26/11/2024

exploiting
d knowledge
distance -> size
can be solved

Two-view geometry

Epipolar geometry

3D reconstruction

F-matrix comp.

Structure comp.

Intrisic ambiguity in 3D world → 2D image projection

↳ in multi-view geometry we have to solve intrinsic 3D world ambiguity...

even if camera calibrated, we don't know the distance! Just the direction..

Courtesy slide S.Lazebnik

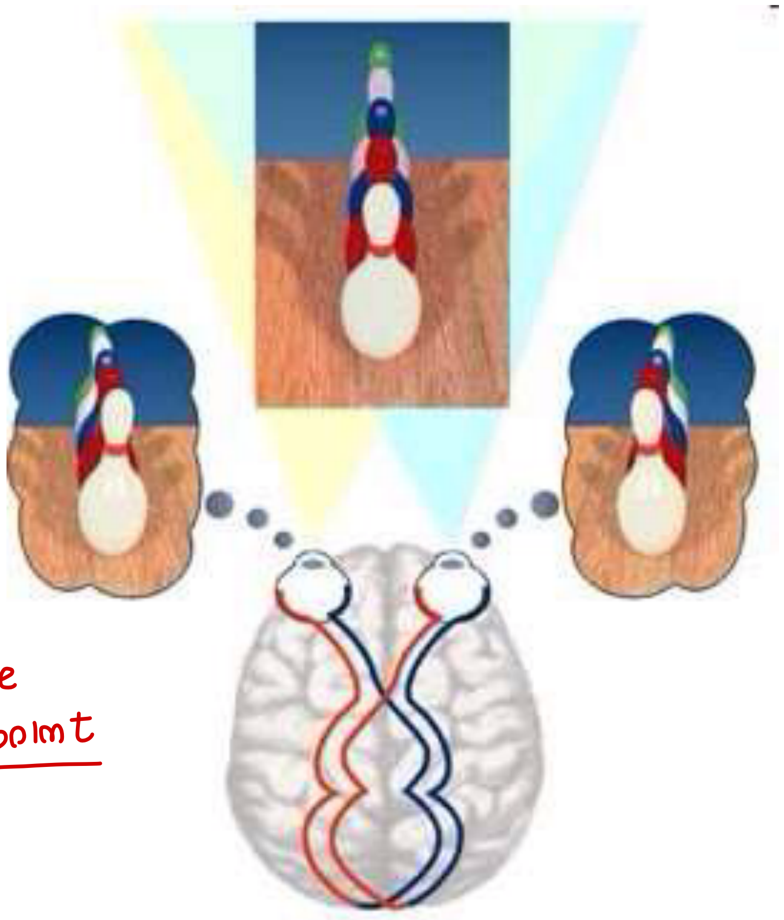
from this camera view we don't reconstruct size!

⇒ if we use more than one image, from different viewpoints! ⇒

After projection, different depths cannot be distinguished in the image.

ambiguity is
different with
different viewpoint!

⇑
we can reconstruct the
scene from different viewpoint



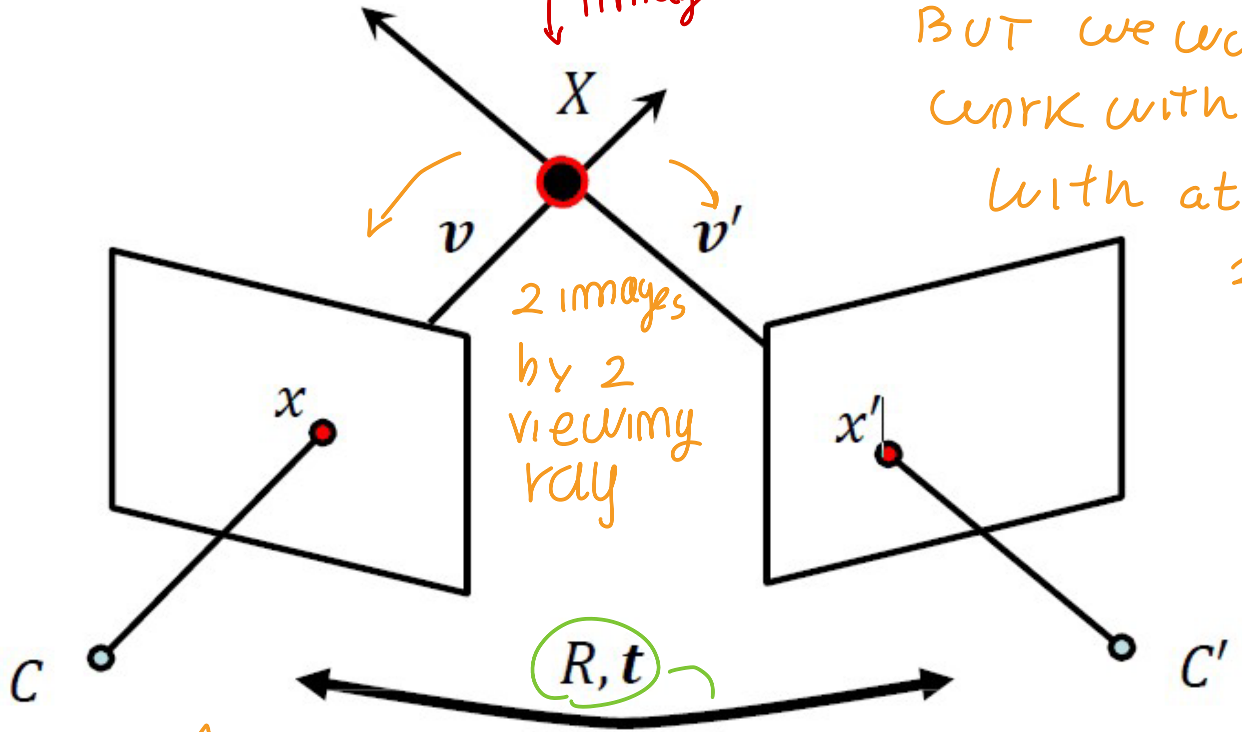
Multiple views of the same scene can help to solve such ambiguities:
In images taken from different viewpoints, ambiguities are not the same

We need to understand geometry of this to develop techniques to reconstruct set of points in 3D →

Multiple views of a point X in 3D

(we use 2 cameras) X in 3D world generate two images

up to now we deal with 2D planar scene... BUT we want to work with 3D points with at least 2 views of same scene!

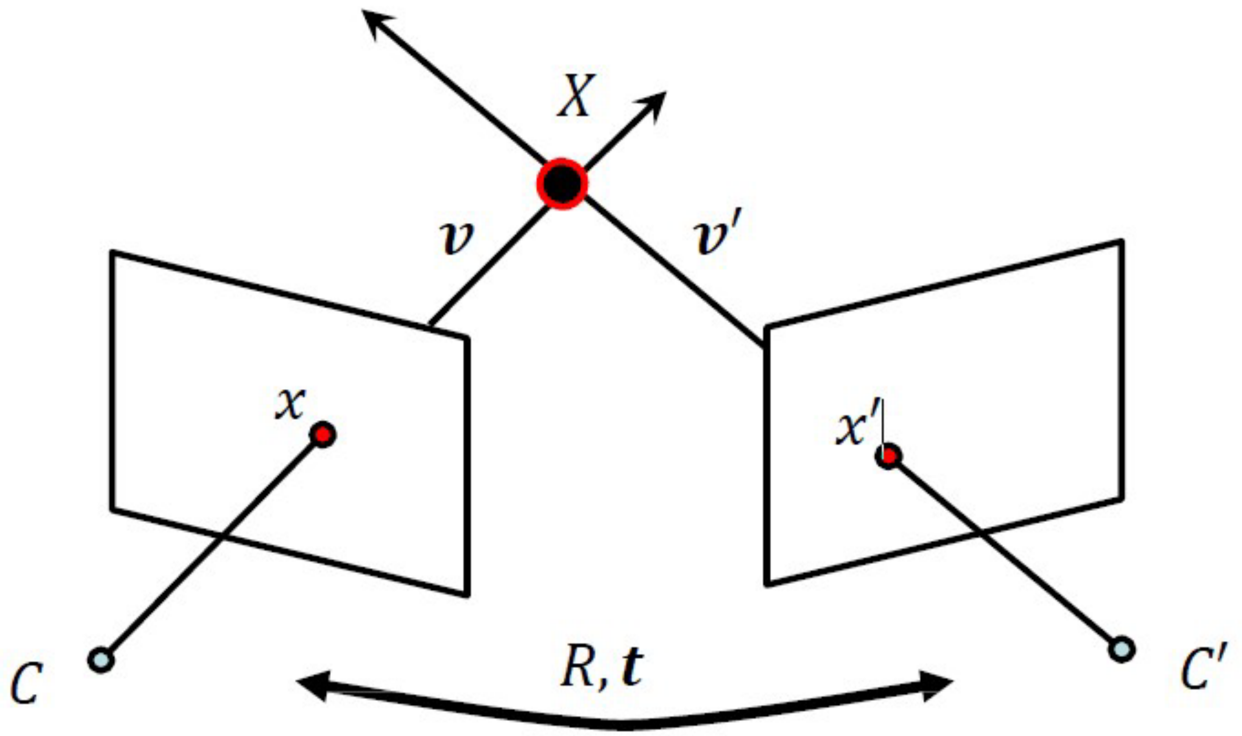


↑ the viewing rays intersect in X point

R : relative rotation matrix

relative pose of the two cameras... (rotation / translation)

Multiple views of a point X in 3D: scenarios



(SIMPLE CASE)

scenario 1: STEREO vision

x ←→ x' known; R,t known; K and K' known;

→ find **X**

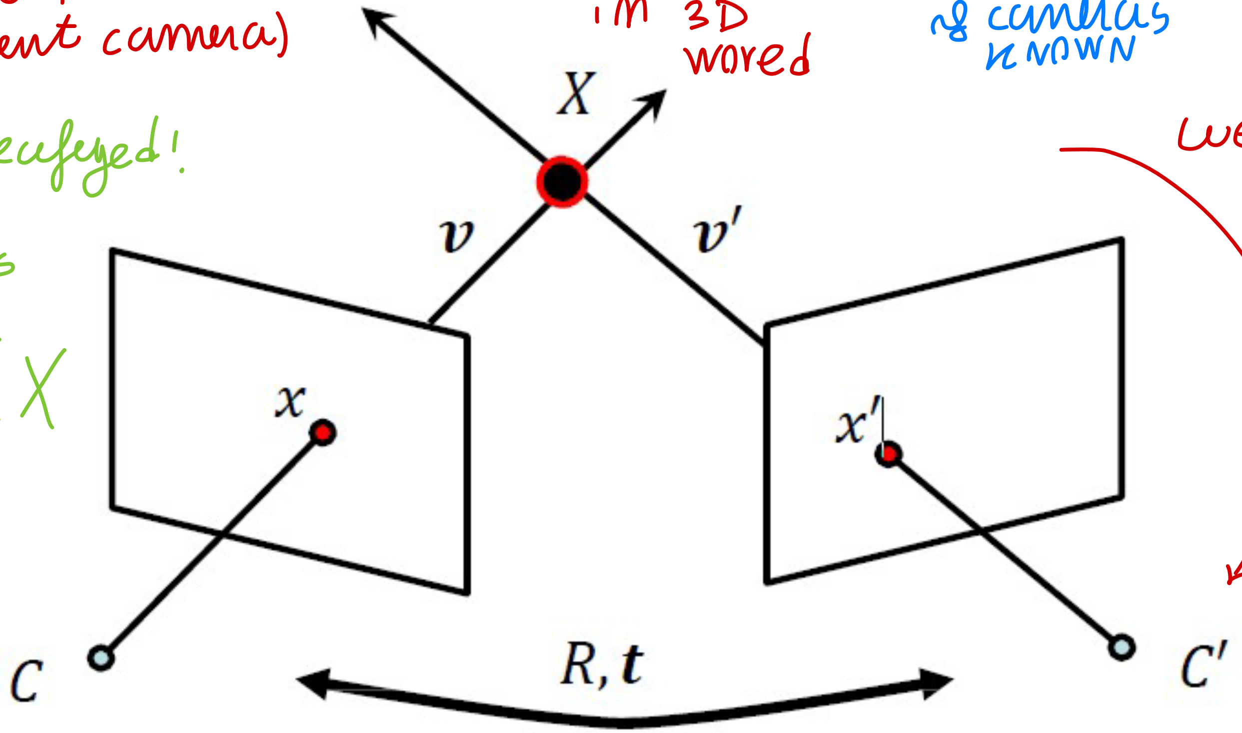
in 3D world

x, x' corresponding points! (reproduced by same point in 3D on two different camera)

relative position of cameras KNOWN

camera calibrate

correspondance specified! NOT trivial, techniques for points matching s.t both x, x' are images of X



we can compute the viewing ray associated to this points x, x'

since we KNOW R, t position, v, v' easy to compute

and intersect

old techniques, used by boats to recognize stars and compute directions

from K(K') and x(x') we compute viewing ray $\boldsymbol{v}(\boldsymbol{v}')$ →

triangulation: $\boldsymbol{X} = \boldsymbol{v} \cap \boldsymbol{v}'$

concept of triangulation

scenario 1: STEREO vision ≃ based on triangulation

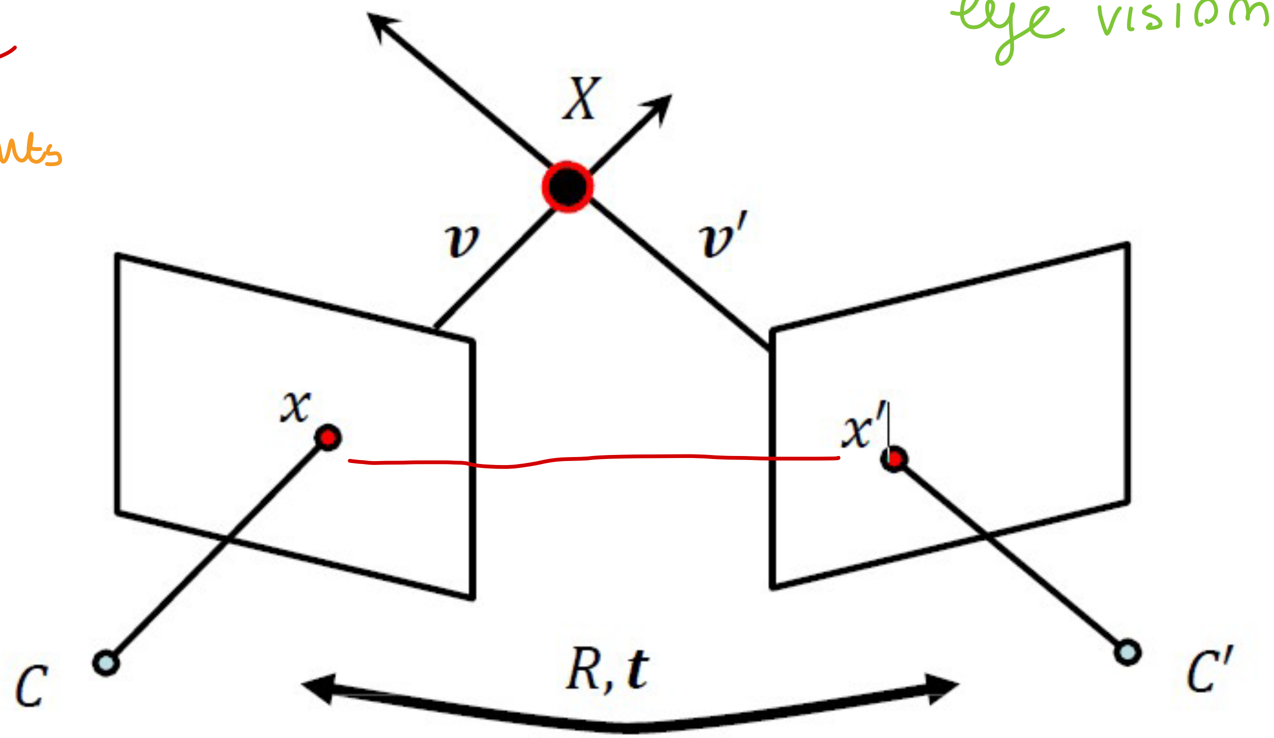
x ←→ x' known; R,t known; K and K' known;

→ find **X**

as in human eye vision

having 2 lines not parallel, you intersect it as 2 segments of triangle

if you know segments and orientation, INTERSACTION can be found as viewing rays point of contact



from K(K') and x(x') we compute viewing ray $\boldsymbol{v}(\boldsymbol{v}')$ →

triangulation: $\boldsymbol{X} = \boldsymbol{v} \cap \boldsymbol{v}'$

INTERMEDIATE

scenario 2: calibrated structure from motion

(we move the camera!)

x ←→ x' known; **R,t unknown**; K and K' known;

→ find **X** and **R,t**

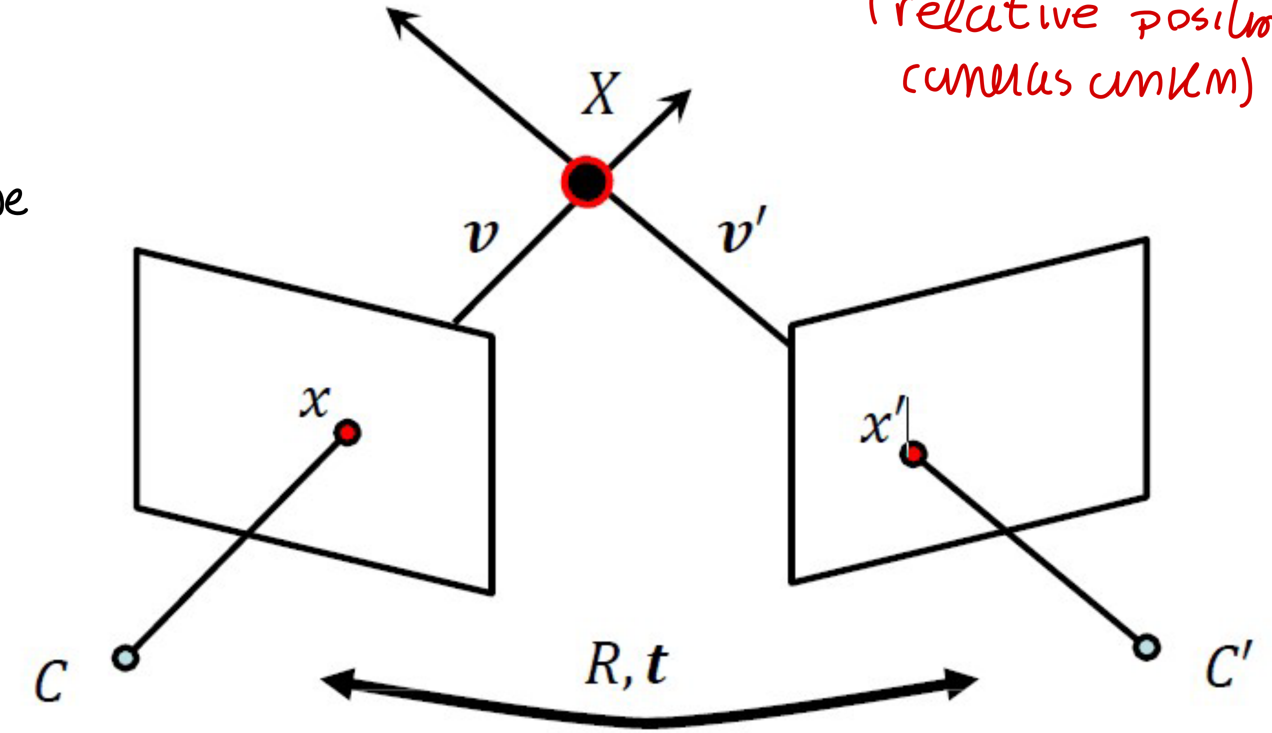
I want to move camera freely!

(relative position of cameras unknown)

↓

take more images without tracking positions (having stereo vision is hard)

similar to typical cv problem... I don't know relative position, I want to reconstruct shape



use **epipolar constraint** to estimate R, t; → compute viewing rays $\boldsymbol{v}$, $\boldsymbol{v}'$

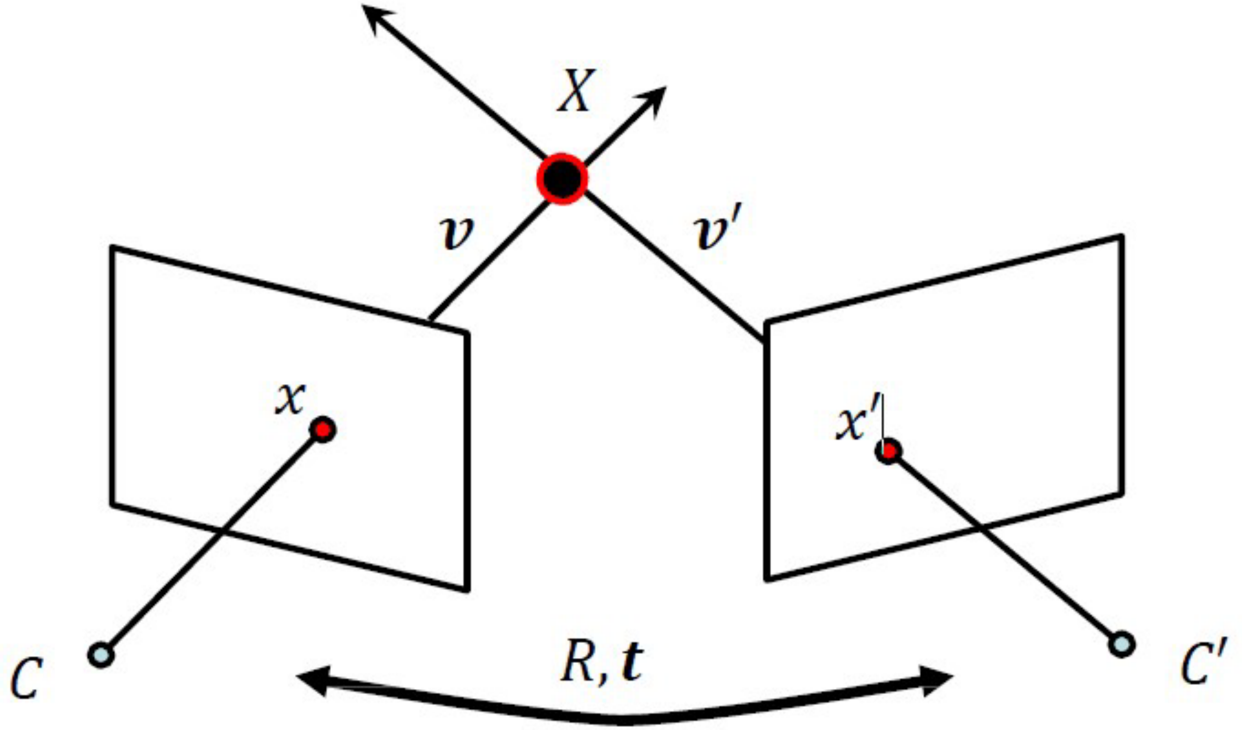
→ triangulation: $\boldsymbol{X} = \boldsymbol{v} \cap \boldsymbol{v}'$

scenario 3: uncalibrated structure from motion

x ←→ x' known; **R,t unknown**; **K and K' unknown;**

↑harder case!
MORE COMPLEX!

→ find X, **R,t** and **K, K'**

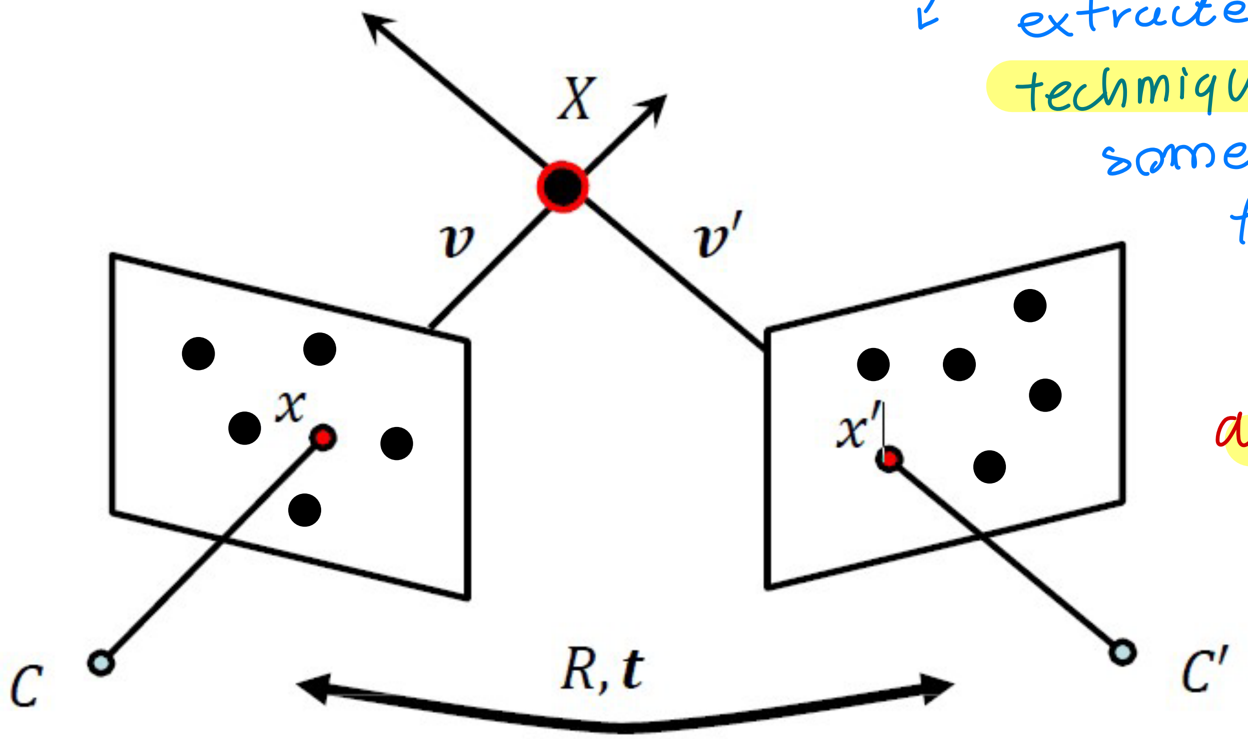


use **epipolar constraint** & partial information on scene and/or cameras to estimate K, K', R, t; then compute viewing rays $\boldsymbol{v}$, $\boldsymbol{v}'$ → triangulation: $\boldsymbol{X} = \boldsymbol{v} \cap \boldsymbol{v}'$

1 First of all: given many features in both images find pairs of **corresponding** features

determine points extracted correspondance... techniques based on some descriptor of features of images

We suppose corresp. are KNOWN...



Feature extraction, feature descriptor computation and feature matching

Three questions:

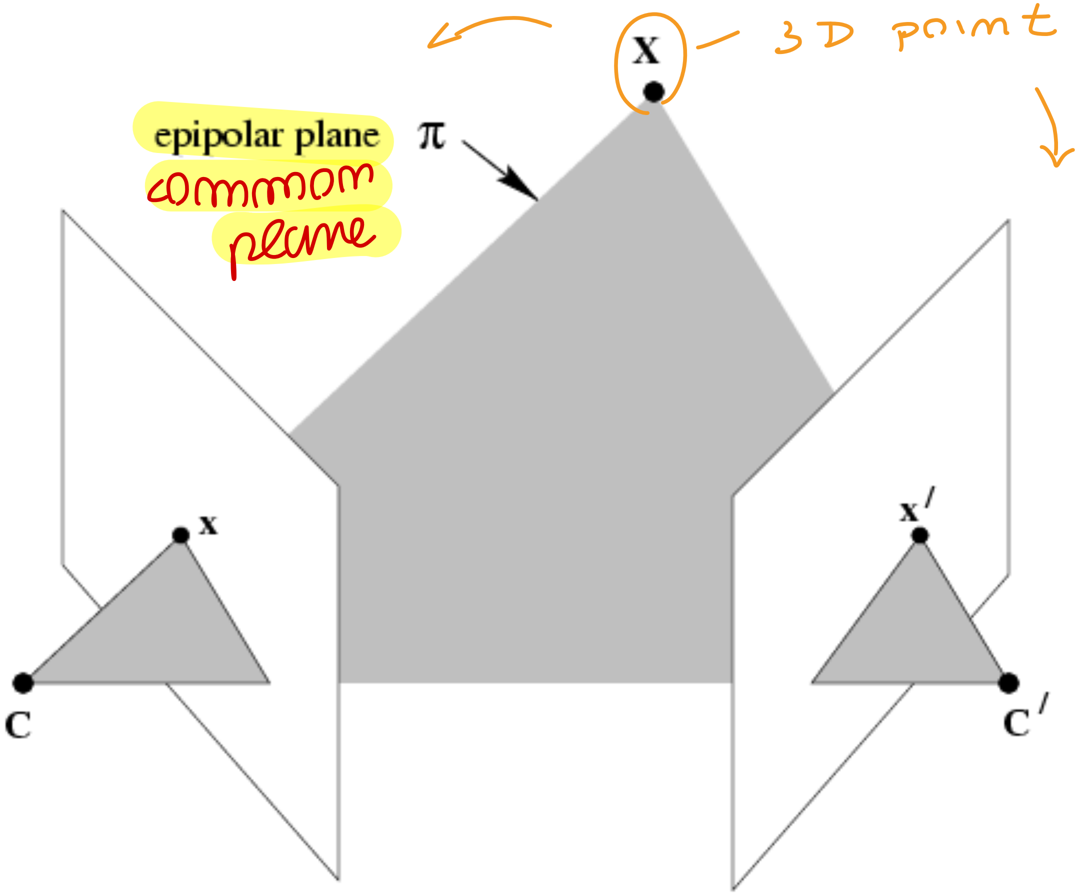
(i) Correspondence geometry: Given an image point x in the first view, how does this constrain the position of the corresponding point x' in the second image?

(ii) Camera geometry (motion): Given a set of pairs of corresponding image points $\{x_i \leftrightarrow x'_i\}$, $i=1,\ldots,n$, what are the cameras P and P' for the two views?

(iii) Scene geometry (structure): Given corresponding image points $x_i \leftrightarrow x'_i$ and cameras P, P', what is the position of (their backprojection) X in space?

↓ phenomenom of multi-view geom... the same ∀ scenario

The epipolar geometry



3D point ⇐ x, c, c', x', X

identify plane!

⇓

all 5 points are co-planar

epipolar plane

Scene point, its images and camera centers are coplanar

C,C',x,x' and X are coplanar

we define also: → from this we derive many properties ⟹

The epipolar constraint: viewing rays of corresponding image points must intersect in 3D space ⚠

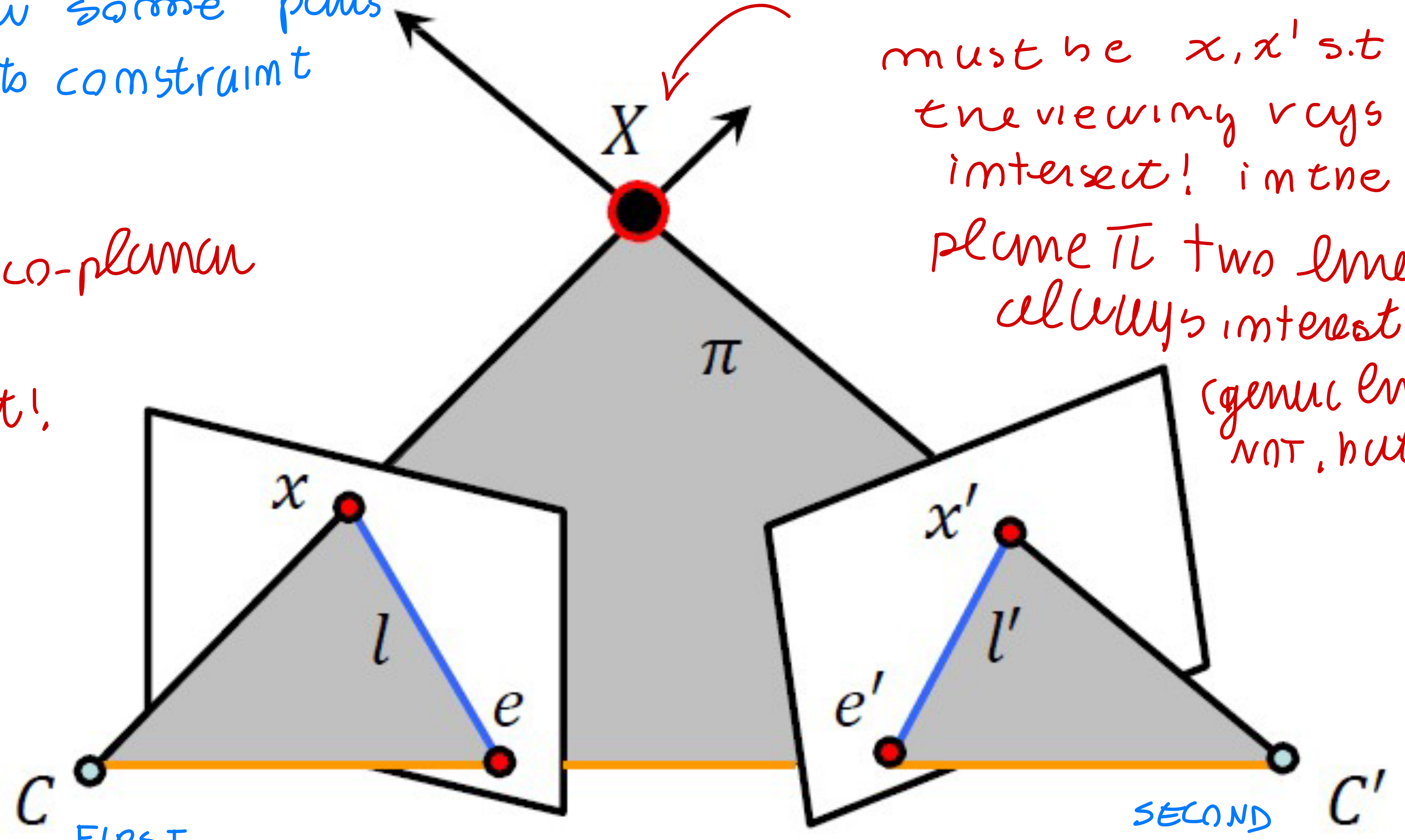
special situation

suppose we know some points and we want to constraint others...

those points co-planar are related by same constraint!

must be x, x' s.t the viewing rays intersect! in the plane π two lines co-planar always interest!

(generic line in 3D NOT, but yes in 2D)



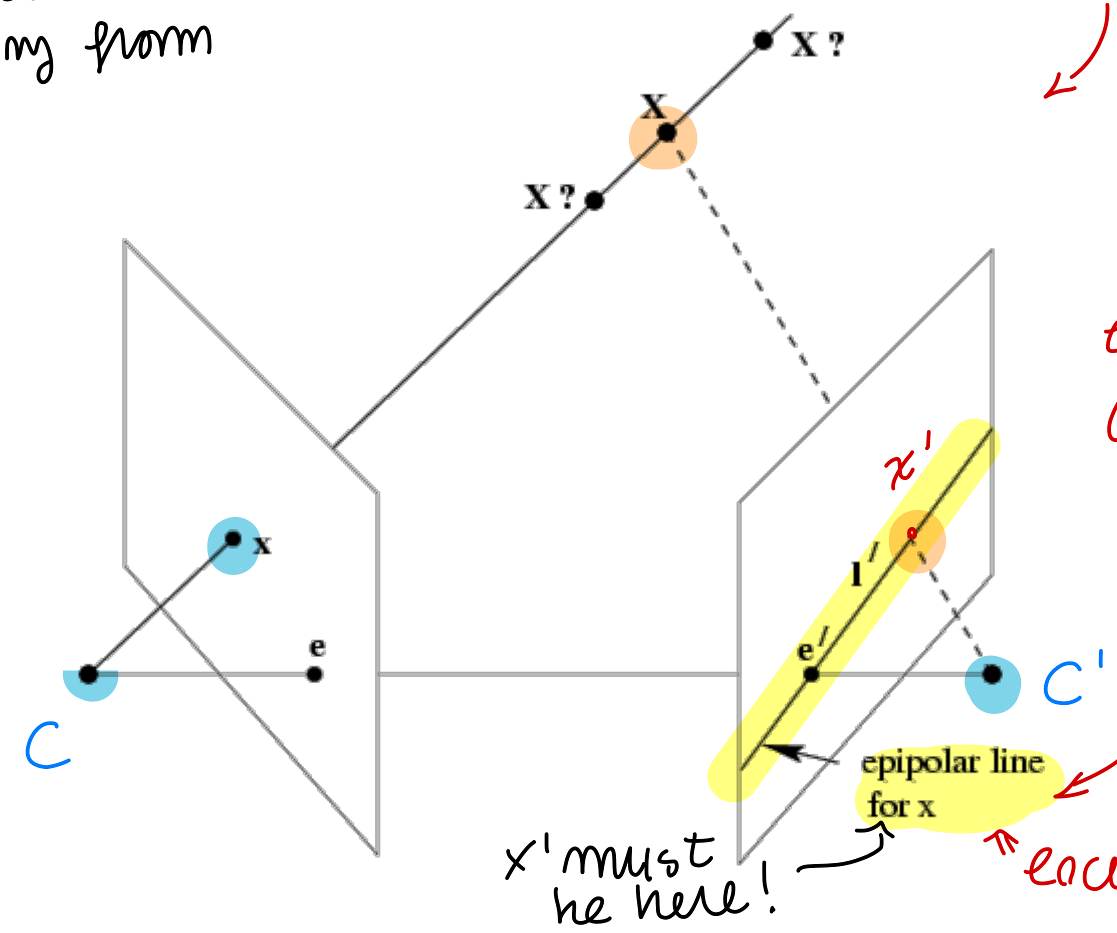
FIRST CAMERA

SECOND CAMERA

The epipolar constraint: viewing rays of corresponding points must intersect in 3D space

if we want to know something from components

since we know X produce x, we know viewing ray; we can have X on v, correspondingly x' is projection of X



the points X, x' cannot be wherever we want or the constraint will not be respected

second image has to be in the line where X should be

locus of all possible position of II image x'

What if only C,C',x are known?

The epipolar constraint: viewing rays of corresponding points must intersect in 3D space

this point e' belongs to any epipolar line.

any epipolar line must go through it.

"image taken by II camera of the center of I camera"

epipole

also, this line cannot be a generic line... should include first center... consider image of I camera center C taken by II camera, this image is part of epipolar line

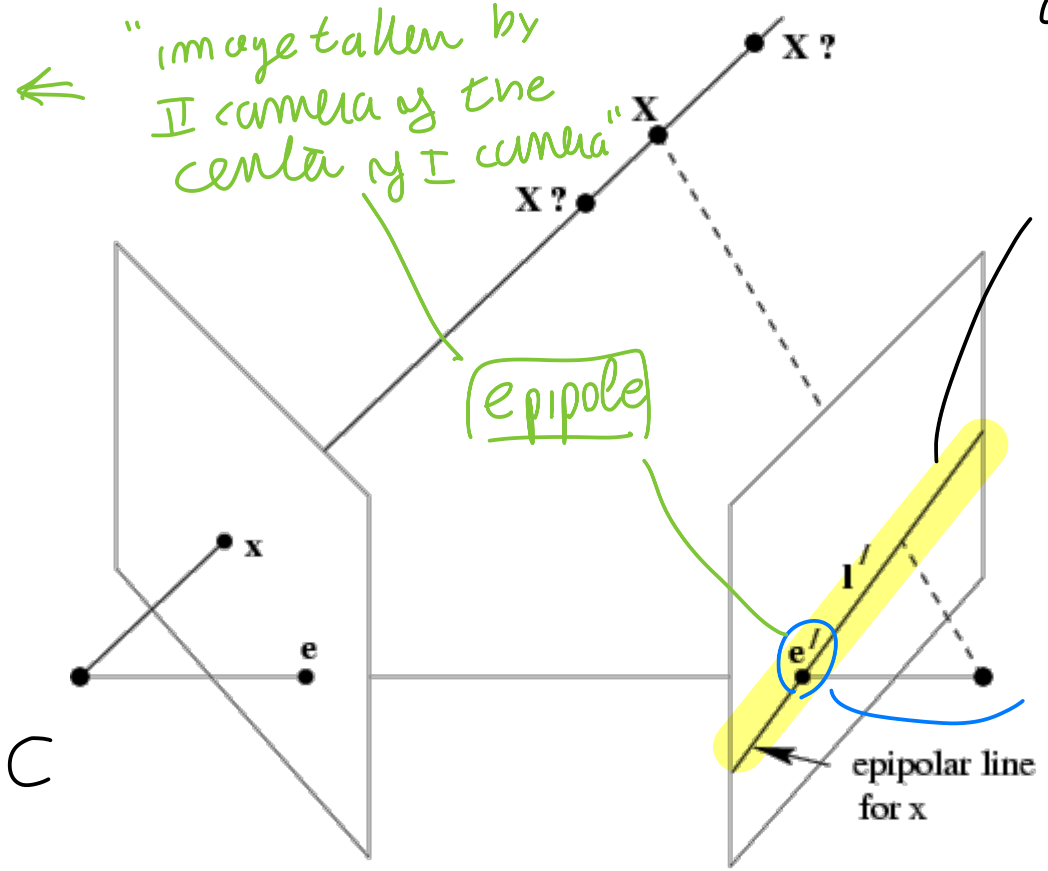
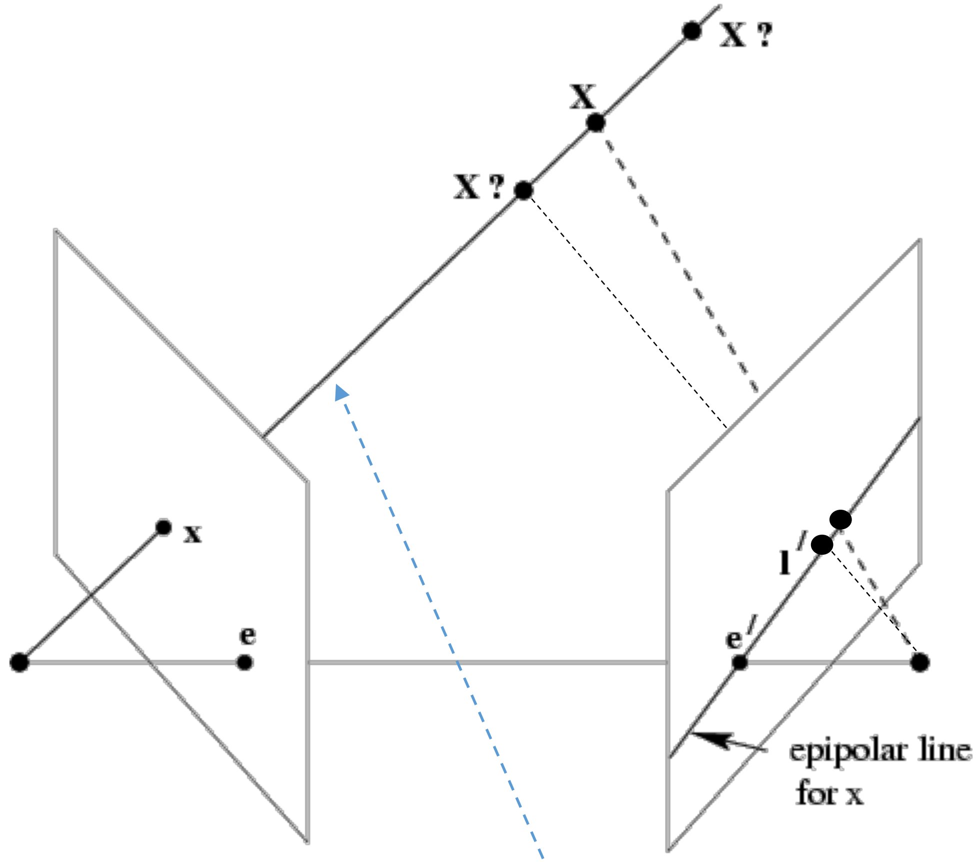


image of C, first camera

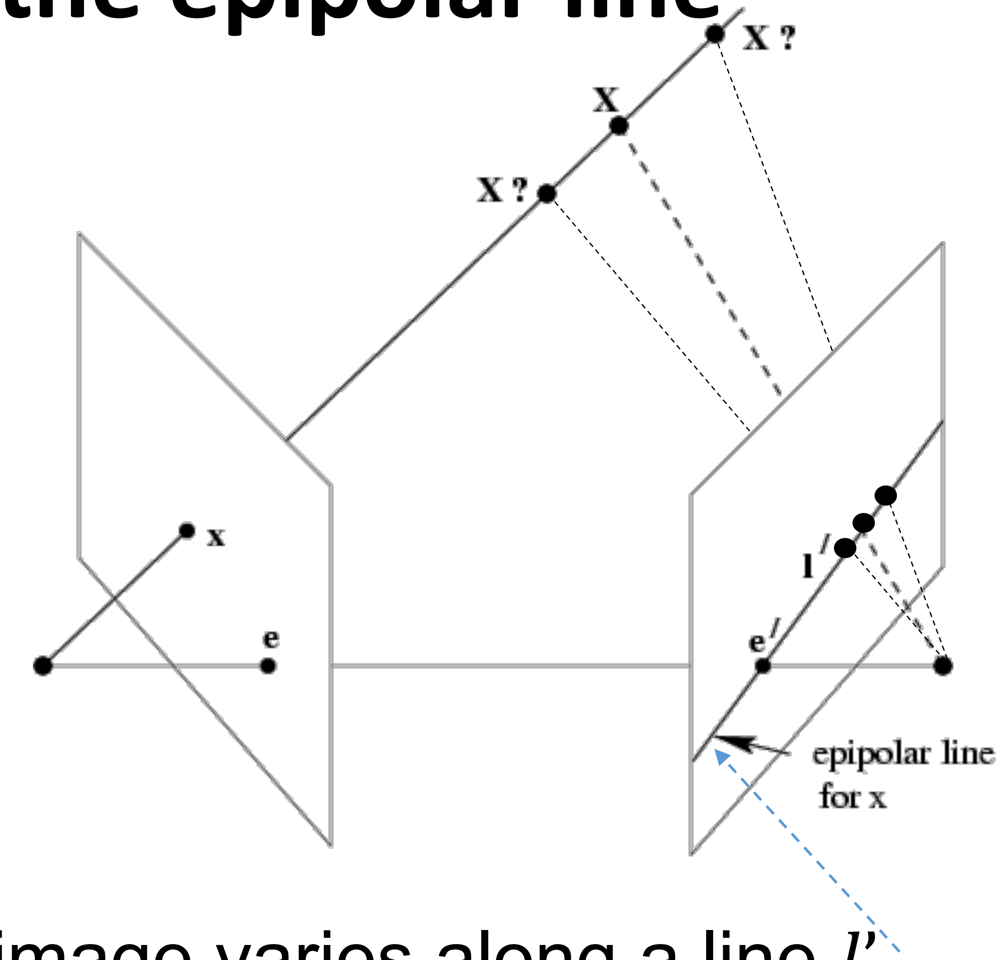
What if only C,C',x are known?

The epipolar constraint



possible position of X in the 3D space
varies along the viewing ray associated to x

The epipolar constraint: the epipolar line



their image varies along a line l'.

l': image projection of the viewing ray

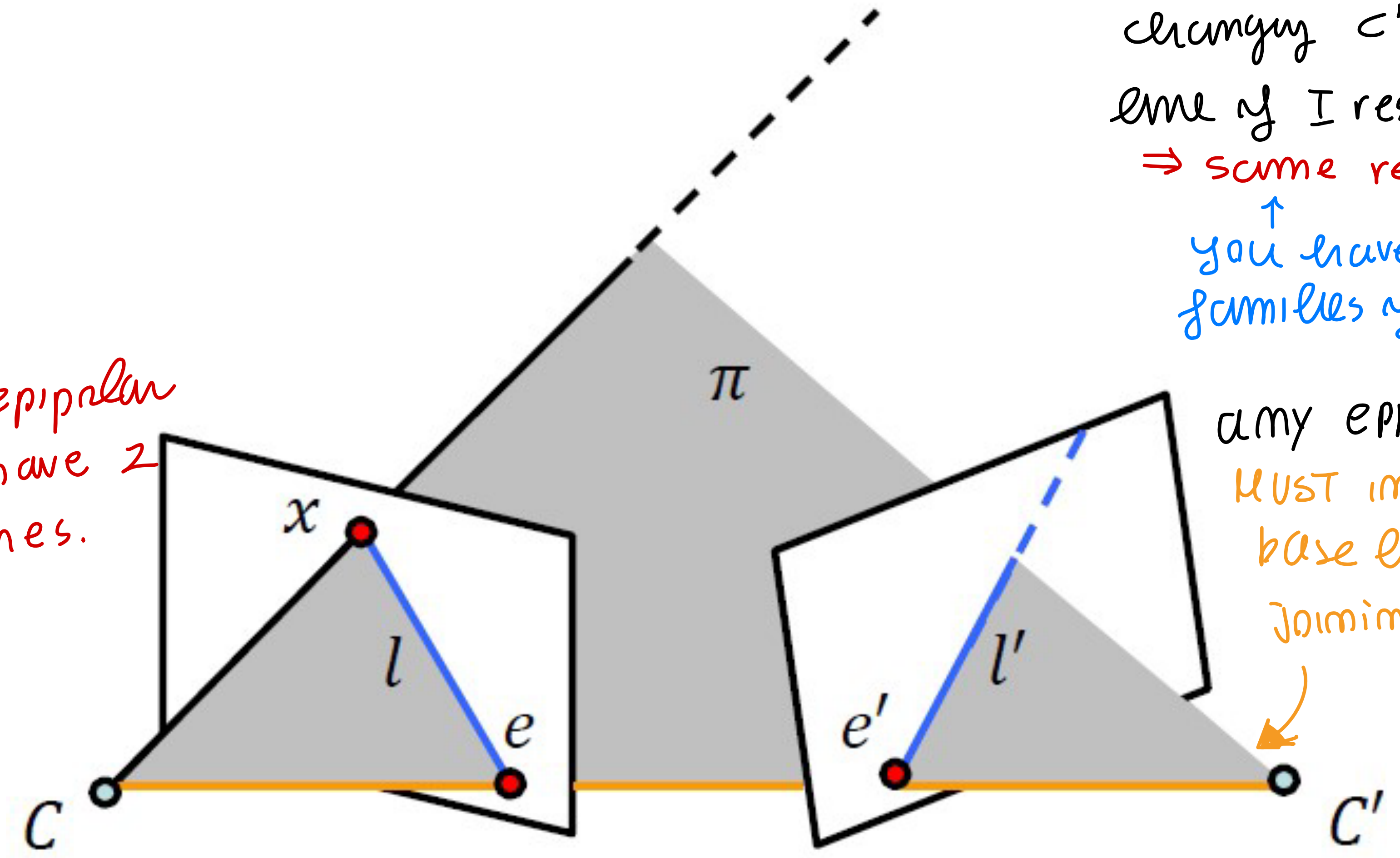
Correspondence between epipolar lines

if you do the symmetric case....

changing c', c epipolar line of I respect II camera

⇒ same result

you have two families of epipolar line



for each epipolar plane you have 2 epipolar lines.

any epipolar plane MUST include base line (CC') joining centers

≃ pencil of planes around base line, 1 DOF planes!

The epipolar constraint: the epipole

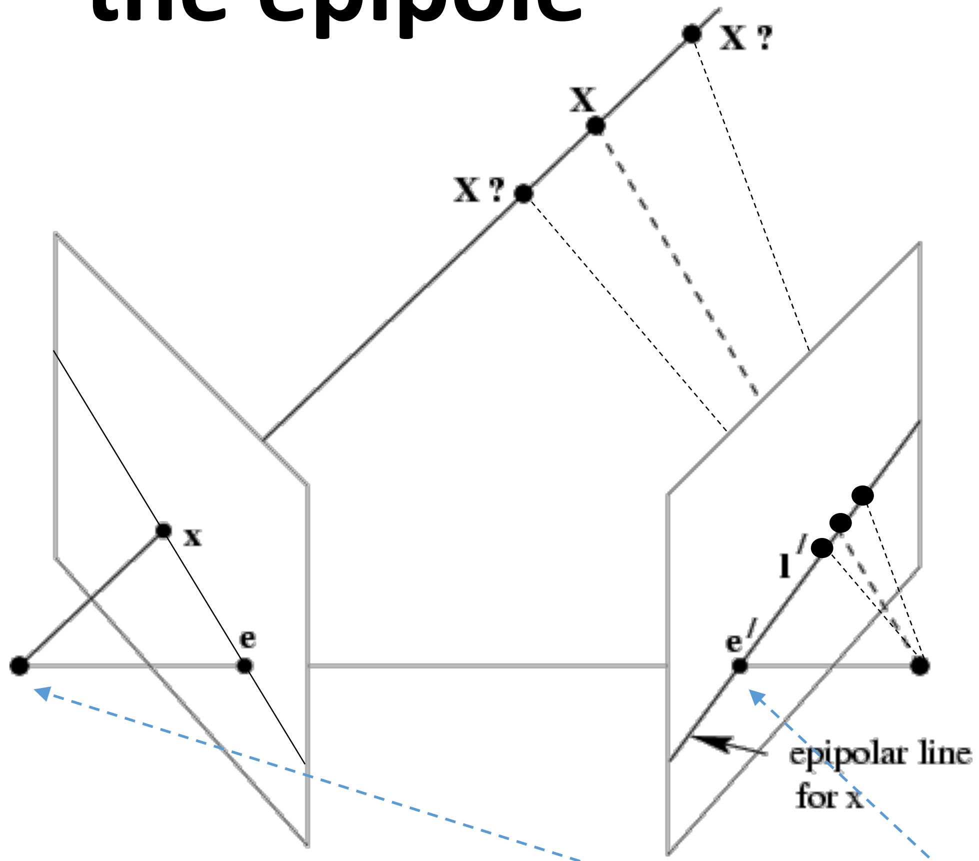
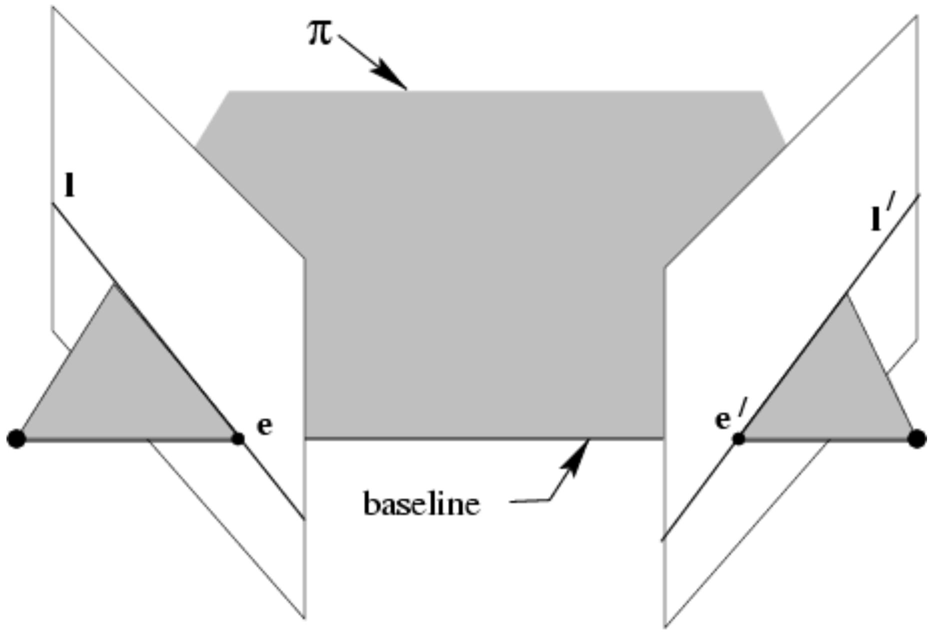


image projection of the viewing ray

the viewing ray is a line through the 1st camera center

→its image l' always goes through the image $\mathbf{e'}$ of the 1st camera center

The epipolar constraint

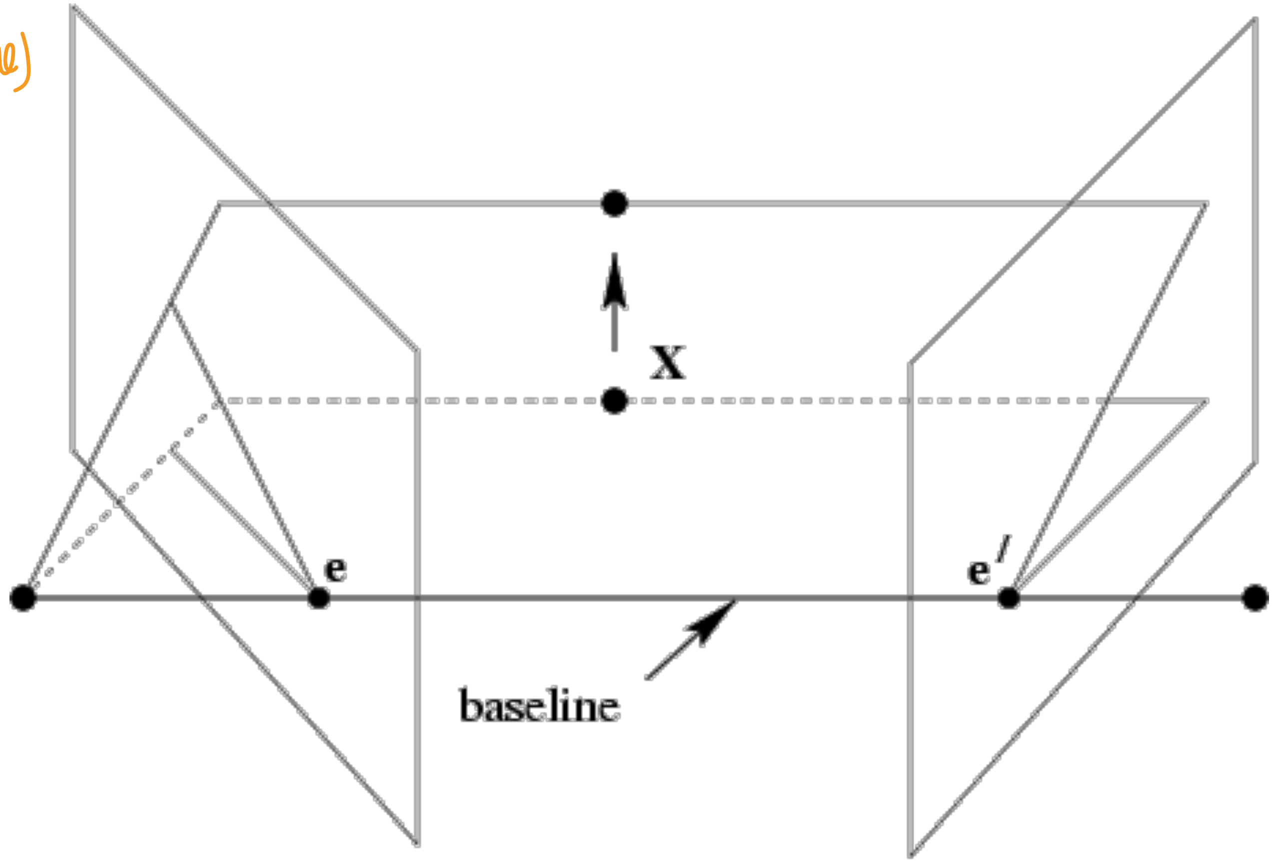


All points on π project on l and l'

The epipolar constraint

concurrent planes
letting it vary...
you can define,
CROSS RATIO!
(2D CR, OR 3D for plane)

families of
epipolar lines
are projectively
corresponding

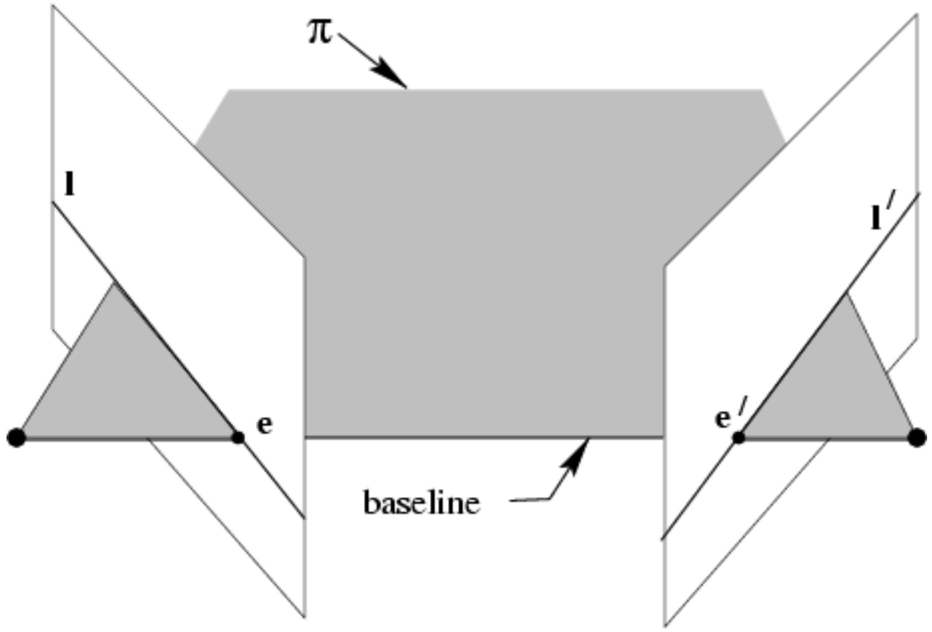


Family of coaxial planes π (with baseline as axis) cross the image planes at families of epipolar lines l and l'

Family of lines l concur at e and family of lines l' concur at e'

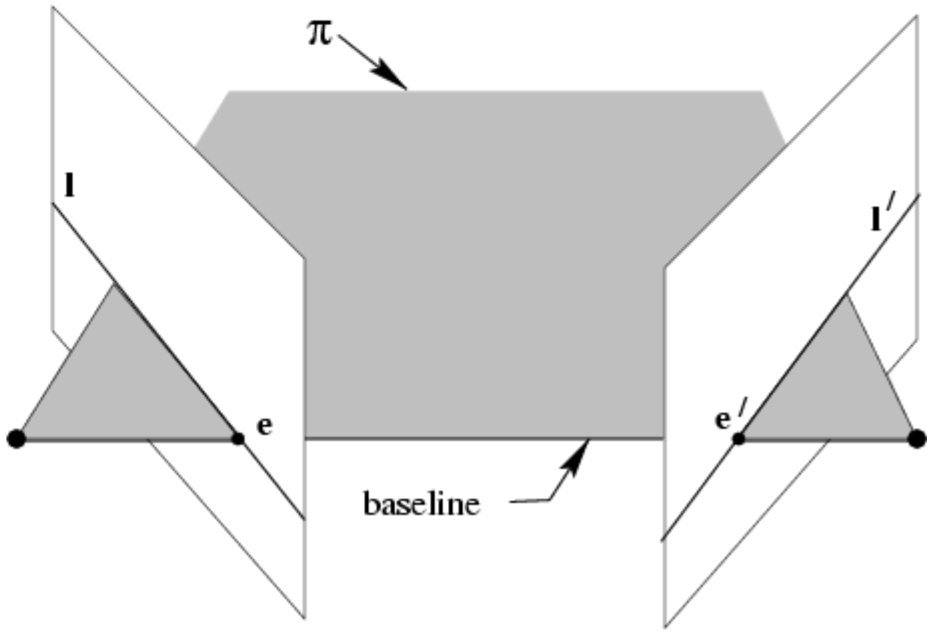
Epipoles and epipolar lines

epipoles e,e’
= intersection of baseline with image plane
= projection of projection center in other image
= vanishing point of camera motion direction



an epipolar plane = plane containing baseline (1-D family)

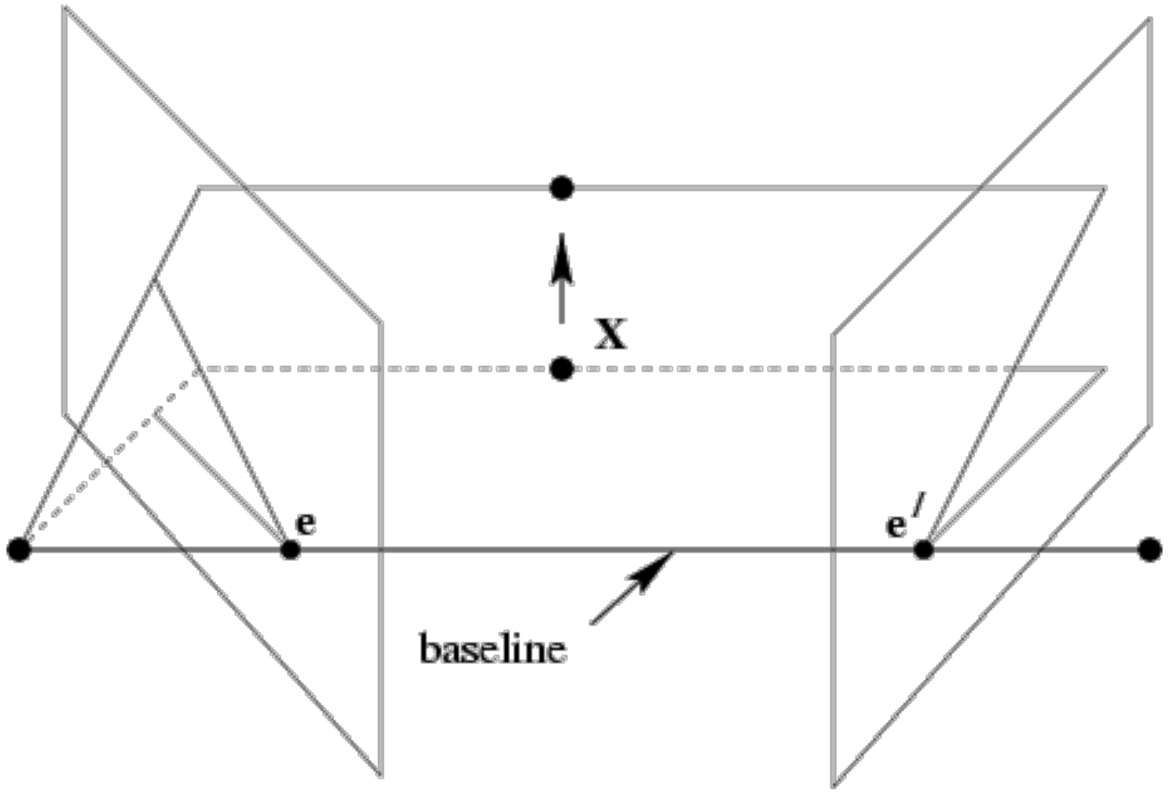
an epipolar line = intersection of epipolar plane with image
(always come in corresponding pairs)



an epipolar plane = plane containing baseline (1-D family)

an **epipolar line** = intersection of epipolar plane with image plane
(always come in corresponding pairs), goes through epipole

→ 1-D family of coplanar, concurrent lines
→ projectively related (same cross ratio in the two images)



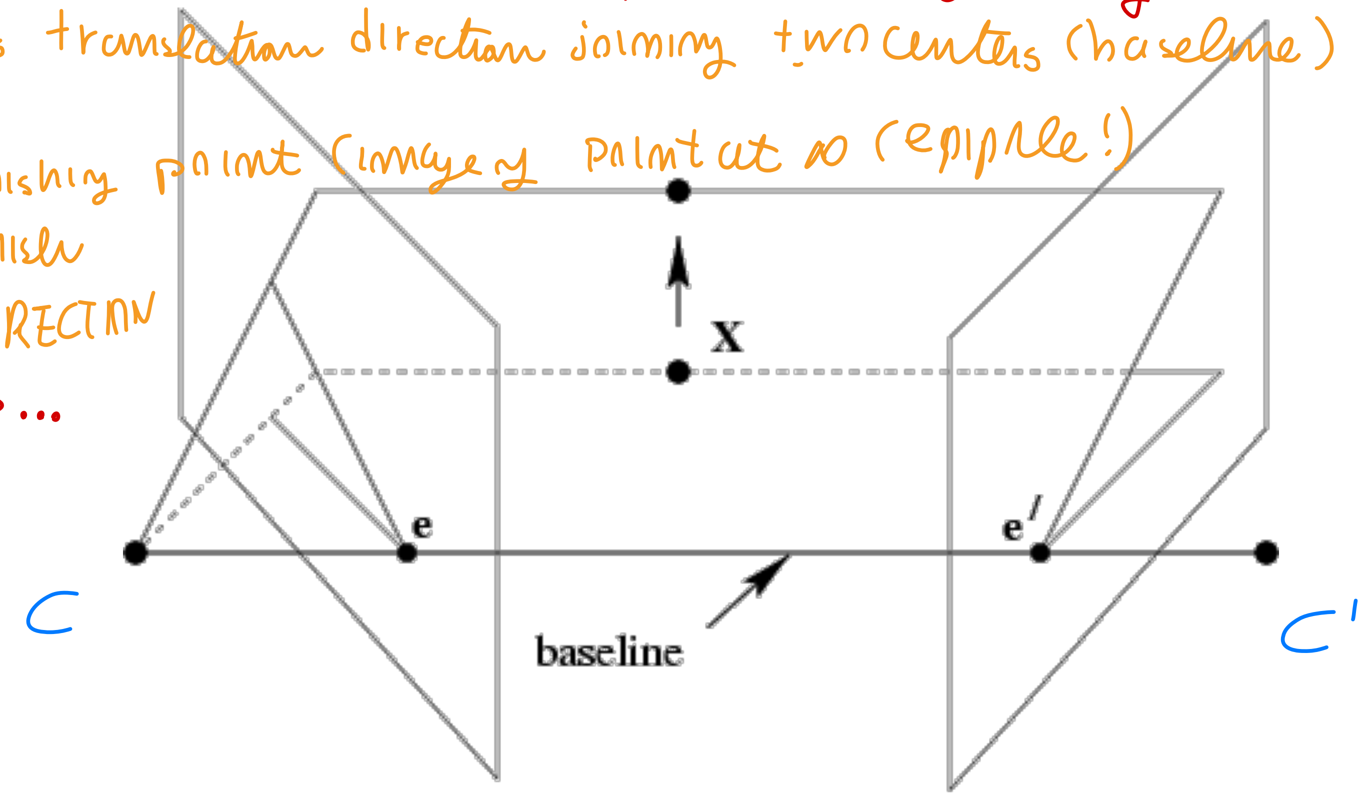
- SUMMARIZE:

→ **epipolar lines**: 1-D family of coplanar, concurrent lines

→ projectively related (same cross ratio in the two images = 3D cross ratio of the epipolar planes)

→ concur at the **epipoles**

IF we are moving the camera. (NOT 2 cameras, SAME camera moving!)
relative motion has translation direction joining two centers (baseline)
the motion
direction has vanishing point (image of point at ∞ (epipole!)
epipole are also vanish
point of MOTION DIRECTION
(*) => ...



- **epipole** e,e' additional property (4TH)
  = intersection of baseline with image plane
  = projection of camera center in other image (other camera)
  = vanishing point of camera relative motion direction
  = concurrency point for epipolar lines

(*) when moving from first to II position, the translation direction connecting camera centers

↑

the image of the point at ∞ X_∞ are the epipoles...
≃ vanishing point

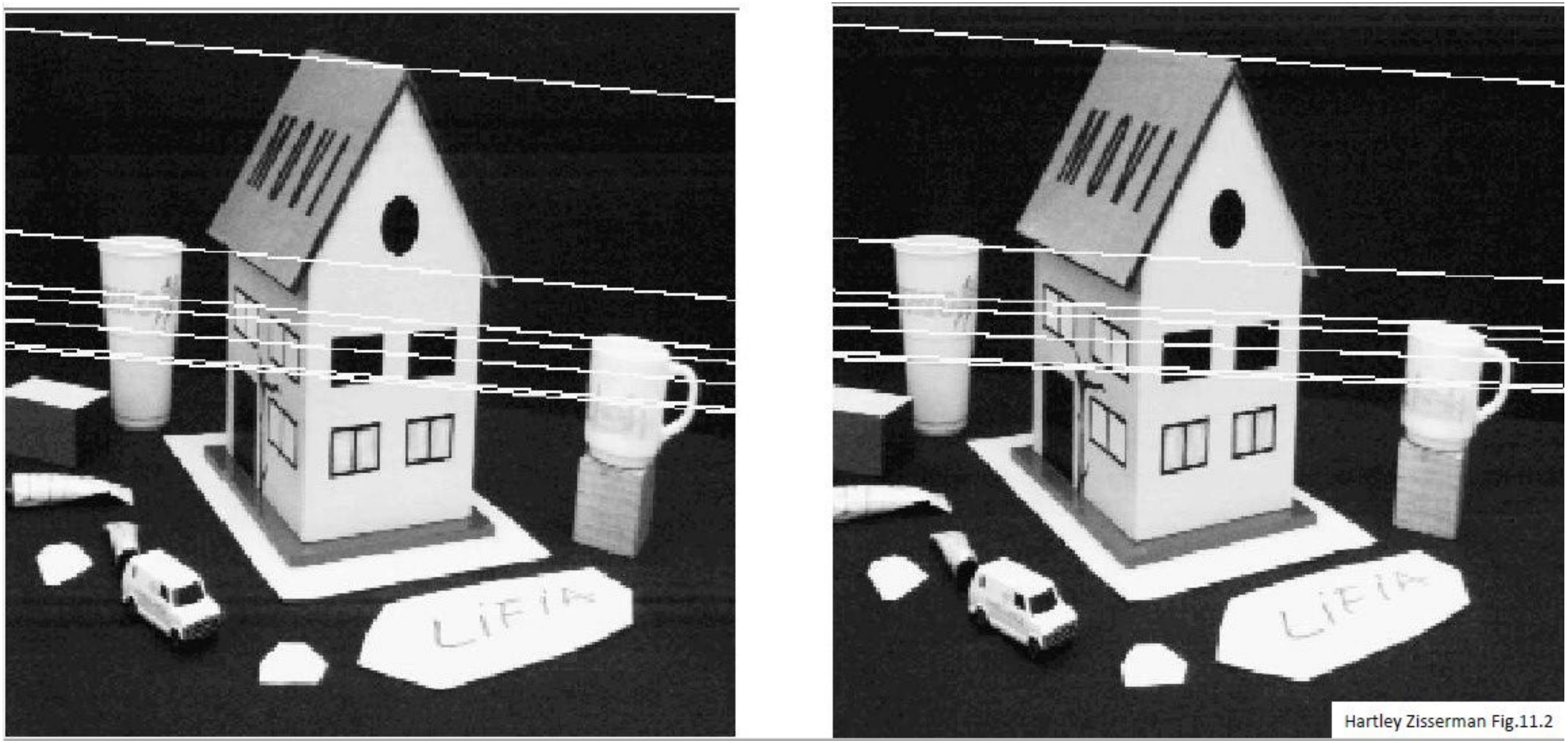
↑

when just translating it is easy to understand

↑

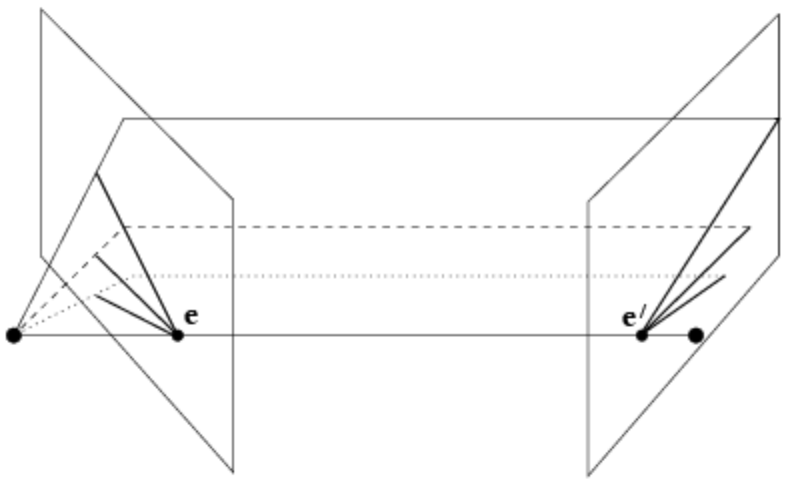
vanishing point is epipole!

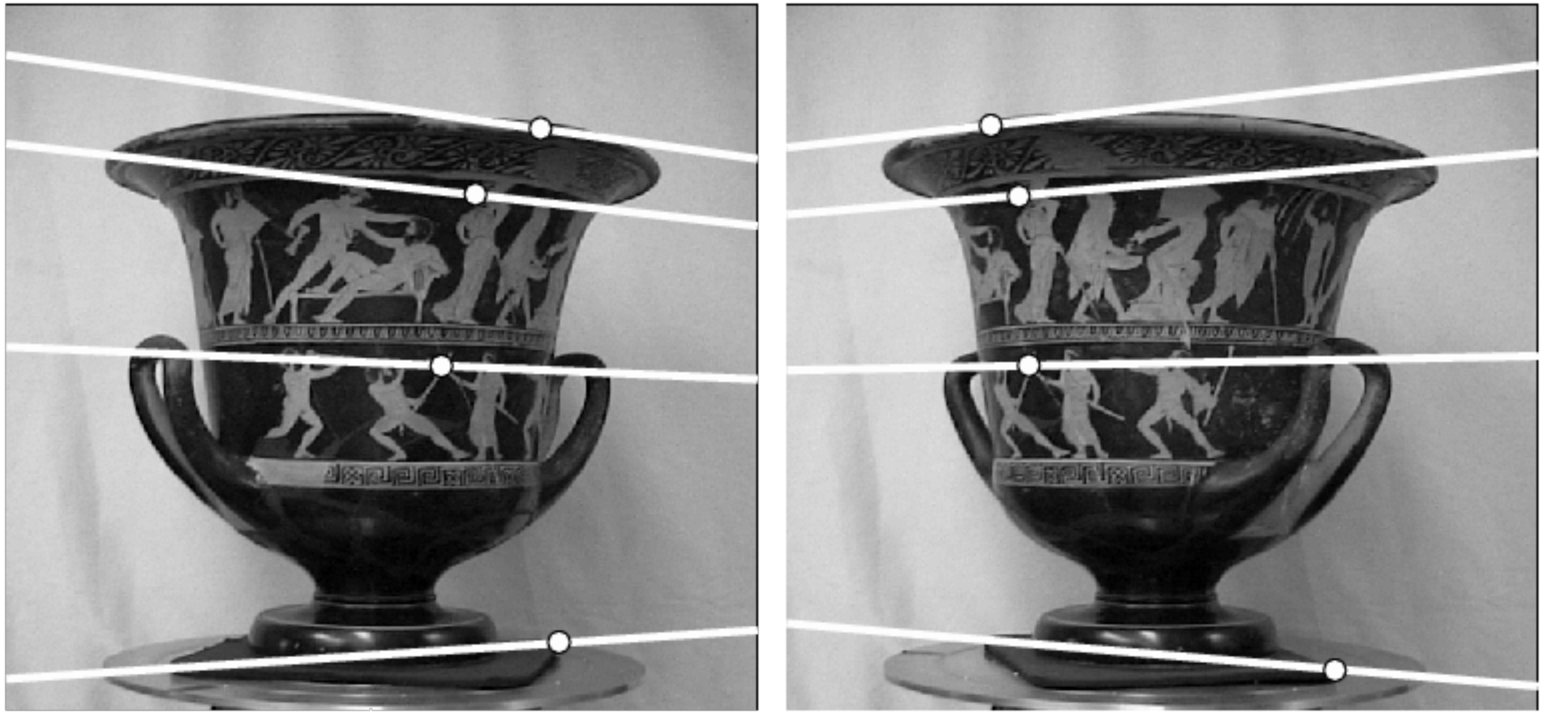
Epipolar lines



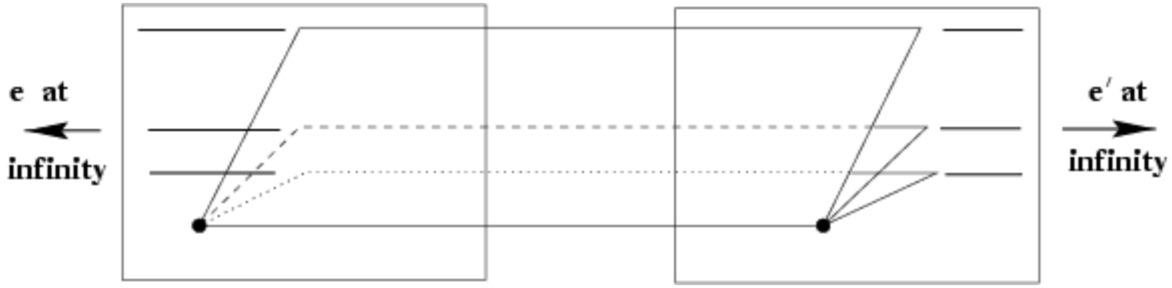
Example: converging cameras

epipolar lines



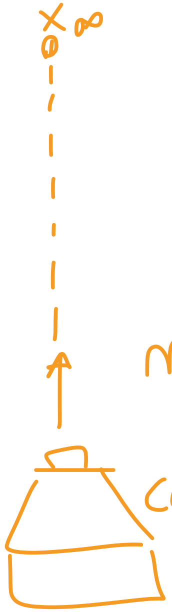


Example: motion parallel to image plane

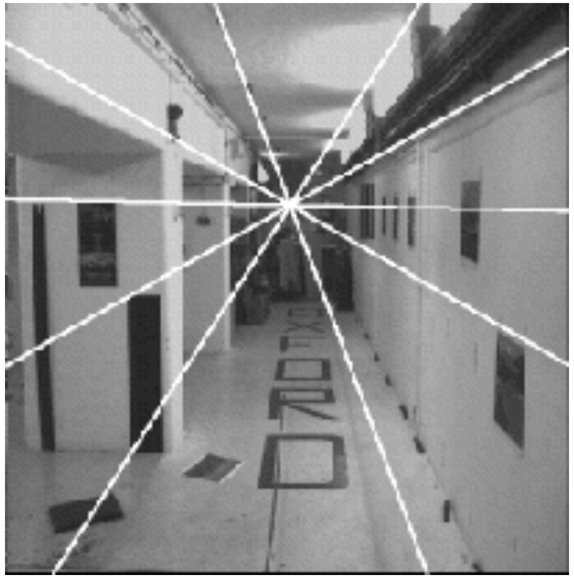


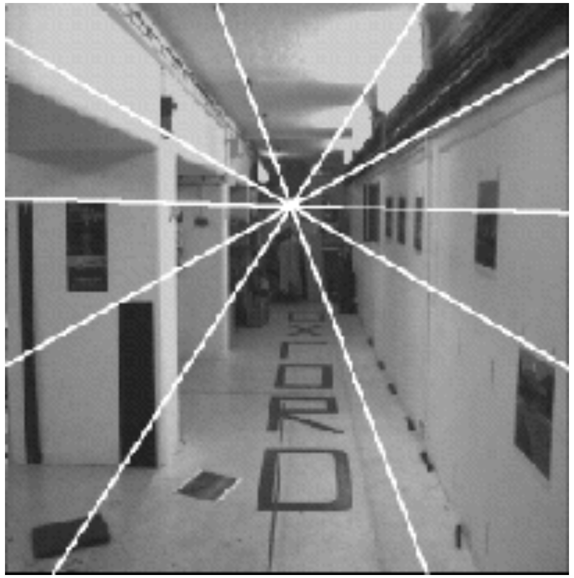
Example: forward motion (translation)

you move along the baseline



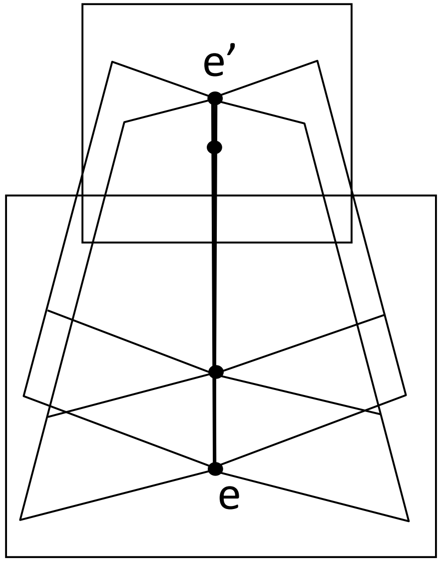
same epipole when just translating





camera moving, just translating no orientation change...

IF camera is just translating, epipoles are images... without orient change the epipoles are the same



pixel coord are the same

image of point at ∞ is same... equal vectors, move on same line, no change of internal camera param

3/12

The fundamental matrix F (Faugeras, 1992)

↳ formalize relationship between an image point in I image and corresponding epipolar line in II image

algebraic representation of epipolar geometry

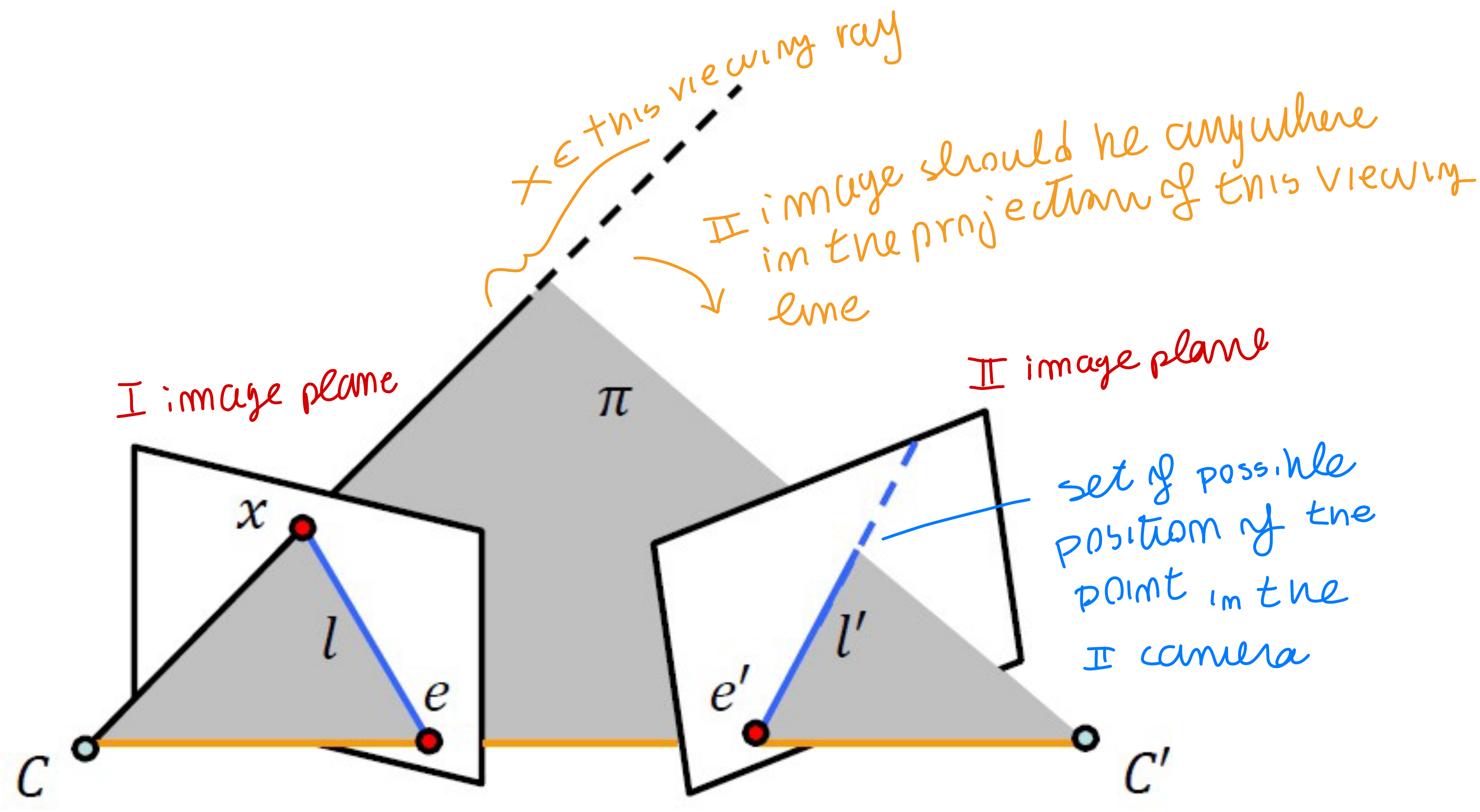
the fact that the point has image, CONSTRAINTS second image somehow ⇒ describe algebraically what happen:

$$\underbrace{x \mapsto l' = Fx}_{\text{CONSTRAINT between two images of same 3D point taken by I, II camera}}$$

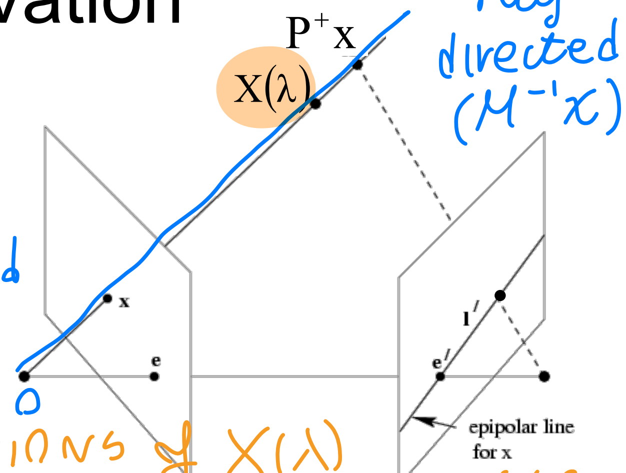
(Foundamental matrix)

we will see that mapping is (singular) correlation (i.e. projective mapping from points to lines) represented by the fundamental matrix F

@ geometric level



The fundamental matrix F derivation



viewing ray directed as $(M^{-1}x)\ \lambda$

Using the equations of camera and 3D geometry:

First image of a point X in 3D $\qquad \mathrm{x} = \mathrm{PX} = [\mathrm{M}|\mathrm{m}]\,\mathrm{X}$

⇓ given x, possible position is the viewing ray of x → all viewing rays should go through center O

Locus of points X: viewing ray through x

$$\mathrm{X} = \underbrace{\mathrm{O}}_{\text{camera center}} + \lambda \begin{bmatrix} \mathrm{M}^{-1}\mathrm{x} \\ 0 \end{bmatrix}$$

→ as linear combination

LOCUS OF POSITIONS of $X(\lambda)$ given x image

Project locus of X onto second camera P′

$$\mathrm{x}' = \mathrm{P}'\mathrm{X} = \mathrm{P}'\mathrm{O} + \lambda[\mathrm{M}'|\mathrm{m}'] \begin{bmatrix} \mathrm{M}^{-1}\mathrm{x} \\ 0 \end{bmatrix} = \mathrm{e}' + \lambda \mathrm{M}'\mathrm{M}^{-1}\mathrm{x}$$

l' is the image of the viewing ray through x

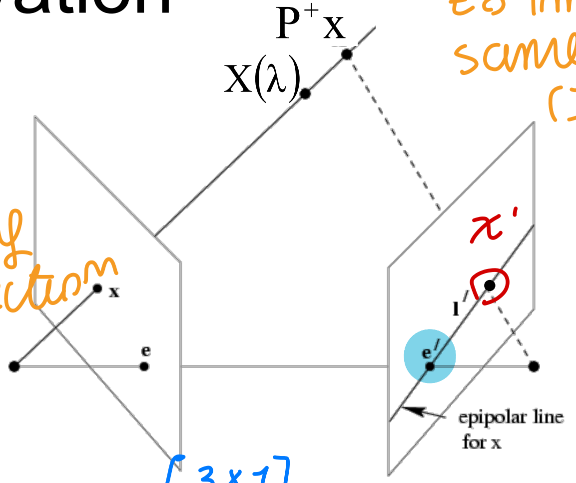
$$\mathrm{x}' \in l' = \mathrm{e}' \times \mathrm{M}'\mathrm{M}^{-1}\mathrm{x} = [\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1}\mathrm{x}$$

x′ is the 2nd image of X

$$\mathrm{x}' \in l' \Rightarrow \mathrm{x}'^T l' = 0 \rightarrow \mathrm{x}'^T [\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1}\mathrm{x} = 0$$

$$\boxed{\mathrm{x}'^T F\,\mathrm{x} = 0} \qquad \text{with} \qquad F = [\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1}$$

The fundamental matrix F derivation



this correspond to image of same point x (II image)

First image of a point X $\quad \mathrm{x} = \mathrm{PX} = [\mathrm{M}|\mathrm{m}]\,\mathrm{X}$

Locus of points X: viewing ray through x $\quad \mathrm{X} = \mathrm{O} + \lambda \begin{bmatrix} \mathrm{M}^{-1}\mathrm{x} \\ 0 \end{bmatrix}$

point @ infinity of the direction of view ray

x' is projection of X into II camera (by P')

[3x1]

Project locus of X onto second camera P′ $\quad \mathrm{x}' = \mathrm{P}'\mathrm{X} = \mathrm{P}'\mathrm{O} + \lambda[\mathrm{M}'|\mathrm{m}'] \begin{bmatrix} \mathrm{M}^{-1}\mathrm{x} \\ 0 \end{bmatrix} = \mathrm{e}' + \boxed{\lambda \mathrm{M}'\mathrm{M}^{-1}\mathrm{x}}$

the image of I camera center to the II image plane, we get the epipole e'

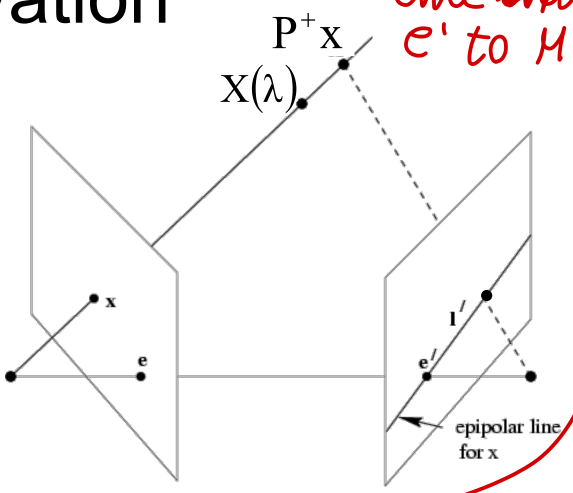
l' is the image of the viewing ray through x $\quad \mathrm{x}' \in l' = \mathrm{e}' \times \mathrm{M}'\mathrm{M}^{-1}\mathrm{x} = [\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1}\mathrm{x}$

x′ is the 2nd image of X $\quad \mathrm{x}' \in l' \Rightarrow \mathrm{x}'^T l' = 0 \rightarrow \mathrm{x}'^T [\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1}\mathrm{x} = 0$

$$\boxed{\mathrm{x}'^T F\,\mathrm{x} = 0} \quad \text{with} \quad F = [\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1}$$

The fundamental matrix F derivation

linear comb... x'
set of position is a
line on
e' to $M'M^{-1}x$



First image of a point X $\quad \mathrm{x} = \mathrm{PX} = [\mathrm{M}|\mathrm{m}]\,\mathrm{X}$

Locus of points X:
viewing ray through x $\quad \mathrm{X} = \mathrm{O} + \lambda \begin{bmatrix} \mathrm{M}^{-1}\mathrm{x} \\ 0 \end{bmatrix}$

Project locus of X onto
second camera P′

$$\mathrm{x}' = \mathrm{P}'\mathrm{X} = \mathrm{P}'\mathrm{O} + \lambda[\mathrm{M}'|\mathrm{m}'] \begin{bmatrix} \mathrm{M}^{-1}\mathrm{x} \\ 0 \end{bmatrix} = \underbrace{\mathrm{e}'}_{\text{NOT dependent on x}} + \underbrace{\lambda \mathrm{M}'\mathrm{M}^{-1}\mathrm{x}}_{\text{point which depends on the I image}}$$

as CROSS PRODUCT formulation

l' is the image of the
viewing ray through x

$$\mathrm{x}' \in l' = \underbrace{\mathrm{e}' \times \mathrm{M}'\mathrm{M}^{-1}\mathrm{x}}_{\text{vector product is linear on components}} = [\mathrm{e}']_{\times}\mathrm{M}'\mathrm{M}^{-1}\mathrm{x}$$

MATRIX for vector product

the components are linear in e' part

x′ is the 2nd image of X $\quad \mathrm{x}' \in l' \Rightarrow \mathrm{x}'^{T} l' = 0 \rightarrow \mathrm{x}'^{T}[\mathrm{e}']_{\times}\mathrm{M}'\mathrm{M}^{-1}\mathrm{x} = 0$

$$\boxed{\mathrm{x}'^{T} F\, \mathrm{x} = 0} \quad \text{with} \quad F = [\mathrm{e}']_{\times}\mathrm{M}'\mathrm{M}^{-1}$$

$$x' \in l' = F\,x = [e']_\times M' M^{-1} x$$

where

constructed to compute
CROSS PRODUCT

$$[e']_\times = \begin{bmatrix} 0 & -e'_z & e'_y \\ e'_z & 0 & -e'_x \\ -e'_y & e'_x & 0 \end{bmatrix}$$

is a skew-symmetric matrix used to compute a cross-product by means of matrix multiplication

$[e']_\times$ is singular

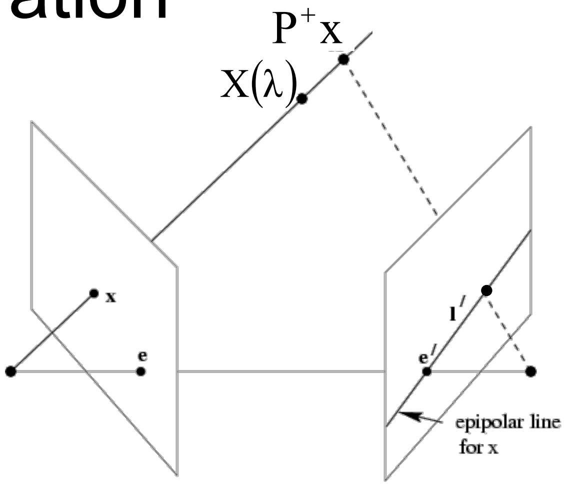
simple way to compute vector product as MATRIX multiplication

$$\boldsymbol{a} \times \boldsymbol{b} = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = ([a]_\times b)$$

Thus the matrix associated to the cross product against $\boldsymbol{e}'$ is

$$[\boldsymbol{e}']_\times = \begin{bmatrix} 0 & -e'_z & e'_y \\ e'_z & 0 & -e'_x \\ -e'_y & e'_x & 0 \end{bmatrix}$$

The fundamental matrix F derivation



First image of a point X $\qquad \mathrm{x} = \mathrm{PX} = [\mathrm{M}|\mathrm{m}]\,\mathrm{X}$

Locus of points X:
viewing ray through x $\qquad \mathrm{X} = \mathrm{O} + \lambda \begin{bmatrix} \mathrm{M}^{-1}\mathrm{x} \\ 0 \end{bmatrix}$

Project locus of X onto
second camera P′ $\qquad \mathrm{x}' = \mathrm{P}'\mathrm{X} = \mathrm{P}'\mathrm{O} + \lambda[\mathrm{M}'|\mathrm{m}'] \begin{bmatrix} \mathrm{M}^{-1}\mathrm{x} \\ 0 \end{bmatrix} = \mathrm{e}' + \lambda \mathrm{M}'\mathrm{M}^{-1}\mathrm{x}$

lines containing all possible set of images of x in II image plane

MATRIX of vector product

l' is the image of the
viewing ray through x $\qquad \mathrm{x}' \in l' = \mathrm{e}' \times \mathrm{M}'\mathrm{M}^{-1}\mathrm{x} = [\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1}\mathrm{x}$

means that: (constraint of x∈l)

x′ is the 2nd image of X $\qquad \mathrm{x}' \in l' \Rightarrow \mathrm{x}'^T l' = 0 \rightarrow \mathrm{x}'^T \underbrace{[\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1}}_{F}\mathrm{x} = 0$

$$\boxed{\mathrm{x}'^T F\,\mathrm{x} = 0}$$ with $$F = [\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1}$$

Foundamental matrix

The fundamental matrix $F = [e']_\times M' M^{-1}$

correspondence condition

two points are CORRESPONDING IFF this $x'^T F x = 0$ is satisfyed

The fundamental matrix satisfies the condition that for any pair of corresponding points $x \leftrightarrow x'$ in the two images $x'^T F x = 0$ $\left(x'^T \, l' = 0 \right)$

bi-linear equation in each of the two image point

as algebra constr

ALGEBRAIC FORMULATION of epipolar constraint (that we saw at geometric level)

two viewing rays must intersect

even if cameras are not calibrated, i.e., P and P' are unknown, the fundamental matrix F can still be computed by imposing correspondence conditions: $x'^T F x = 0$

When we KNOW both cameras, it can be explicitly computed by formula

even without KNOWING camera, we can rely on images!

→ even when P, P' unkn..

IF we have enough pairs of corresponding points

⇓

we can try to select pairs of corresponding image points

↓

in each pair $x \leftrightarrow x'$ of same point X

IF we can extract enough $x_i \leftrightarrow x'_i$ pairs

we can compute F without P KNOWLEDGE

use lots of $x_i \leftrightarrow x'_i$ $i = 1, 2, .., N$ and write lots of equations

$$\boxed{\underline{x'^T_i F x_i = 0}} \quad i = 1, .., N$$

Collect sufficient $i = 1..N$ and we can estimate F!

↪ properties of F ⇒

being product of skewsymm $[e']_\times$

singular $\det([e']_\times) = 0$

then $\det(F) = 0 \cdot ... = 0$

⇓

also F is SINGULAR

The fundamental matrix F: properties

(SINGULAR)

- $F = [e']_\times M' M^{-1}$ is the only 3x3 rank 2 matrix satisfying $x'^T F x = 0$ for all $x \leftrightarrow x'$

this pairs of camera — reverse pair of camera

- **Transpose**: if F is fundamental matrix for (P,P'), then F^T is fundamental matrix for (P',P)

we can compute foundamental matrix of reversed pair ← $x^T F^T x' = 0$ constraint get reversed

- **Epipolar lines:** $l' = F x$ & $l = F^T x'$

Given img point x in I camera, epipolar line e' is easy to find by equations of foundament matrix

- **Epipoles are LNS of F**: on all epipolar lines, thus $e'^T F x = 0$, $\forall x \Rightarrow \boxed{e'^T F = 0}$, similarly $Fe = 0$

LEFT NULL SPACE → IF we consider $= e'$ ⇒ $e'^T \cdot e' = 0$ is point of epipolar line

whatever x we take, it holds $e'^T F = 0$! e' is LNS of F

- **F** has 7 d.o.f. , i.e. 3x3 -1(homogeneous) -1(rank-2 constraint)

- **F** is a correlation, projective mapping from a point x to a line l'=Fx (not a proper correlation, because it is not invertible)

computation of F from pairs of corresponding image points

use $x'_i F x_i = 0$ equations, linear in coeff. F

8 point pairs (linear), 7 point pairs (non-linear), 8+ (least-squares)

The fundamental matrix F: properties

- F $=[e']_{\times}M'M^{-1}$ is the only 3x3 rank 2 matrix satisfying $x'^{T}Fx=0$ for all $x \leftrightarrow x'$

- **Transpose:** if F is fundamental matrix for (P,P'), then F^T is fundamental matrix for (P',P)

you can reverse it by Fe = 0, other epipoles

- **Epipolar lines:** $l' = F\,x$ & $l = F^{T}x'$

- **Epipoles are LNS of F:** on all epipolar lines, thus $e'^{T}Fx=0$, $\forall x \Rightarrow e'^{T}F=0$, similarly $Fe=0$

homogeneous matrix, λF keep same equation

SINGULAR

- **F** has 7 d.o.f. , i.e. 3x3 -1(homogeneous) -1(rank-2 constraint) = 9 - 1 - 1 = 7 dof

as linear constraint, you can use 7 equations in bi-linear form

- **F** is a correlation, projective mapping from a point x to a line l'=Fx (not a proper correlation, because it is not invertible)

computation of F from pairs of corresponding image points

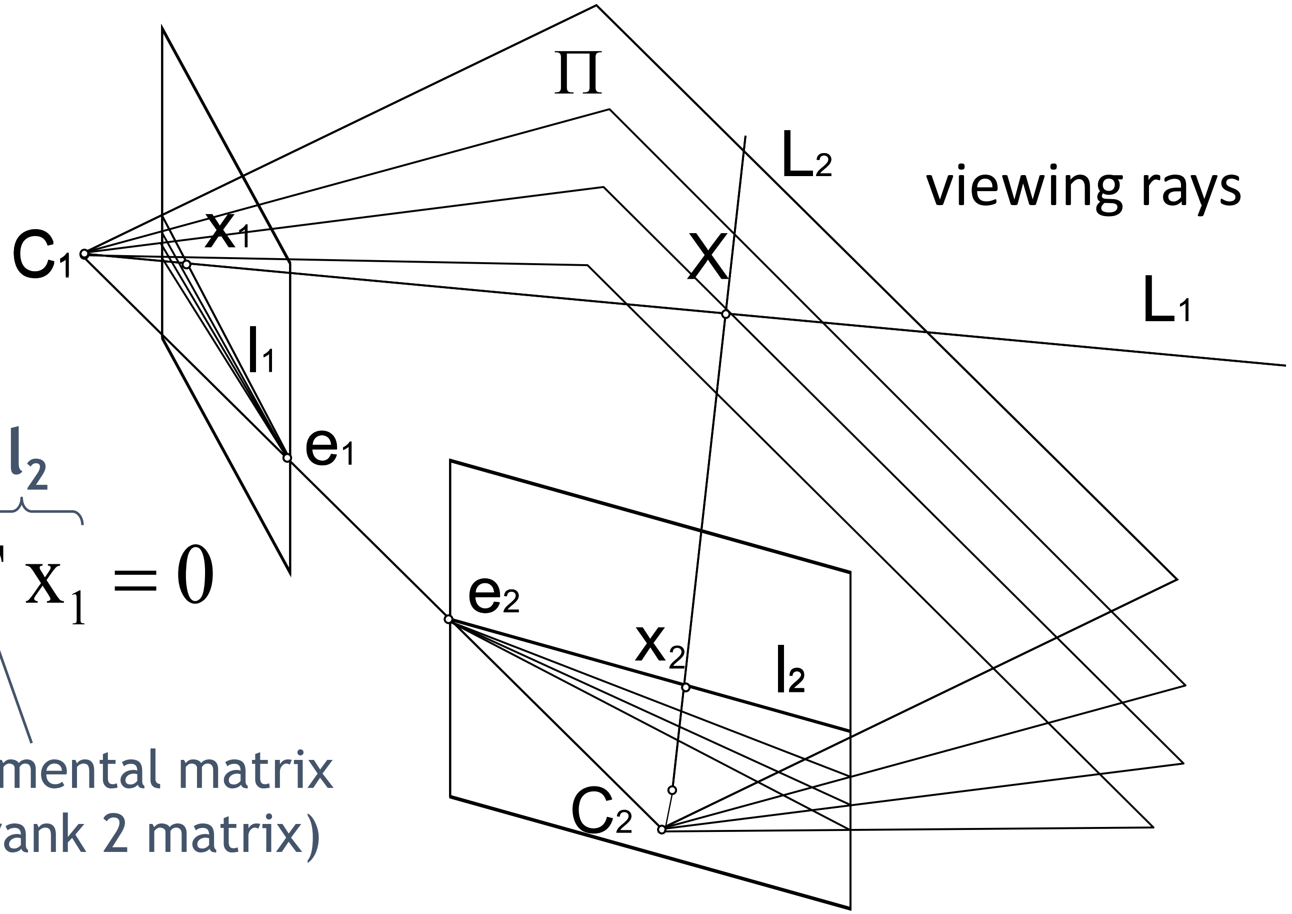
use $x'_i F x_i = 0$ equations, linear in coeff. F

8 point pairs (linear), 7 point pairs (non-linear), 8+ (least-squares)

better estimate reducing numerical error

Epipolar geometry

Underlying structure in set of matches for rigid scenes



$$\underbrace{\mathbf{x}_2^T \overbrace{\mathbf{F}}}_{\mathbf{l}^T_1} \; \overbrace{\mathbf{x}_1}^{} = 0$$

$\mathbf{l}^T_1$ $\mathbf{l}_2$

Fundamental matrix (3x3 rank 2 matrix)

1. Computable from corresponding points
2. Simplifies matching
3. Allows to detect wrong matches
4. Related to calibration

Canonical representation: (see later)

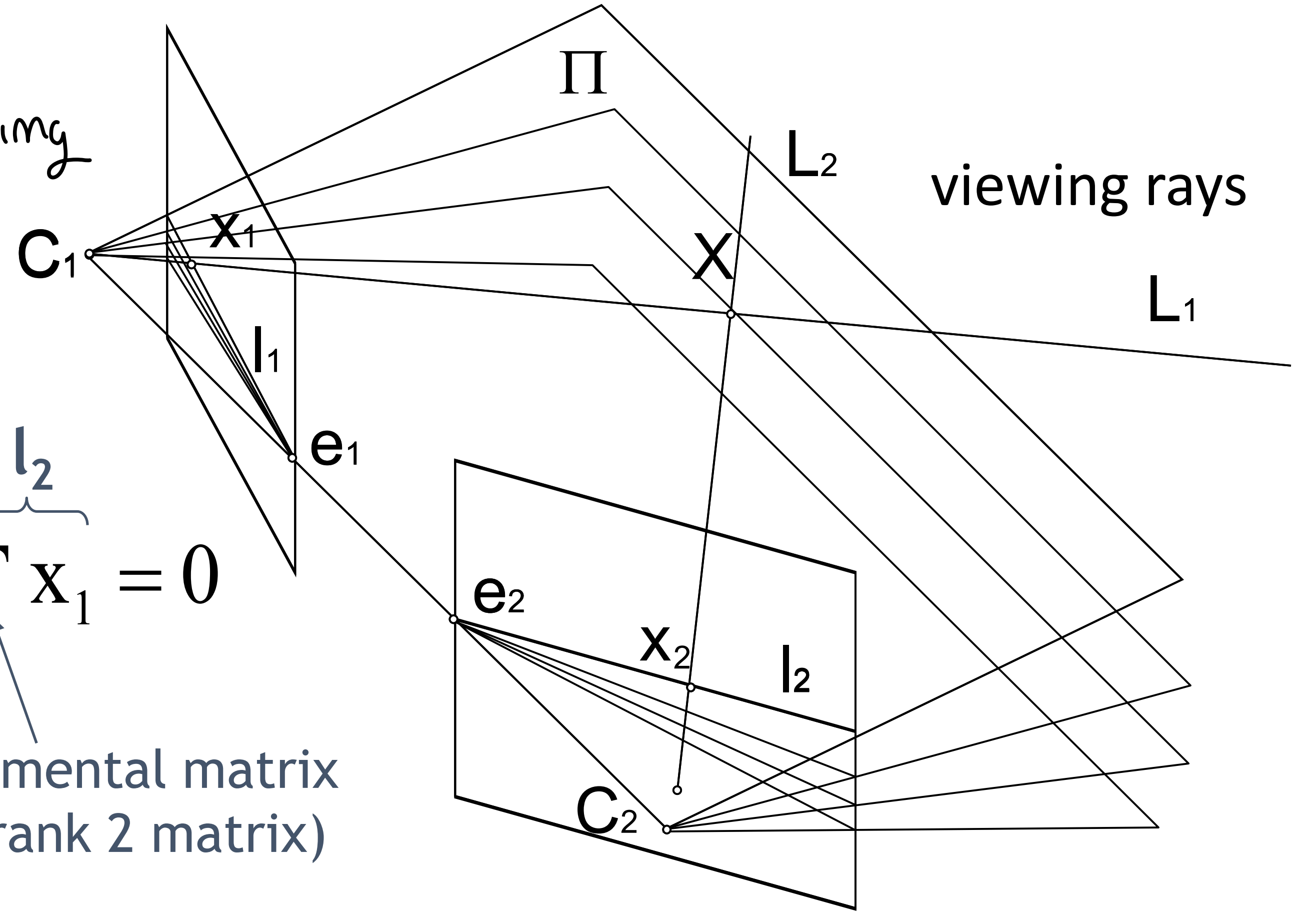
$$P = [I \,|\, 0] \quad P' = [[e']_\times F + e' v^T \,|\, \lambda e']$$

in SUMMARY...

letting epipolar plane vary, you identify 2D CROSS RATIO! ⇒ projectively corresponding

Epipolar geometry

Underlying structure in set of matches for rigid scenes



$$\underbrace{x_2^T}_{l^T_1}\underbrace{\mathbf{F}\, x_1}_{l_2} = 0$$

Fundamental matrix (3x3 rank 2 matrix)

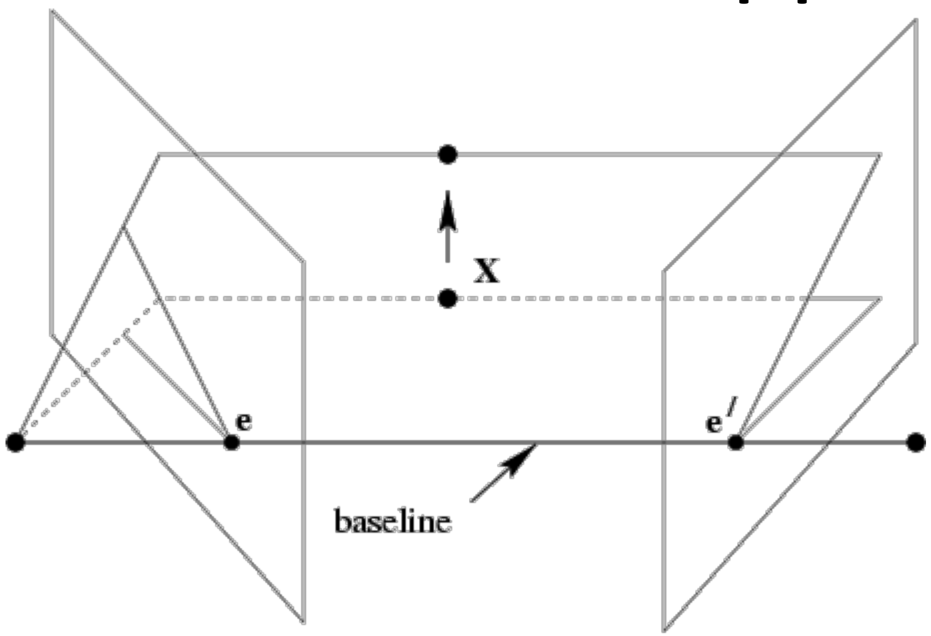
Epipolar planes : 3D cross ratio

Epipolar lines : same 2D cross ratio on both images

→ projectively related (1D)

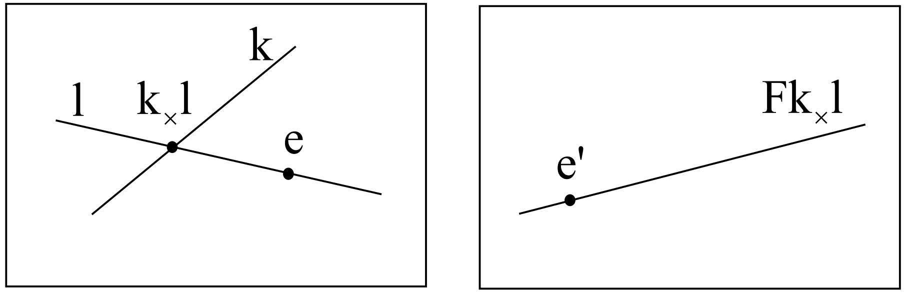
1. Computable from corresponding points
2. Simplifies matching
3. Allows to detect wrong matches
4. Related to calibration

epipolar lines are projectively related: algebraic proof



l epipolar lines through e

take a line k not through e : e.g., take k = e



Consider the point $\mathrm{x} = k \cap l = \mathrm{e} \times l = [\mathrm{e}]_{\times}\, l$

Then $l' = F\mathrm{x} = F[\mathrm{e}]_{\times} l = Hl$ with $H = F[\mathrm{e}]_{\times}$

Notice that rank H = 2 < 3. this is OK since
the relation between epipolar lines is 1D projective

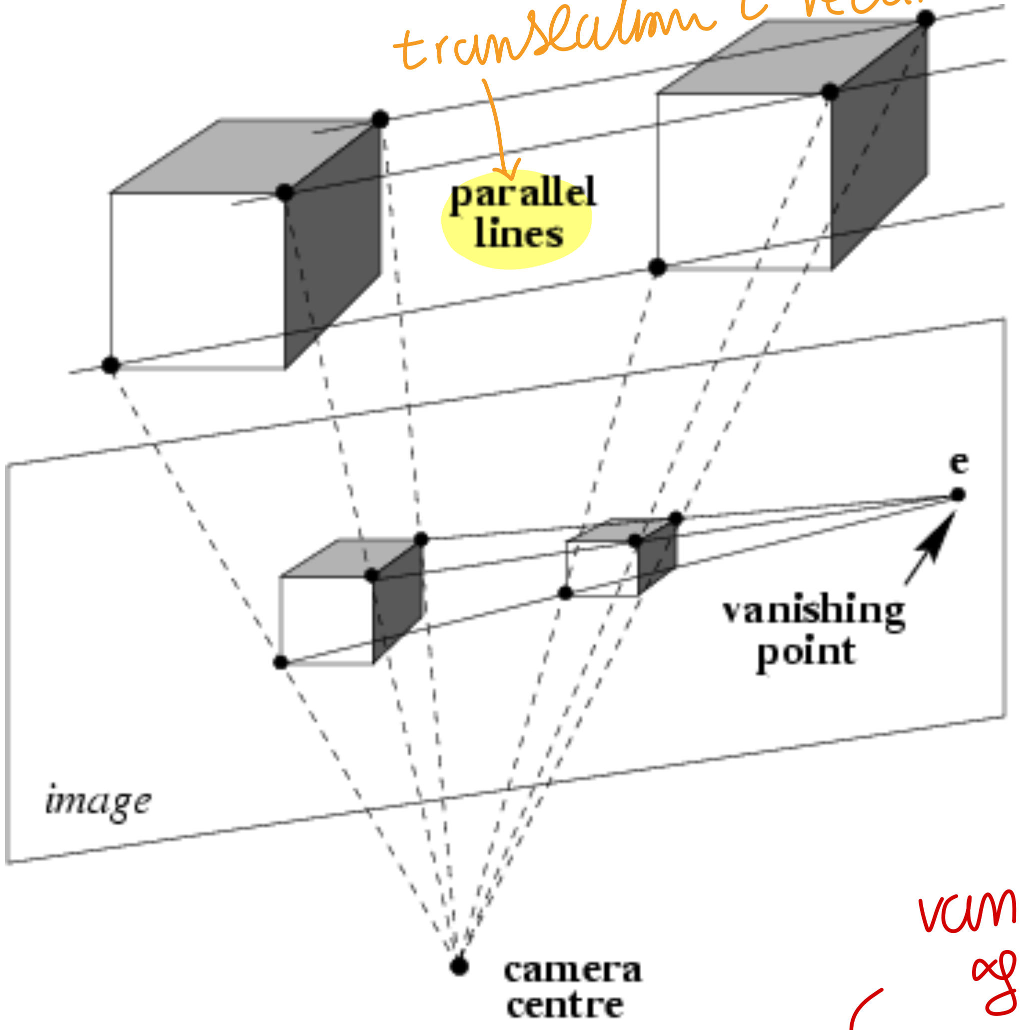
particular case

Fundamental matrix for pure translation (NOT ROTATION)

navigating along rectilinear path

II camera as just translation of the I camera

translation t vector



3D direction can be defined as

$K^{-1}e$

$R = I_3$ identity

$M = [K R] = K$

$e = Kt$ relationship

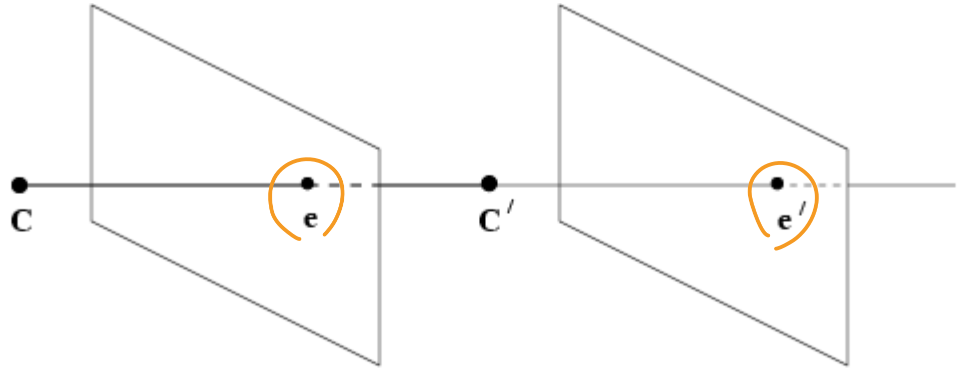
vanishing point of translation direction

Viewing ray of epipole is parallel to the translation direction

$$K^{-1}e = \mathbf{t} \rightarrow \boxed{e = K\mathbf{t}}$$

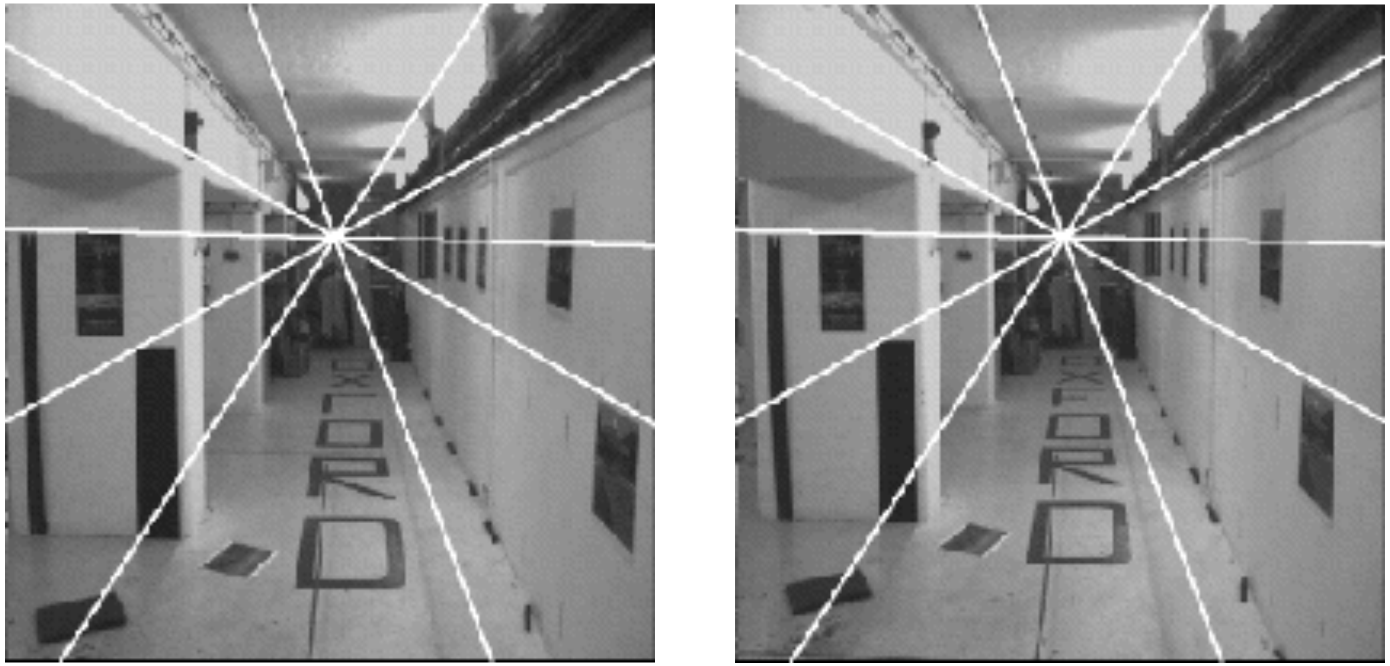
Fundamental matrix for pure translation

the two epipoles are equal



when just translating, epipole remain the same!

concurrency point is the epipole...



Fundamental matrix for pure translation

$$F = [e']_\times M' M^{-1} = [e']_\times$$

F only 2 d.o.f., $x^T[e]_x x=0 \Rightarrow$ auto-epipolar

example: $e' = (1,0,0)^T \quad F = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}_\times \qquad x'^T F x = 0 \Leftrightarrow y = y'$

$$x = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = PX = K[I|0]X = K\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} u \\ v \\ Z \end{bmatrix} = Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Z x_{cart} \rightarrow \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = ZK^{-1}x_{cart}$$

$$x' = P'X = K[I|t]\begin{bmatrix} ZK^{-1}x_{cart} \\ 1 \end{bmatrix} = K[I|t]\begin{bmatrix} K^{-1}x_{cart} \\ 1/Z \end{bmatrix} = x_{cart} + Kt/Z$$

$$x' = x_{cart} + Kt/Z \text{ this } x' = x'_{cart} \text{ if } t_z = 0$$

motion starts at x and moves towards (or away from) e, faster depending on *Z*

Fundamental matrix for pure translation

$$F = [e']_\times M' M^{-1} = [e']_\times$$

F only 2 d.o.f., $x^T[e]_x x=0 \Rightarrow$ auto-epipolar

<u>example</u>: $\mathbf{e}' = (1,0,0)^\top \quad \mathbf{F} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}_\times \quad \mathbf{x}'^\top \mathbf{F}\mathbf{x} = 0 \Leftrightarrow y = y'$

see next slide

$$x = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = PX = K[I|0]X = \mathbf{K}\begin{bmatrix} \boldsymbol{X} \\ \boldsymbol{Y} \\ \boldsymbol{Z} \end{bmatrix} = \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \\ \boldsymbol{Z} \end{bmatrix} = Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Z\mathrm{x}_{cart} \rightarrow \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = ZK^{-1}\mathrm{x}_{cart}$$

$$x' = P'X = K[I|t]\begin{bmatrix} ZK^{-1}\mathrm{x}_{cart} \\ 1 \end{bmatrix} = K[I|t]\begin{bmatrix} K^{-1}\mathrm{x}_{cart} \\ 1/Z \end{bmatrix} = \mathrm{x}_{cart} + Kt/Z$$

$$x' = \mathrm{x}_{cart} + Kt/Z \quad \text{this } x' = x'_{cart} \text{ if } t_z = 0$$

motion starts at x and moves towards (or away from) e, faster depending on *Z*

why is $\mathrm{K}\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} u \\ v \\ Z \end{bmatrix}$?

$$\mathrm{K}\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} f_x & (s) & U_o \\ 0 & f_y & V_o \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} * \\ * \\ Z \end{bmatrix}$$

Fundamental matrix for pure translation

$$\mathrm{F} = [\mathrm{e}']_\times \mathrm{M}'\mathrm{M}^{-1} = [\mathrm{e}']_\times$$

F only 2 d.o.f., $x^T[e]_x x=0 \Rightarrow$ auto-epipolar

<u>example</u>: $\mathbf{e}' = (1,0,0)^{\mathrm{T}}$ $\quad \mathbf{F} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}_\times$ $\quad \mathbf{x}'^{\mathrm{T}}\mathbf{F}\mathbf{x} = 0 \Leftrightarrow y = y'$

$$\mathrm{x} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathrm{PX} = \mathrm{K}[\mathrm{I}|0]\mathrm{X} = \mathrm{K}\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} u \\ v \\ Z \end{bmatrix} = Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Z\mathbf{x}_{\boldsymbol{cart}} \rightarrow \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z\mathrm{K}^{-1}\mathrm{x}_{cart}$$

$$\mathrm{x}' = \mathrm{P}'\mathrm{X} = \mathrm{K}[\mathrm{I}|\mathrm{t}]\begin{bmatrix} ZK^{-1}\mathrm{x}_{cart} \\ 1 \end{bmatrix} = \mathrm{K}[\mathrm{I}|\mathrm{t}]\begin{bmatrix} K^{-1}\mathrm{x}_{cart} \\ 1/Z \end{bmatrix} = \mathrm{x}_{cart} + \mathrm{K}t/Z$$

$\mathrm{x}' = \mathrm{x}_{cart} + \mathrm{Kt}/Z$ this $\mathrm{x}' = \mathrm{x}'_{cart}$ if $\mathrm{t}_z = 0$

motion starts at x and moves towards (or away from) e, faster depending on *Z*

Fundamental matrix for pure translation

$$F = [e']_\times M' M^{-1} = [e']_\times$$

F only 2 d.o.f., $x^T[e]_x x=0 \Rightarrow$ auto-epipolar

example: $\mathbf{e}' = (1,0,0)^T \quad \mathbf{F} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}_\times \quad \mathbf{x}'^T \mathbf{F}\mathbf{x} = 0 \Leftrightarrow y = y'$

see next slide

$$x = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = PX = K[I|0]X = K \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} u \\ v \\ Z \end{bmatrix} = Z \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Z\mathbf{x}_{cart} \rightarrow \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = ZK^{-1}x_{cart}$$

$$x' = P'X = K[I|t] \begin{bmatrix} ZK^{-1}x_{cart} \\ 1 \end{bmatrix} = K[I|t] \begin{bmatrix} K^{-1}x_{cart} \\ 1/Z \end{bmatrix} = x_{cart} + Kt/Z$$

$x' = x_{cart} + Kt/Z$ this $x' = x'_{cart}$ if $t_z = 0$

motion starts at x and moves towards (or away from) e, faster depending on *Z*

In $\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$, x and y are the cartesian coordinates of the image point x, therefore we indicate

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \text{ as } \mathrm{x}_{cart}$$

Fundamental matrix for pure translation

$$F = [e']_\times M' M^{-1} = [e']_\times$$

F only 2 d.o.f., $x^T[e]_x x=0 \Rightarrow$ auto-epipolar

<u>example</u>: $\mathbf{e}' = (1,0,0)^T \quad \mathbf{F} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}_\times \quad \mathbf{x}'^T \mathbf{F}\mathbf{x} = 0 \Leftrightarrow y = y'$

$$x = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = PX = K[I|0]X = K \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} u \\ v \\ Z \end{bmatrix} = Z \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Z x_{cart} \rightarrow \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z K^{-1} x_{cart}$$

$$x' = P'X = \mathbf{K}[\mathbf{I}|\mathbf{t}] \begin{bmatrix} Z K^{-1} x_{cart} \\ 1 \end{bmatrix} = K[I|t] \begin{bmatrix} K^{-1} x_{cart} \\ 1/Z \end{bmatrix} = x_{cart} + Kt/Z$$

$x' = x_{cart} + Kt/Z$ this $x' = x'_{cart}$ if $t_z = 0$

motion starts at x and moves towards (or away from) e, faster depending on *Z*

Fundamental matrix for pure translation

$$F = [e']_\times M' M^{-1} = [e']_\times$$

F only 2 d.o.f., $x^T[e]_x x=0 \Rightarrow$ auto-epipolar

<u>example</u>: $\mathbf{e}' = (1,0,0)^T \quad \mathbf{F} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}_\times \quad \mathbf{x}'^T \mathbf{F}\mathbf{x} = 0 \Leftrightarrow y = y'$

$$\mathrm{x} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathrm{PX} = \mathrm{K}[\mathrm{I}|0]\mathrm{X} = \mathrm{K}\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} u \\ v \\ Z \end{bmatrix} = Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Z\mathrm{x}_{cart} \rightarrow \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z\mathrm{K}^{-1}\mathrm{x}_{cart}$$

see next slide

$$\mathrm{x}' = \mathrm{P}'\mathrm{X} = \mathbf{K}[\mathbf{I}|\mathbf{t}]\begin{bmatrix} Z\mathrm{K}^{-1}\mathrm{x}_{cart} \\ 1 \end{bmatrix} = \mathrm{K}[\mathrm{I}|\mathrm{t}]\begin{bmatrix} \mathrm{K}^{-1}\mathrm{x}_{cart} \\ 1/Z \end{bmatrix} = \mathrm{x}_{cart} + \mathrm{Kt}/Z$$

$\mathrm{x}' = \mathrm{x}_{cart} + \mathrm{Kt}/Z$ this $\mathrm{x}' = \mathrm{x}'_{cart}$ if $\mathrm{t}_z = 0$

motion starts at x and moves towards (or away from) e, faster depending on *Z*

Yet another expression of P

$$P = [\mathrm{M} \quad \mathrm{m}] \quad O = \begin{bmatrix} o \\ 1 \end{bmatrix} = \mathrm{RNS(P)}$$

$$\mathrm{M} = \mathrm{KR}_{cam\to world} \quad o = t_{world\to cam}$$

From

$$PO = [\mathrm{M} \quad \mathrm{m}] \begin{bmatrix} t_{world\to cam} \\ 1 \end{bmatrix} = \mathrm{M}t_{world\to cam} + \mathrm{m} = 0$$

is $m = -\mathrm{M}t_{world\to cam} = -\mathrm{KR}_{cam\to world} t_{world\to cam}$

But since

$$t_{cam\to world} = -\mathrm{R}t_{world\to cam},$$

Then $m = Kt_{cam\to world}$

Hence $P = [\mathrm{M} \quad \mathrm{m}] = [K\mathrm{R}_{cam\to world} \quad Kt_{cam\to world}]$

$$\boxed{P = K[R \quad t]}$$

where $R \stackrel{\text{def}}{=} \mathrm{R}_{cam\to world}$ and $t \stackrel{\text{def}}{=} t_{cam\to world}$

Fundamental matrix for pure translation

$$F = [e']_\times M' M^{-1} = [e']_\times$$

F only 2 d.o.f., $x^T[e]_x x=0 \Rightarrow$ auto-epipolar

<u>example</u>: $\mathbf{e}' = (1,0,0)^T \quad \mathbf{F} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}_\times \quad \mathbf{x}'^T \mathbf{F} \mathbf{x} = 0 \Leftrightarrow y = y'$

$$\mathrm{x} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \mathrm{PX} = \mathrm{K}[\mathrm{I}|0]\mathrm{X} = \mathrm{K}\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} u \\ v \\ Z \end{bmatrix} = Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Z\mathrm{x}_{cart} \rightarrow \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z\mathrm{K}^{-1}\mathrm{x}_{cart}$$

$$\mathrm{x}' = \mathrm{P}'\mathrm{X} = \mathrm{K}[\mathrm{I}|\mathrm{t}]\begin{bmatrix} Z\mathrm{K}^{-1}\mathrm{x}_{cart} \\ 1 \end{bmatrix} = \mathrm{K}[\mathrm{I}|\mathrm{t}]\begin{bmatrix} \mathrm{K}^{-1}\mathrm{x}_{cart} \\ 1/Z \end{bmatrix} = \mathrm{x}_{cart} + \mathrm{K}t/Z$$

$\mathrm{x}' = \mathrm{x}_{cart} + \mathrm{K}t/Z$ this x' is cartesian, i.e., $\mathrm{x}' = \mathrm{x}'_{cart}$, if $\mathrm{t}_z = 0$ see next slide

motion starts at x and moves towards (or away from) e, faster depending on *Z*

$$\mathrm{K}\mathbf{t}/\mathrm{Z} = \mathrm{K}\begin{bmatrix} t_X \\ t_Y \\ t_Z \end{bmatrix}/Z == \begin{bmatrix} f_x & (s) & U_o \\ 0 & f_y & V_o \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} t_X \\ t_Y \\ t_Z \end{bmatrix}/Z = \begin{bmatrix} \because \\ \because \\ t_Z/Z \end{bmatrix}$$

if $t_Z = 0$,

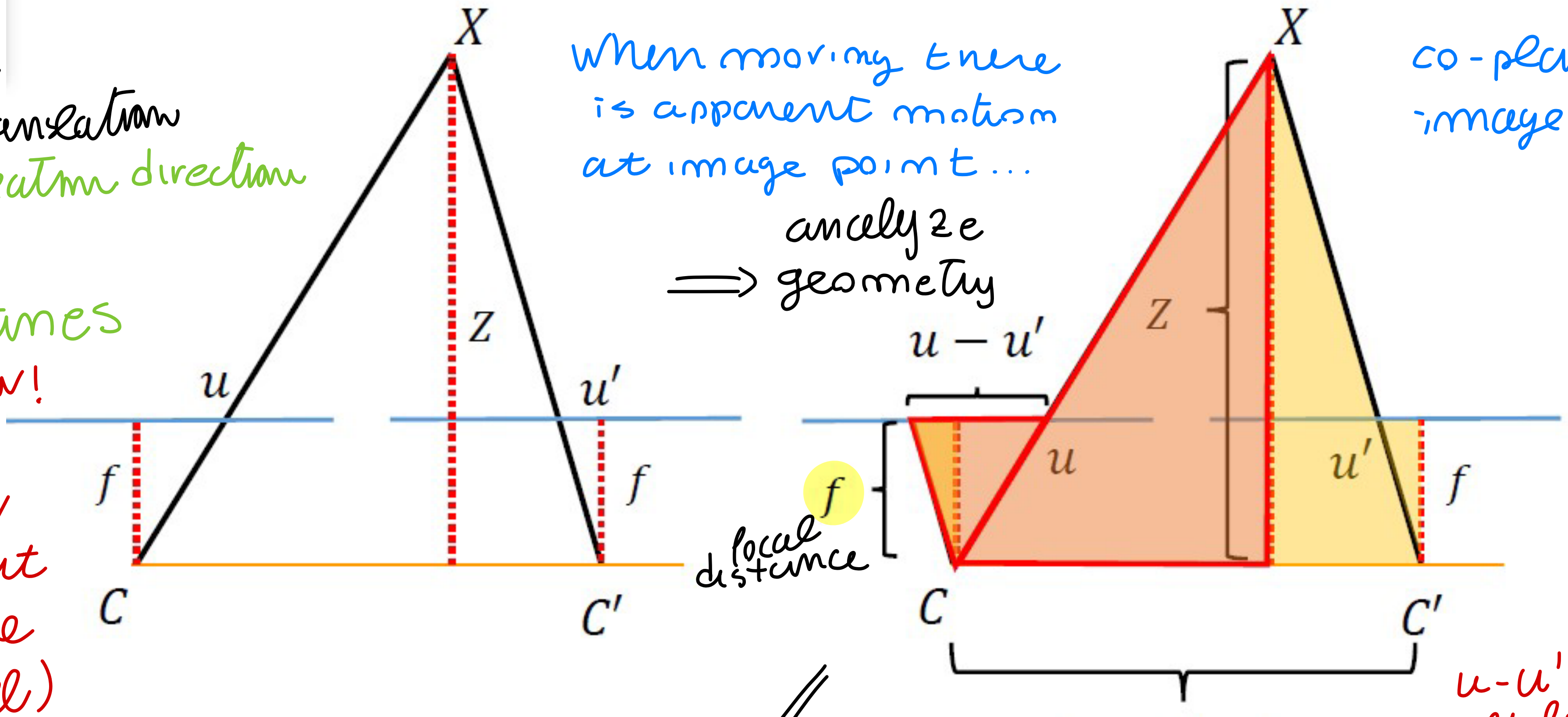
$$\mathrm{x}' = \mathrm{x}_{cart} + \mathrm{K}\mathbf{t}/\mathrm{Z} = \mathrm{x}_{cart} + \begin{bmatrix} * \\ * \\ 0 \end{bmatrix} = \begin{bmatrix} \cdot \\ \cdot \\ 1 \end{bmatrix} + \begin{bmatrix} \because \\ \because \\ 0 \end{bmatrix} = \begin{bmatrix} * \\ * \\ 1 \end{bmatrix} = \mathrm{x}'_{cart}$$

For equal cameras with coplanar image planes ($t_z = 0$) disparity is inversely proportional to the depth

additional effect of translation when translation direction s.t parallel image planes

(NO ROTATION! camera translation, means that img plane are parallel)

if co-planar ~ same plane



when moving there is apparent motion at image point...

analyze ⟹ geometry

co-planar image planes

focal distance

similar triangles → $\frac{u-u'}{f} = \frac{b}{Z}$

u-u' inversely proportional to the vector z of depth when taking two images with moving camera...

$$x' = x_{cart} + Kt/Z$$

the image of closer objects move faster than the image of distant objects

the image of closer objects move faster

than the image of distant objects

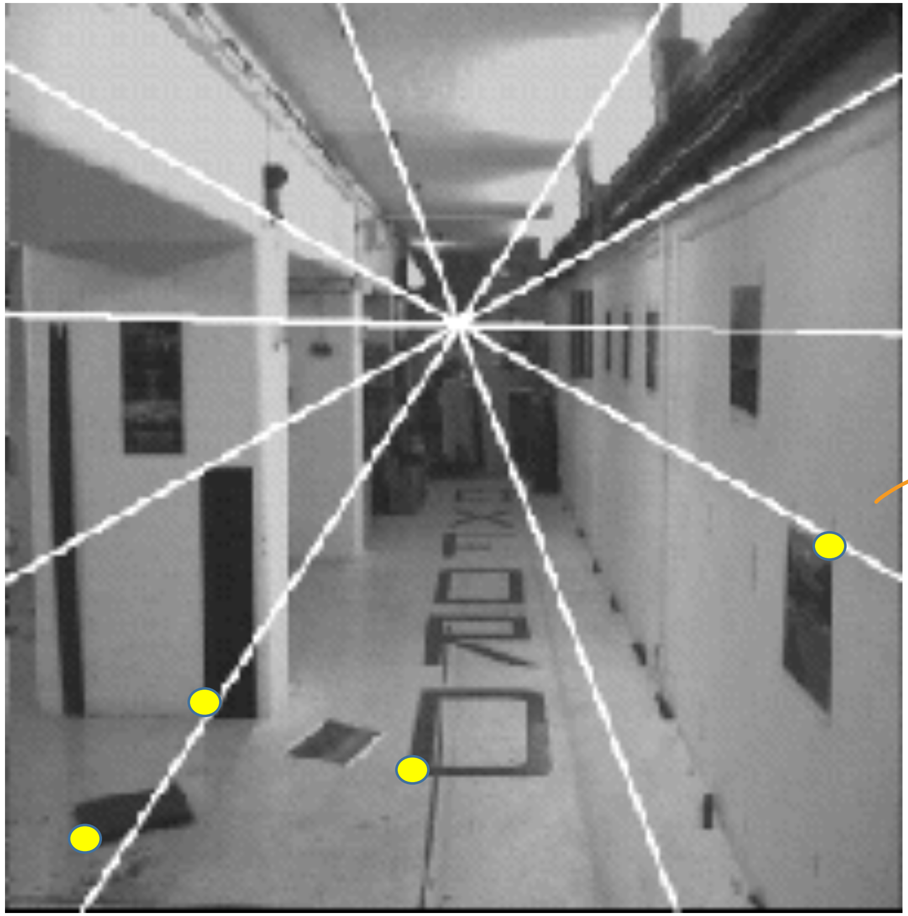
when motion is translation ...

interesting phenomenon

Time to impact

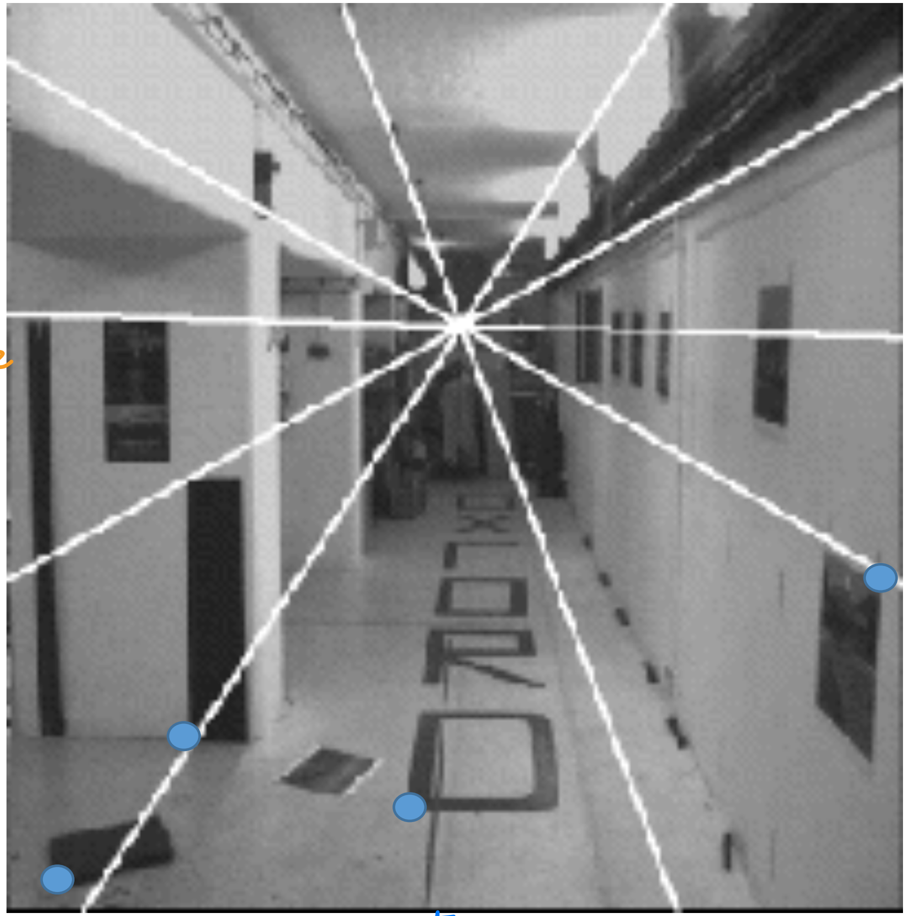
approaching all points by same distance ...

t = 0



we observe feature movement

t = Δt



since points go along lines pure translation ... displacement along line with point@infinity as translation VANISHING point

Time to impact

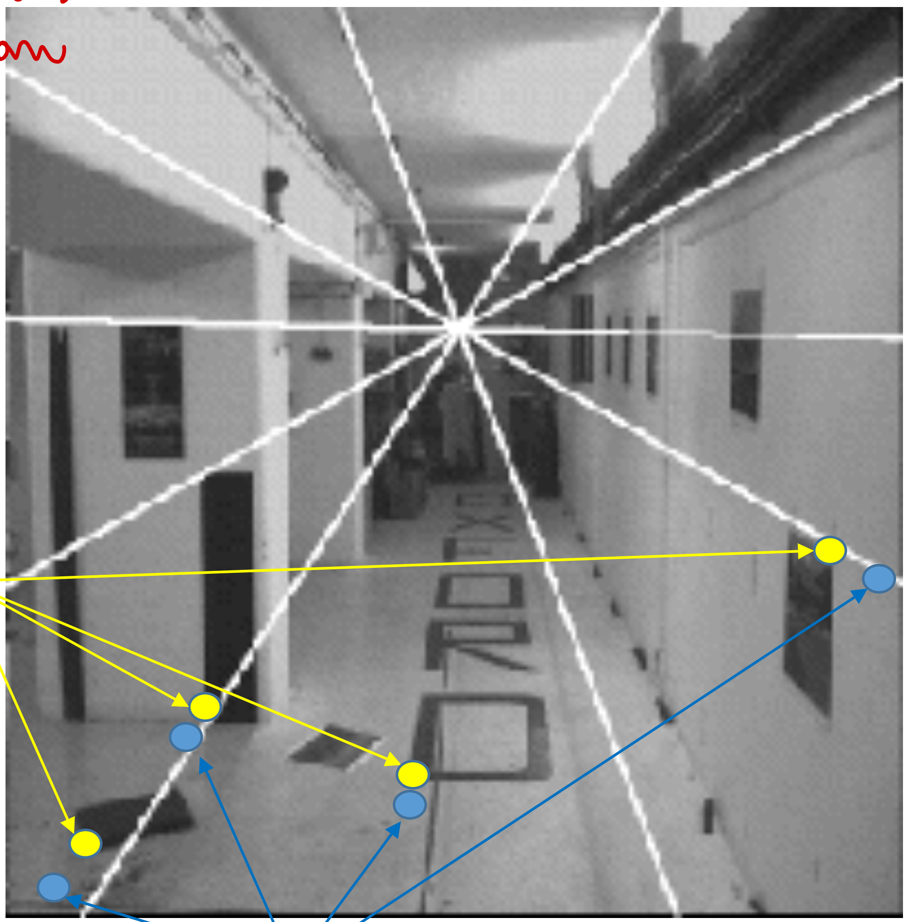
Hyp: constant speed → at certain time a feature cross the lens plane // image plane

space proportional to time

the line joining points are lines along translation as images of small segments in some direction

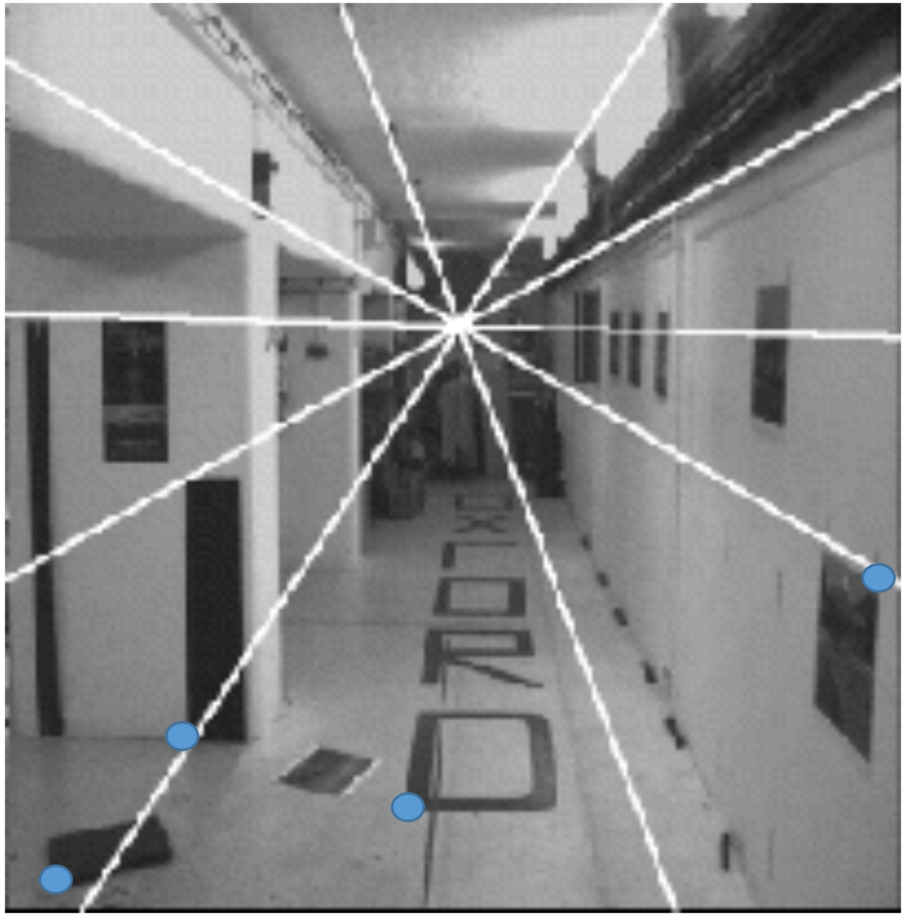
t = 0

t = 0

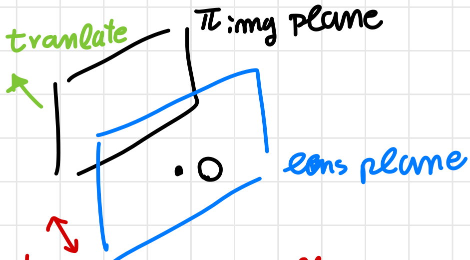


t = Δt

t = Δt



When we translate...
↓
at a certain point features crossed by img plane



d very small usually

↳ time to impact

$\underbrace{\text{time to impact}}$
how many secs take for a point to hit the image plane...

↑
it depends on the point chosen.

↓

when motion at constant speed v

⇐ then space $s \sim vt$
proportional to time

so time to impact
d distance along translation direction

↘ if I can compute the time to impact I can reconstruct

$\underbrace{\text{depth } z}$ // parallel to translation direction

3D feature coord along translation direction !

+ calibrated camera ⇒ metric reconstruction

IF points are co-linear ... all images of this feature move while translate ↘

↓ I can compute CROSS RATIO, having more than 4 linear

space proportional to time →

cross ratio of times = cross ratio of distances

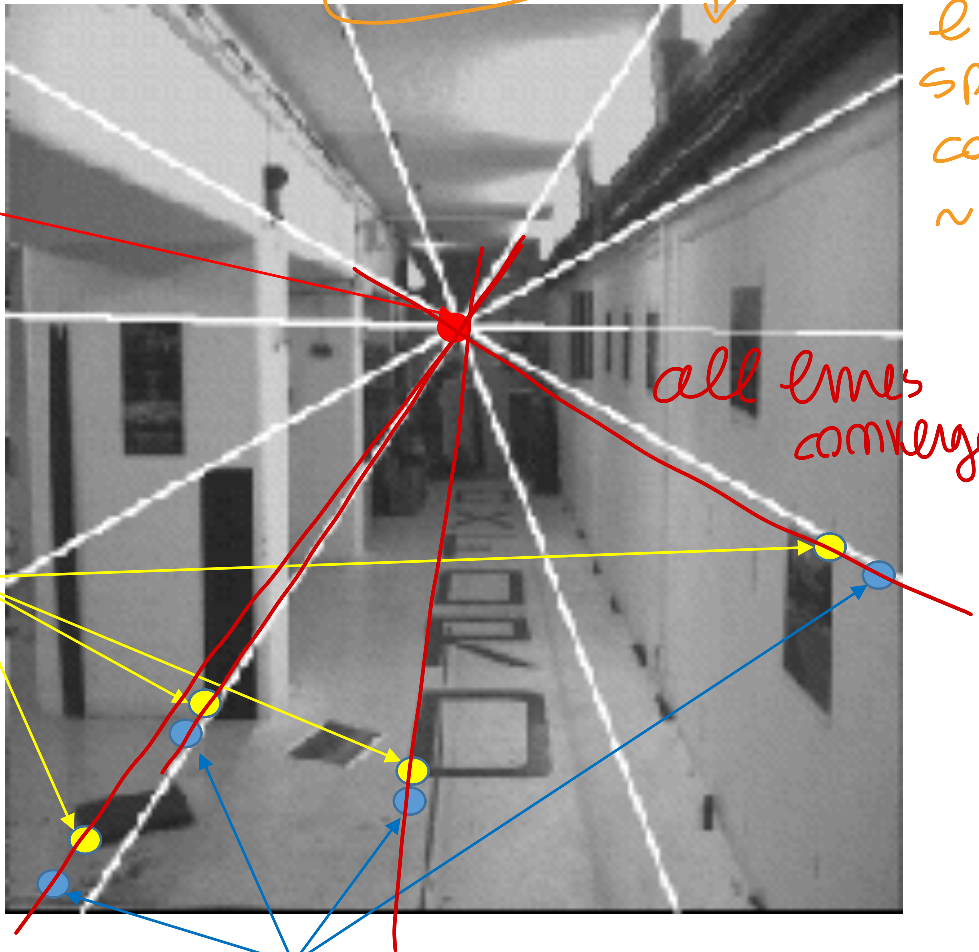
= cross ratio of image coordinates

t = 0

planar CROSS RATIO equal to 3D line space coordinate ~ time

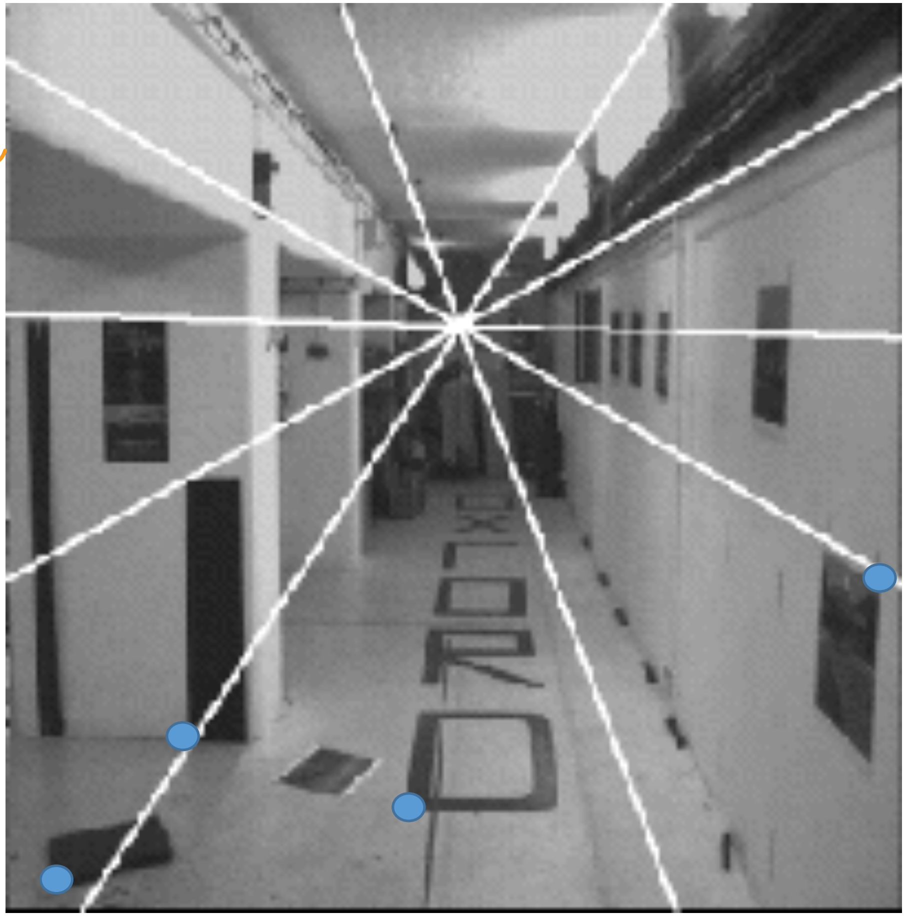
t = Δt

t = ∞



t = 0

t = Δt



⇒ IN PRACTICE...
I equate CROSS RATIO of image coord and distance over time.. I NEED 4 co-linear points

cross ratio of times = cross ratio of image coordinates

what is the fourth time instant?

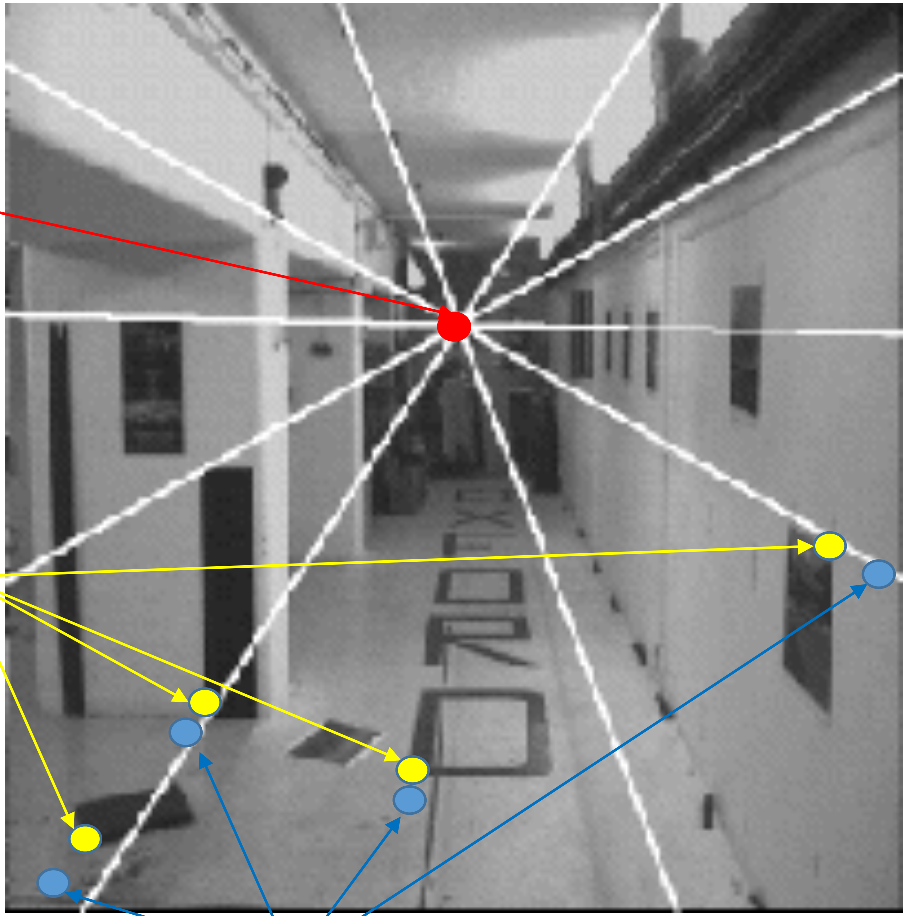
time to impact ~ space
⇓ we can compute

t = 0

t = Δt

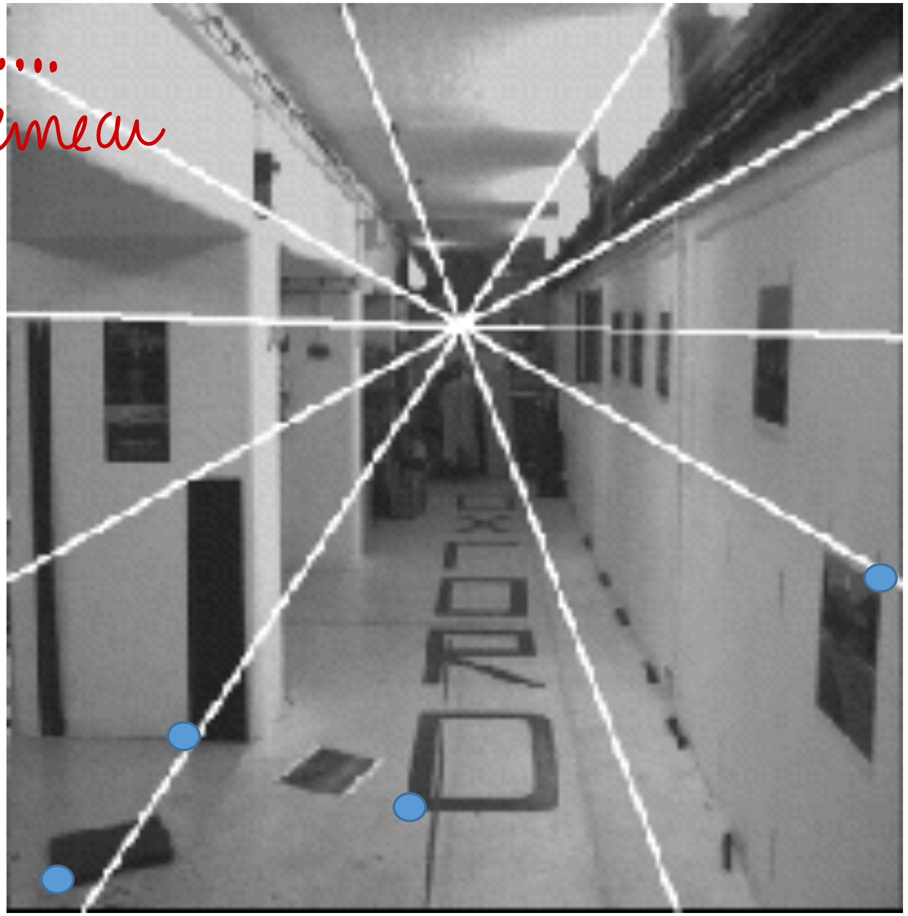
t = ∞

t = 0



t = Δt

time of impact.... from co-linear points
⇓
image of point at impact..
↑
when feature hits image plane
↑
intersect lens plane ⇒ at that point $z=0$ depth null



this will reach @ t = ∞

what is the fourth time instant?

time of impact *ti* with lens plane

Each feature has its own impact time instant

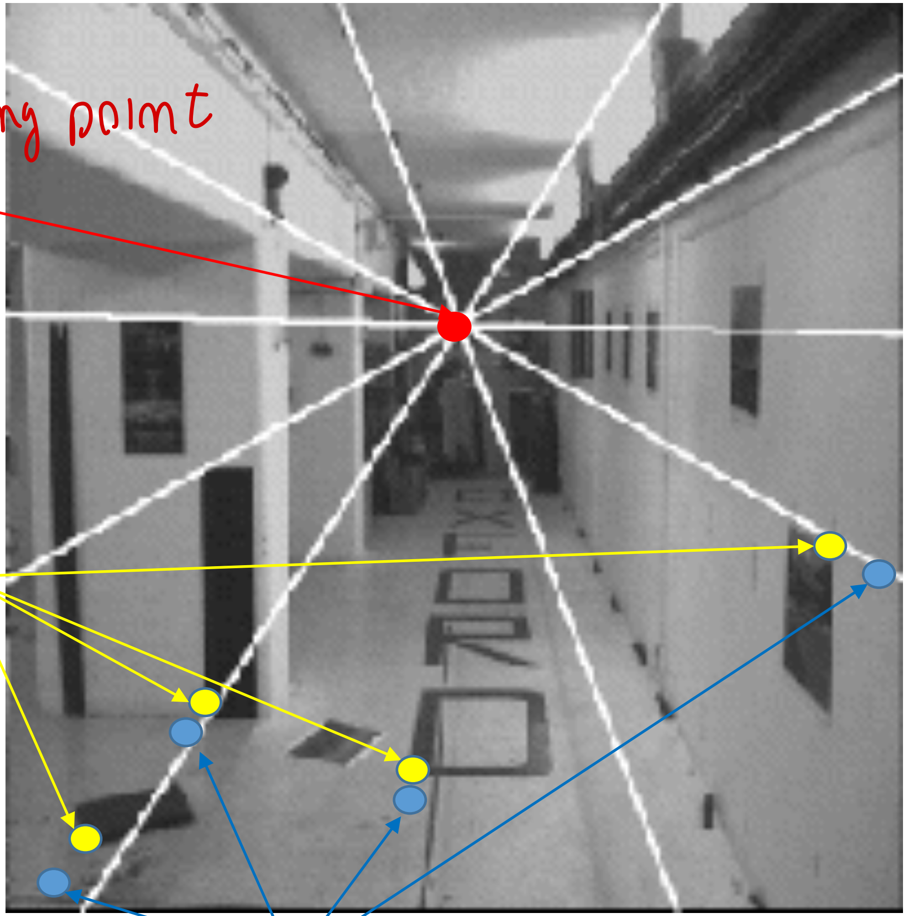
image of the point @ infinity...

vanishing point

t = 0

t = ∞

t = 0

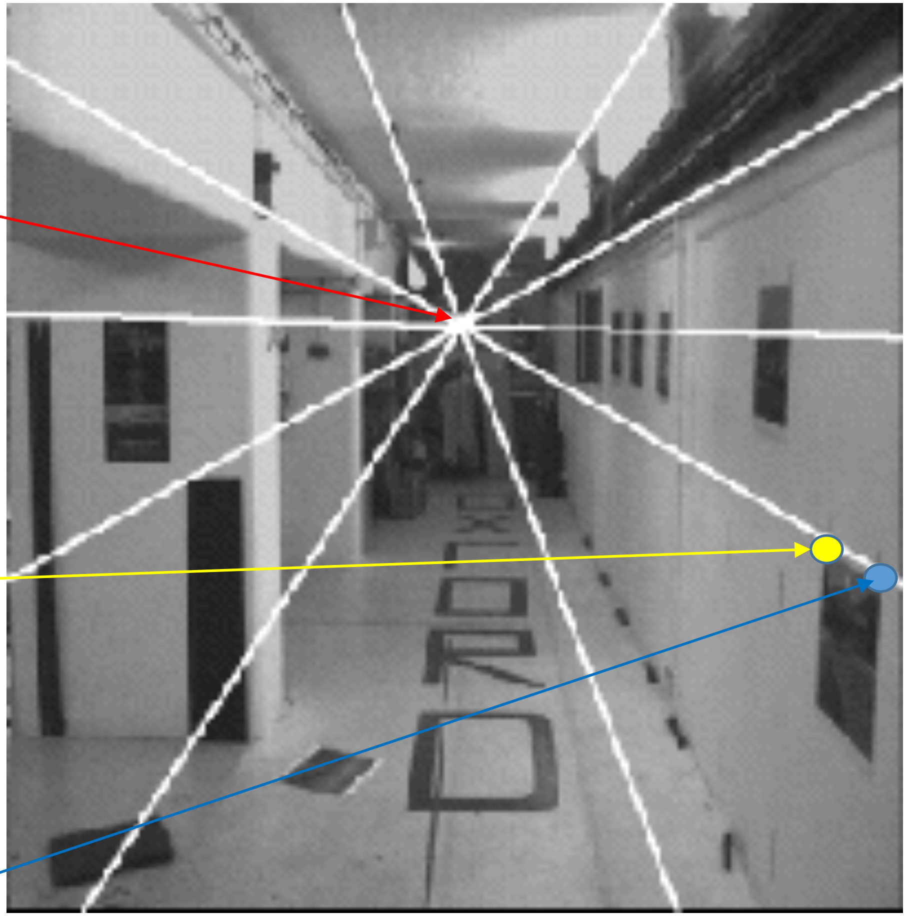


t = Δt

t = Δt

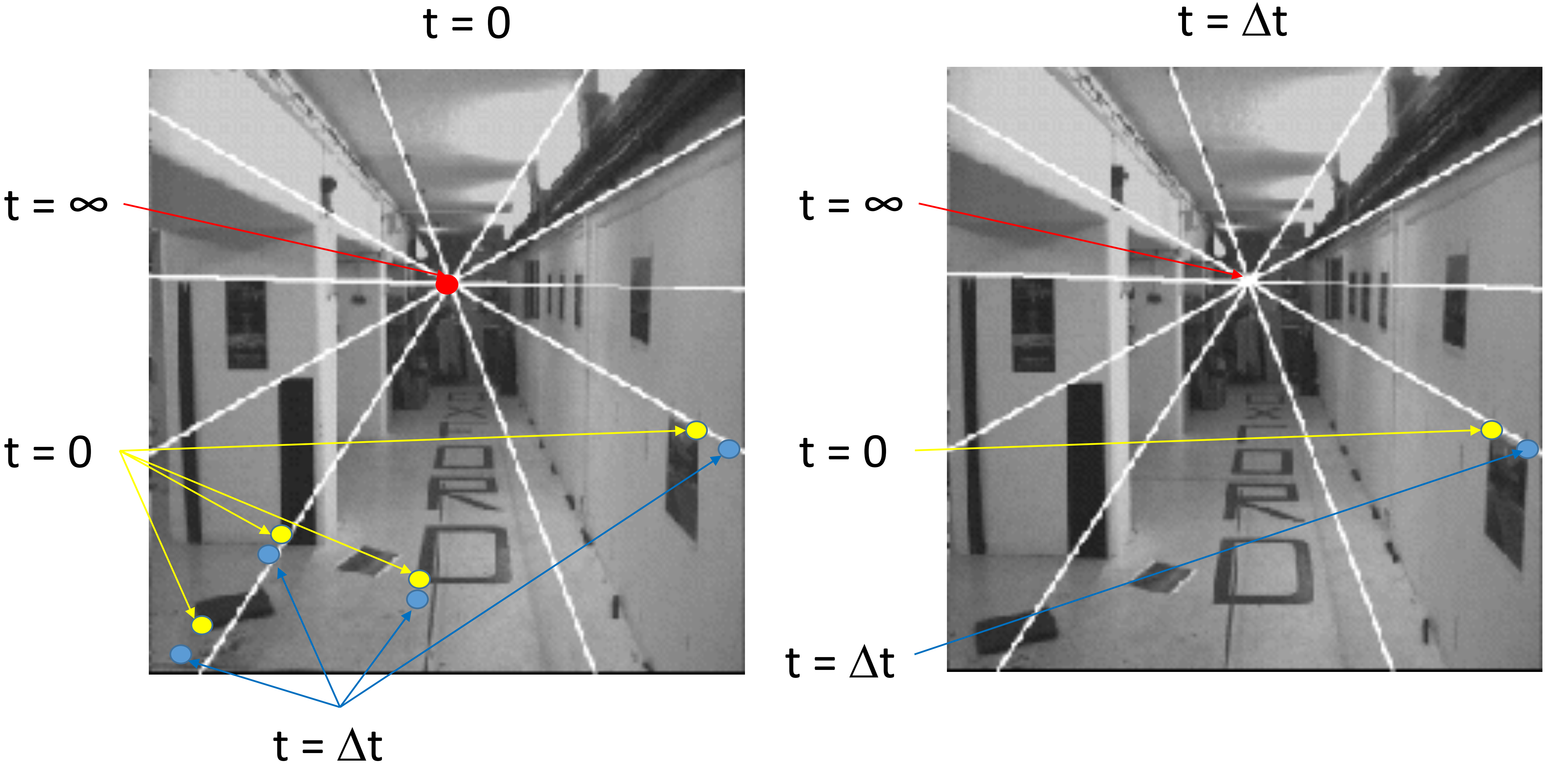
t = ∞

t = 0



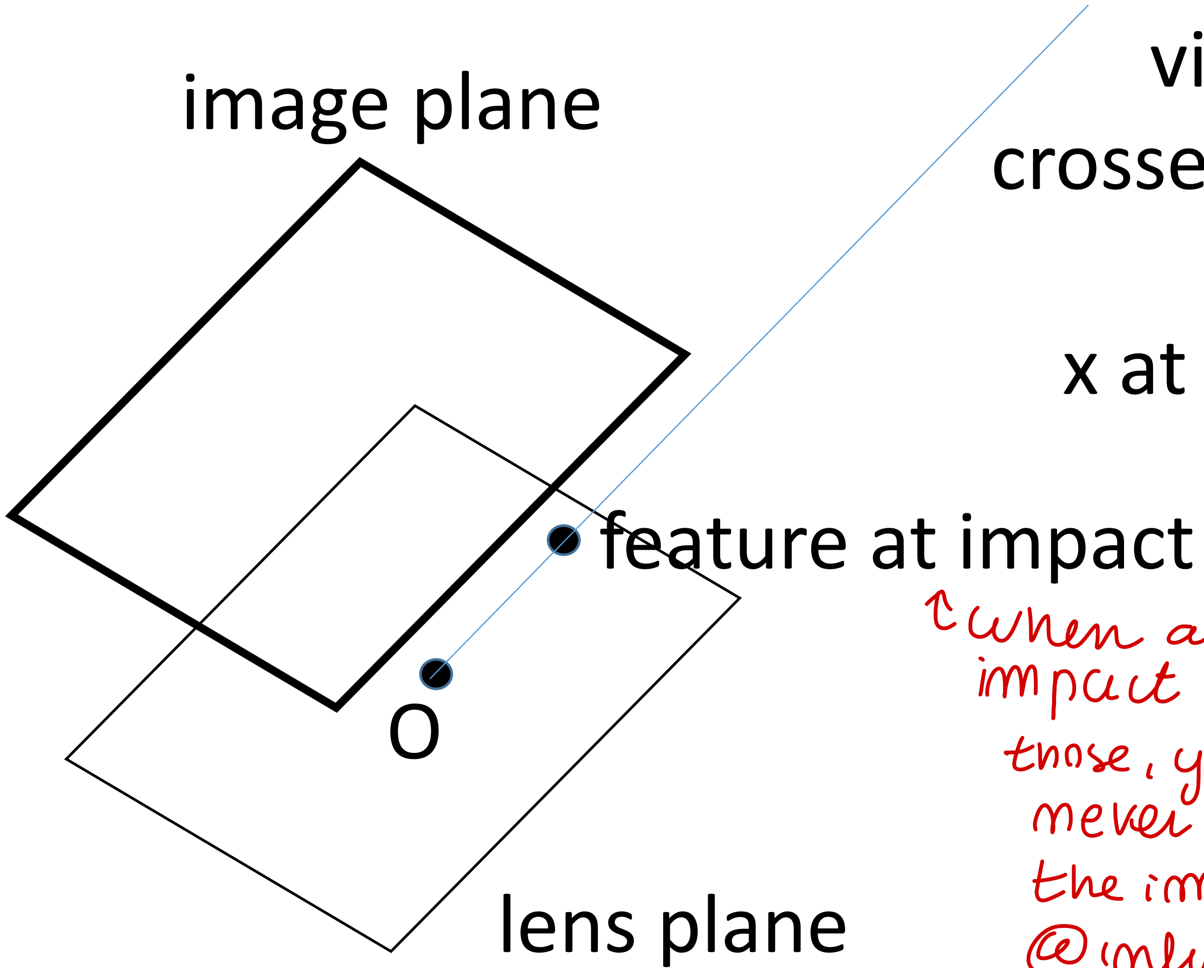
t = Δt

what is the fourth image coordinate?
at time of impact *ti* with lens plane
the image feature is at the ∞



so @ impact time,
we know everything,
when impact,
time $t = t_i$
and image is @ ∞,
when $t = t_i$ depth
$Z = 0$

all 4 elements
in time-space
domain
for img-space
are settled



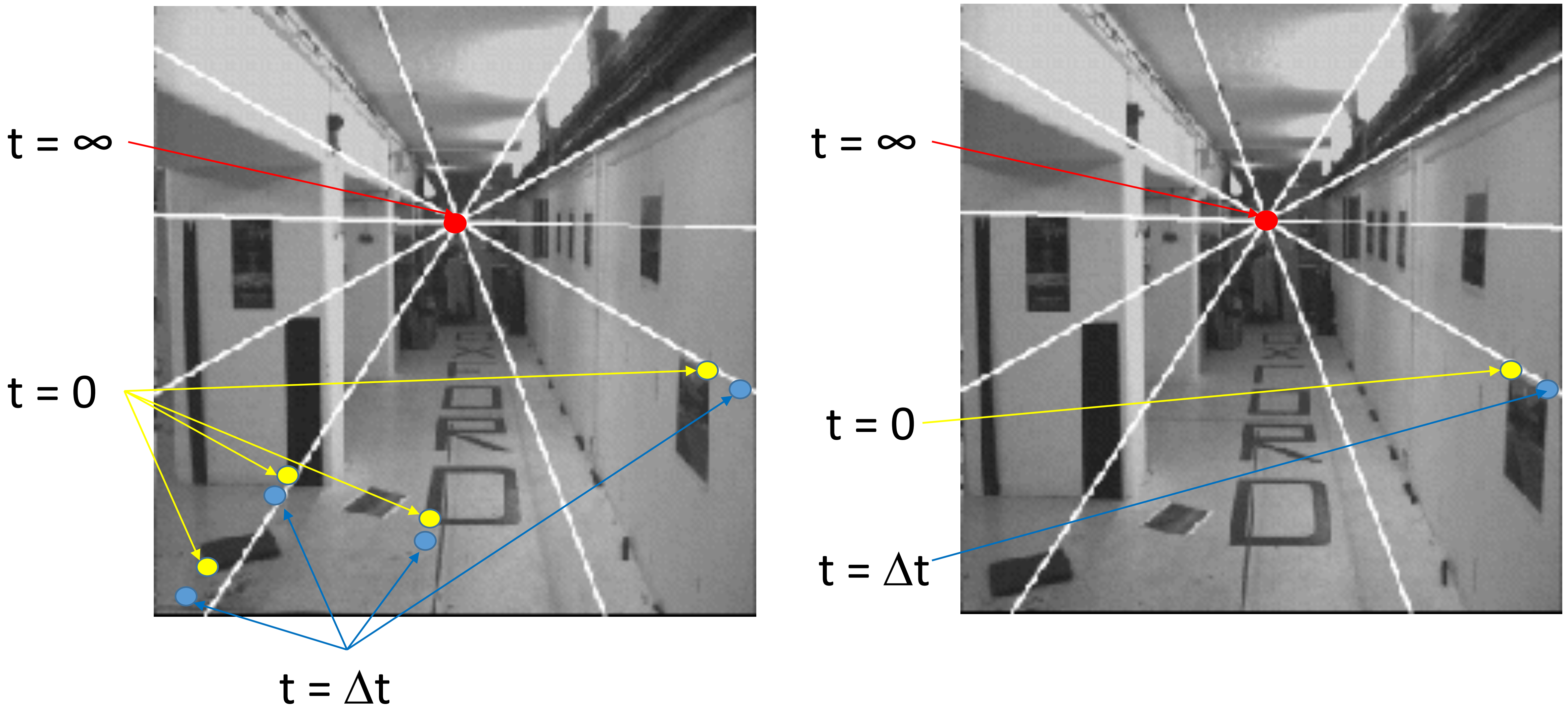
viewing ray:
crosses image plane
at ∞ →
x at imapact is ∞

when at
impact, joining
those, you cross
never...
the image is
@ infinity when
crossing at
lens plane

equal cross ratios

$$\frac{t_i-\Delta t}{t_i-0}\Big/\underbrace{\cancel{\frac{\infty-\Delta t}{\infty-0}}}_{1} = \underbrace{\cancel{\frac{\infty-\mathbf{x}'}{\infty-\mathbf{x}}}}_{1}\Big/\frac{\mathbf{x}_\infty-\mathbf{x}'}{\mathbf{x}_\infty-\mathbf{x}}$$

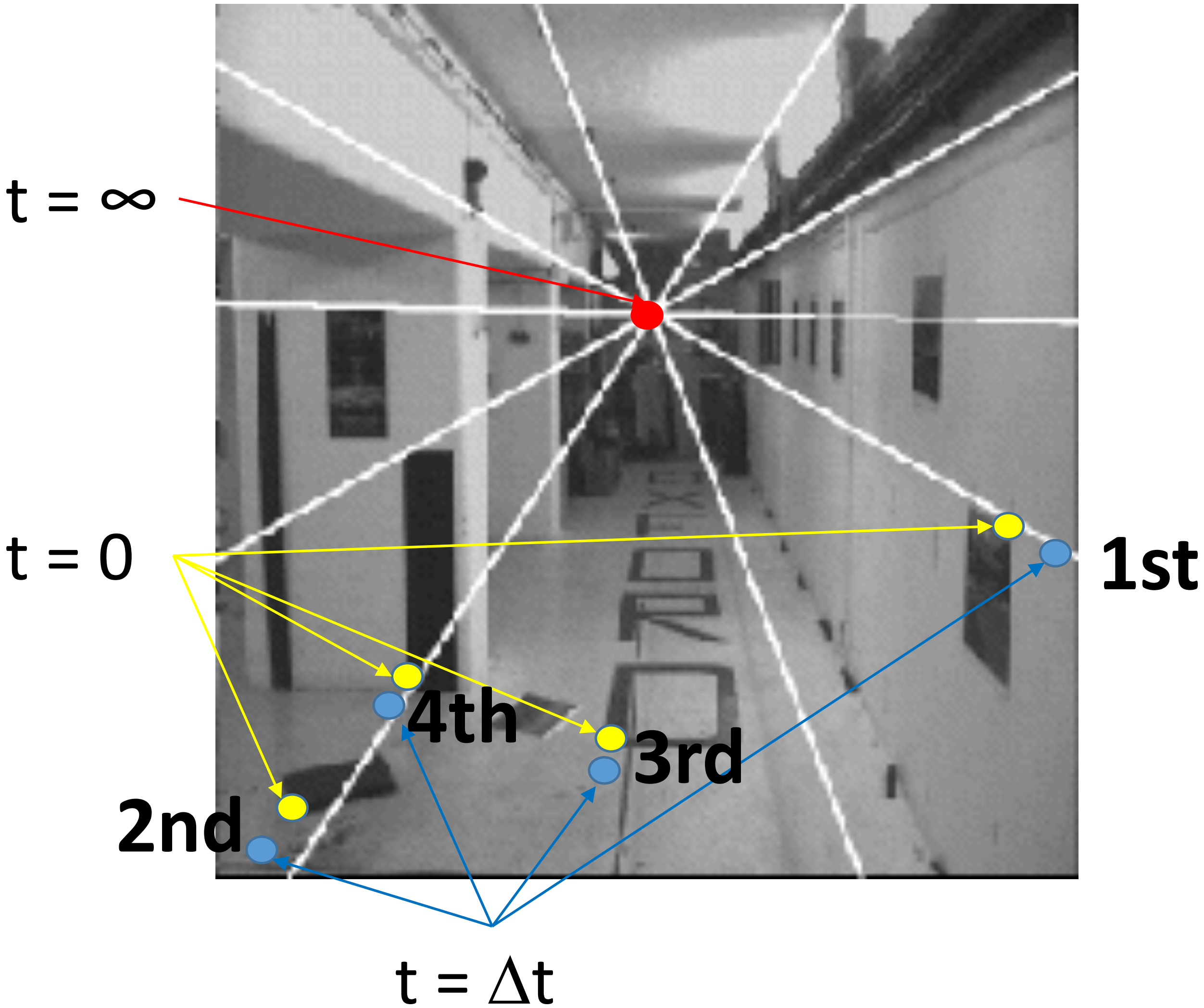
same operations at impact...



equal cross ratios

$$t_i = \Delta t \frac{\mathbf{x}_\infty - \mathbf{x}'}{\mathbf{x} - \mathbf{x}'}$$

time to impact relation, proportional to Δt time ~ velocity



sort features by increasing time to impact

computing for each features we can extract depth

full Reconstruction
if we have calibrated camera

partial 3D
reconstruction ~

Translational motion

time to impact of a feature
proportional to the **depth** of the feature
(i.e. distance from lens plane along motion direction)

→

it allows a partial 3D reconstruction

in the case of **calibrated** images (K known)
it allows a **full** (euclidean) 3D reconstruction,
since, for each moving feature, we know both the depth (from its time-to-impact) and the direction (from its viewing ray)

Fundamental matrix for pure translation

$$F = [e']_\times M' M^{-1} = [e']_\times$$

F only 2 d.o.f., $x^T[e]_x x=0 \Rightarrow$ auto-epipolar

<u>example</u>: $\mathbf{e}' = (1,0,0)^T \quad \mathbf{F} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}_\times \quad \mathbf{x}'^T \mathbf{F}\mathbf{x} = 0 \Leftrightarrow y = y'$

$$x = \begin{bmatrix} u \\ v \\ w \end{bmatrix} = PX = K[I|0]X = K\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} u \\ v \\ Z \end{bmatrix} = Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = Z x_{cart} \rightarrow \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z K^{-1} x_{cart}$$

$$x' = P'X = K[I|t]\begin{bmatrix} ZK^{-1}x_{cart} \\ 1 \end{bmatrix} = K[I|t]\begin{bmatrix} K^{-1}x_{cart} \\ 1/Z \end{bmatrix} = x_{cart} + Kt/Z$$

$x' = x_{cart} + Kt/Z$ and if $t_z = 0$ then $x' = x'_{cart}$ it is cartesian

motion starts at x and moves towards (or away from) e, faster depending on *Z*

By the way $\quad x' = x_{cart} + Kt/Z$ is a multiple of x'_{cart}

$$x' = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} + K\mathbf{t}/Z = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} + \begin{bmatrix} f_x & (s) & U_o \\ 0 & f_y & V_o \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} /Z = \begin{bmatrix} * \\ * \\ 1 + t_z/Z \end{bmatrix}$$

$\rightarrow$

$$x'_{cart} = \frac{x'}{1+t_z/Z} \rightarrow x'_{cart}(1 + t_z/Z) = x' = x_{cart} + K\,t/Z$$

but $K\mathbf{t}/t_z = x_\infty$ is the common epipole e of the two images, hence

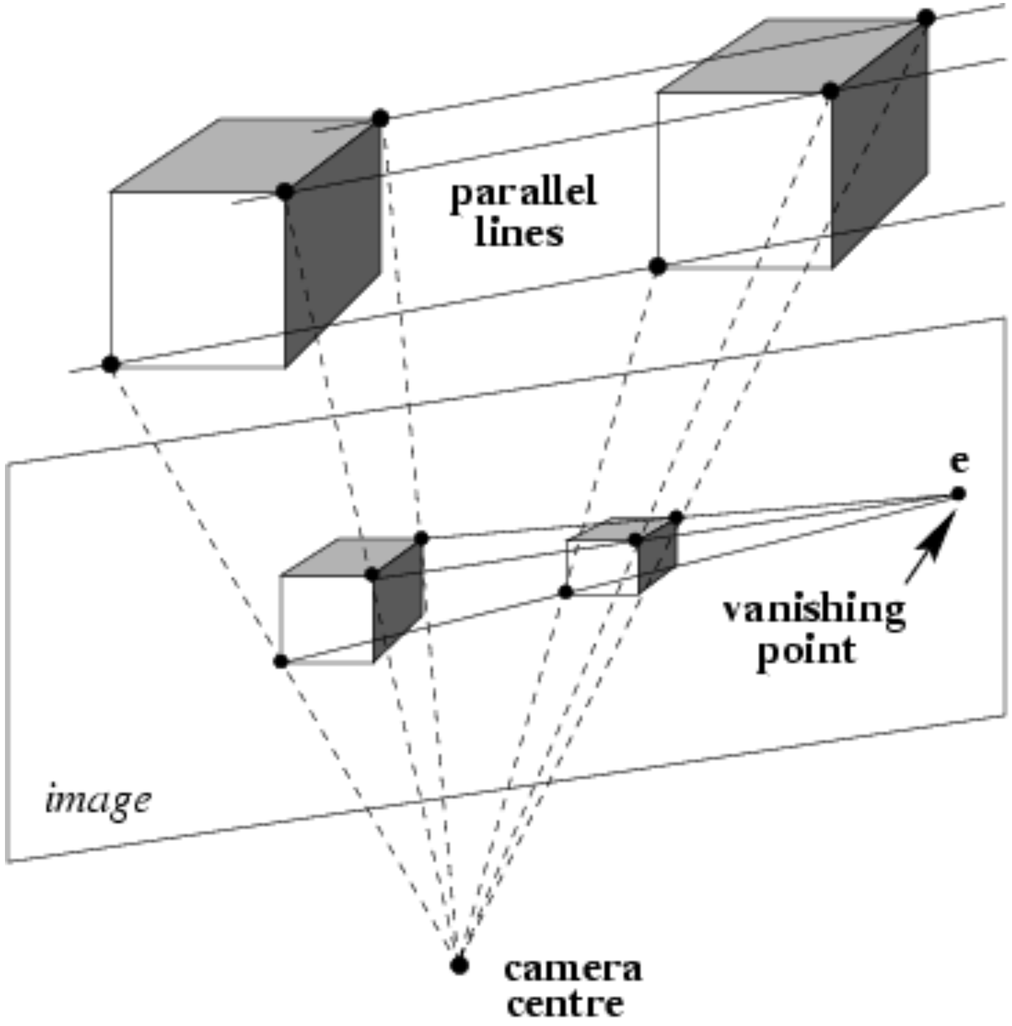
see next slide $\quad x'_{cart} + x'_{cart}(t_z/Z) = x_{cart} + x_\infty(t_z/Z)$

$\rightarrow$

$$Z = t_z \frac{x_\infty - x'_{cart}}{x'_{cart} - x_{cart}}$$

(ratio of two colinear vectors)

Fundamental matrix for pure translation



Viewing ray of epipole is parallel to the translation direction

$$K^{-1}e = \mathbf{t} \rightarrow e = K\mathbf{t}$$

By the way $\quad x' = x_{cart} + Kt/Z$ is a multiple of x'_{cart}

$$x' = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} + K\mathbf{t}/Z = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} + \begin{bmatrix} f_x & (s) & U_o \\ 0 & f_y & V_o \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} /Z = \begin{bmatrix} * \\ * \\ 1 + t_z/Z \end{bmatrix}$$

$\rightarrow$

$$x'_{cart} = \frac{x'}{1+t_z/Z} \rightarrow \underline{x'_{cart}(1 + t_z/Z)} = x' = \underline{x_{cart} + K\,t/Z}$$

but $K\mathbf{t}/t_z = x_\infty$ is the common epipole e of the two images, hence

$$\underline{x'_{cart} + x'_{cart}(t_z/Z)} = \underline{x_{cart} + x_\infty(t_z/Z)}$$

$\rightarrow$

$$Z = t_z \frac{x_\infty - x'_{cart}}{x'_{cart} - x_{cart}}$$

(ratio of two colinear vectors)

By the way $\mathrm{x}' = \mathrm{x}_{cart} + \mathrm{Kt}/Z$ is a multiple of x'_{cart}

$$\mathrm{x}' = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} + \mathrm{Kt}/Z = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} + \begin{bmatrix} f_x & (s) & U_o \\ 0 & f_y & V_o \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} /Z = \begin{bmatrix} * \\ * \\ 1 + t_z/Z \end{bmatrix}$$

→

$$\mathrm{x}'_{cart} = \frac{\mathrm{x}\prime}{1+t_z/Z} \rightarrow \mathrm{x}'_{cart}(1 + t_z/Z) = \mathrm{x}' = \mathrm{x}_{cart} + \mathrm{K\,t}/Z$$

but $\mathrm{Kt}/t_z = \mathrm{x}_\infty$ is the common epipole of the two images, hence

$$\mathrm{x}'_{cart} + \mathrm{x}'_{cart}(t_z/Z) = \mathrm{x}_{cart} + \mathrm{x}_\infty(t_z/Z)$$

→

$$Z = t_z \frac{\mathrm{x}_\infty - \mathrm{x}\prime_{cart}}{\mathrm{x}\prime_{cart} - \mathrm{x}_{cart}}$$

(ratio of two colinear vectors)

depth can be usefule in many application

Observation

t_z : camera displacement along Z during Δt

you can compute Z coord using translation displacement instead of Δt

$$Z = t_z \frac{x_\infty - x'_{cart}}{x'_{cart} - x_{cart}}$$

given Z depth, we need viewing direction to achieve full reconstruction

Calibrated camera

and

$$t_i = \Delta t \frac{x_\infty - x'_{cart}}{x_{cart} - x'_{cart}}$$

are of opposite sign:

this is because the time to impact is positive only if Z decreases with time, which can only occur if the speed $t_z / \Delta t$ is negative depth Z provide distance coord along translation respect lens plane...

by depth, moving along Z direction, we can reconstruct points depth

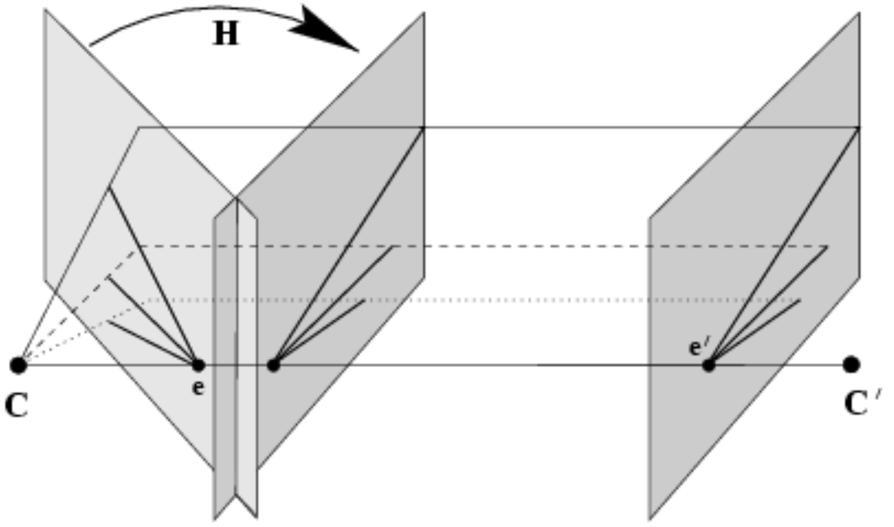
$\forall$ feature depth

BUT NOT full $\underbrace{\text{3D reconstruction}}$

to specify position you need also viewing direction of that point!

intersect viewing ray with plane π_Z at depth Z to reconstruct the scene

General motion



$$\mathrm{x}'^{\mathsf{T}} \left[\mathrm{e}'\right]_\times \hat{\mathrm{x}} = 0$$

$$\mathrm{x}'^{\mathsf{T}} \left[\mathrm{e}'\right]_\times \mathrm{H}\mathrm{x} = 0$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z\mathrm{K}^{-1}\mathrm{x}_{cart}$$

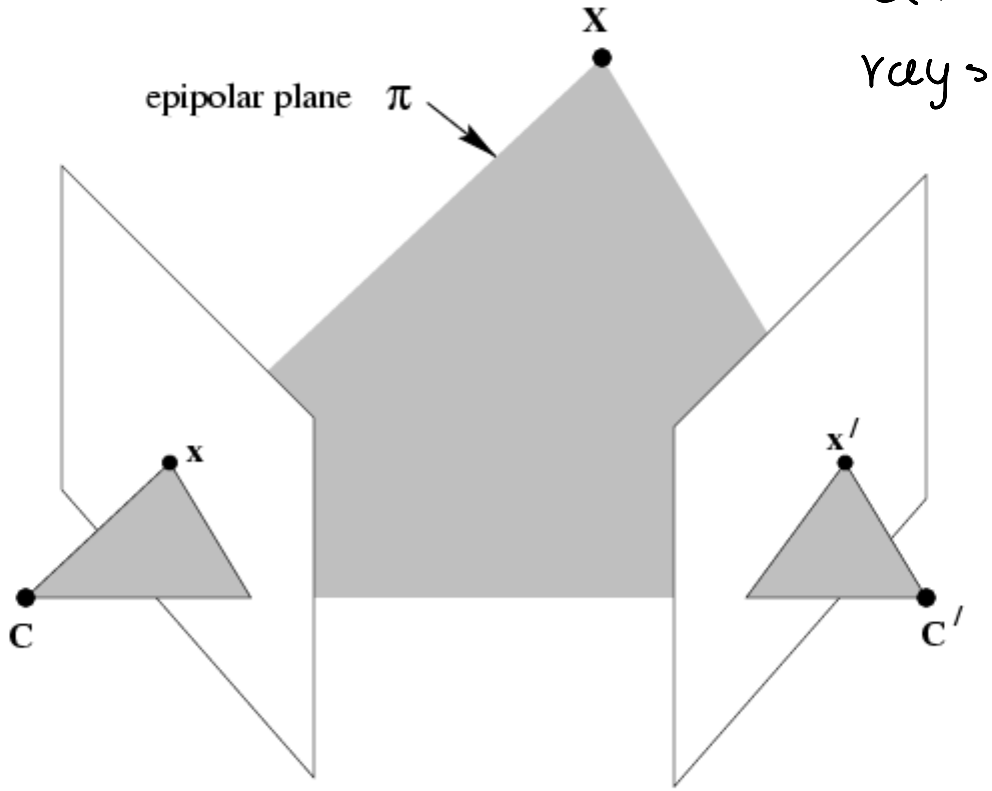
$$\mathrm{x}' = \mathrm{P}'\mathrm{X} = \mathrm{K}'[\mathrm{R}|\mathrm{t}]\begin{bmatrix} Z\mathrm{K}^{-1}\mathrm{x}_{cart} \\ 1 \end{bmatrix} = \mathrm{K}'[\mathrm{R}|\mathrm{t}]\begin{bmatrix} \mathrm{K}^{-1}\mathrm{x}_{cart} \\ \frac{1}{Z} \end{bmatrix} = \mathrm{K}'\mathrm{R}\mathrm{K}^{-1}\mathrm{x}_{cart} + \mathrm{K}'t/Z$$

$$\mathrm{x}' = \mathrm{K}'\mathrm{R}\mathrm{K}^{-1}\mathrm{x}_{cart} + \mathrm{K}'\mathrm{t}/Z$$

to do this, a technique is TRIANGULATION

3D reconstruction by triangulation

extracting viewing rays and finding point X



X = intersection of the viewing rays:

- viewing ray associated to x with camera P
- viewing ray associated to x′ with camera P′

Attention: change of notation

under some conditions you can reconstruct $\Rightarrow$

$$\mathrm{P}, \mathrm{P}' \rightarrow \mathrm{P}_1, \mathrm{P}_2$$

let's change notation

I camera II camera

$$\mathrm{P}_1', \mathrm{P}_2'$$

$\leftarrow$ in reconstruction, we modify (refine) camera until we get the correct

will be used to indicate modified camera pairs

iteratively try to fix it

Remember: (in New notation)

Fundamental matrix of cameras $\mathrm{P}_1 = [\mathrm{M}_1|\mathrm{m}_1], \mathrm{P}_2 = [\mathrm{M}_2|\mathrm{m}_2]$:

$$\mathrm{F} = [\mathrm{e}_2]_\times \mathrm{M}_2 \mathrm{M}_1{}^{-1}$$

$$\mathrm{e}_2 = \mathrm{LNS(F)} \rightarrow \mathrm{e}_2{}^T \mathrm{F} = 0$$

Projective Ambiguity Theorem

give you two equivalent facts! ⇓

A fundamental matrix F_{12} is compatible with camera pairs (P_1, P_2) and (P_1', P_2')

From P_1, P_2 F is univoquely determined, while from F I have ∞ number of P_1', P_2' compatible with it

both first cameras couples 1, 2 and adjusted 1', 2' cameras

⇕ It holds that:

camera pairs are projectively related: i. e. ∃ an invertible matrix H_{4x4} such that

$$\begin{cases} P_1' = P_1 H^{-1} \\ P_2' = P_2 H^{-1} \end{cases}$$

both transformed in same way!

this holds in both direction

Projective Ambiguity Theorem

proof of ↑ direction

this is "easy" to proof

camera pairs (P_1, P_2) and (P_1', P_2') have the same fundamental matrix F_{12}

can be computed from pairs of image points correspond.

↑

given camera pairs related, then

camera pairs are projectively related: i.e. $\exists$ an invertible matrix H_{4x4} such that

points co-planar remain such tat, co-linearity, CR order of contact etc.. preserved!

$$\begin{cases} P_1' = P_1 H^{-1} \\ P_2' = P_2 H^{-1} \end{cases}$$

→ compatible with $\underbrace{\text{same}}$ F

because F from pairs of img points

⇓

compatible with same images...

also lot of related cameras are fine ⇒

Projective ambiguity: ↑ proof summary

Suppose that a Fundamental matrix F is compatible with a camera pair (P_1, P_2). Consider a second camera pair $(P_1{}', P_2{}')$ such that:

we have ∞ pairs of cameras compatible...

$$\begin{cases} P_1' = P_1 H^{-1} \\ P_2' = P_2 H^{-1} \end{cases}$$

given camera matrix

Triangulating the viewing rays of corresponding pairs of image points $(\mathrm{x}_1, \mathrm{x}_2)$ according to cameras (P_1, P_2), the 3D points X are obtained. by intersection

Now apply a projective transformation H to points X : points $X' = HX$ are obtained.

Project the transformed points $X' = HX$ onto cameras $(P_1{}', P_2{}')$:

The same image points (→ **same Fundamental matrix** F) are obtained by image-projecting

the transformed 3D points $X' = HX$ onto the new cameras $(P_1{}', P_2{}')$

reconstruction related to original ..

by apply transformation

in fact

false cameras P_1', P_2' gives triangulation that still keep same image, projectively related

$$\begin{cases} \mathrm{x}'_1 = P_1' X' = P_1\, H^{-1} X' = P_1\, H^{-1} H X = P_1\, X = \mathrm{x}_1 \\ \mathrm{x}'_2 = P_2' X' = P_2\, H^{-1} X' = P_2\, H^{-1} H X = P_2\, X = \mathrm{x}_2 \end{cases}$$

same image!

→

given images pairs are compatible not only with true 3D points X

but also with any projectively transformed points $X' = HX$

← FIRST STEP of 3D reconstruction

2) from false triangulation, I reduce generality to affine, then similarity
↓
reach desired reconstruction

Projective transformation and invariance

Derivation based purely on projective concepts

$$\hat{x} = Hx, \hat{x}' = H'x' \Rightarrow \hat{F} = H'^{-T} F H^{-1}$$

F invariant to transformations of projective 3D-space

$$x_1 = P_1 X = \left(P_1 H^{-1}\right)(HX) = P_1' X'$$

$$x_2 = P_2 X = \left(P_2 H^{-1}\right)(HX) = P_2' X'$$

$(P_1, P_2) \mapsto F$ unique

$F_{12} \mapsto (P_1, P_2)$ not unique

canonical form

$P_1 = [I \mid 0]$

$P_2 = [M \mid m]$

$$e_2 = m \rightarrow F_{12} = [m]_\times M$$

useful fact learnt from the ↑ proof

A set of images of a given scene

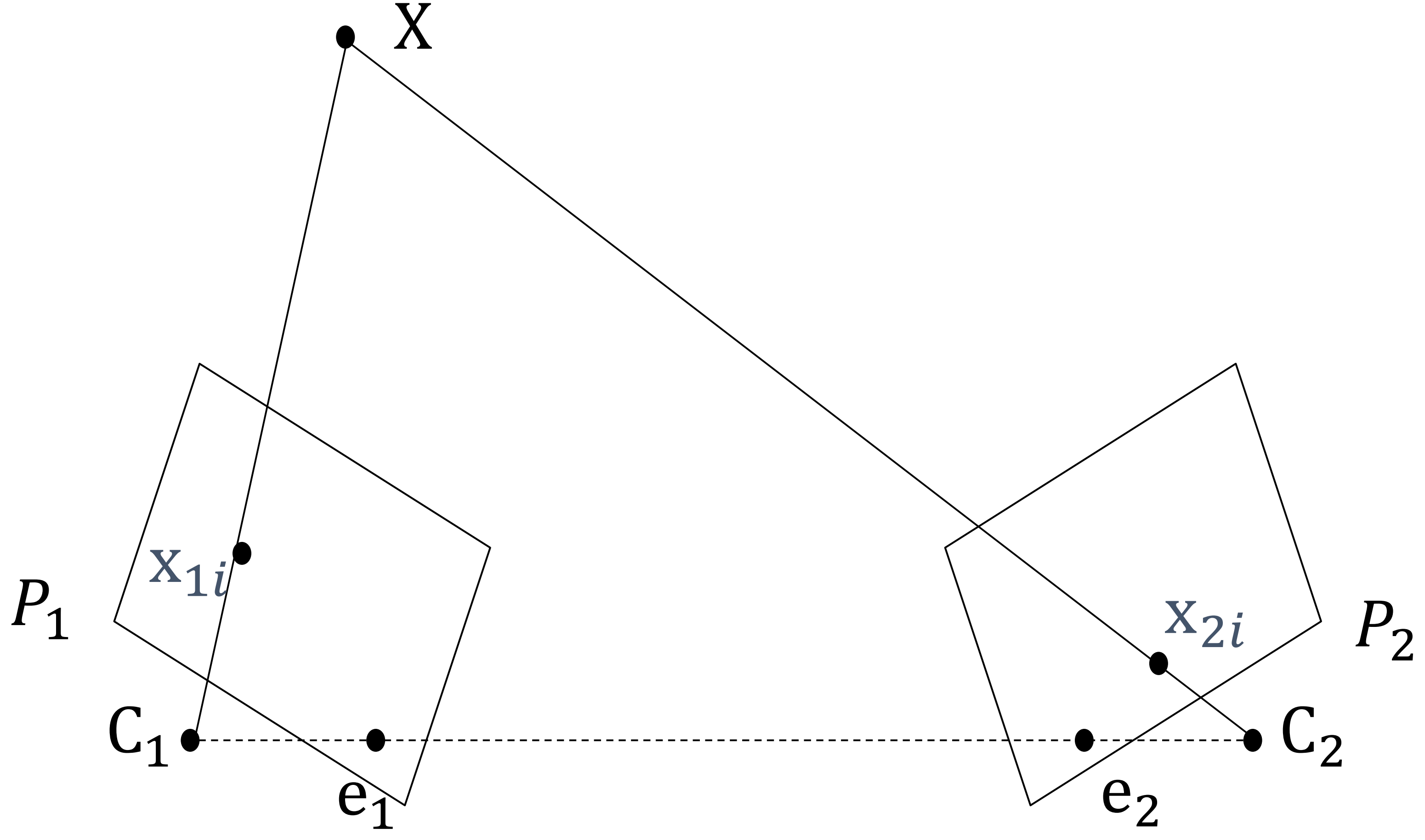
are also compatible with

any projective transformation of the given scene

or

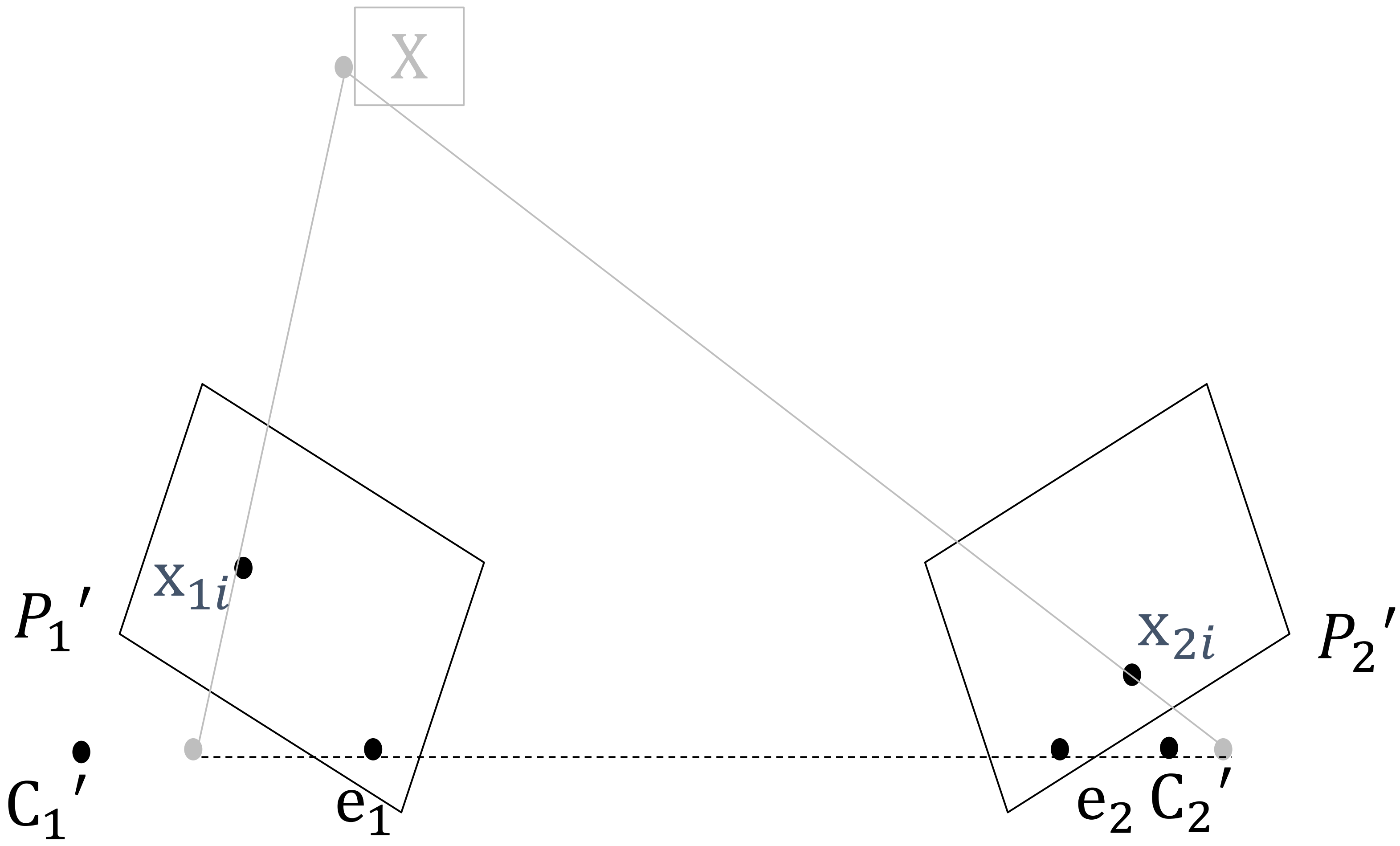
any two projectively related 3D scenes can produce exactly the same images:
by adapting the cameras, i.e. by letting the camera matrices vary

Projective ambiguity F=F (P_1, P_2): given corresponding image points

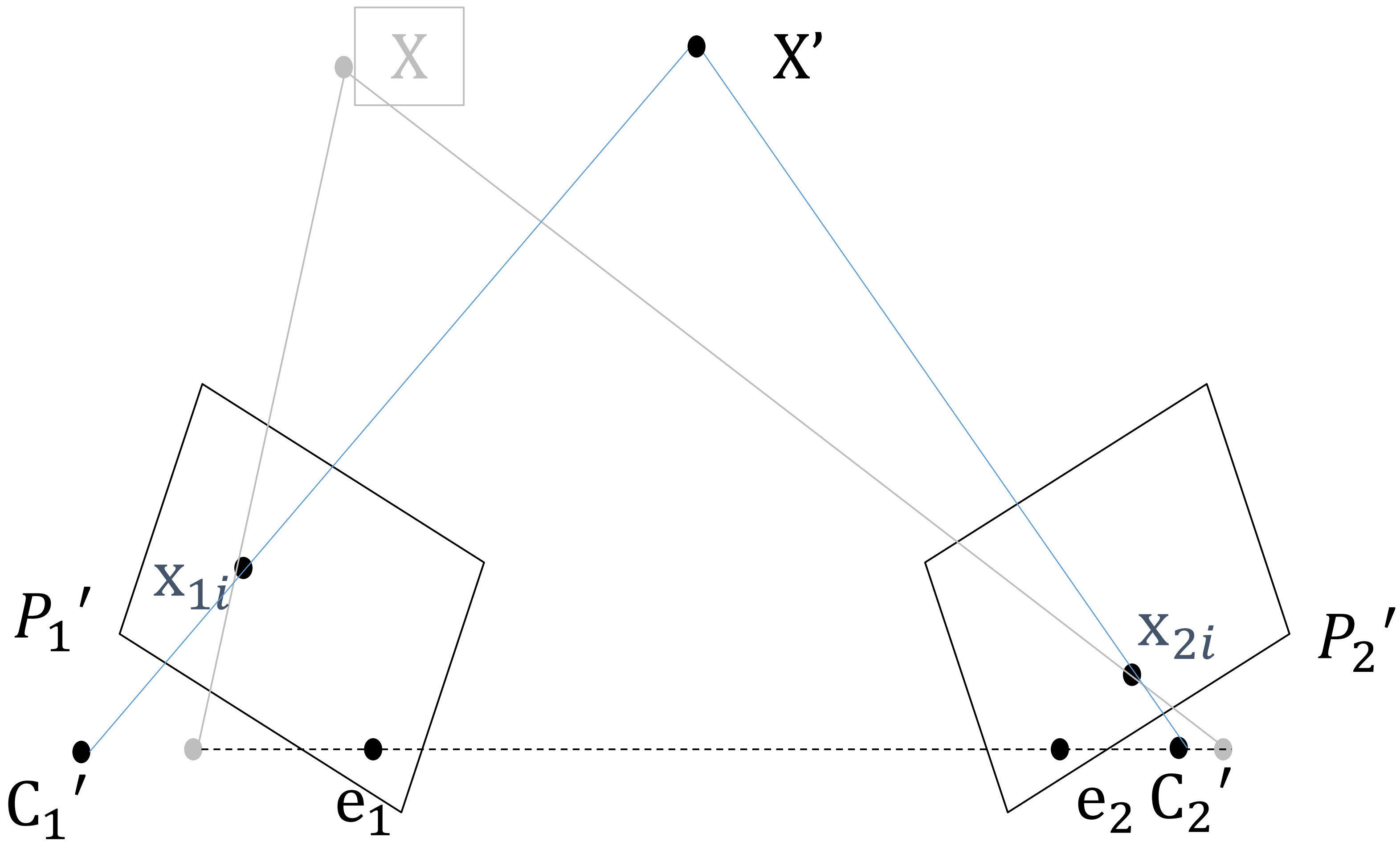


triangulated points X

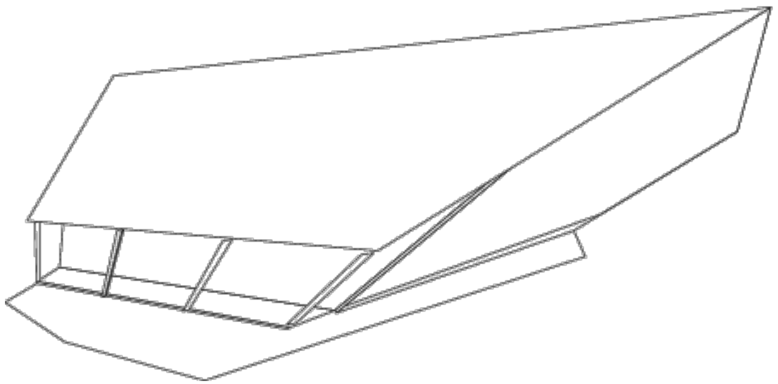
Projective ambiguity F=F (P_1', P_2')=F (P_1, P_2): other cameras with the same F: $P_1' = P_1\mathrm{H}^{-1}$ $P_2' = P_2\mathrm{H}^{-1}$



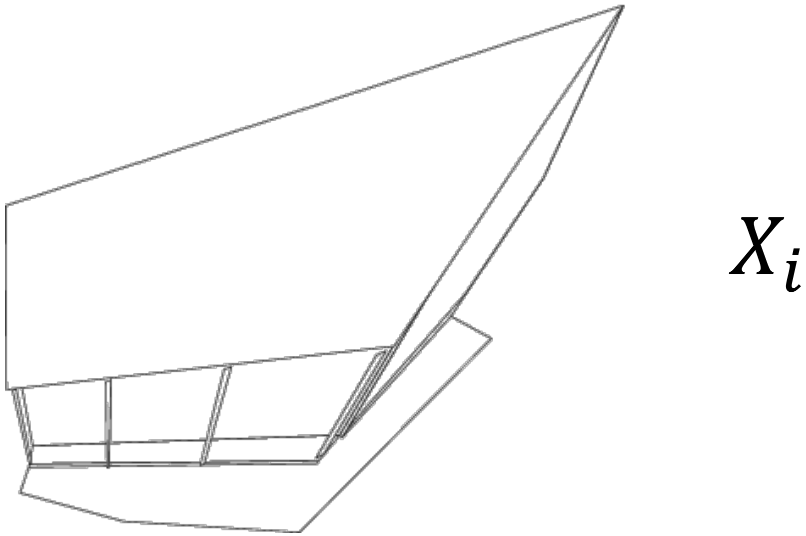
Projective ambiguity F=F (P_1', P_2')=F (P_1, P_2): other cameras with the same F: $P_1' = P_1 \mathrm{H}^{-1}$ $P_2' = P_2 \mathrm{H}^{-1}$



triangulated points $\mathrm{X}' = \mathrm{HX}$



X_i



X_i'

Camera pairs in canonical form

From ↑ part of the theorem

two camera pairs that are projectively related have the same F

→

if camera pair (P_1^o, P_2^o) has F as fundamental matrix, with $P_1^o = [\mathrm{M} \quad \mathrm{m}]$,

then take $\mathrm{H} = \begin{bmatrix} & \mathrm{M} & & \mathrm{m} \\ 0 & 0 & 0 & 1 \end{bmatrix}$: $P_1^o = [\mathrm{M} \quad \mathrm{m}] = [\mathrm{I} \quad 0]\mathrm{H}$

therefore, also $\boldsymbol{P_1} \stackrel{\text{def}}{=} [\mathbf{I} \quad \mathbf{0}] = P_1^o H^{-1}$ and $\boldsymbol{P_2} \stackrel{\text{def}}{=} [\mathbf{A} \quad \mathbf{a}] = P_2^o H^{-1}$

have the same F as fundamental matrix!! Cameras in **canonical form** for F :

$$\boxed{(P_1, P_2) = ([\mathrm{I} \quad 0], [\mathrm{A} \quad \mathrm{a}])}$$

In addition, the epipole of pair $(P_1, P_2) = (([\mathrm{I} \quad 0], [\mathrm{A} \quad \mathrm{a}])$ is $[\mathrm{A} \quad \mathrm{a}] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \mathrm{a}$

and $F = [\mathrm{e}_2]_\times \mathrm{M}_2 {\mathrm{M}_1}^{-1} = [a]_\times \mathrm{AI}^{-1} = [a]_\times \mathrm{A}$

image by P_2 of the center of P_1

Projective Ambiguity Theorem: proof of ↓ direction

A fundamental matrix F_{12} is compatible with camera pairs (P_1, P_2) and (P_1', P_2')

$$\downarrow$$

$$\text{camera pairs are projectively related: i. e.} \exists \text{ an invertible matrix } H_{4x4} \text{ such that}$$

$$P_1' = P_1 H^{-1}$$
$$P_2' = P_2 H^{-1}$$

Projective ambiguity of cameras given F

↓ easier proof with cameras in canonical form

if F is the same both for (P_1, P_2) and for $(P_1{}', P_2{}')$ ➔

∃ a projective transformation H so that $P_1' = P_1 H^{-1}$ and $P_2' = P_2 H^{-1}$

$$\mathrm{P}_1 = [\mathrm{I}|0]\ ;\ \mathrm{P} = \mathrm{P}_2 = [\mathrm{A}|\mathrm{a}] \quad \text{and} \quad \mathrm{P}_1' = [\mathrm{I}|0];\ \mathrm{P}' = \mathrm{P}_2' = [\widetilde{\mathrm{A}}|\tilde{\mathrm{a}}]$$

$$\mathrm{F} = [\mathrm{a}]_\times \mathrm{A} = [\tilde{\mathrm{a}}]_\times \widetilde{\mathrm{A}}$$

lemma: $\tilde{\mathbf{a}} = \boldsymbol{k}\mathbf{a}$ **and** $\widetilde{\mathbf{A}} = \boldsymbol{k}^{-1}(\mathbf{A} + \mathbf{a}\mathbf{v}^{\mathrm{T}})$

proof: $\mathrm{a}^T\mathrm{F} = \mathrm{a}^T[\mathrm{a}]_\times\mathrm{A} = 0$ and similarly $\tilde{\mathrm{a}}\mathrm{F} \Rightarrow 0 \quad \tilde{\mathrm{a}} = \mathrm{ka}$ since they are both LNS of the rank 2 matrix F

$[\mathrm{a}]_\times\mathrm{A} = [\tilde{\mathrm{a}}]_\times\widetilde{\mathrm{A}} \Rightarrow [\mathrm{a}]_\times(k\widetilde{\mathrm{A}} - \mathrm{A}) = 0 \Rightarrow$ all columns of $(k\widetilde{\mathrm{A}} - \mathrm{A})$ are multiples of a $\Rightarrow (k\widetilde{\mathrm{A}} - \mathrm{A}) = \mathrm{av}^{\mathrm{T}}$

in fact, $\mathrm{a} \times (k\widetilde{\mathrm{A}} - \mathrm{A}) = 0 \Rightarrow$ a is «parallel» to each column vector in $(k\widetilde{\mathrm{A}} - \mathrm{A})$

take $\mathrm{H}^{-1} = \begin{bmatrix} k^{-1}\mathrm{I} & 0 \\ k^{-1}\mathrm{v}^{\mathrm{T}} & k \end{bmatrix}$ $\quad \mathrm{P}_1\mathrm{H}^{-1} = [I|0] = \mathrm{P}_1'$ and from $(k\widetilde{\mathrm{A}} - \mathrm{A}) = \mathrm{av}^{\mathrm{T}}$ is $\widetilde{\mathrm{A}} = k^{-1}(\mathrm{A} + \mathrm{av}^{\mathrm{T}})$

$$\mathrm{P}_2\mathrm{H}^{-1} = [\mathrm{A}|\mathrm{a}]\begin{bmatrix} k^{-1}I & 0 \\ k^{-1}\mathrm{v}^{\mathrm{T}} & k \end{bmatrix} = [k^{-1}(\mathrm{A} + \mathrm{av}^{\mathrm{T}}) \mid k\mathrm{a}] = [\widetilde{\mathrm{A}}|\tilde{\mathrm{a}}] = \mathrm{P}_2{}'$$

(22-15=7, ok)

useful fact learnt from the ↓ proof

if

cameras $\mathrm{P}_1 = [\mathrm{I}|0]$ and $\mathrm{P}_2 = [\mathrm{A}|\mathrm{a}]$ are compatible with a Fundamental matrix F_{12}

then

also cameras $\mathbf{P_1}' = [\boldsymbol{I}|\mathbf{0}]$ and $\mathbf{P_2}' = [\,\mathbf{A+av}^{\boldsymbol{T}}\big|\boldsymbol{\lambda a}]$
are compatible with F_{12} , for any vector v and scalarλ

Most general canonical cameras given $\boldsymbol{F}$

A possible choice of cameras compatible with a given F:

$$\mathrm{P}_1 = [\mathrm{I}|0] \quad \mathrm{P}_2 = [[\mathrm{e}_2]_\times F \mid \mathrm{e}_2], \quad (\mathrm{e}_2 = \mathrm{LNS}\ F)$$

since the fundamental matrix of $(\mathrm{P}_1, \mathrm{P}_2)$ is F: in fact for canonical cameras is

$$F = [\mathrm{m}]_\times \mathrm{M} = [\mathrm{e}_2]_\times \mathrm{M} = [\mathrm{e}_2]_\times [\mathrm{e}_2]_\times F = -\|\mathbf{e_2}\|^2 \boldsymbol{F} \Leftrightarrow F$$

However, $\mathrm{M} = [\mathrm{e}_2]_\times F$ is singular → camera P_2 is degenerate

Canonical representation:

from theorem proof with $A = [\mathrm{e}_2]_\times F$, $\mathrm{a} = \mathrm{e}_2$, divide by *k* and take $\lambda = 1/k^2$

$$\mathrm{P}_1 = [\mathrm{I}|0] \quad \mathrm{P}_2 = [[\mathrm{e}_2]_\times F + \mathrm{e}_2 \mathrm{v}^\mathrm{T} | \lambda \mathrm{e}_2]$$

is the **most general camera pair** in canonical form, that is compatible with F

Most general canonical cameras given $\boldsymbol{F}$

A possible choice of cameras compatible with a given F:

$$P_1 = [I|0] \quad P_2 = [[e_2]_\times F \mid e_2], \quad (e_2 = \mathrm{LNS}\ F)$$

since the fundamental matrix of (P_1, P_2) is F: in fact for canonical cameras is

$$F = [m]_\times M = [e_2]_\times M = [e_2]_\times [e_2]_\times F = -\|\mathbf{e_2}\|^2 \boldsymbol{F} \Leftrightarrow F$$

see next slides

However, $M = [e_2]_\times F$ is singular → camera P_2 is degenerate

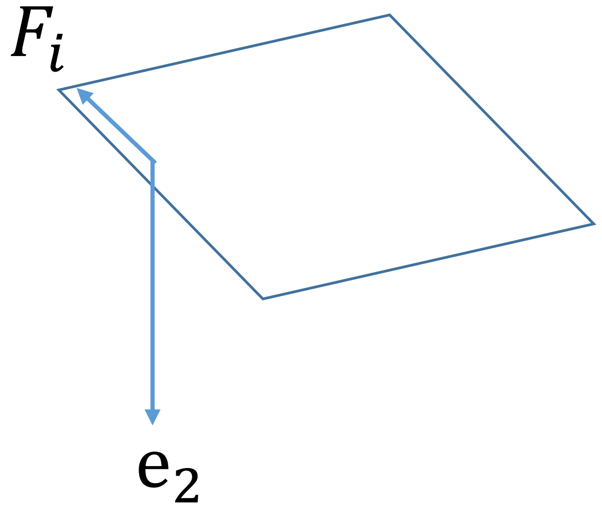
Canonical representation:

from theorem proof with $A = [e_2]_\times F$, $a = e_2$, divide by *k* and take $\lambda = 1/k^2$

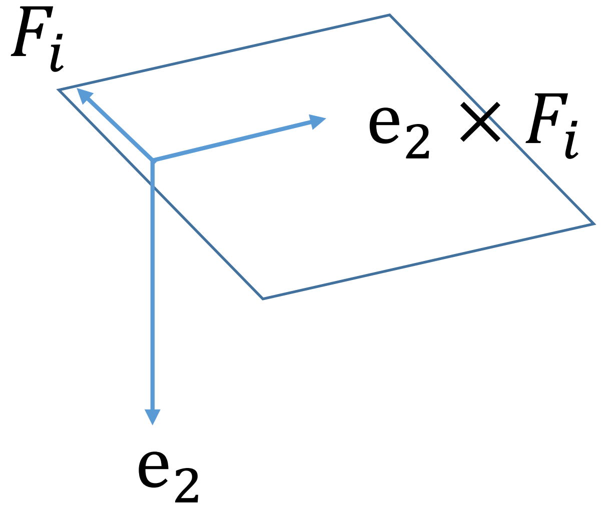
$$P_1 = [I|0] \quad P_2 = [[e_2]_\times F + e_2 v^T | \lambda e_2]$$

is the **most general camera pair** in canonical form, that is compatible with F

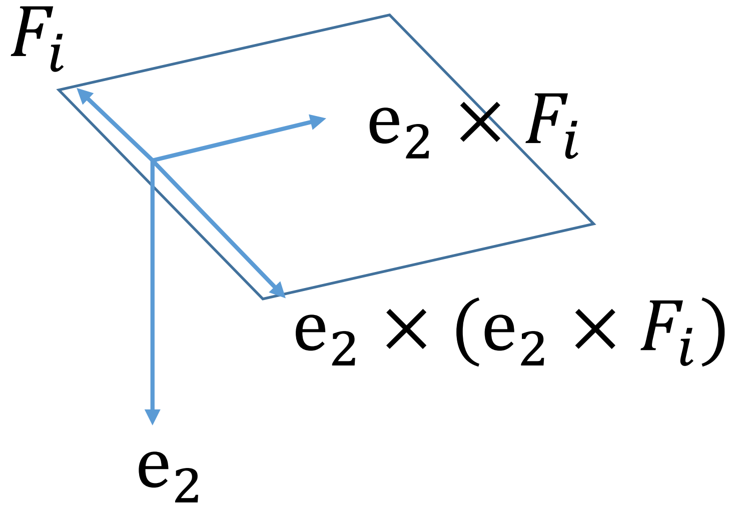
e_2 is LNS() → $\mathrm{e}_2^T\ \mathrm{F} = \mathrm{e}_2^T\ [F_1 \quad F_2 \quad F_3] = 0 \rightarrow \mathrm{e}_2 \perp F_i$



e_2 is LNS() → $e_2^T\,F = e_2^T\,[F_1 \quad F_2 \quad F_3] = 0 \rightarrow e_2 \perp F_i$



e_2 is LNS() → $\mathrm{e}_2^T \,\mathrm{F} = \mathrm{e}_2^T \,[F_1 \quad F_2 \quad F_3] = 0 \rightarrow \mathrm{e}_2 \perp F_i$



$$[\mathrm{e}_2]_\times [\mathrm{e}_2]_\times F_i = -\|\mathrm{e}_2\|^2 F_i$$

$$\downarrow$$

$$\boxed{[\mathrm{e}_2]_\times [\mathrm{e}_2]_\times \mathrm{F} = -\|\mathrm{e}_2\|^2 \mathrm{F}}$$

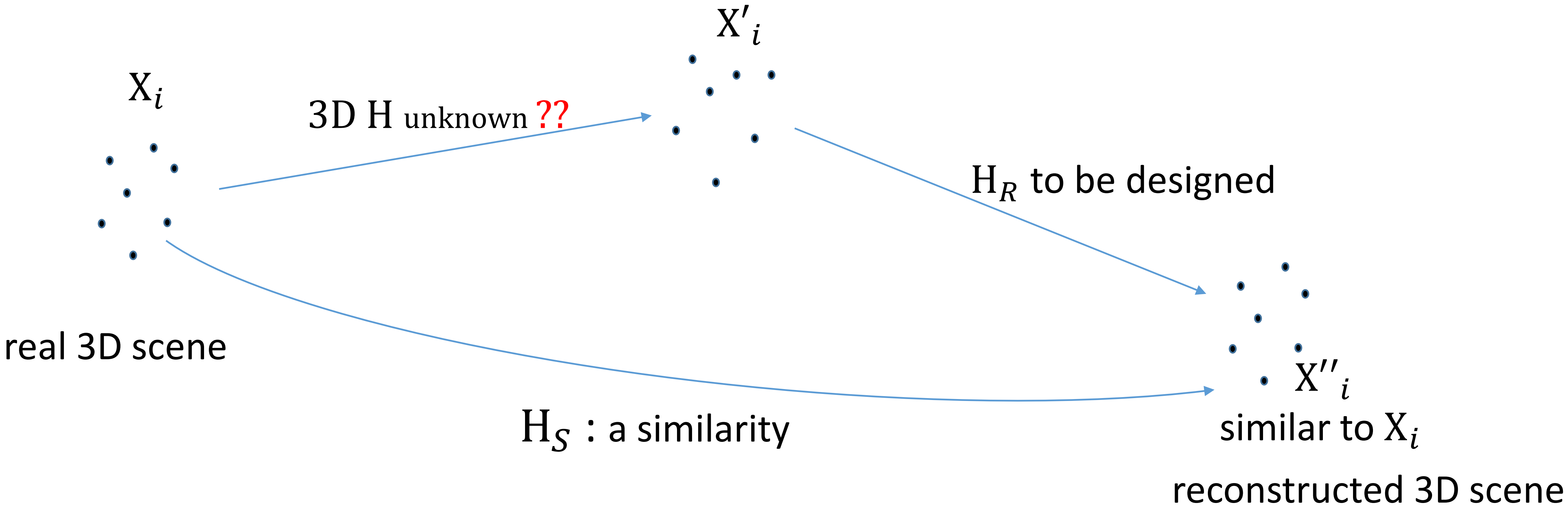
Most general canonical cameras given F

$$e_2 = \text{LNS } F$$

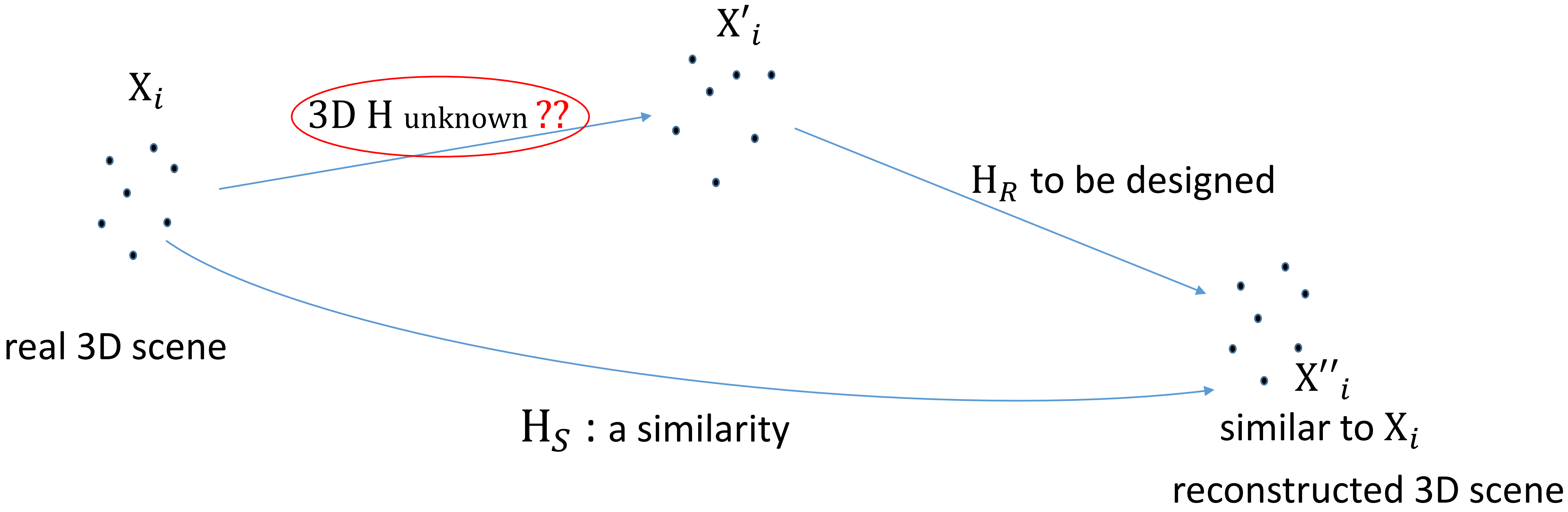
$$P_1 = [I|0] \quad P_2 = [[e_2]_\times F + e_2 v^T | \lambda e_2]$$

is the **most general camera pair** in canonical form, that is compatible with F

3D shape reconstruction from images



3D shape reconstruction from images



3D reconstruction problem

A step was left behind, while studying 3D projective Geometry:

- How to construct an observation of the 3D scene, that is a 3D projectivity of it?
- Take 2D images of 3D scene points X_i and find correspondences $\mathrm{x}_1^i \leftrightarrow \mathrm{x}_2^i$
- Find fundamental matrix(es) F_{12} by solving $\begin{cases} \mathrm{x}_2^{Ti}\mathrm{F}_{12}\mathrm{x}_1^i = 0 \\ i = 1 \dots N \end{cases}$ for F_{12}
- Take a tentative pair of cameras compatible with F_{12}

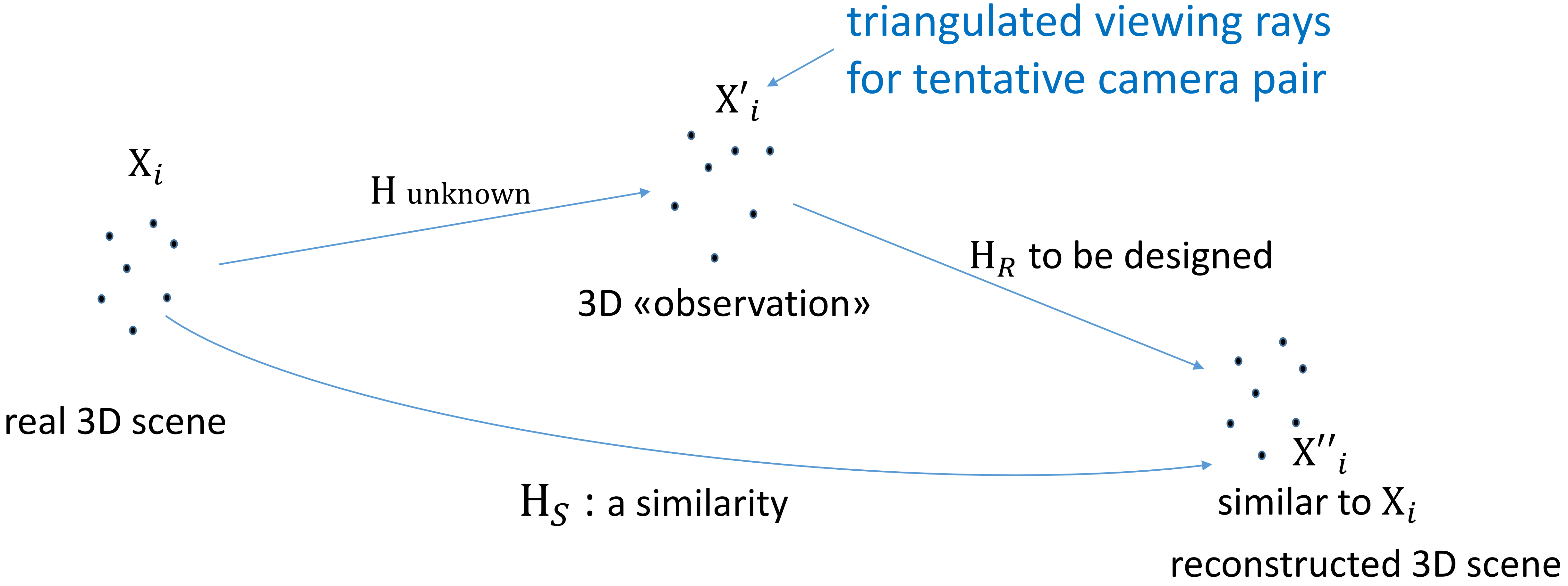
$$\mathrm{P}_1 = [\mathrm{I} \,|\, 0] \quad \mathrm{P}_2 = [[\mathrm{e}_2]_\times \mathrm{F}_{12} + \mathrm{e}_2\mathrm{v}^\mathrm{T} \,|\, \lambda \mathrm{e}_2]$$

where e_2 is the LNS of F_{12}, λ and v are any nonzero scalar and vector

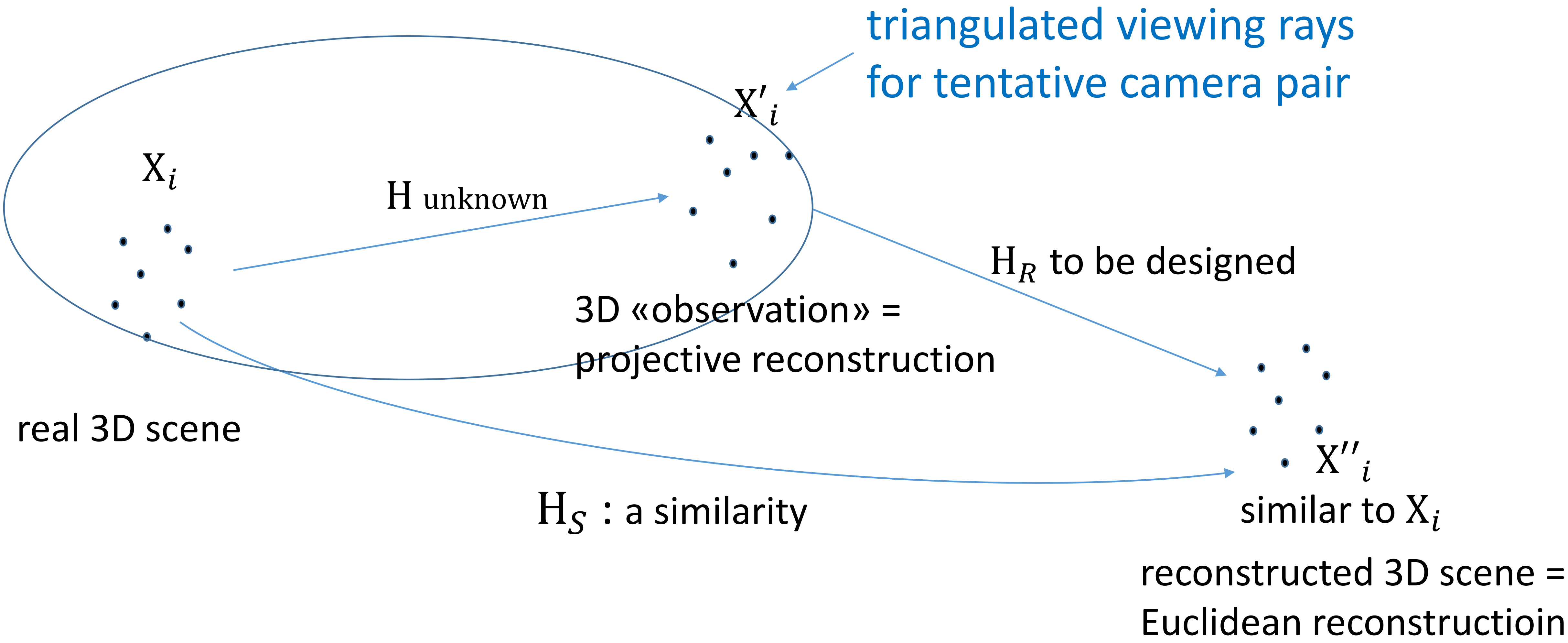
- Triangulate the pairs of viewing rays associated to $\mathrm{x}_1^i \leftrightarrow \mathrm{x}_2^i$ through cameras $\mathrm{P}_1\,\mathrm{P}_2$
- Take X_i' as the points resulting from the triangulations:

THESE X_i' ARE AN UNKNOWN PROJECTIVE TRANSFORMATION OF THE TRUE 3D POINTS X_i

3D shape reconstruction from images

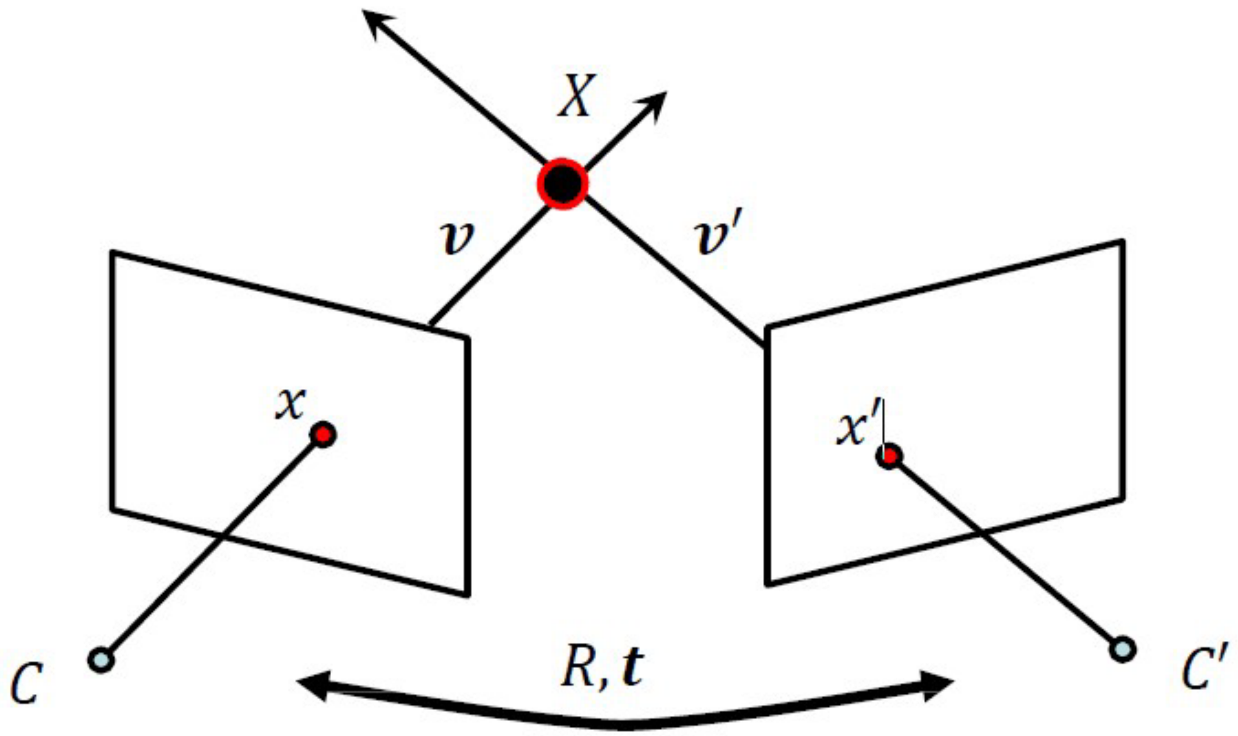


3D shape reconstruction: first step



scenario 2: **calibrated** structure from motion

x ←→x’ known; **R,t unknown**; K and K’ known;



use **epipolar constraint** to estimate R, t; → compute viewing rays v, v’

→ triangulation: $\boldsymbol{X} = \boldsymbol{v} \cap \boldsymbol{v}'$

Yet another expression of P

$$P = [\mathrm{M} \quad \mathrm{m}] \quad O = \begin{bmatrix} o \\ 1 \end{bmatrix} = \mathrm{RNS(P)}$$

$$\mathrm{M} = \mathrm{KR}_{cam\to world} \quad o = t_{world\to cam}$$

From

$$PO = [\mathrm{M} \quad \mathrm{m}] \begin{bmatrix} t_{world\to cam} \\ 1 \end{bmatrix} = \mathrm{M}t_{world\to cam} + \mathrm{m} = 0$$

is $m = -\mathrm{M}t_{world\to cam} = -\mathrm{KR}_{cam\to world} t_{world\to cam}$

But since

$$t_{cam\to world} = -\mathrm{R}_{cam\to world} t_{world\to cam},$$

Then $m = Kt_{cam\to world}$

Hence $P = [\mathrm{M} \quad \mathrm{m}] = [K\mathrm{R}_{cam\to world} \quad Kt_{cam\to world}]$

$$P = K[R \quad t]$$

where $R \stackrel{\text{def}}{=} \mathrm{R}_{cam\to world}$ and $t \stackrel{\text{def}}{=} t_{cam\to world}$

3D shape reconstruction from calibrated images

Calibrated cameras observe a same scene from unknown relative positions

Change image coordinates

$$\mathrm{x}_1 = \mathrm{K}_1\hat{\mathrm{x}}_1$$
$$\mathrm{x}_2 = \mathrm{K}_2\hat{\mathrm{x}}_2$$

$x_1 = P_1X \rightarrow K_1\hat{\mathrm{x}}_1 = K_1[\mathrm{R}_1 \quad \mathrm{t}_1]X \rightarrow \quad \hat{\mathrm{x}}_1 = [\mathrm{R}_1 \quad \mathrm{t}_1]X$

and, similarly, $\quad \hat{\mathrm{x}}_2 = [\mathrm{R}_2 \quad \mathrm{t}_2]X$

Place world reference = camera-1 reference: $R_1 = I$ and $t_1 = 0$ (and call $R \stackrel{\text{def}}{=} \mathrm{R}_2$, $t \stackrel{\text{def}}{=} t_2$)

$\hat{\mathrm{x}}_1 = \underbrace{[I \quad 0]}_{\hat{P}_1}X$ and $\hat{\mathrm{x}}_2 = \underbrace{[R \quad t]}_{\hat{P}_2}X$

Essential matrix of two calibrated images (Longuet-Higgins 1980)

$$\hat{\mathrm{x}}_2{}^T E_{12} \hat{\mathrm{x}}_1 = 0$$

$$E_{12} = [\hat{\mathrm{e}}_2]_\times \widehat{\mathrm{M}}_2 \widehat{\mathrm{M}}_1^{-1}$$

The epipole is the image projection of the center of the first camera center (the origin) onto the second camera $\hat{P}_2 = [R \quad t]$: $\hat{\mathrm{e}}_2 = [R \quad t] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = t$

thus $\boxed{E_{12} = [t]_\times \mathrm{R}}$

Relationship between Fundamental matrix F and Essential matrix E

$$\mathrm{x}_2^T F_{12} \mathrm{x}_1 = \hat{\mathrm{x}}_2{}^T \mathrm{K}_2^T F_{12} \mathrm{K}_1 \hat{\mathrm{x}}_1 = 0 = \hat{\mathrm{x}}_2{}^T E_{12} \hat{\mathrm{x}}_1 \dashrightarrow \mathrm{K}_2^T F_{12} \mathrm{K}_1 = E_{12}$$

Derive R and t first, then triangulate

- **Property**: any 3x3 skew-symmetric matrix S can be written as

$$S = hU\begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}U^T = hUZU^T$$ where $Z \triangleq \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ and U is an orthogonal matrix

Let $W \triangleq \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ be another orthogonal matrix: → it is $Z = \pm\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}W$

Thus, since matrix $[t]_\times$ is skew-symmetric, $[t]_\times = \pm UZU^T = \pm U\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}WU^T$

Therefore $E_{12} = [t]_\times \mathrm{R} = \pm U\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}\underbrace{WU^T\mathrm{R}}_{V^T} = \pm U\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}V^T$ with V also orthogonal

this is svd(E) !

svd(E_{12}) $= \pm U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} W U^T \mathrm{R} = \pm U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T$ gives U and V as output matrixes

Two solutions: + sign and – sign

+ sign: → $V^T = W U^T R$ therefore $\mathrm{R}_+ = U W^T V^T = U \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} V^T$

$-$ sign: $-Z = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} W^T$ → $-E_{12} = U \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} W^T U^T \mathrm{R}$ → $V^T = W^T U^T \mathrm{R}$

→ second solution $\mathrm{R}_- = U W V^T = U \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} V^T$

From $E_{12} = [t]_\times \mathrm{R}$, for each solution for R, there are two solutions for t: $[t]_\times = \pm E_{12} \mathrm{R}^T$

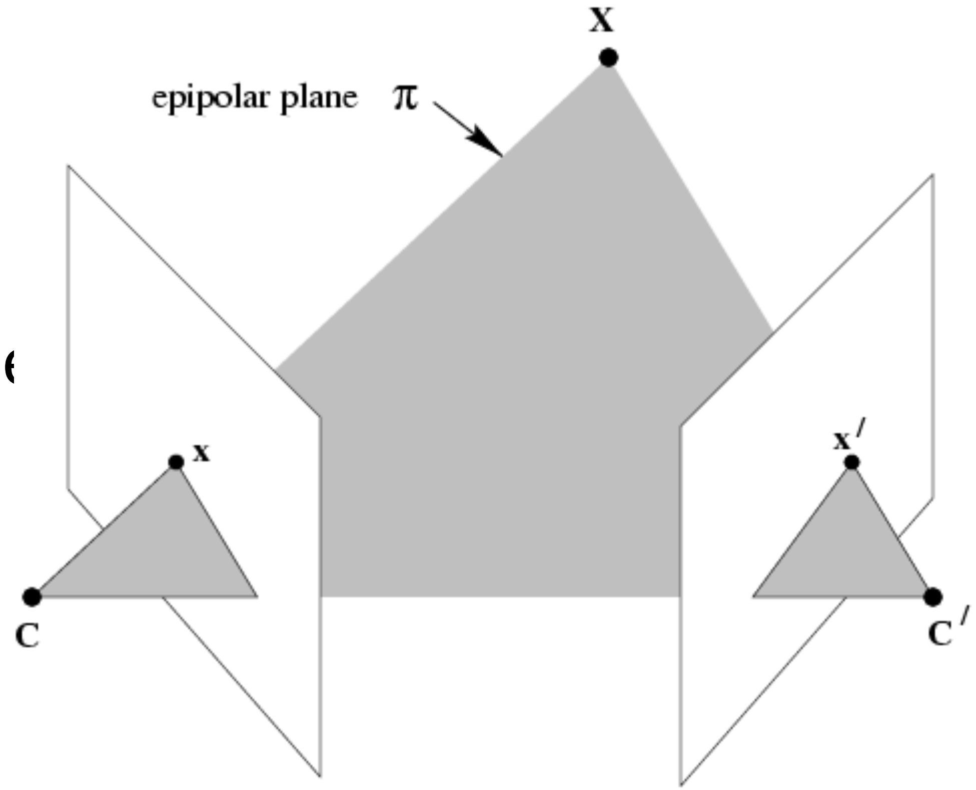
Once R and t have been derived, the relative pose of the two cameras is known, together with K_1 and K_2 → 3D reconstruction by TRIANGULATION

Modulo scale-translation (module of $\boldsymbol{t}$):

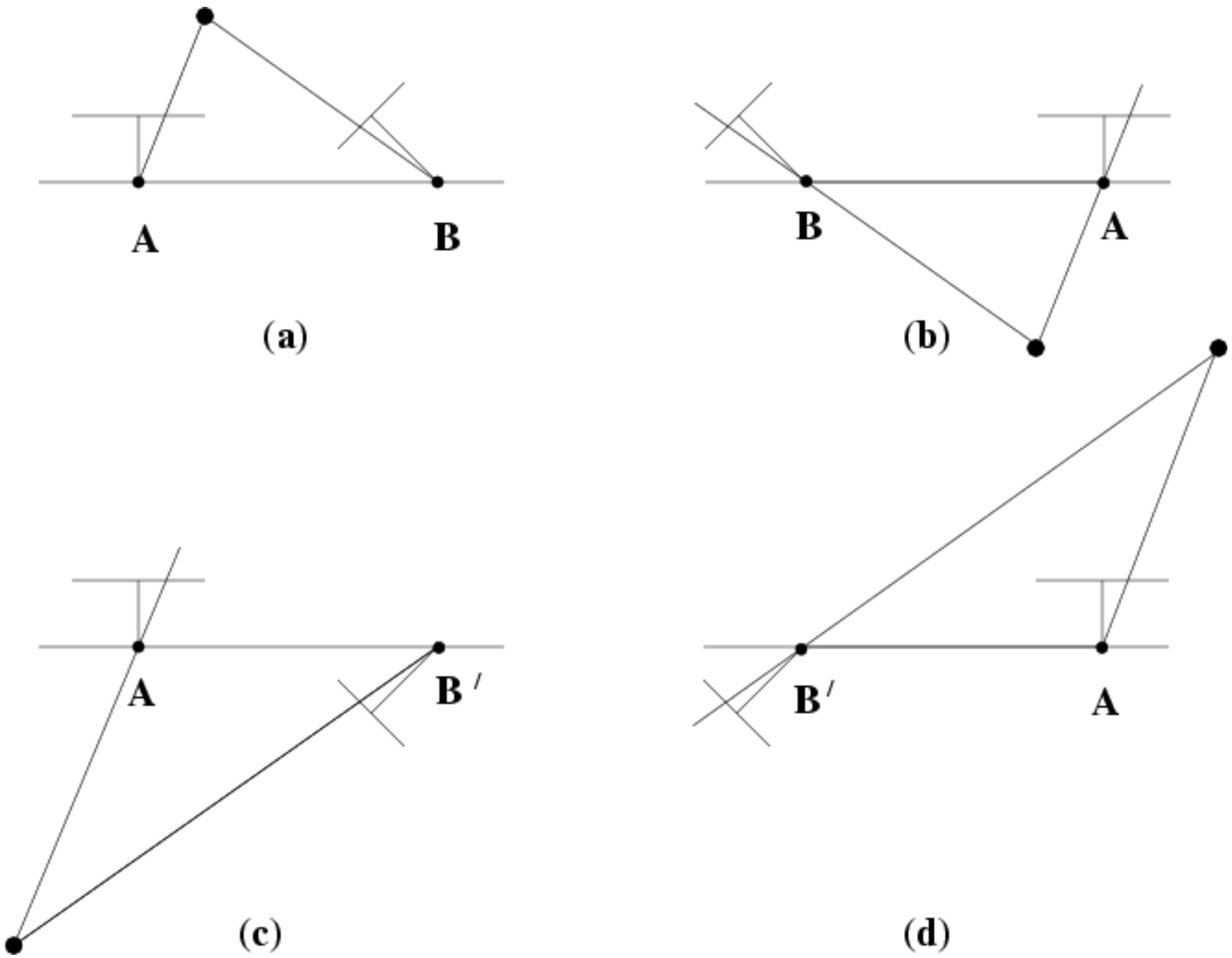
Scale, or module of translation, can not be determined from just images

→Additional information needed, such as, e

e.g., a size in the scene or $|\boldsymbol{t}|$



Four possible reconstructions from E



(**only one solution** where the triangulated point is **in front** of **both cameras**)

3D reconstruction

from multiple images

Three questions:

(i) **Correspondence geometry:** Given an image point x_1 in the first image, how does this constrain the position of the corresponding point x_2 in the second image?

(ii) **Camera geometry (motion):** Given a set of corresponding image points $\{\mathrm{x}_{1i} \leftrightarrow \mathrm{x}_{2i}\}$, $i = 1, \ldots, n$, what are the cameras P_1 and P_2 for the two views?

(iii) **Scene geometry (structure):** Given corresponding image points $\mathrm{x}_{1i} \leftrightarrow \mathrm{x}_{2i}$ and cameras P_1 , P_2 , what is the position of (their pre-image) X in space?

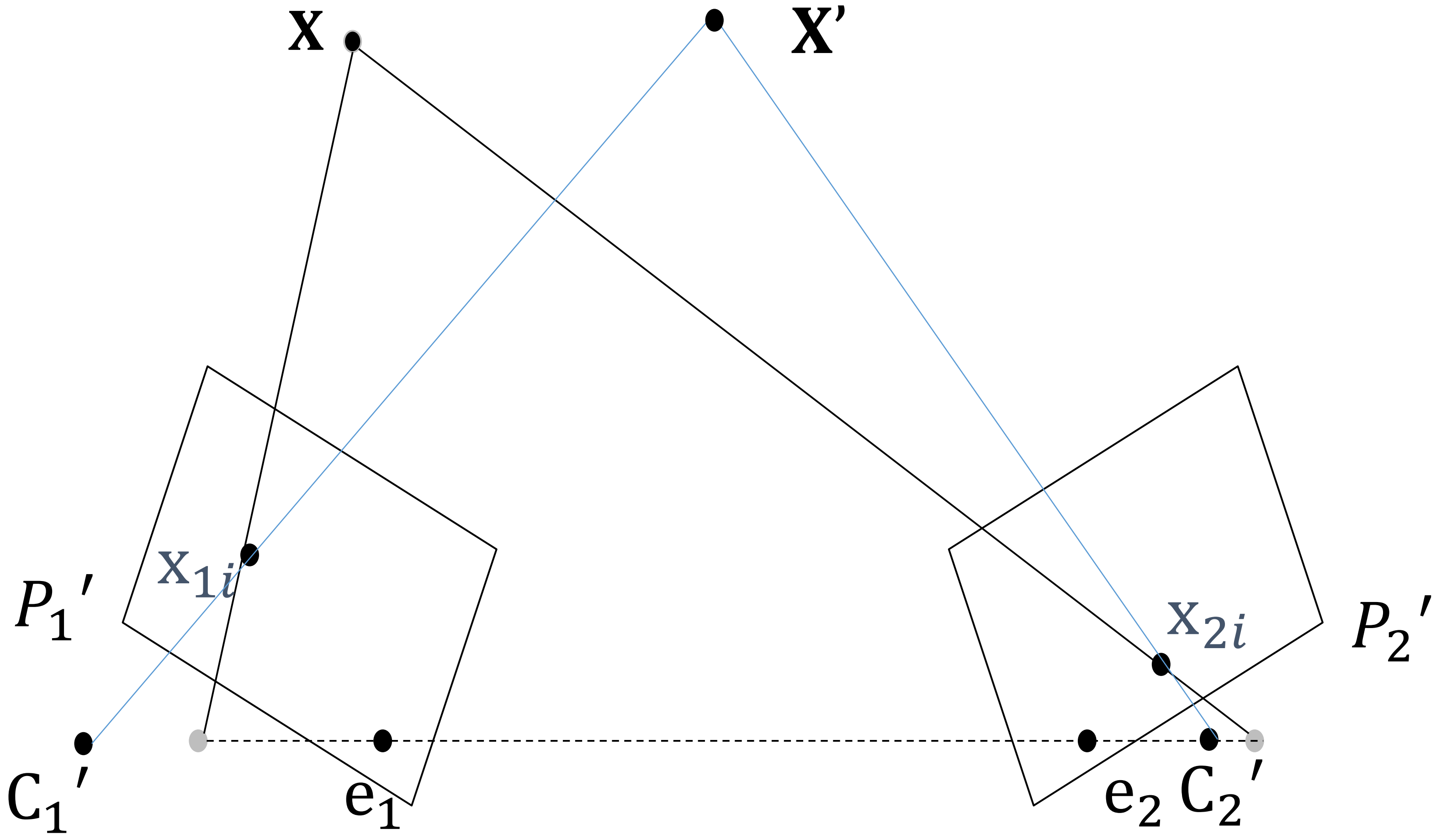
3D reconstruction of cameras and structure

reconstruction problem:

given $\mathrm{x}_{1i} \leftrightarrow \mathrm{x}_{2i}$, compute P_1, P_2 and X_i

$$\mathrm{x}_{1i} = P_1 \mathrm{X}_i \quad \mathrm{x}_{2i} = P_2 \mathrm{X}_i \qquad \text{for all } i$$

without additional information, reconstruction is only possible up to projective ambiguity (projective reconstruction)



X
X'
x_{1i}
P_1'
x_{2i}
P_2'
C_1'
e_1
e_2
C_2'

outline of **projective** reconstruction

(i) Compute F from correspondences
(ii) Compute tentative camera matrices from F
(iii) Compute 3D point for each pair of corresponding points

(i) computation of F

use $\mathrm{x}_{2i}^{T}\mathrm{F}\mathrm{x}_{1i} = 0$ equations, linear in coeff. F

8 points (linear), 7 points (non-linear), 8+ (least-squares)

(ii) computation of tentative camera matrices

use $\mathrm{P}_1 = [\mathrm{I}|0] \quad \mathrm{P}_2 = [[\mathrm{e}_2]_\times \mathrm{F}_{12} + \mathrm{e}_2\mathrm{v}^\mathrm{T}|\lambda \mathrm{e}_2]$

(iii) triangulation

compute intersection of two backprojected rays

3D projective reconstruction

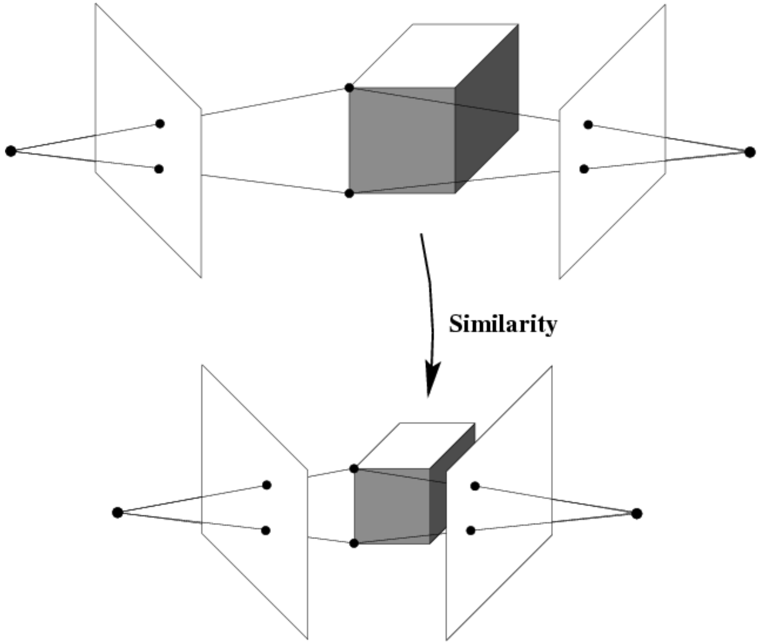
Construct observation of the 3D scene, projective mapping of it

- Take images of 3D scene points X_i and find correspondences $\mathrm{x}_1^i \leftrightarrow \mathrm{x}_2^i$
- Find fundamental matrix(es) F_{12} by solving $\begin{cases} \mathrm{x}_2^{Ti}\mathrm{F}_{12}\mathrm{x}_1^i = 0 \\ i = 1 \dots N \end{cases}$ for F_{12}
- Take a tentative pair of cameras compatible with F_{12} P_1 P_2

  where e_2 is the LNS of F_{12}, λ and v are any nonzero scalar and vector

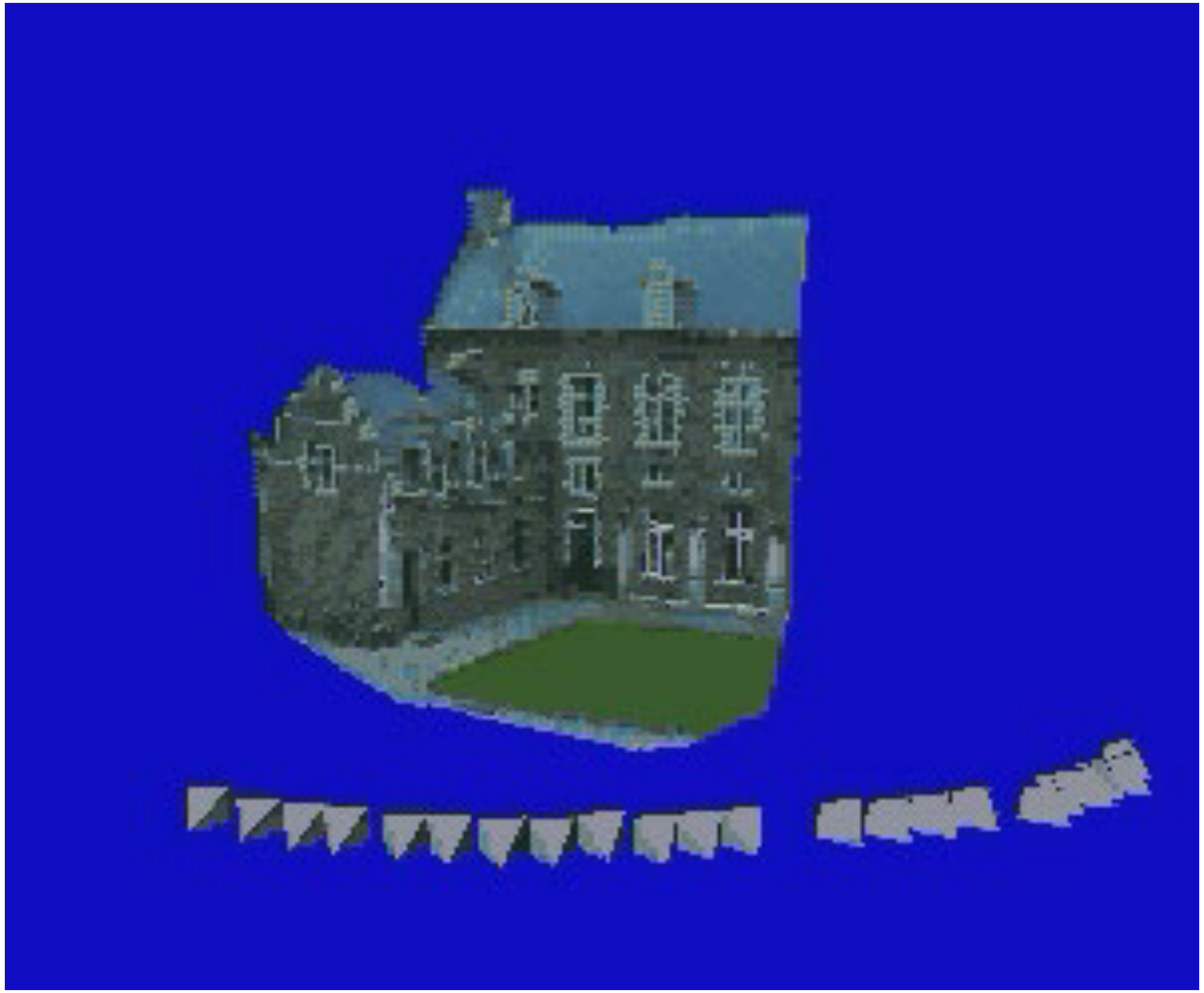
$$\mathrm{P}_1 = [\mathrm{I}|0] \quad \mathrm{P}_2 = [[\mathrm{e}_2]_\times \mathrm{F}_{12} + \mathrm{e}_2 \mathrm{v}^\mathrm{T} | \lambda \mathrm{e}_2]$$

Purpose:
reconstruction ambiguity up to a similarity

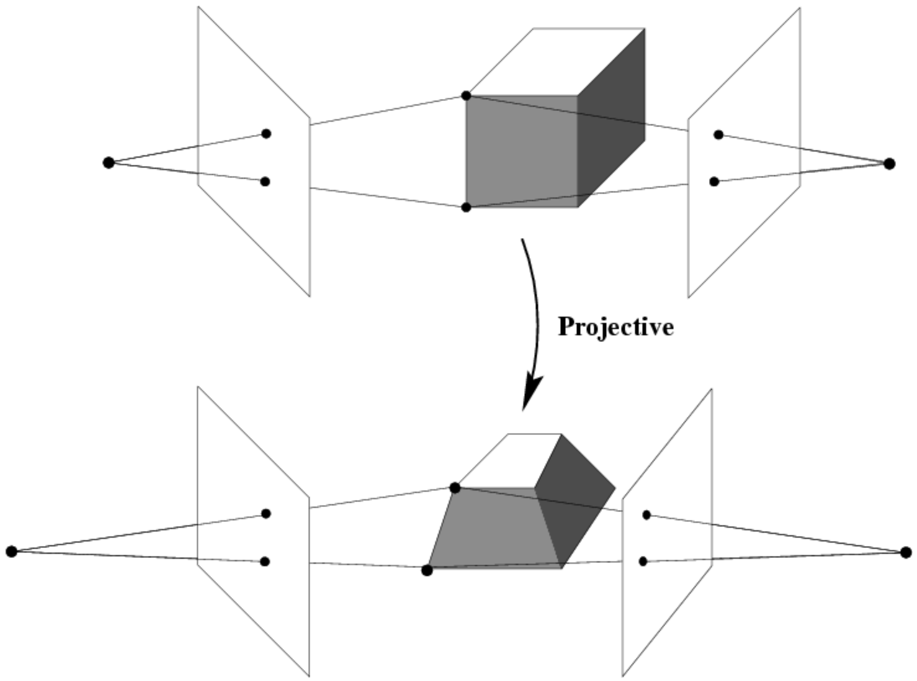


$$\mathbf{x}_i = \mathrm{P}\mathbf{X}_i = \left(\mathrm{P}\mathrm{H}_S^{-1}\right)\left(\mathrm{H}_S\mathbf{X}_i\right)$$

$$\mathrm{P}\mathrm{H}_S^{-1} = \mathrm{K}[\mathrm{R}|\,\mathrm{t}]\begin{bmatrix}\mathrm{R}'^{\mathrm{T}} & -\mathrm{R}'^{\mathrm{T}}\mathrm{t}' \\ 0 & \lambda\end{bmatrix} = \mathrm{K}[\mathrm{R}\mathrm{R}'^{\mathrm{T}}| - \mathrm{R}\mathrm{R}'^{\mathrm{T}}\mathrm{t}' + \lambda\mathrm{t}]$$



Starting point:
reconstruction ambiguity up to a projective mapping



$$x_i = PX_i = \left(PH_P^{-1}\right)\left(H_P X_i\right)$$

Terminology

Pairs of corresponding image points $\mathrm{x}_{1i} \leftrightarrow \mathrm{x}_{2i}$

Original unknown scene $\overline{X_i}$

Projective, affine, similarity reconstruction
= reconstruction that is identical to original up to projective, affine, similarity transformation

Literature: Metric and Euclidean reconstruction
= similarity reconstruction

The projective reconstruction theorem

If a set of point correspondences in two views determine the fundamental matrix uniquely, then the scene and cameras may be reconstructed from these correspondences alone, and any two such reconstructions from these correspondences are projectively equivalent

$$x_{1i} \leftrightarrow x_{2i} \qquad (P_1, P_2, \{X_i\}) \qquad (P_1', P_2', \{X_i'\})$$

$$\underbrace{P_1' = P_1 H^{-1} \quad P_2' = P_2 H^{-1}}_{\text{theorem from last class}} \qquad X_i' = H X_i \qquad \left(\begin{array}{c}\text{except:}\\ F x_{1i} = x^T{}_{2i} F = 0\end{array}\right)$$

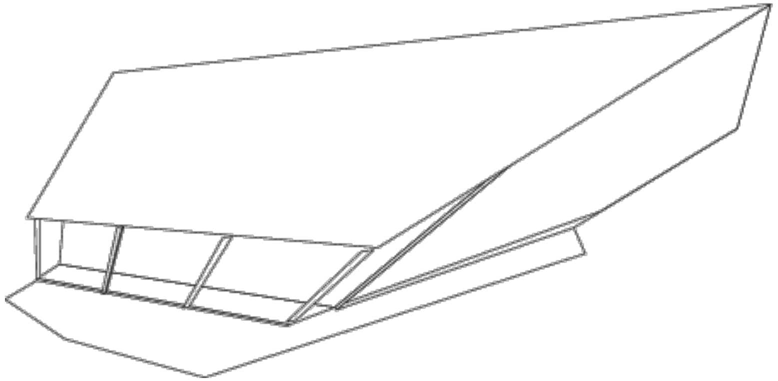
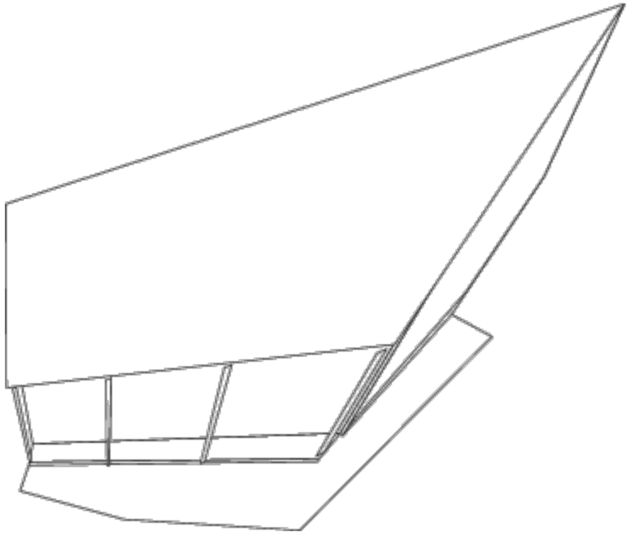
$$P_1' X_i' = P_1'(H X_i) = P_1 H^{-1} H X_i = P_1 X_i = x_{1i} = P_1' X_i'$$

same images idem $P_2 X_i = x_{2i} = P_2' X_i'$

two possibilities: $X_i' = H X_i$, or points along baseline

key result:
allows projective reconstruction from pair of uncalibrated images

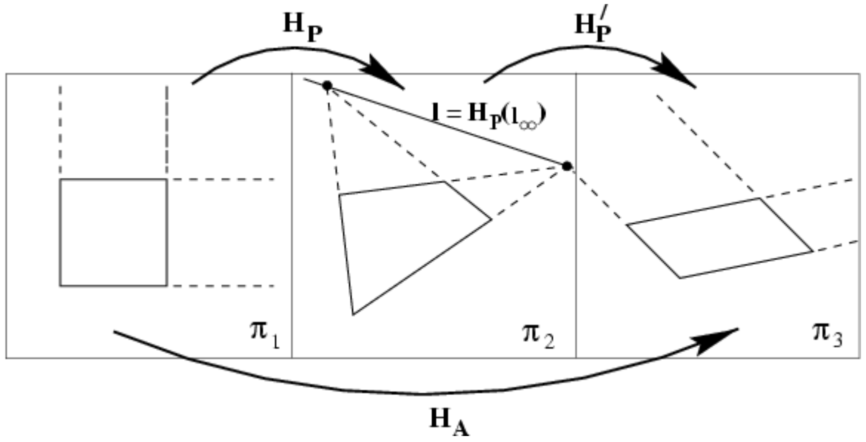
 

Stratified reconstruction

(i) Projective reconstruction
(ii) Affine reconstruction
(iii) Metric reconstruction

Projective to affine

remember 2-D case



A theorem useful for Affine reconstruction

Theorem on an affine invariant

Theorem. *A projective transformation* H *maps the plane at the infinity* $\boldsymbol{\pi}_\infty$ *onto itself, i.e.,* $\boldsymbol{\pi}_\infty$ *is **invariant** under a projective transformation*

$$\Updownarrow$$

H *is **affine***

Application to affine reconstruction

Given 3D points obtained by an uknown projective mapping of an unknown original scene (set of points in 3D space)

$\rightarrow$

the plane $\boldsymbol{\pi}'_\infty$ (i.e. the transformed $\boldsymbol{\pi}_\infty$) is in general $\neq \boldsymbol{\pi}_\infty$!!

Use $\boldsymbol{\pi}'_\infty$ as additional information: if we apply to the transfomed set a second mapping $H_{\boldsymbol{AR}}$ which maps $\boldsymbol{\pi}'_\infty$ back to $\boldsymbol{\pi}_\infty$, we obtain a new, reconstructed model

The composed mapping of $\boldsymbol{\pi}_\infty$ is again $\boldsymbol{\pi}_\infty$ $\rightarrow$

From the theorem, the obtained model is an affine mapping of the original scene

$\rightarrow$

The obtained model is an **affine reconstruction** of the scene

Use of $\boldsymbol{\pi}'_\infty$ in affine reconstruction

….

apply to the transformed point set a second projective mapping $\boldsymbol{H_{AR}}$ that maps $\boldsymbol{\pi}'_\infty$ back to $\boldsymbol{\pi}_\infty$,

how can we find such a projective mapping H_{AR} ?

$$H_{AR} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ & \boldsymbol{\pi}'^{T}_{\infty} & & \end{bmatrix},$$

such that H_{AR} it is invertible

To sum up: affine rectification from $\boldsymbol{\pi}'_\infty$

- Find three points $\mathbf{X_1}, \mathbf{X_2}, \mathbf{X_3}$ that result from mapping three points **at the infinity**
- Fit the transformed plane $\boldsymbol{\pi}'_\infty$ to them: $\boldsymbol{\pi}'_\infty = \mathbf{RNS}\left(\begin{bmatrix} \mathbf{X_1}^T \\ \mathbf{X_2}^T \\ \mathbf{X_3}^T \end{bmatrix}\right)$
- Affine rectification matrix

$$H_{AR} = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ & \boldsymbol{\pi}'_\infty{}^T & & \end{bmatrix}$$

- Affine reconstructed model $M_A = \{\mathbf{X}'_{\boldsymbol{i}}\} = H_{AR}\ \mathrm{given}_{\mathrm{points}} = H_{AR}\ \{\mathbf{X}_{\boldsymbol{i}}\}$

Theorem
A projective transformation $\mathcal{H}$
maps $\boldsymbol{\pi}_\infty$ to itself

$$\Updownarrow$$

$\mathcal{H}$ is an affinity

if π'_∞ known → apply mapping from π'_∞ to π_∞:
combined mapping from real to reconstructed
is an affinity

Projective to affine: additional information π'_∞

$(P_1, P_2, \{X_i\})$ initial cameras, e.g., canonical form

$$\pi'_\infty = (A, B, C, D)^{\mathrm{T}} \mapsto (0,0,0,1)^{\mathrm{T}} = \pi_\infty$$

$$\mathrm{H}^{-\mathrm{T}}\pi'_\infty = (0,0,0,1)^{\mathrm{T}}$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{I}|\mathbf{0} \\ \boldsymbol{\pi}'_\infty \end{bmatrix}$$ (if $D\neq 0$)

theorem says up to a projective transformation,
but a projective transformation with fixed π_∞ is **affine**

new cameras: $P'_1 = P_1 H^{-1}$ and $P'_2 = P_2 H^{-1}$
new reconstruction: $X'_i = H\ X_i$ affine mapping of the true points

Affine reconstruction can be sufficient for some applications,
e.g. mid-point, centroid, parallellism

constraints on π'_∞: examples

Projective to affine

Costraints on π'_∞ from translational motion

points at infinity are fixed for a pure translation

$\Rightarrow$ Reconstruction (triangulation) of $x_i \leftrightarrow x_i$ is on π_∞

$$F = [e']_\times M M^{-1} = [e']_\times = [e]_\times$$

e': vanishing point of motion direction

$$P = [I|0]$$

$$P' = [I|e']$$

Constraints on π'_∞ from scene

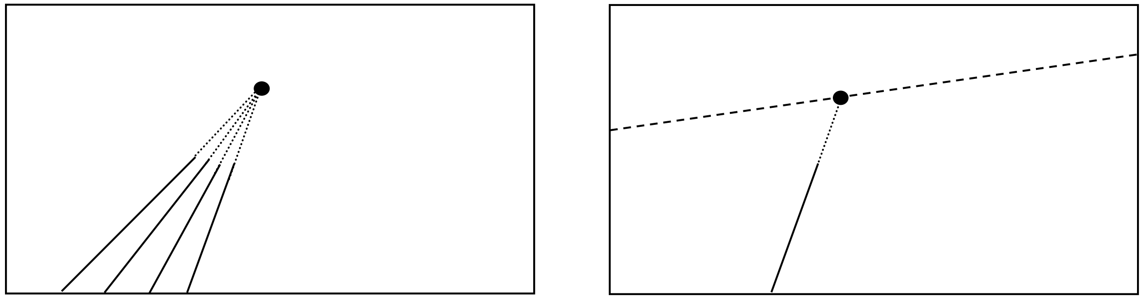
Parallel lines observed in the images

parallel lines intersect at infinity
reconstruction of corresponding vanishing point yields
a point whose **true position** is on the **plane at infinity**

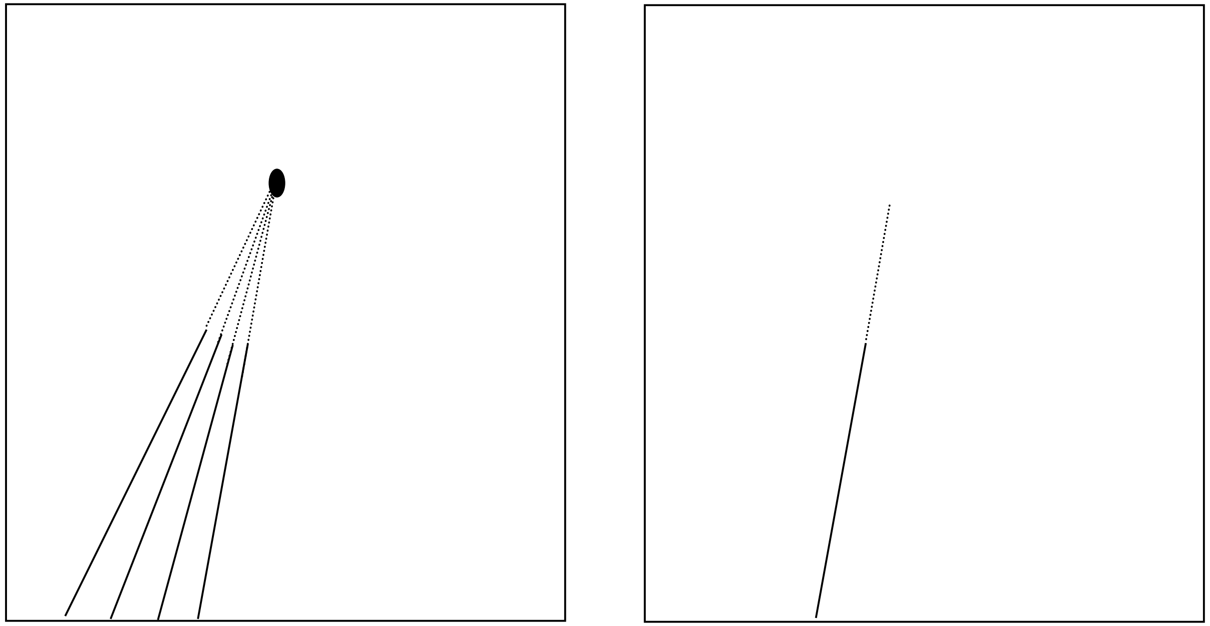
Images of 3 sets of parallel lines allow to uniquely determine π'_∞

remark: in presence of noise determining the intersection of parallel lines is a delicate problem

remark: obtaining vanishing point in one image, and just one line in the other image, can be sufficient

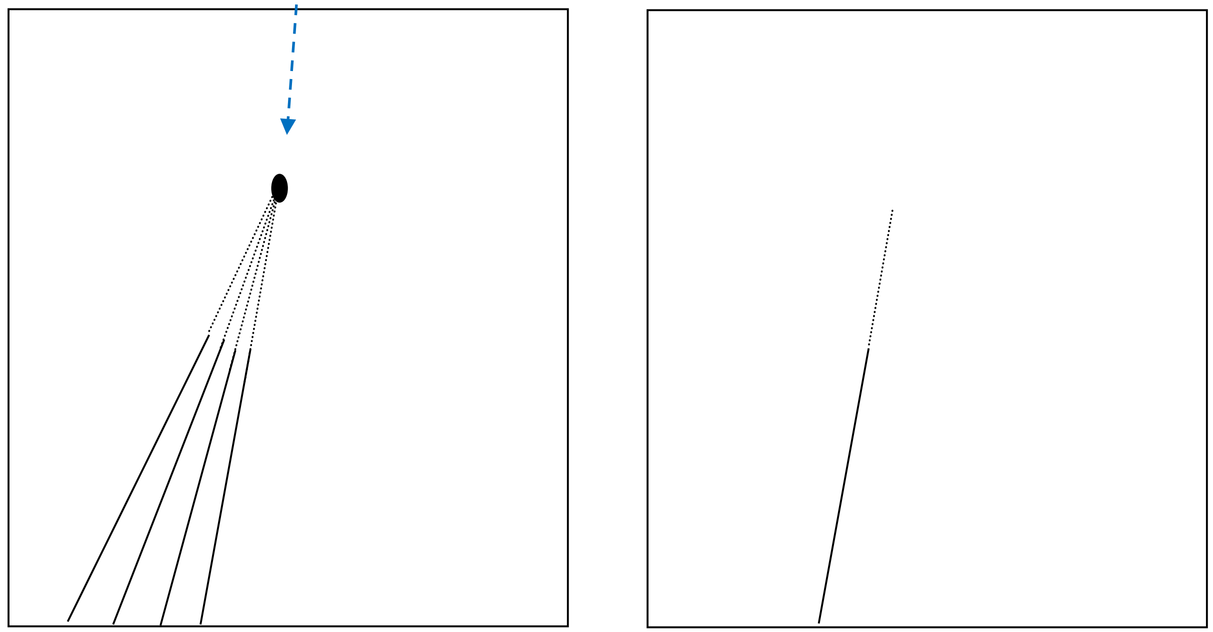


Projective to affine

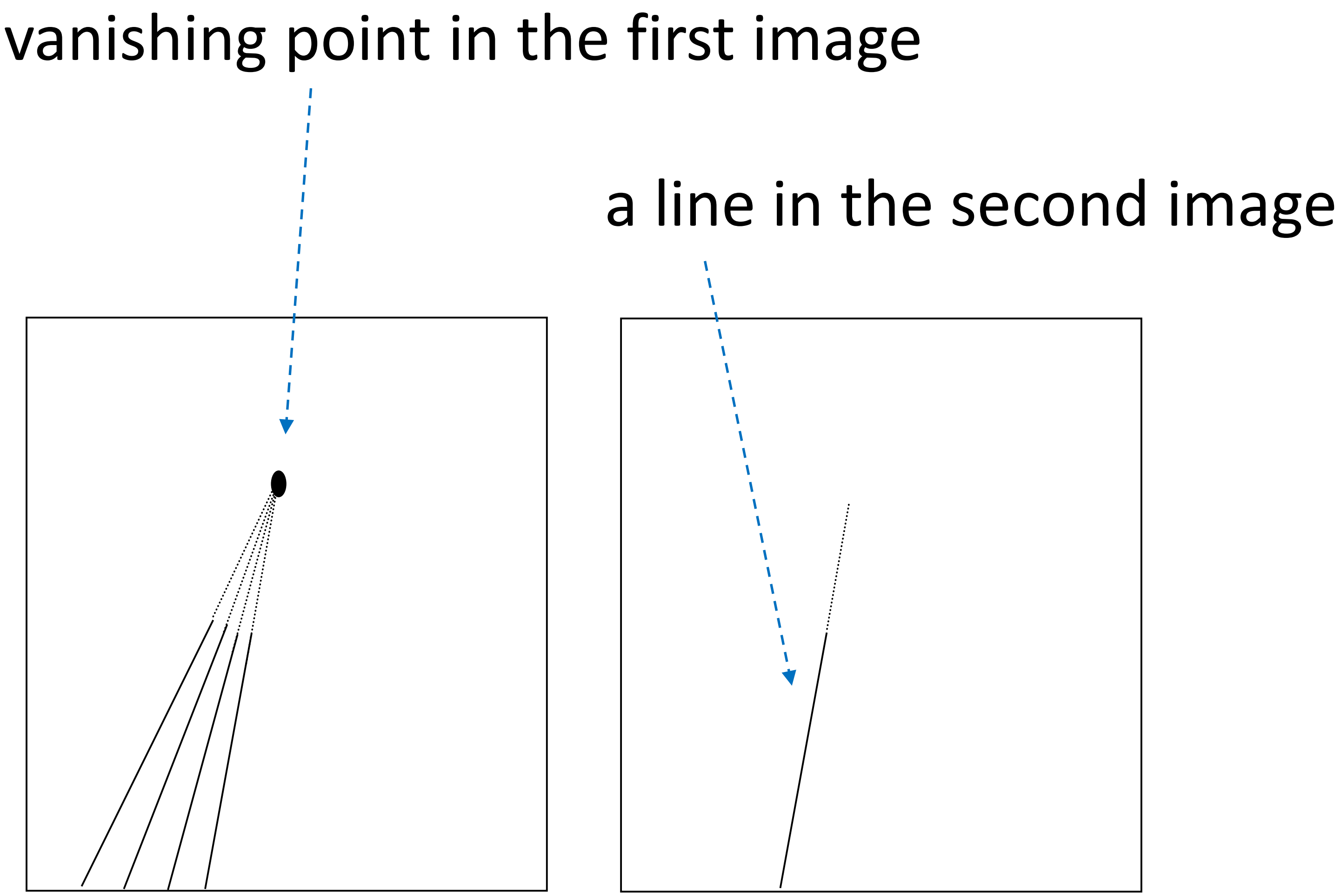


remark: obtaining vanishing point in one image, and just one line in the other image, can be sufficient

vanishing point in the first image



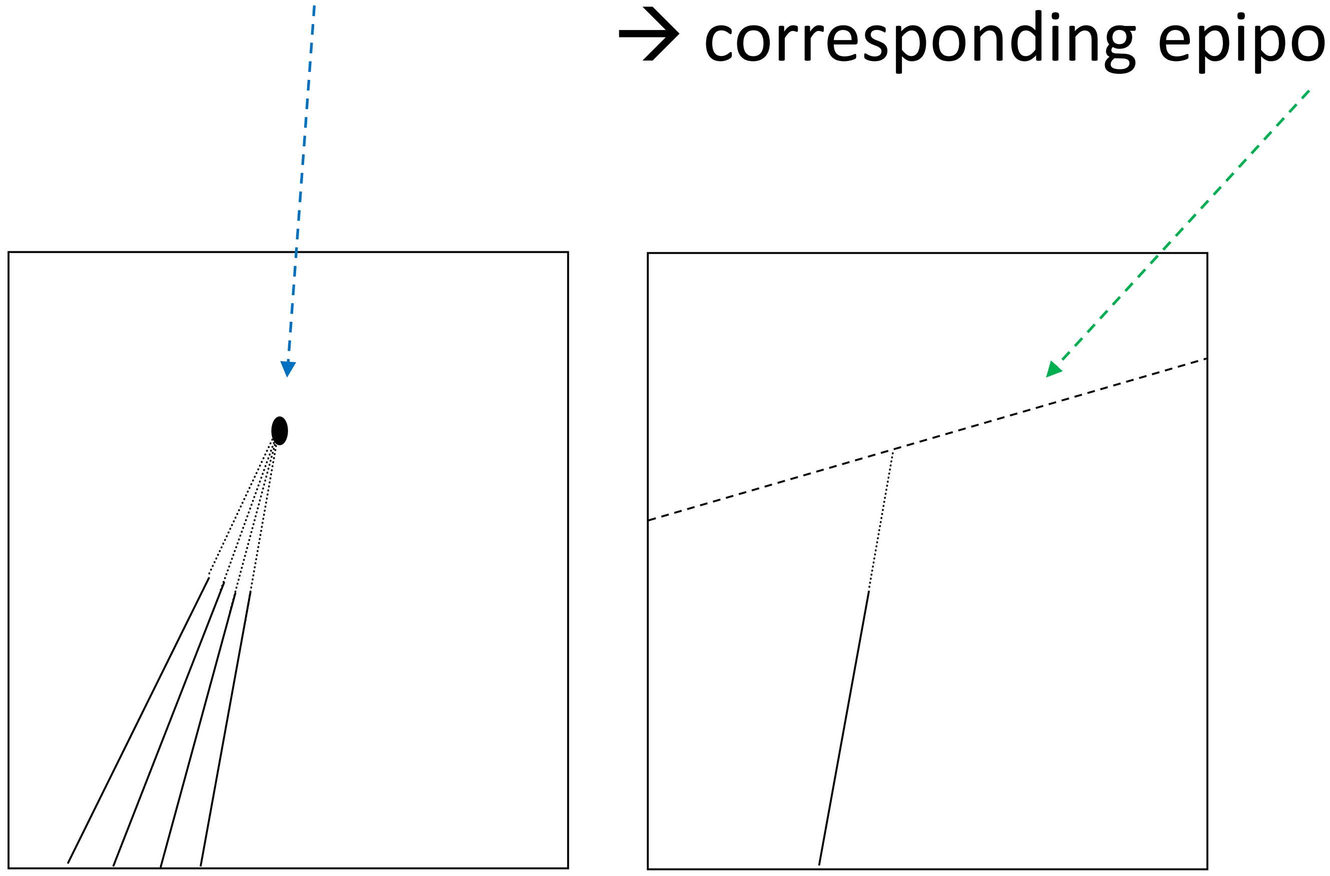
remark: obtaining vanishing point in one image, and just one line in the other image, can be sufficient



remark: obtaining vanishing point in one image, and just one line in the other image, can be sufficient

vanishing point in the first image

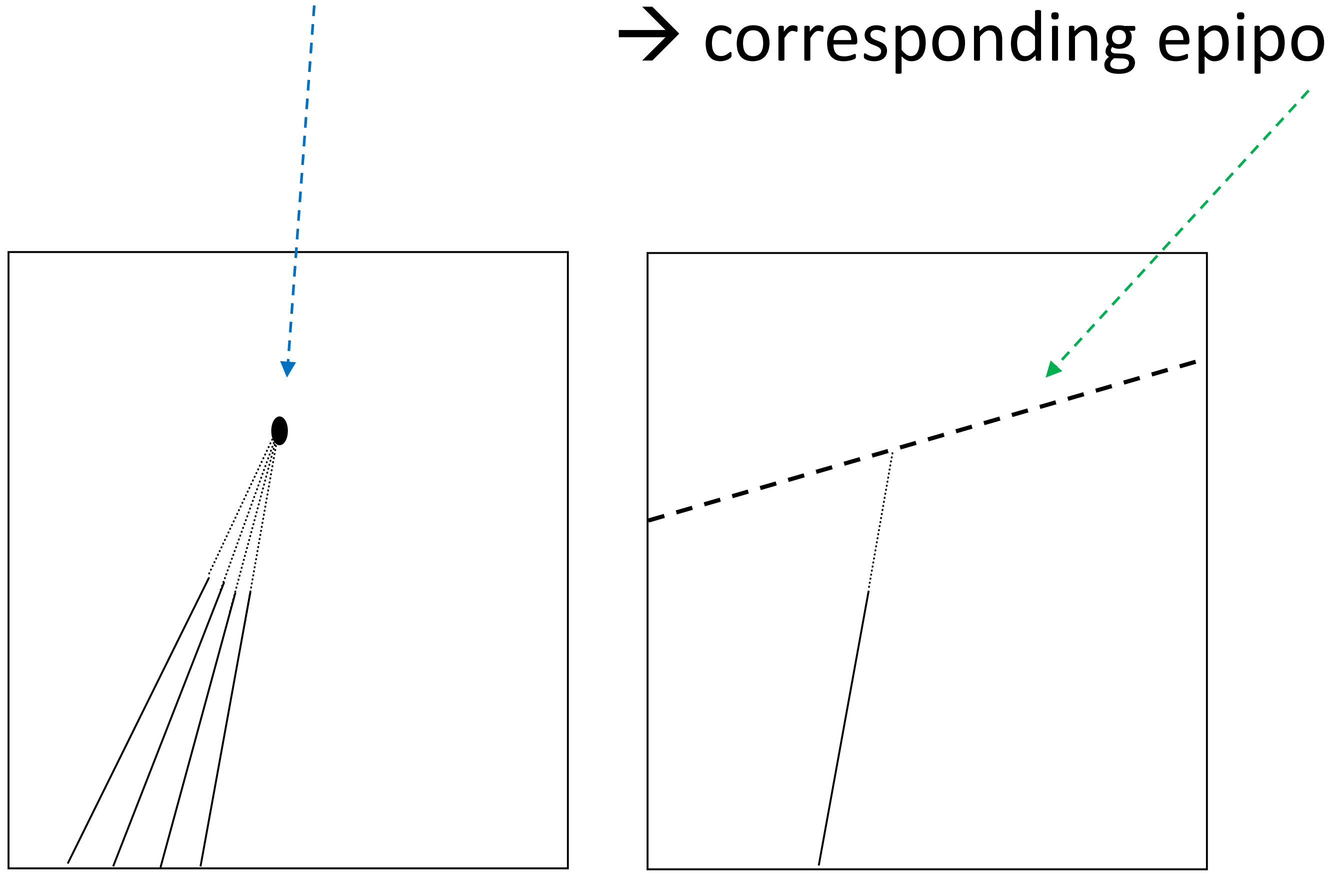
→ corresponding epipolar line



remark: obtaining vanishing point in one image, and just one line in the other image can be sufficient

vanishing point in the first image

→ corresponding epipolar line

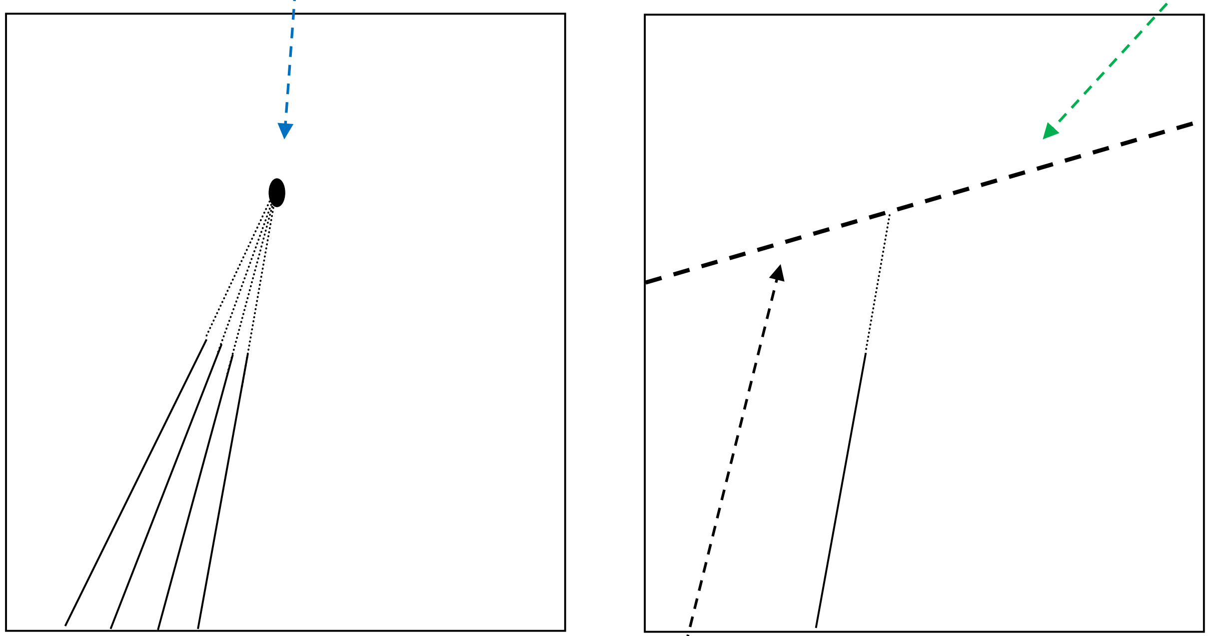


The vanishing point in the second image …

remark: obtaining vanishing point in one image, and just one line in the other image can be sufficient

vanishing point in the first image

→ corresponding epipolar line

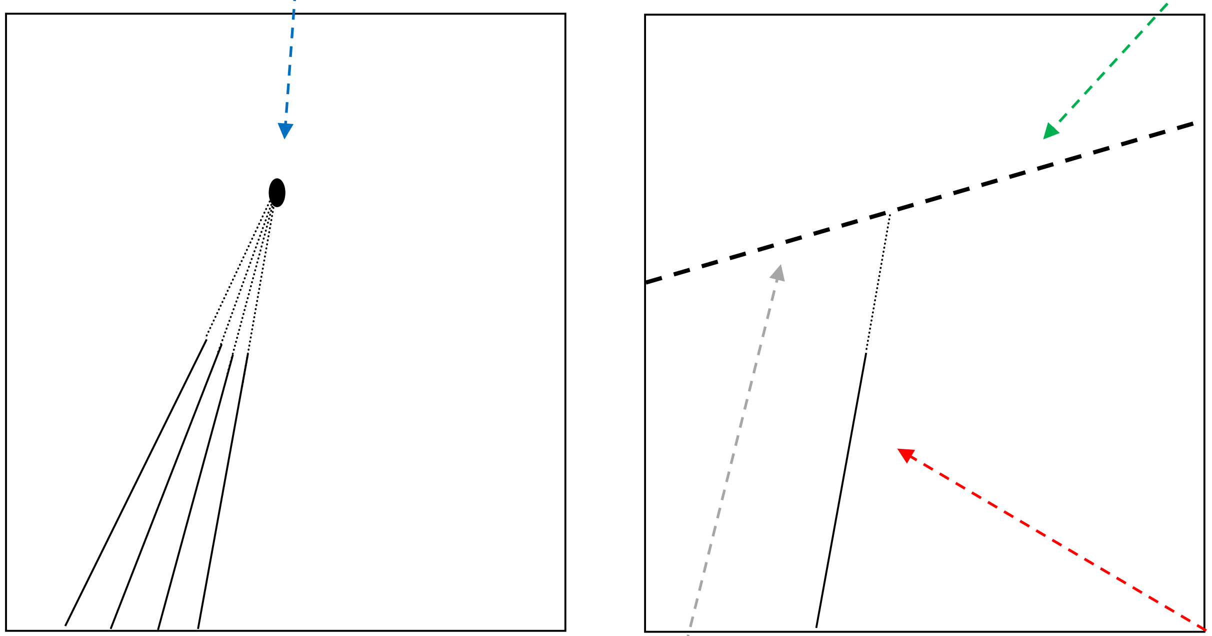


The vanishing point in the second image must lie (i) on this epipolar line

remark: obtaining vanishing point in one image, and just one line in the other image can be sufficient

vanishing point in the first image

→ corresponding epipolar line

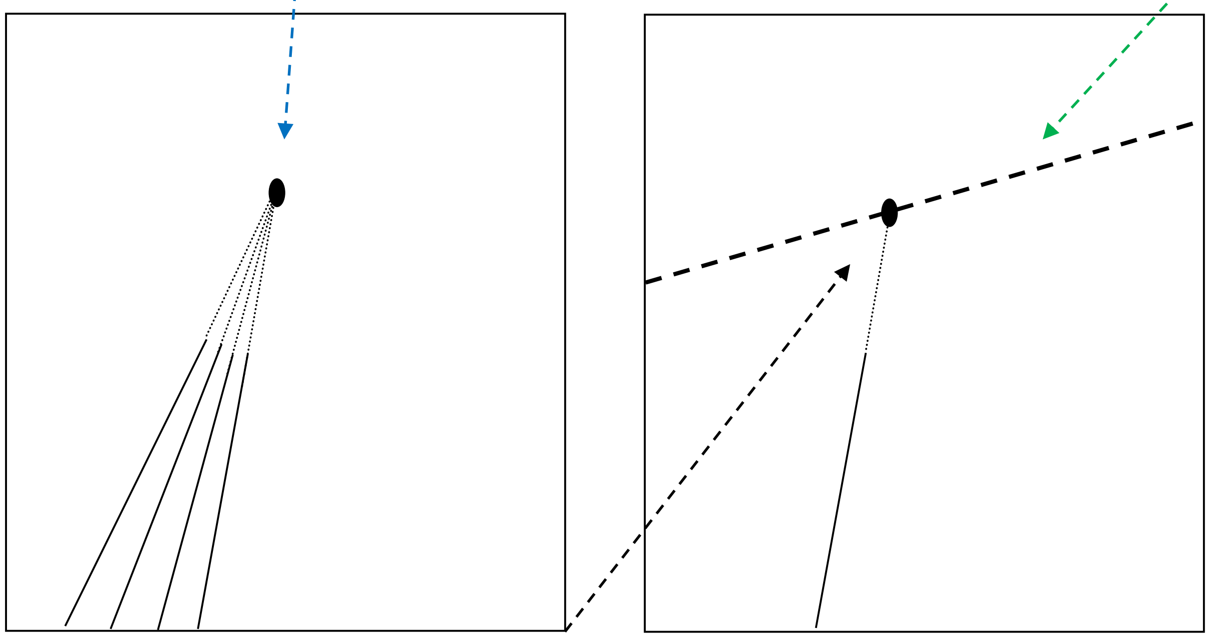


The vanishing point in the second image must lie (i) on this epipolar line, and (ii) on this image line

remark: obtaining vanishing point in one image, and just a line in the other image can be sufficient

vanishing point in the first image

→ corresponding epipolar line

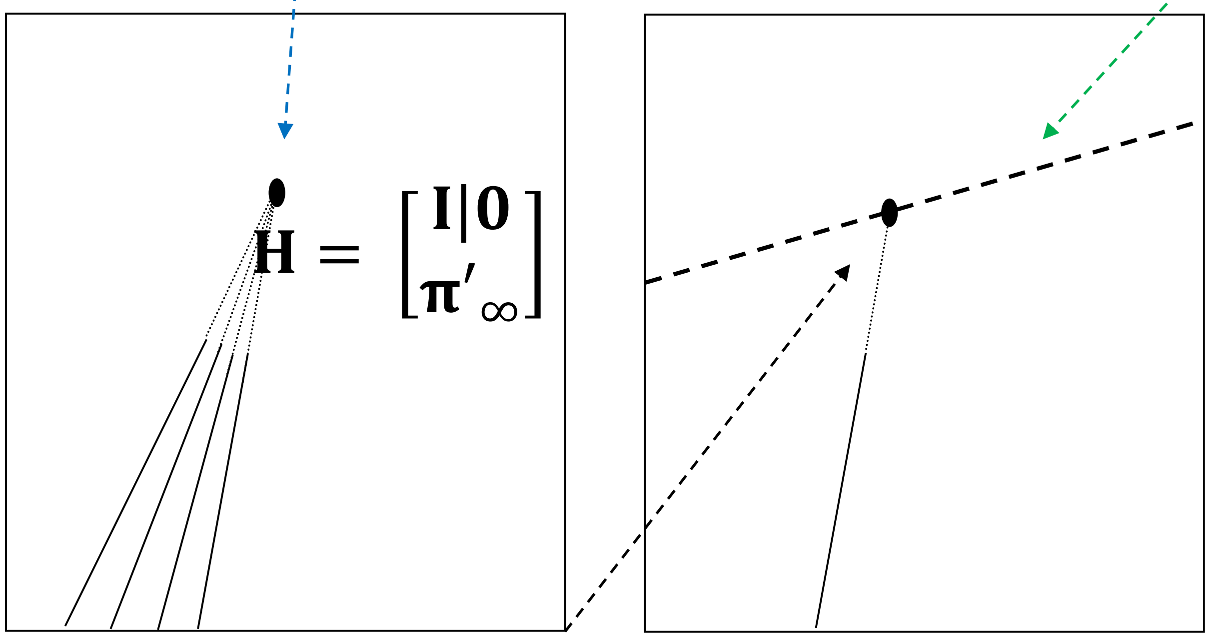


The vanishing point in the second image must lie (i) on this epipolar line, and (ii) on this image line

remark: obtaining vanishing point in one image, and just a line in the other image can be sufficient

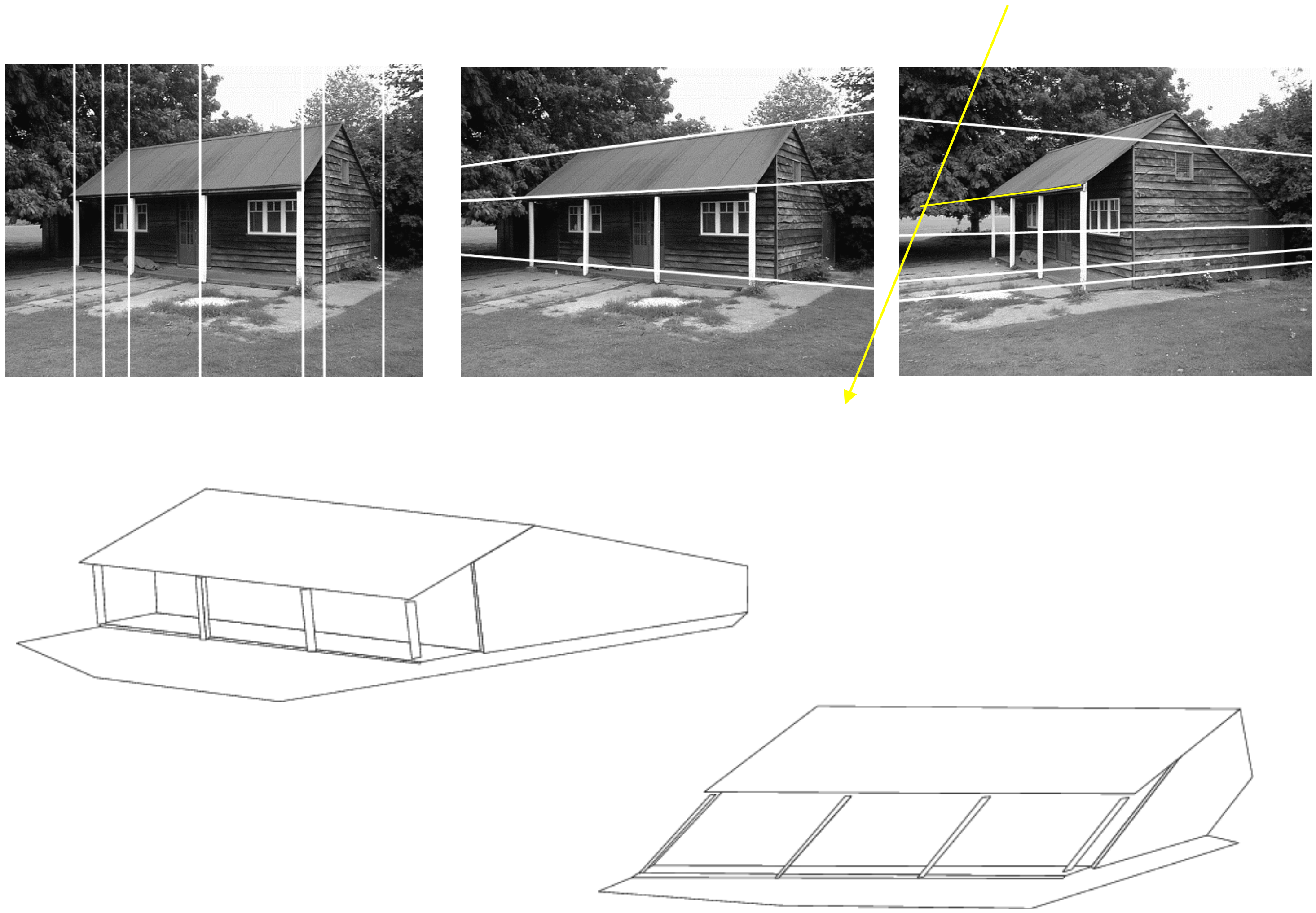
vanishing point in the first image

→ corresponding epipolar line



From 3 such pairs of corresponding vanishing points triangulate them → find 3 3D points → find π'_∞ through them → apply $\mathbf{H} = \begin{bmatrix} \mathbf{I}|\mathbf{0} \\ \boldsymbol{\pi}'_\infty \end{bmatrix}$

Scene constraints: parallel lines

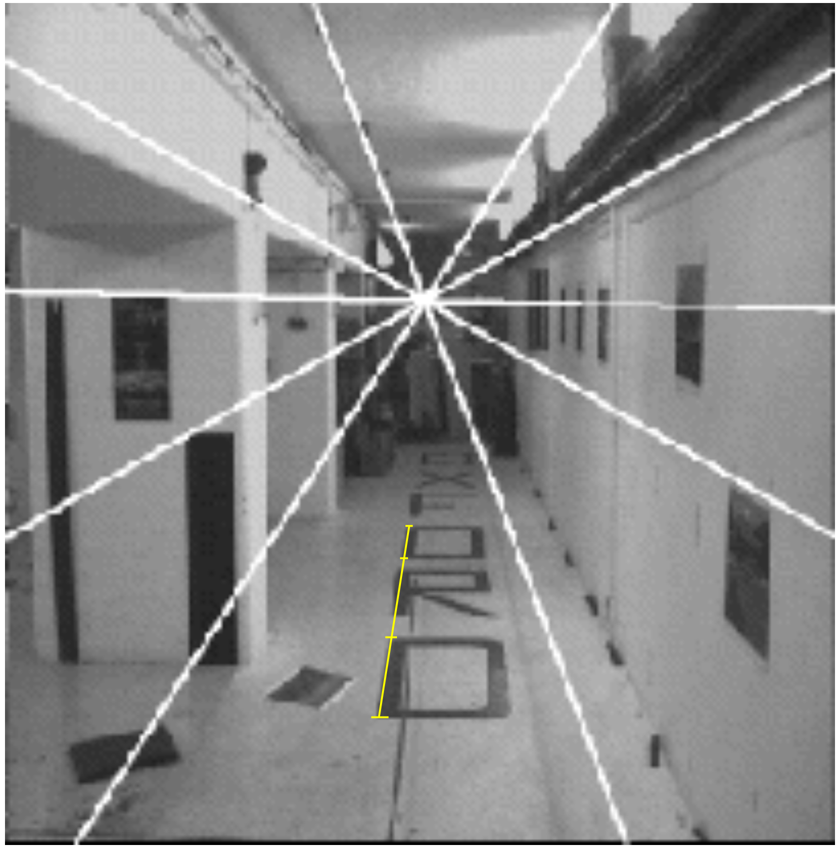


Projective to affine

other constraints on π'_∞ from scene

known distance ratios on a line

known distance ratio along a line allow to determine point at infinity (same as 2D case) : from cross ratio

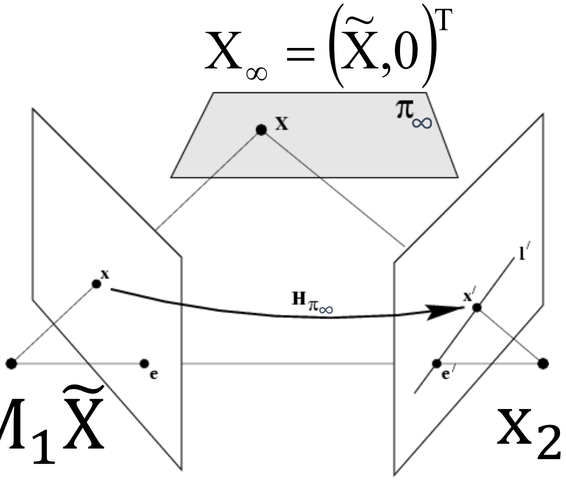


Projective to affine

Alternative cameras
(for affine reconstruction)

The homography induced by the plane at the ∞

$$P_1 = [M_1|\mathbf{m}_1] \quad P_2 = [M_2|\mathbf{m}_2]$$



$$H_\infty = M_2 M_1{}^{-1}$$

$$x_1 = M_1\widetilde{X} \qquad x_2 = M_2\widetilde{X}$$

unchanged under affine transformations

$$P_i = [M_i|m_i]\begin{bmatrix} A & a \\ 0 & 1 \end{bmatrix} = [M_i A|M_i a + m_i]$$

$$\rightarrow H_\infty = M_2 A A^{-1} M_1{}^{-1} = M_2 M_1{}^{-1}$$

affine reconstruction: alternative cameras $P_1 = [I\,|\,0] \quad P_2 = [H_\infty\,|\,e]$

Proof: use affine transformation with H^{-1} where $H \stackrel{\text{def}}{=} \begin{bmatrix} A & a \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} M & m \\ 0 & 1 \end{bmatrix}$

Projective to affine

The homography induced by the plane at the ∞

$$P_1 = [M_1|\mathbf{m}_1] \quad P_2 = [M_2|\mathbf{m}_2] \qquad\qquad H_\infty = M_2 M_1^{-1}$$

affine reconstruction: alternative cameras $\quad P'_1 = [I|0] \quad P'_2 = [H_\infty|e]$

Proof: use affine transformation with H^{-1} where $H \stackrel{\text{def}}{=} \begin{bmatrix} M_1 & \mathbf{m}_1 \\ 0 & 1 \end{bmatrix}$ →

$$H^{-1} = \begin{bmatrix} M_1^{-1} & -M_1^{-1}\mathbf{m}_1 \\ 0 & 1 \end{bmatrix}$$

→ $P'_1 = P_1 H^{-1} = [I|0]$

and

→ $P'_2 = P_2 H^{-1} = \left[M_2 M_1^{-1}\right| - M_2 M_1^{-1}\mathbf{m}_1 + \mathbf{m}_2] = [H_\infty|e]$

in fact the first camera center is $\mathbf{o}_1 = -M_1^{-1}\mathbf{m}_1$

and its image is

$$e = P_2 \begin{bmatrix} \mathbf{o}_1 \\ 1 \end{bmatrix} = [M_2|\mathbf{m}_2] \begin{bmatrix} \mathbf{o}_1 \\ 1 \end{bmatrix} = -M_2 M_1^{-1}\mathbf{m}_1 + \mathbf{m}_2$$

Alternative cameras
(for affine reconstruction)

Alternative cameras
(for affine reconstruction)

Remember:

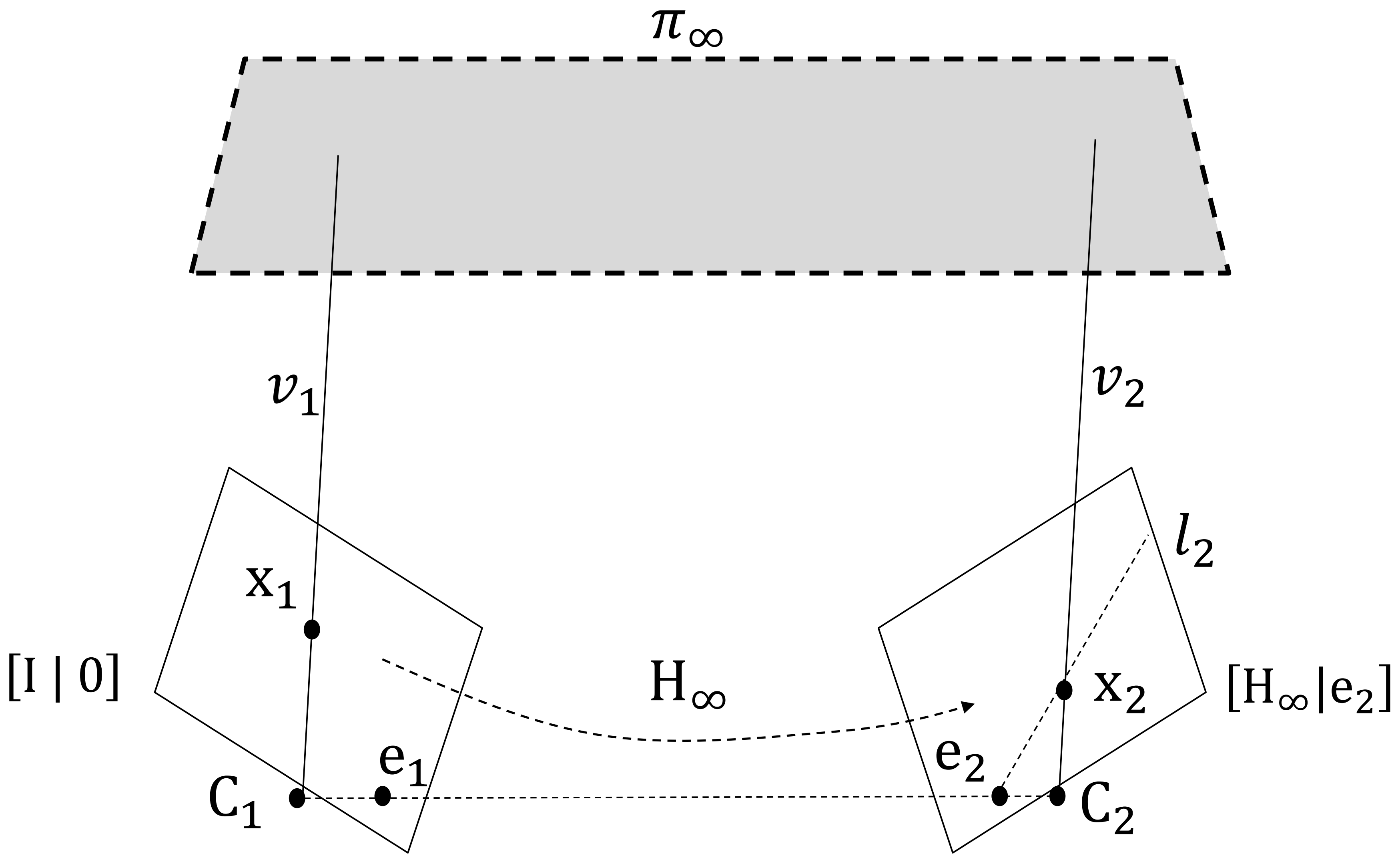
if 3D points are on a plane π, their two images are related by a homography H_π induced by plane π

Alternative cameras (for affine reconstruction)

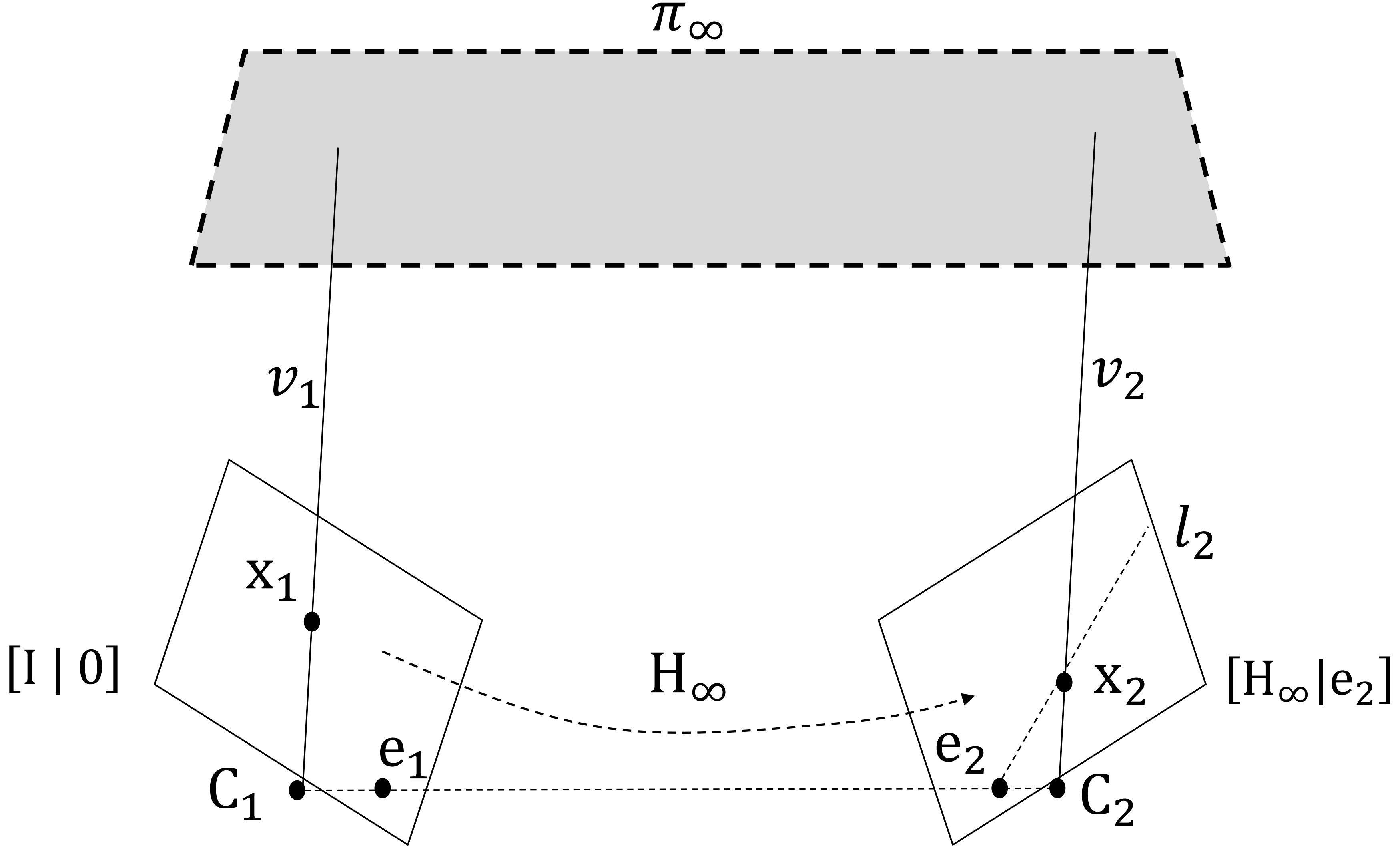
if 3D points are known to be at the ∞, i.e., on the plane π_∞, their two images are related by the homography H_∞ induced by plane π_∞

This homography H_∞ can be computed by three correspondences $\mathrm{x}_1 \leftrightarrow \mathrm{x}_2$ + one $\mathrm{e}_1 \leftrightarrow \mathrm{e}_2$

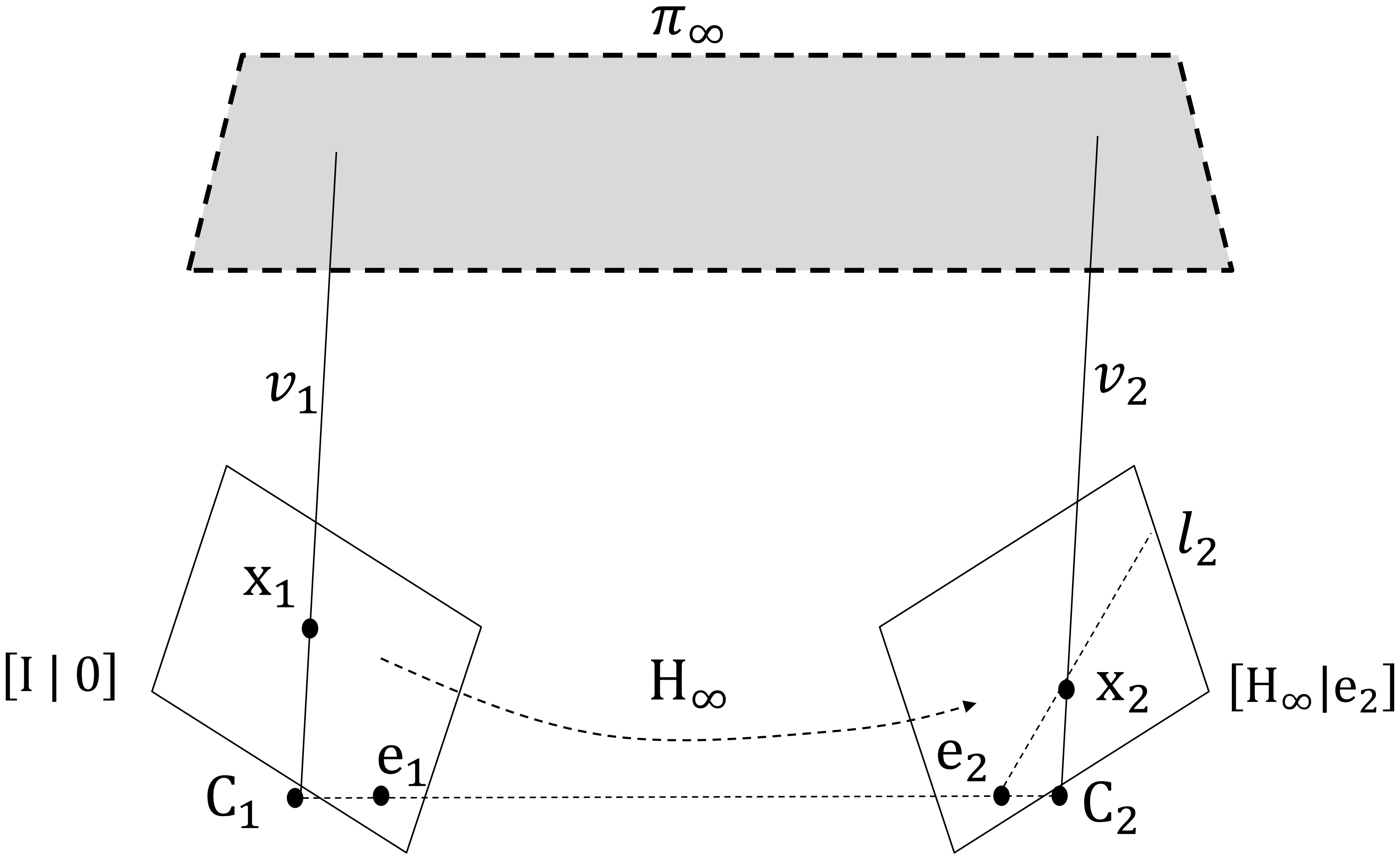
Now check camera pair $([\mathrm{I} \mid 0]\ ,\ [\mathrm{H}_\infty | \mathrm{e}_2])$



triangulate vewing rays of $x_1 \leftrightarrow x_2 = H_\infty x_1$using cameras
$[I \mid 0]$: $\quad$ $\mathrm{dir}(v_1) = I^{-1}x_1 = x_1$ and …



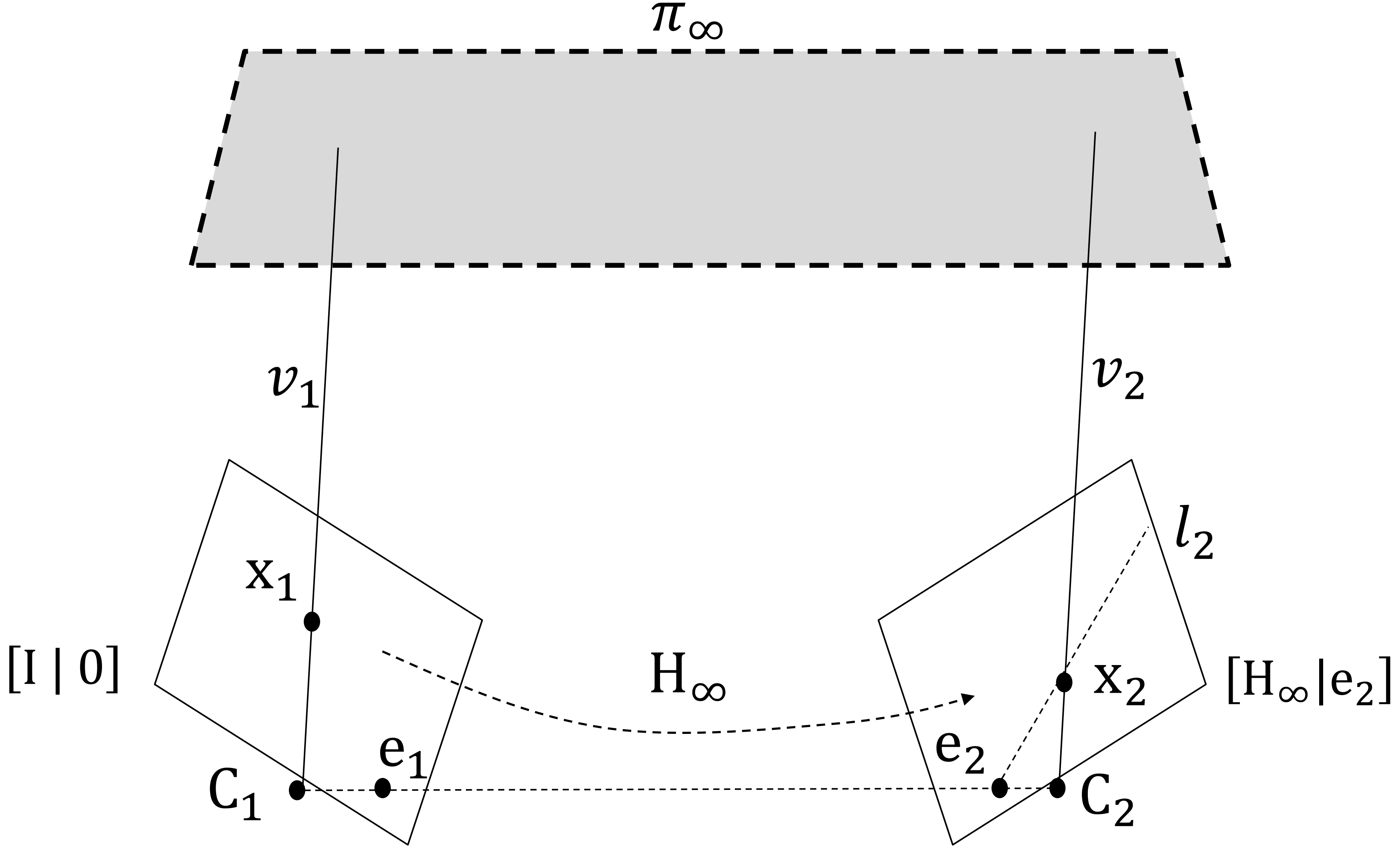
triangulate vewing rays of $\mathrm{x}_1 \leftrightarrow \mathrm{x}_2 = \mathrm{H}_\infty \mathrm{x}_1$using cameras
$[\mathrm{I} \mid 0]$: $\quad \mathrm{dir}(v_1) = \mathrm{I}^{-1}\mathrm{x}_1 = \mathrm{x}_1$ and
$[\mathrm{H}_\infty|\mathrm{e}_2]$: $\mathrm{dir}(v_2) = \mathrm{H}_\infty{}^{-1}\mathrm{x}_2 = \mathrm{H}_\infty{}^{-1}\mathrm{H}_\infty \mathrm{x}_1 = \mathrm{x}_1 = \mathrm{dir}(v_1)$



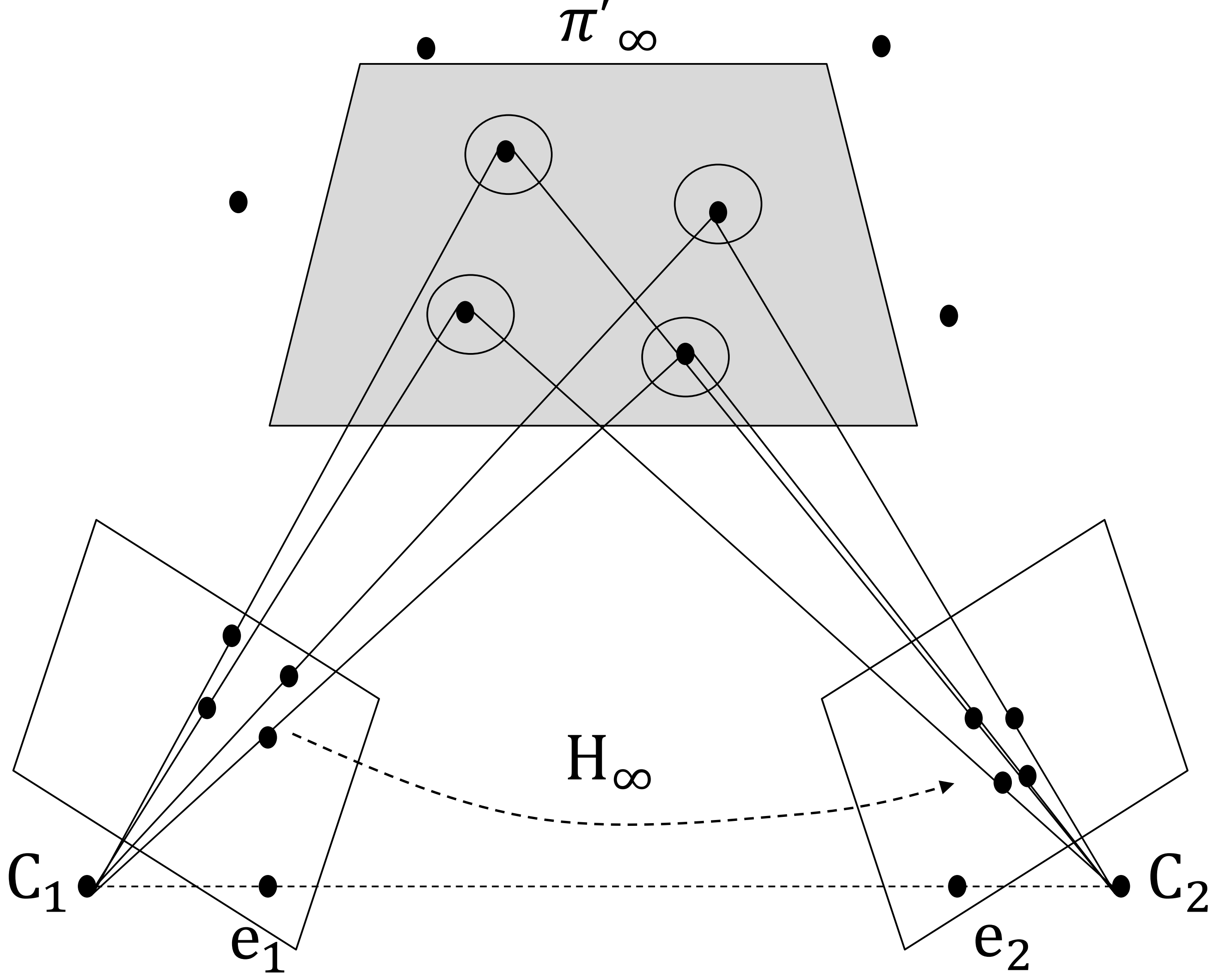
triangulate vewing rays of $x_1 \leftrightarrow x_2 = H_\infty x_1$ using cameras

$[I \mid 0]$: dir(v_1) $= x_1$ and $[H_\infty|e_2]$: dir(v_2) $= x_1 =$ dir(v_1)

viewing rays v_1 , v_2 are parallel → their ∩ X' is at the ∞ !!



true point X on π_∞, reconstructed point X’ on π_∞: AFFINE RECONSTRUCTION



Affine reconstruction: from 4 pairs of corresponding **vanishing** points compute homography H_π (H_∞). Use cameras $[I \mid 0]$ and $[H_\infty | e_2]$ to triangulate 3D points

To compute the homography induced by the plane at the infinity

- find three pairs of corresponding vanishing points $\mathrm{v}_{1\mathrm{j}} \leftrightarrow \mathrm{v}_{2\mathrm{j}}$
  $j = 1..3$ from two images

- use epipoles $\mathrm{e}_1 \leftrightarrow \mathrm{e}_2$ as the fourth pair of points (the epipoles can not be used to triangulate a point in 3D)

- → compute homography using these four pairs of points

- → this homography is H_∞

then use cameras $\mathrm{P}_1 = [\mathrm{I}|0]$ and $\mathrm{P}_2 = [\mathrm{H}_\infty|\mathrm{e}_2]$ as new cameras

affine to metric

Affine to metric

Given $P_1 = [M_1|m_1]$ and $\omega_1 = \left(KK^T\right)^{-1}$ of **just one** of the cameras, a possible transformation from affine to metric is

$$H = \begin{bmatrix} A^{-1} & 0 \\ 0 & 1 \end{bmatrix}$$ with $A =$ Cholesky factorisation of $AA^T = \left(M_1{}^T\omega_1 M_1\right)^{-1}$

Proof: new camera $P_1' = P_1 H^{-1} = [M_1 A|m_1]$ is OK if

$$\rightarrow$$

$$\omega_1{}^{-1} = KK^T = (KR)(KR)^T = M'M'^T = (M_1A)(M_1A)^T = M_1AA^TM_1{}^T$$

$$\rightarrow$$

$$M_1{}^{-1}\omega_1{}^{-1}M_1{}^{-T} = AA^T$$ Q.E.D

Then, the new (first) camera is $P_1{}' = P_1H^{-1} = [M_1A|m_1]$

and new reconstruction is $X' = HX = \begin{bmatrix} A^{-1} & 0 \\ 0 & 1 \end{bmatrix} X$

Affine to metric: the new (other) camera P_2'

From $P_2 = [M_2|m_2]$, new camera is $P_2' = P_2 H^{-1} = [M_2 A|m_2]$

(since $H = \begin{bmatrix} A^{-1} & 0 \\ 0 & 1 \end{bmatrix}$) with $AA^T = \left(M_1{}^T \omega_1 M_1\right)^{-1}$

The ω matrix of the other camera is

$$\omega_2^{-1} = (M_2 A)(M_2 A)^T = M_2 A A^T M_2{}^T = M_2 \left(M_1{}^T \omega_1 M_1\right)^{-1} M_2{}^T =$$
$$= M_2\ M_1{}^{-1}\ \omega_1^{-1}\ M_1{}^{-T}\ M_2{}^T = H_\infty\ \omega_1^{-1}\ H_\infty{}^T$$

Transfer of the ω matrix:

$$\omega_2 = H_\infty{}^{-T} \omega_1\ H_\infty{}^{-1}$$

through the homography H_∞ induced by the plane at the ∞

Affine to metric

If, after affine reconstruction, K is known for camera $\mathrm{P}_1 = [\mathrm{I}\,|0\,]$, then this camera is transformed to $\mathrm{P'}_1 = [\mathrm{K}\,|0\,]$ by applying the reconstructing mapping

$$\mathrm{H} = \begin{bmatrix} \mathrm{K}^{-1} & 0 \\ 0 & 1 \end{bmatrix}$$

the other cameras P_2, P_3… are mapped to $\mathrm{P'}_2$, $\mathrm{P'}_3$ …

$$\mathrm{P'}_2 = \mathrm{P}_2\mathrm{H}^{-1} = [\mathrm{M}_2|\mathrm{m}_2]\begin{bmatrix} \mathrm{K} & 0 \\ 0 & 1 \end{bmatrix} = [\mathrm{M}_2\mathrm{K}|\mathrm{m}_2]$$

while the updated (metric) reconstruction is

$$\mathrm{X''} = \mathrm{HX'} = \begin{bmatrix} \mathrm{K}^{-1} & 0 \\ 0 & 1 \end{bmatrix}\mathrm{X'}$$

Constraints on ω from ortogonality

vanishing points corresponding to orthogonal directions

$$\mathrm{v}_1^{\mathrm{T}} \omega \mathrm{v}_2 = 0$$

vanishing line and vanishing point corresponding to plane and normal direction

$$\mathrm{l} = \omega \mathrm{v}$$

Constraints on ω from known internal parameters

$$\omega = K^{-T}K^{-1}$$

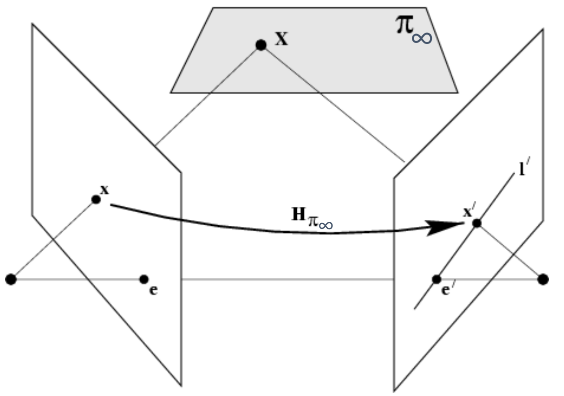
rectangular pixels (zero skew factor)

$$s = 0 \qquad \omega_{12} = \omega_{21} = 0$$

+ square pixels (or known aspect ratio)

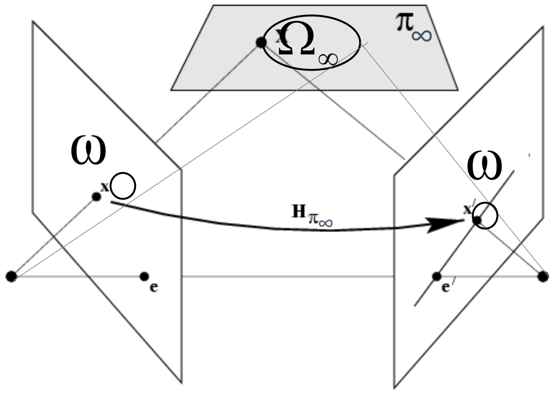
$$f_x = f_y \qquad \omega_{11} = \omega_{22}$$

Remember: **Homography induced by plane at the infinity**



$$H_\infty = M_2 M_1^{-1}$$

transfer of vanishing point: $v_2 = H_\infty v_1 = M_2 M_1^{-1} v_1$



transfer of the ω matrix $\omega_2 = H_\infty^{-T} \omega_1 H_\infty^{-1}$

Example: same camera for all images → constraints on ω from affine reconstruction (H_∞)

same intrinsics $\mathrm{K} \Rightarrow$ same ω matrix, e.g. moving camera

$$\omega = \omega' = \mathrm{H}_\infty{}^{-\mathrm{T}} \omega \mathrm{H}_\infty{}^{-1}$$

given enough images there is in general only one matrix ω that transfers to itself in all images,

This approach is called *self-calibration,* see later

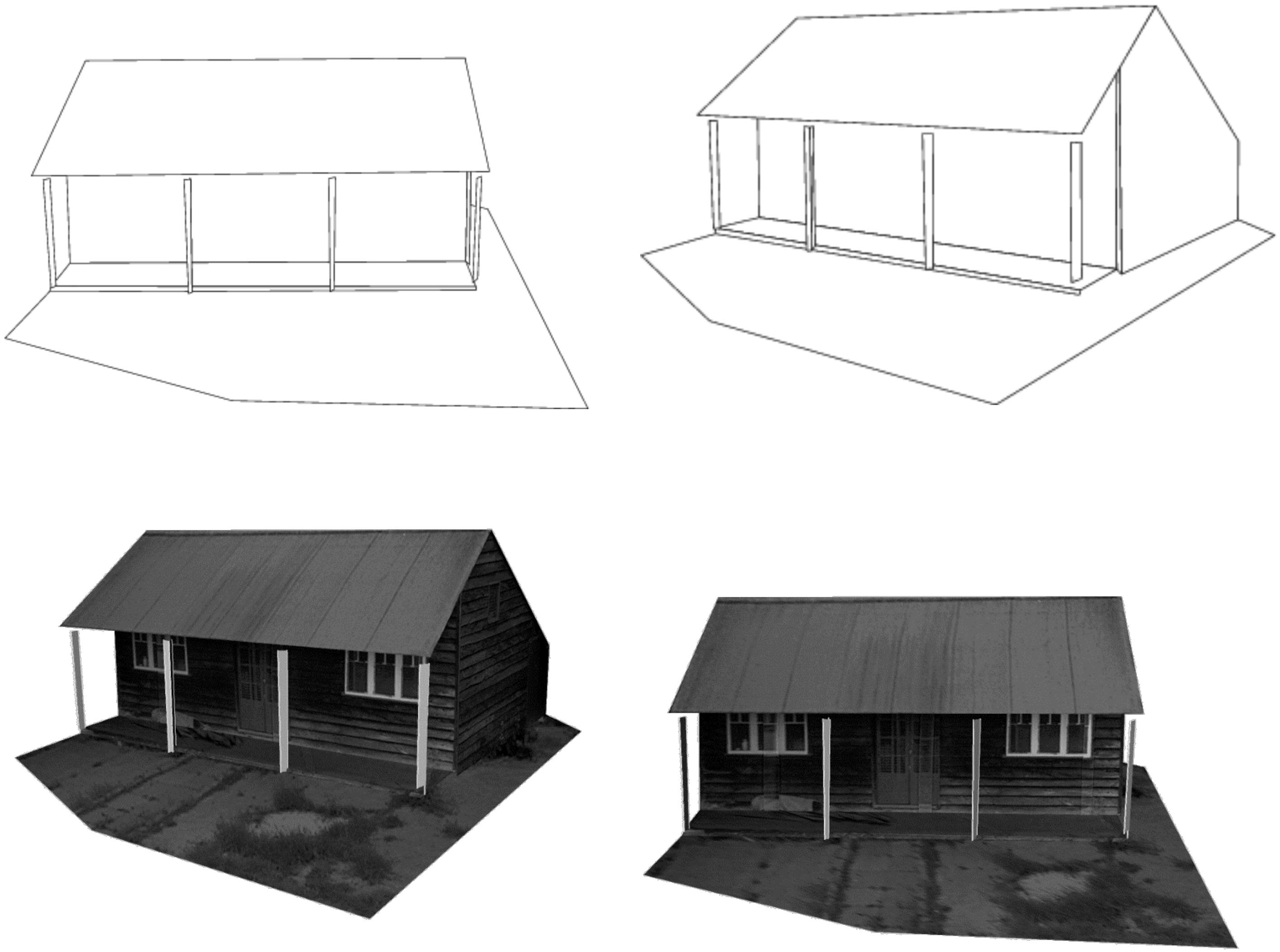
transfer of ω matrix : $\omega' = \omega = \mathrm{H}_\infty{}^{-\mathrm{T}} \omega\, \mathrm{H}_\infty{}^{-1}$

Direct metric reconstruction using ω of **all** cameras

$$\omega = K^{-T}K^{-1} \Rightarrow K$$

calibrated reconstruction: Essential matrix

→ first relative pose and then stereo triangulation



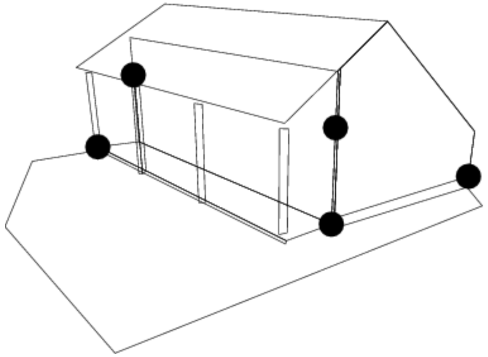
Direct reconstruction using ground truth

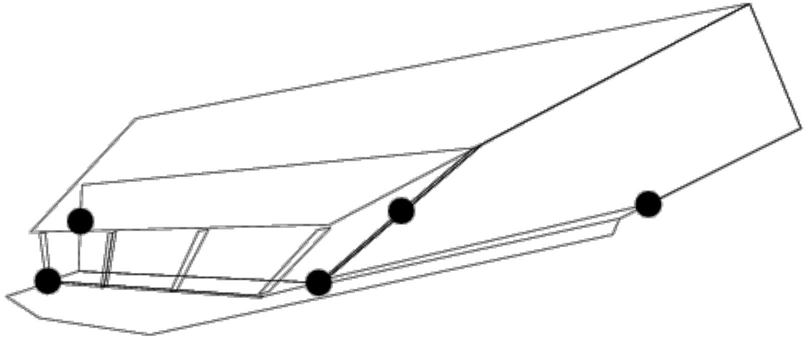
use control points X_{Ei} with known coordinates
to go from projective to metric

$$X_{Ei} = HX_i$$

$$x_i = PH^{-1}X_{Ei}$$

(2 lin. eq. in H^{-1} per view,
3 for two views)





3D Shape Reconstruction

Data

Enough pairs of corresponding image points

$$\mathrm{x}_{1i} \leftrightarrow \mathrm{x}_{2i}$$

taken by two uncalibrated images

Purpose

Compute

$$(\mathrm{P}_1^{\mathrm{M}}, \mathrm{P}_2^{\mathrm{M}}, \{\mathrm{X}_\mathrm{i}^{\mathrm{M}}\})$$

(i.e. within similarity of original scene and cameras)

Algorithm

(i) Compute projective reconstruction $(P_1^o, P_2^o, \{X_i^o\})$

(a) Compute F from $x_{1i} \leftrightarrow x_{2i}$

(b) Compute P_1^o, P_2^o from F: $P_1^o = [I|0]$ $P_2^o = [[e_2]_\times F \mid e_2]$, ($e_2$=LNS F)

(c) Triangulate X_i from $x_{1i} \leftrightarrow x_{2i}$

(ii) Rectify reconstruction from projective to metric

Direct method: compute H from control points $X_{Ei} = HX_i$

then $P_1^M = P_1^o H^{-1}, P_1^M = P_1^o H^{-1}, X_{Ei} = HX_i$

Stratified method:

(a) **Affine reconstruction**: find (fit) π_∞ and apply $H = \begin{bmatrix} I|0 \\ \pi_\infty \end{bmatrix}$ **OR**

find (fit) H_∞ and set new cameras $P_1^A = [I|0]$ and $P_2^A = [H_\infty|e_2]$

(a) **Metric reconstruction**: find K → (ω matrix) for one of the cameras

apply $H = \begin{bmatrix} A^{-1} & 0 \\ 0 & 1 \end{bmatrix}$ with $AA^T = \left(M^T \omega M\right)^{-1}$ Cholesky factorisation

Image information provided	View relations and projective objects	3-space objects	reconstruction ambiguity
point correspondences	F		projective
point correspondences including vanishing points	F, H_∞	π_∞	affine
Points correspondences and internal camera calibration	F, H_∞ ω, ω'	π_∞ Q_∞	metric

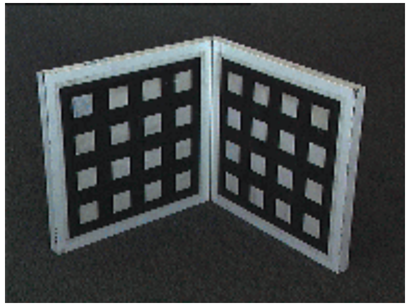
SELF-CALIBRATION

Outline

- Introduction
- Self-calibration
- Absolute Conic

Motivation

- Avoid explicit calibration procedure
  - Complex procedure
  - Need for calibration object
  - Need to maintain calibration



Motivation

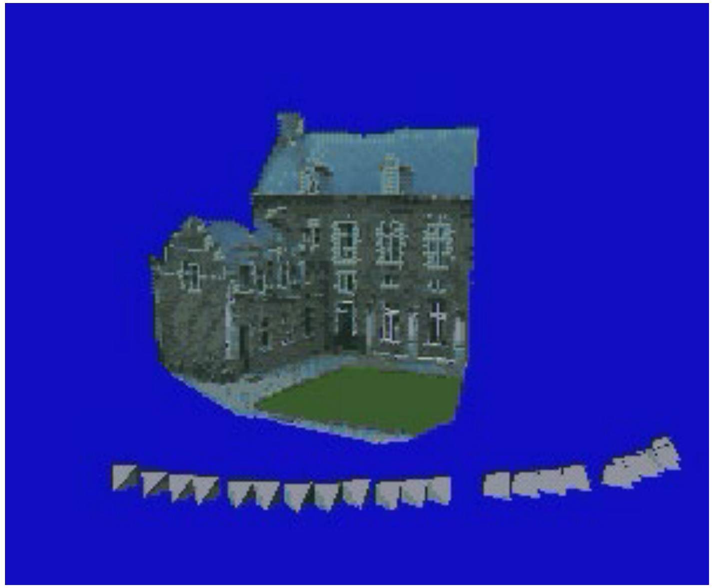
- Allow flexible acquisition
  - No prior calibration necessary
  - Possibility to vary intrinsics
  - Use archive footage

Projective ambiguity

Reconstruction from uncalibrated images

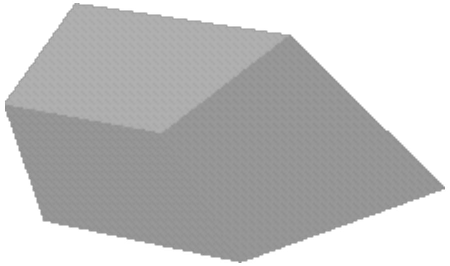
$\Rightarrow$ projective ambiguity on reconstruction

$$\mathrm{x} = \mathbf{P}\mathrm{X} = (\mathbf{P}\mathbf{T}^{-1})(\mathbf{T}\mathrm{X}) = \mathbf{P}'\mathrm{X}'$$



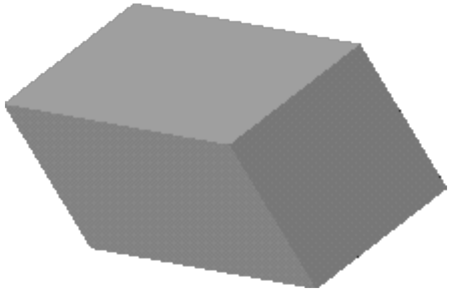
Stratification of geometry

Projective



15 DOF

Affine

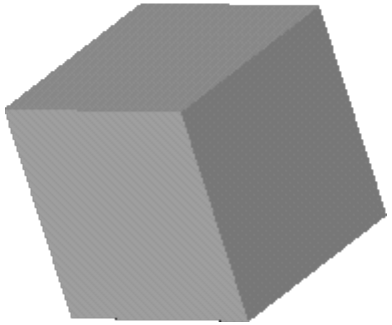


12 DOF

plane at infinity

parallelism

Metric



7 DOF

absolute conic

angles, rel.dist.

More general

More structure

Constraints ?

- Scene constraints
  - Parallellism, vanishing points, horizon, ...
  - Distances, positions, angles, ...

  Unknown scene $\rightarrow$ no constraints
- Camera extrinsics constraints
  - Pose, orientation, motion ...

  Unknown camera motion $\rightarrow$ no constraints
- Camera intrinsics constraints
  - Focal length, principal point, aspect ratio & skew

  Perspective camera model too general
  $\rightarrow$ some constraints

Euclidean projection matrix

Factorization of Euclidean projection matrix

$$\mathbf{P} = \mathbf{K}\left[\mathbf{R}^\top \quad -\mathbf{R}^\top\mathbf{t}\right]$$

Intrinsics: $\mathbf{K} = \begin{bmatrix} f_x & s & u_x \\ & f_y & u_y \\ & & 1 \end{bmatrix}$ (camera geometry)

Extrinsics: $(\mathbf{R}, \mathrm{t})$ (camera motion)

Note: every projection matrix can be factorized,
but only meaningful for euclidean projection matrices

Self-calibration

Upgrade from *projective* structure to *metric* structure using *constraints on intrinsic* camera parameters

- Constant intrinsics

  (Faugeras et al. ECCV´92, Hartley´93, Triggs´97, Pollefeys et al. PAMI´98, ...)

- Some known intrinsics, others varying

  (Heyden&Astrom CVPR´97, Pollefeys et al. ICCV´98,...)

- Constraints on intrincs and restricted motion
  (e.g. pure translation, pure rotation, planar motion)

  (Moons et al.´94, Hartley ´94, Armstrong ECCV´96, ...)

A counting argument

- To go from projective (15DOF) to metric (7DOF) at least 8 constraints are needed
- Minimal sequence length should satisfy

$$n \times (\#\, known) + (n-1) \times (\#\, fixed) \geq 8$$

where n is the number of images

- Independent of algorithm
- Assumes general motion (i.e. not critical)

Dual Image of Absolute Conic
DIAC

From tentative cameras to true cameras

tentative cameras $P_1 = [I \quad 0] \qquad P_i = [A_i \quad a_i]$

true cameras $P_1' = P_1 H^{-1} \quad P_i' = P_i H^{-1}$

with $H^{-1} = \begin{bmatrix} A & t \\ v^T & 1 \end{bmatrix}$

world reference = first-camera reference

$$P_1' = [K_1 \quad 0] = [I \quad 0] \begin{bmatrix} A & t \\ v^T & 1 \end{bmatrix}$$

thus $A = K_1$ and $t = 0$

.. therefore

new camera $P_i' = [K_i R_i \quad K_i t_i] = [A_i \quad a_i]\begin{bmatrix} K_1 & 0 \\ v^T & 1 \end{bmatrix}$

now define $p \quad s.t. \quad v^T = -p^T K_1$

then the new camera $P_i' = [(A_i - a_i p^T)K_1 \quad a_i]$

is a true camera if $(A_i - a_i p^T)K_1 = K_i R_i$

But, since $\omega_i^* = K_i K_i^T = (K_i R_i)(K_i R_i)^T$

Then $\omega_i^* = (A_i - a_i p^T)K_1 K_1^T (A_i - a_i p^T)^T$

Hence $\omega_i^* = (A_i - a_i p^T)\omega_1^* (A_i - a_i p^T)^T$

Selfcalibration equation

$$\omega_i^* = (A_i - a_i p^T)\omega_1^*(A_i - a_i p^T)^T$$

8 unknowns: p and ω_1^* (3 + 5)

provided that only constrained elements of ω_i^* are used (i.e., known or constant)

→ in this way: new equations do not introduce new unknowns

$$\omega_i^* = (A_i - a_i p^T)\, \omega_1^*\, (A_i - a_i p^T)^T$$

$$A_i - a_i p^T = \mathrm{H}_{\infty i}$$

since true cameras are $[K_1|0]$, $[(A_i - a_i p^T)K_1|a_i]$
and the homography induced by the plane π_∞ is

$$\mathrm{H}_{\infty i} = \mathrm{M}_2 \mathrm{M}_1{}^{-1} = A_i - a_i p^T$$

there are 8 unknowns: vector p and matrix K_1

Constraints on intrinsic parameters

$$\mathbf{K} = \begin{bmatrix} f_x & s & u_x \\ & f_y & u_y \\ & & 1 \end{bmatrix}$$

<u>Constant</u>

same cameras: $\mathbf{K}_1 = \mathbf{K}_2 = \cdots$

aspect ratio $f_x / f_y = a$ constant for all cameras

<u>Known</u>

e.g. rectangular pixels: $s = 0$

square pixels: $f_x = f_y, s = 0$

principal point known: $(u_x, u_y) = \left(\frac{w}{2}, \frac{h}{2}\right)$

A counting argument

- To go from projective (15DOF) to metric (7DOF) at least 8 constraints are needed
- Minimal sequence length should satisfy

$$n \times (\#\, known) + (n-1) \times (\#\, fixed) \geq 8$$

- Independent of algorithm
- Assumes general motion (i.e. not critical)

Constraints on DIAC ω^*

$$\omega^* = \begin{bmatrix} f_x^2 + s^2 + u_o^2 & sf_y + u_o v_o & u_o \\ sf_y + u_o v_o & f_y^2 + v_o^2 & v_o \\ u_o & v_o & 1 \end{bmatrix}$$

condition	constraint	type	#constraints
Zero skew	$\omega^*_{12}\omega^*_{33} = \omega^*_{13}\omega^*_{23}$	quadratic	m
Principal point	$\omega^*_{13} = \omega^*_{23} = 0$	linear	$2m$
Zero skew (& p.p.)	$\omega^*_{12} = 0$	linear	m
Fixed aspect ratio (& p.p.& Skew)	$\omega^*_{11}\omega'^*_{22} = \omega^*_{22}\omega'^*_{11}$	quadratic	$m-1$
Known aspect ratio (& p.p.& Skew)	$\omega^*_{11} = \omega^*_{22}$	linear	m
Focal length (& p.p. & Skew)	$\omega^*_{33} = \omega^*_{11}$	linear	m

Note that in the absence of skew the IAC ω can be more practical than the DIAC ω*!

$$\omega_\infty^* = \begin{bmatrix} f_x^2 + s^2 + u_o^2 & sf_y + u_o v_o & u_o \\ sf_y + u_o v_o & f_y^2 + v_o^2 & v_o \\ u_o & v_o & 1 \end{bmatrix}$$

$$\omega = \frac{1}{f_x^2 f_y^2} \begin{bmatrix} f_y^2 & 0 & -f_y^2 u_o \\ 0 & f_x^2 & -f_x^2 v_o \\ -f_y^2 v_o & -f_x^2 u_o & f_x^2 f_y^2 + f_y^2 u_o^2 + f_x^2 v_o^2 \end{bmatrix}$$

Linear algorithm

(Pollefeys et al.,ICCV´98/IJCV´99)

Assume everything known, except focal length

$$\omega^* \approx \begin{bmatrix} \hat{f}^2 & 0 & 0 \\ 0 & \hat{f}^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \propto \mathbf{P}\Omega^*\mathbf{P}^\top$$

$$\left(\mathbf{P}Q^*\mathbf{P}^\top\right)_{11} - \left(\mathbf{P}Q^*\mathbf{P}^\top\right)_{22} = 0$$

$$\left(\mathbf{P}Q^*\mathbf{P}^\top\right)_{12} = 0$$

$$\left(\mathbf{P}Q^*\mathbf{P}^\top\right)_{13} = 0$$

$$\left(\mathbf{P}Q^*\mathbf{P}^\top\right)_{23} = 0$$

Yields 4 constraint per image

Note that rank-3 constraint is not enforced

Linear algorithm revisited

(Pollefeys et al., ECCV'02)

Weighted linear equations

$$\mathbf{K}\mathbf{K}^{\mathsf{T}} \approx \begin{bmatrix} \hat{f}^2 & 0 & 0 \\ 0 & \hat{f}^2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\hat{f} \approx 1$$

$$\frac{1}{0.2}\left(\mathbf{P}\Omega^*\mathbf{P}^{\mathrm{T}}\right)_{11} - \left(\mathbf{P}\Omega^*\mathbf{P}^{\mathrm{T}}\right)_{22} = 0$$

$$\frac{1}{0.01}\left(\mathbf{P}\Omega^*\mathbf{P}^{\mathrm{T}}\right)_{12} = 0$$

$$\frac{1}{0.1}\left(\mathbf{P}\Omega^*\mathbf{P}^{\mathrm{T}}\right)_{13} = 0$$

$$\frac{1}{0.1}\left(\mathbf{P}\Omega^*\mathbf{P}^{\mathrm{T}}\right)_{23} = 0$$

$$\frac{1}{9}\left(\mathbf{P}\Omega^*\mathbf{P}^{\mathrm{T}}\right)_{11} - \left(\mathbf{P}\Omega^*\mathbf{P}^{\mathrm{T}}\right)_{33} = 0$$

$$\frac{1}{9}\left(\mathbf{P}\Omega^*\mathbf{P}^{\mathrm{T}}\right)_{22} - \left(\mathbf{P}\Omega^*\mathbf{P}^{\mathrm{T}}\right)_{33} = 0$$

assumptions

$$\log(\hat{f}) \approx \log(1) \pm \log(3) \qquad c_x \approx 0 \pm 0.1 \qquad s = 0$$

$$\log\left(\frac{\hat{f}_x}{\hat{f}_y}\right) \approx \log(1) \pm \log(1.1) \qquad c_y \approx 0 \pm 0.1$$

Refinement

- Metric bundle adjustment

$$\arg\min_{\mathbf{P}_k, \mathrm{X}_i} \sum_{k=1}^{m} \sum_{i=1}^{n} D\left(\mathrm{x}_{ki}, \mathbf{P}_k\left(\mathrm{X}_i\right)\right)^2$$

Enforce constraints or priors
on intrinsics during minimization
(this is „self-calibration“ for photogrammetrist)