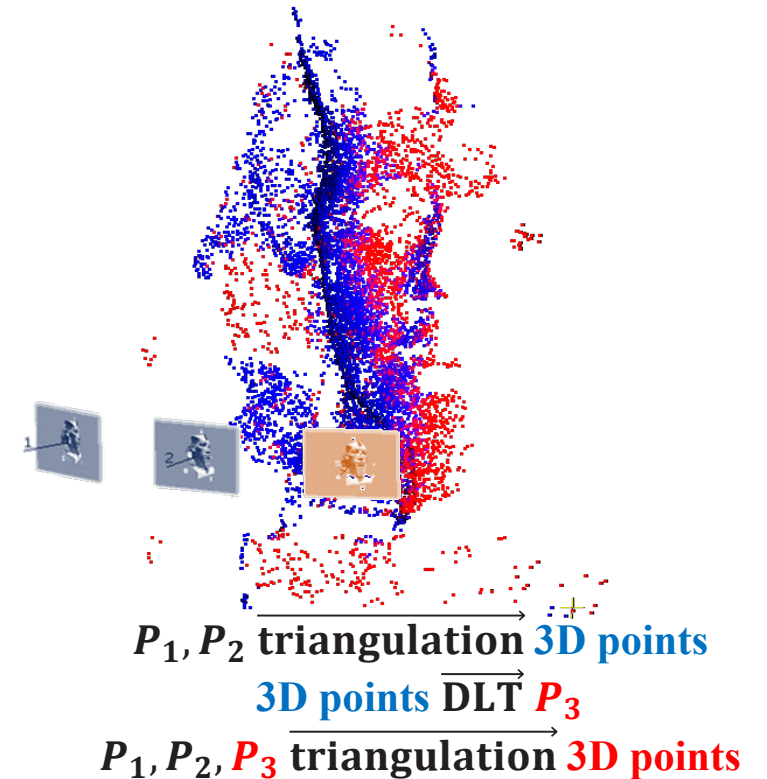
- After finding the feature track, the SfM starts from the first two images
- then we incrementally add the 3<sup>rd</sup> image and check the corresponding points based on the feature track matrix
- Since the 3D points are found from the first two views, we can estimate the  $P$  matrix of the third image using the spatial resection or the PnP method
- Then find new points to reconstruct by the triangulation method and update the total 3D points.

So Incrementally estimate the image motion related to the first image.

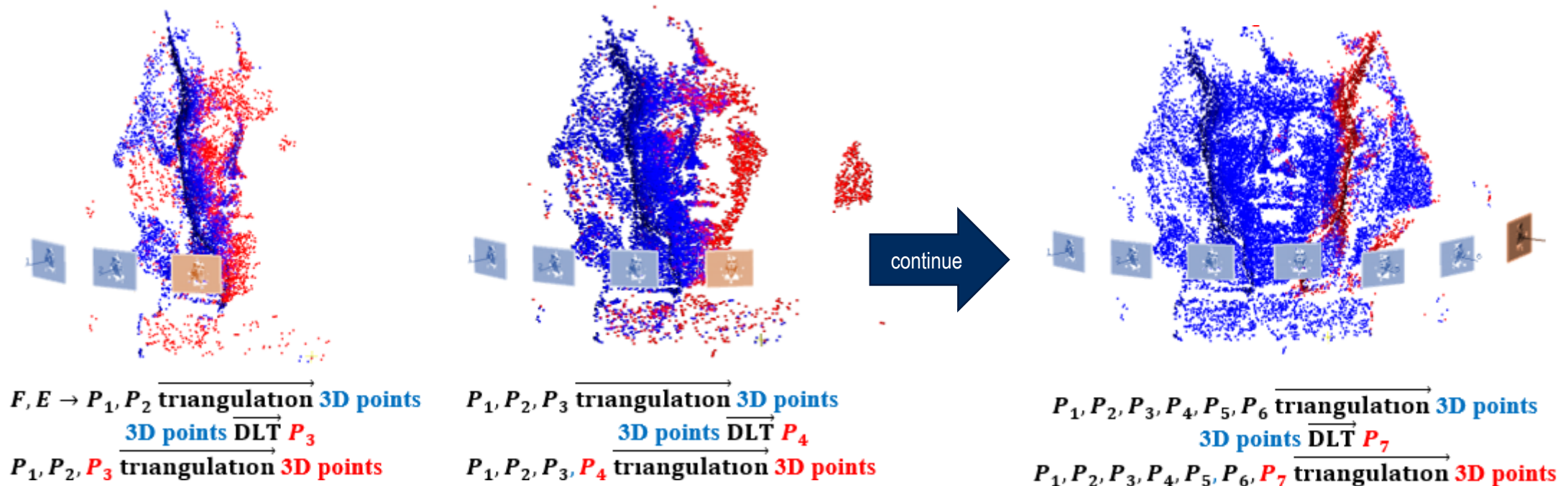


# INCREMENTAL MULTIVIEW SFM

- 3D points are found from the first two views,  $P_1$  &  $P_2$  estimated.
- $P_3$  matrix of the third image can be estimated using the spatial resection like DLT.
- Then find new points to reconstruct by the triangulation method and update the total 3D points.
- Continue incrementally ....
- In the end, we will have the camera motion represented by the projection matrices  $P_i$ ,  $i = 1:m$  and the structure represented by the 3D local coordinates of the tie points



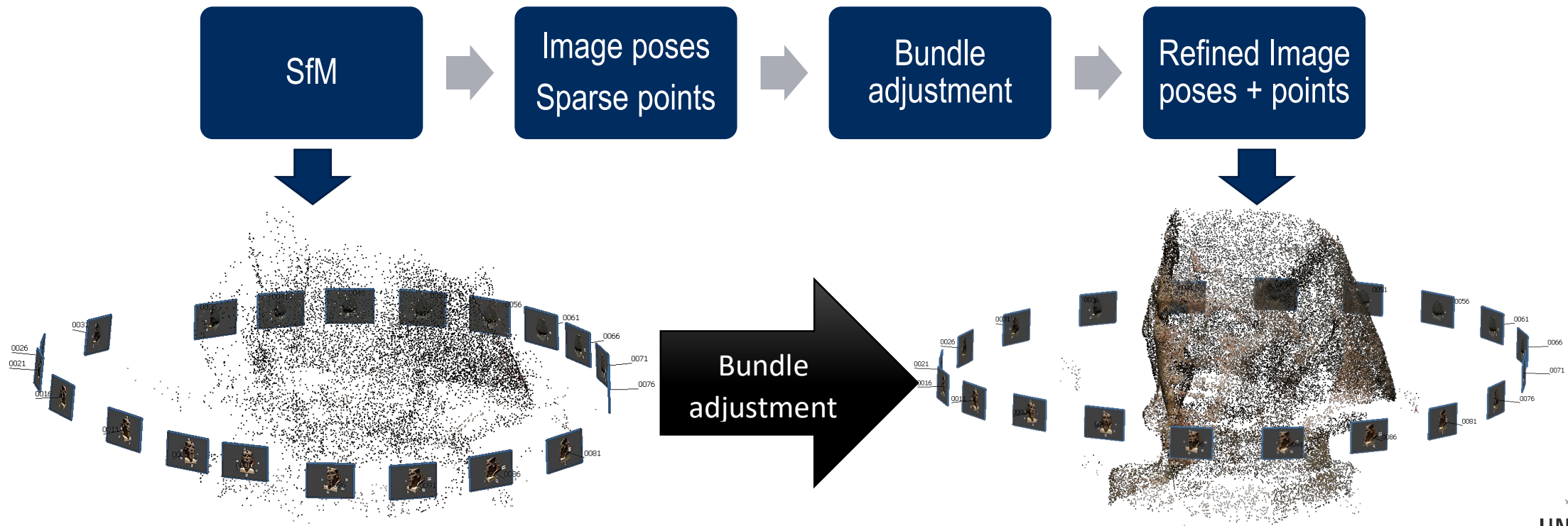
# INCREMENTAL MULTIVIEW SFM



The error will propagate incrementally and accumulate while adding views. What to do? Refine the structure and the motion by **Bundle adjustment**

# BUNDLE ADJUSTMENT BA

BA result in optimized values for the camera poses (camera trajectory), 3D points (3D map) as well as the camera calibration parameters if required.





# BUNDLE ADJUSTMENT BA

The input data will be:

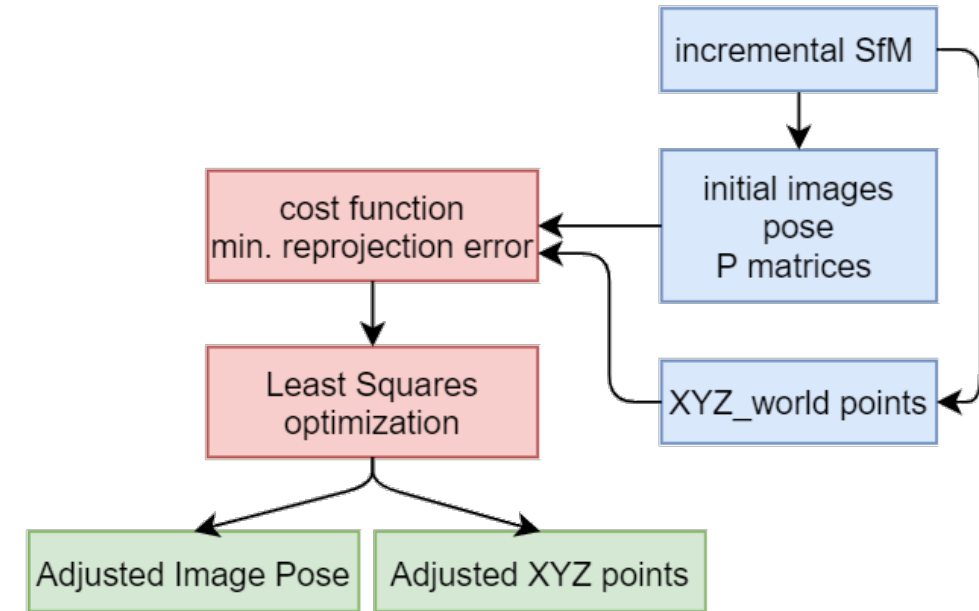
- Projection matrix of every image  $P_{i=1:m}$ .
- 3D coordinates of  $n$  tie points  $X, Y$ , and  $Z$ .
- Camera internal parameters ( $K$  matrix) and lens distortion parameters.
- Observed homologous image points in the viewing images.

Output: Refined motion and structure parameters

Objective: minimizing the total distances between the observed image points and their computed values

$$\phi = \min \sum_{i=1}^n \sum_{j=1}^m ((x_{ij} - \hat{x}_{ij})^2 + (y_{ij} - \hat{y}_{ij})^2)$$

where  $x, y$  are the observed image point coordinates and  $\hat{x}, \hat{y}$  are the projected image point coordinates



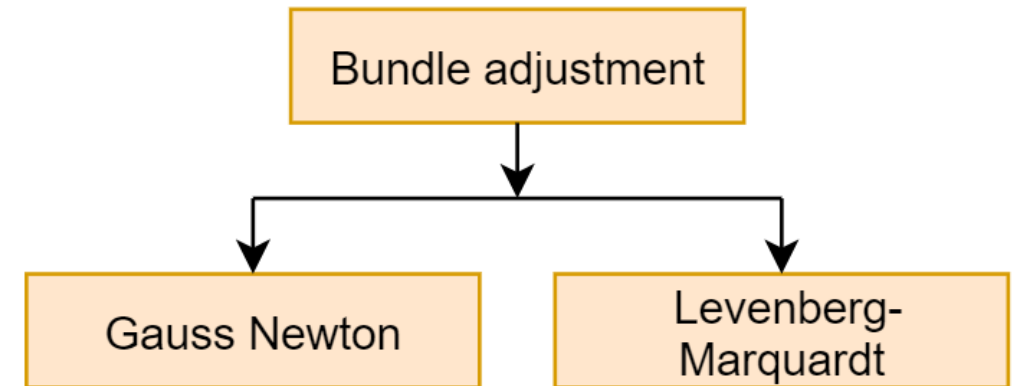


# BUNDLE ADJUSTMENT BA

- There is a need to use the track tensor we explained in the previous chapter or the visibility matrix  $\check{V}$  which indicates whether a point  $i$  is visible in image  $j$  or not

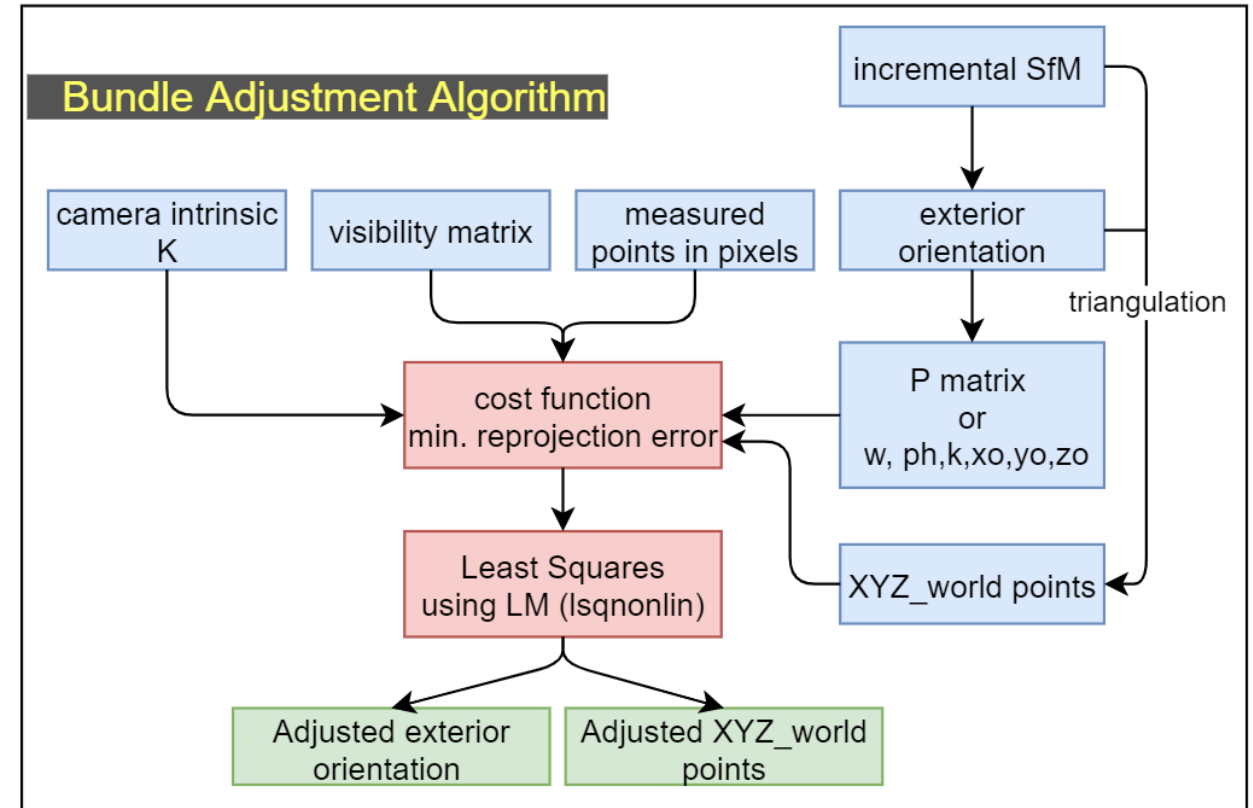
$$\min \sum_{i=1}^n \sum_{j=1}^m \check{V}_{ij} \left( (x_{ij} - \hat{x}_{ij})^2 + (y_{ij} - \hat{y}_{ij})^2 \right)$$

- BA is a nonlinear least squares optimization problem. Two solutions can be used for the problem.
- The Levenberg-Marquardt LM, often known as the damped least-squares method,
- The conventional Gauss-Newton method GN



# BUNDLE ADJUSTMENT BA

- Bundle adjustment packages differ in defining the rotation angles for example some using the quaternions and some the Euler angles.
- Some packages use the rotation  $R$ , and translation  $T$  for the images as an input for the pose while others using the projection matrices.
- Many packages use the Ceres solver <http://ceres-solver.org/> to handle the huge size of sparse normal equation matrices.



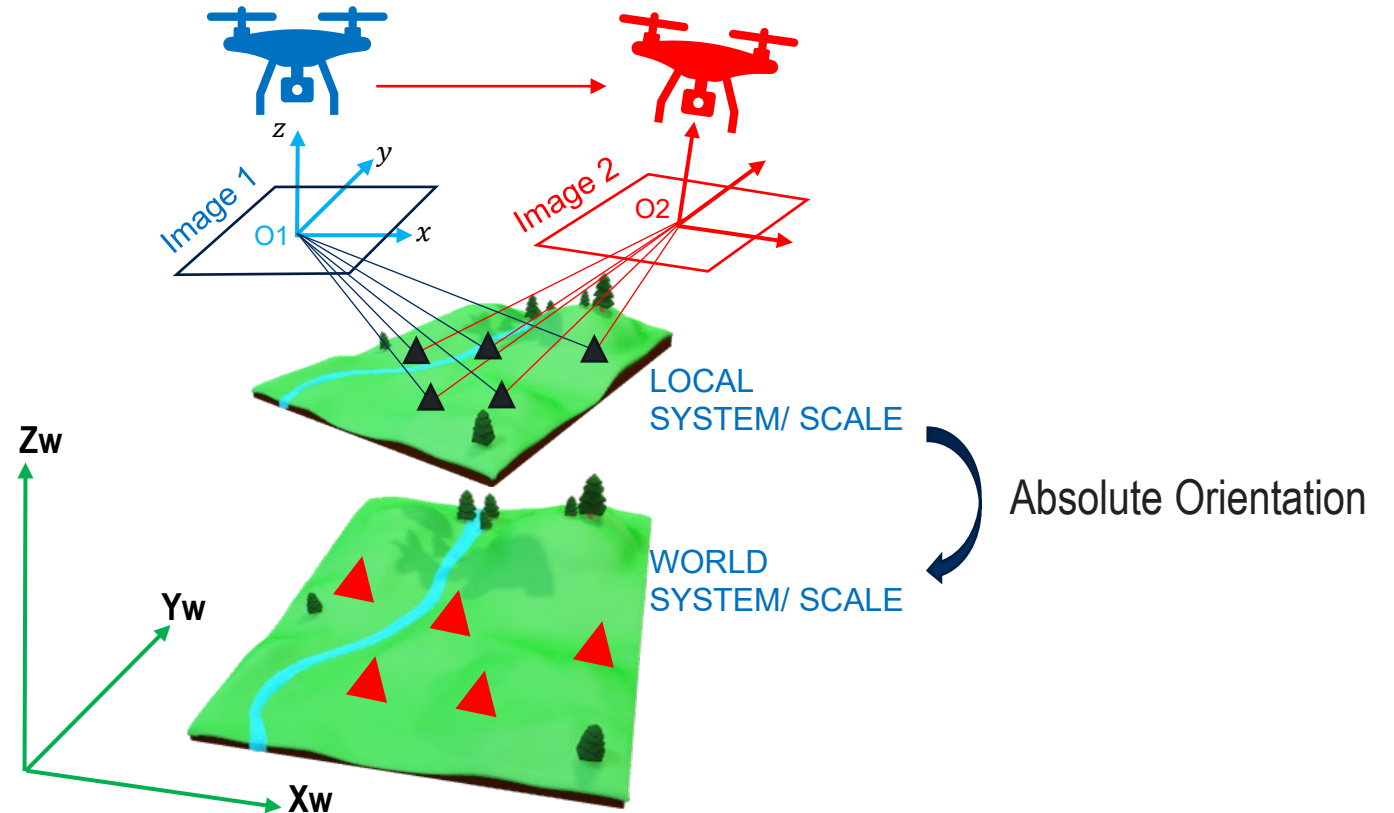
# WHAT HAVE WE LEARNED?

- The concept of SfM using Multiview data.
- The concept of bundle adjustment and its relation in the SfM.

# ABSOLUTE ORIENTATION

Absolute Orientation AO is aimed to transform the output of RO and triangulated object points into the world coordinate system XYZ.

Transformation means correct scale, rotation and translation

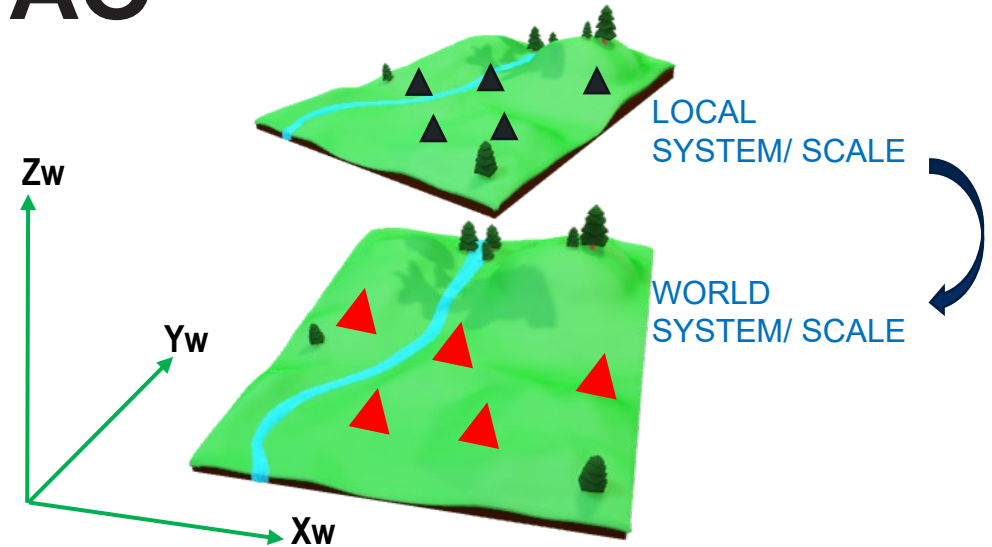


Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)

# ABSOLUTE ORIENTATION AO

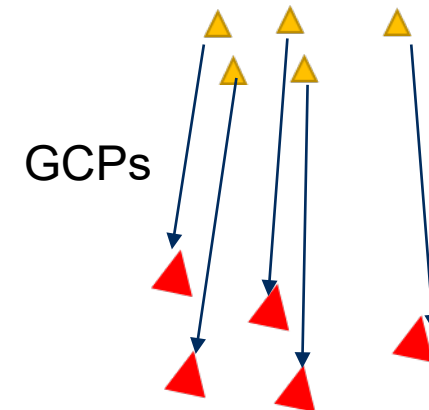
- Absolute orientation is a process of co-registration
- GCPs defined on both systems are required

Absolute orientation problem can be solved by estimating the Rotation, Translation, and Scale.



$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = s \begin{bmatrix} r_{11} & r_{21} & r_{31} \\ r_{12} & r_{22} & r_{32} \\ r_{13} & r_{23} & r_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} Tx \\ Ty \\ Tz \end{bmatrix}$$

Absolute orientation parameters

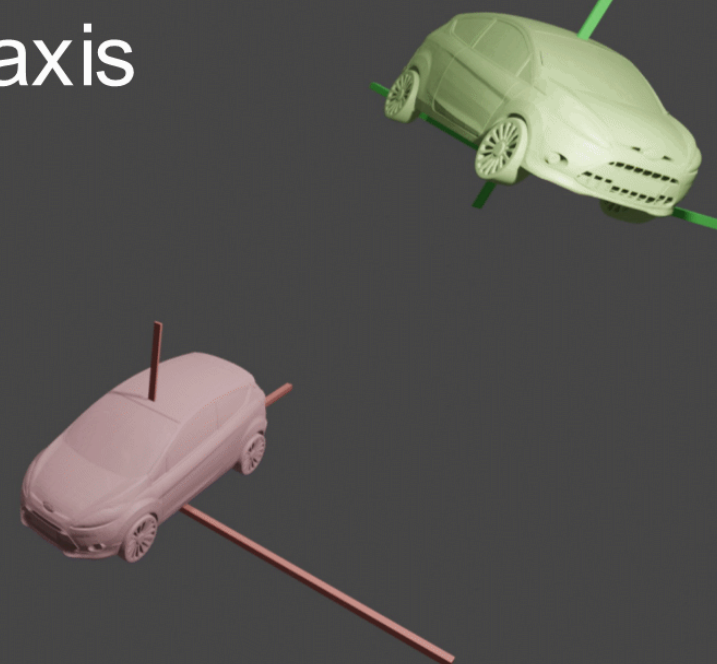


# AO IS A 3D TRANSFORMATION

Absolute orientation is a 3D transformation of 7-parameters  
3 translations, 3 rotations and a scale.

Least-squares adjustment using either iterative solution or closed form direct solution.

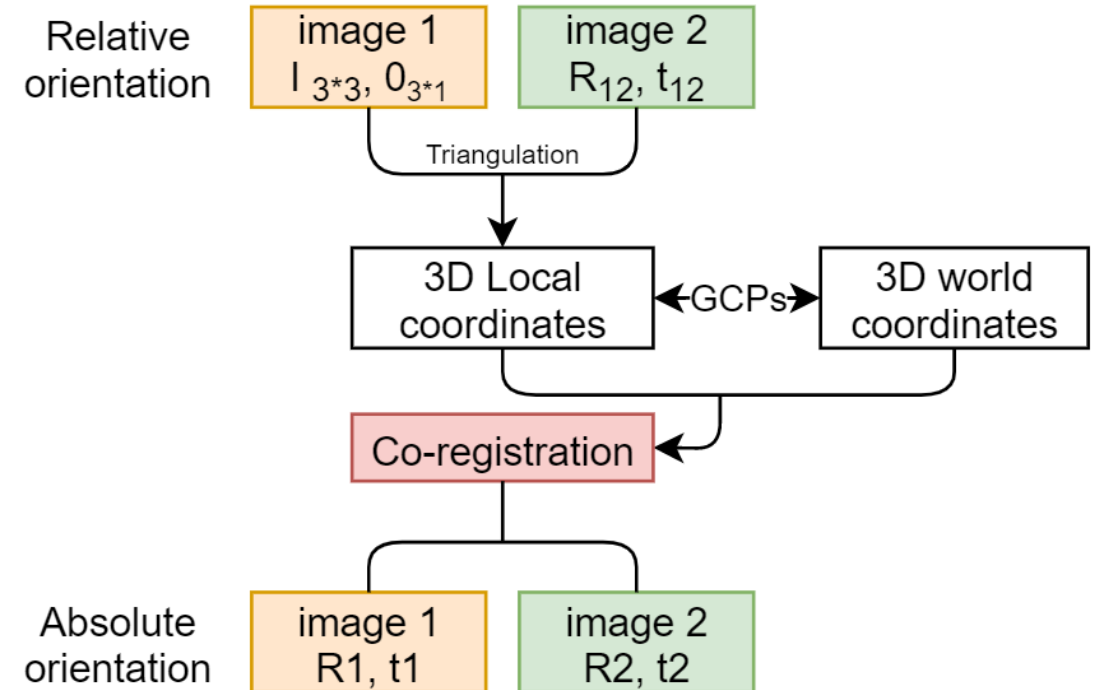
Translation - X axis

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = s \underbrace{\begin{bmatrix} r_{11} & r_{21} & r_{31} \\ r_{12} & r_{22} & r_{32} \\ r_{13} & r_{23} & r_{33} \end{bmatrix}}_{\text{rotation}} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \underbrace{\begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}}_{\text{translation}}$$


# ABSOLUTE ORIENTATION IN A SUMMARY

- Start from the relative orientation by using the fundamental matrix +RANSAC
- Run a triangulation for the scene XYZ points
- Run absolute orientation to make the images oriented with respect to the world coordinate system.
- GCPs are used to achieve the task of 3D coregistration between the local system and the world system

Q/ does this approach accurate enough?  
or we need to apply a refinement?



Source: Alsadik. [www.ltc.nl](http://www.ltc.nl)



# QUIZ

The stereo pair SfM include different steps of (relative orientation, absolute orientation, image matching, triangulation). We want you to select the correct sequence of these computational steps?

- a) relative orientation, absolute orientation, image matching, triangulation
- b) image matching, relative orientation, absolute orientation, triangulation
- c) image matching, relative orientation, triangulation, absolute orientation
- d) relative orientation, triangulation, absolute orientation while image matching is not related.

**THANK YOU**