CIS 5800

Machine Perception

Or *Geometric Computer Vision*

Instructor: Lingjie Liu

Lec 5: Feb 10, 2025

Spring 2025 Tentative Schedule

- HW 1: release last Wed, the deadline is next Wed (Feb 19).
- Midterm
  - March 26 (Wed) during class.
  - Format similar to HW questions, more details soon.

Recap: Projective Transformation = Homography = Collineation=Projectivity

Definition

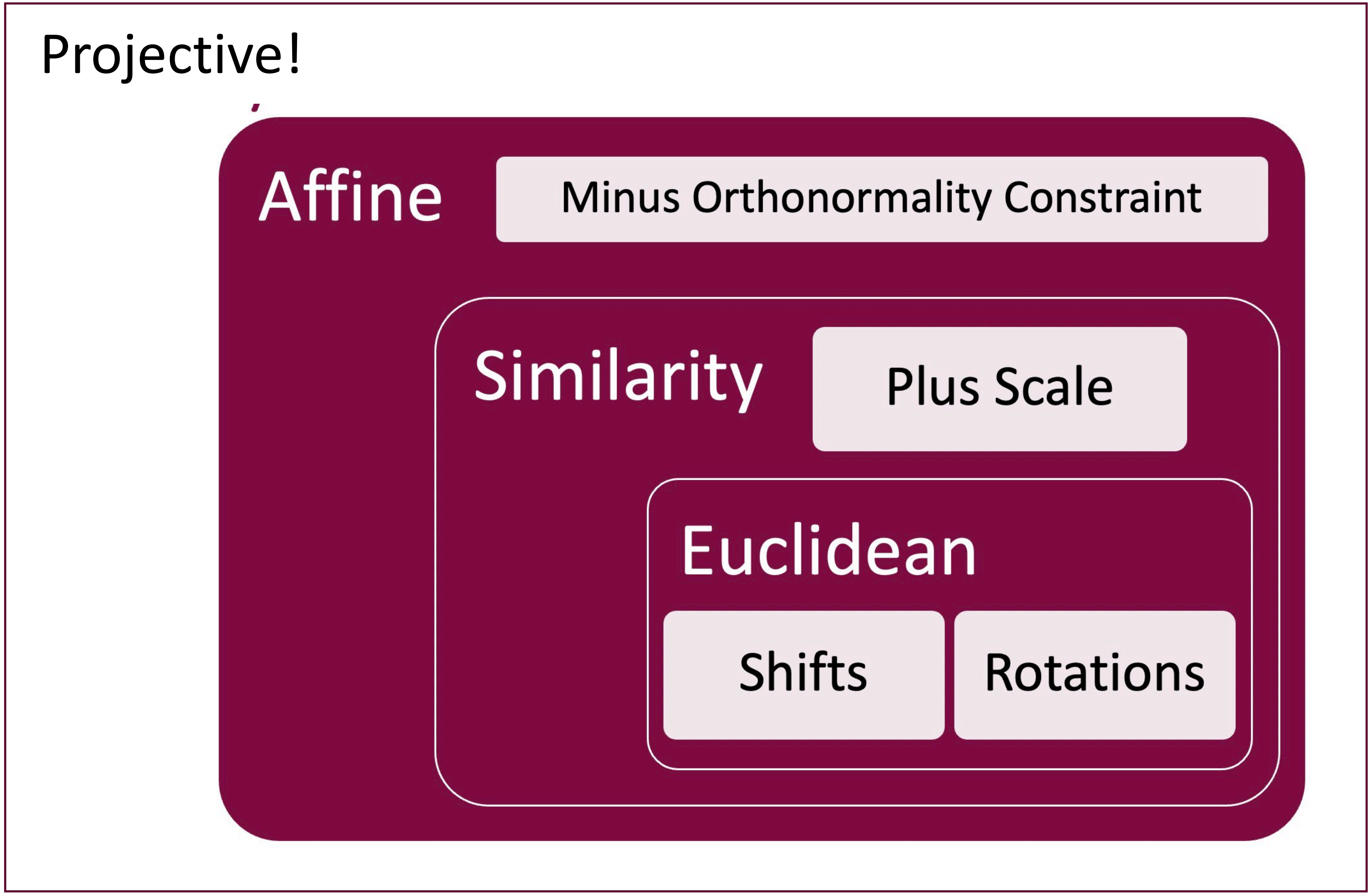
A **projective transformation** is any invertible matrix transformation $\mathbb{P}^2 \to \mathbb{P}^2$.

A projective transformation H maps p to $p' \sim Hp$

Invertibility means that $\det(H) \neq 0$ and that there exists $\lambda \neq 0$ such that $\lambda p' = Hp$

Observe that we will write either $p' \sim Hp$ or $\lambda p' = Hp$

Recap: Place in the Hierarchy of Transformations



Recap: Perspective Projection v.s. Projective Transformation

	Perspective Projection	**Projective Transformation**
Definition	A mapping from 3D space to a 2D plane (e.g., camera image)	A general mapping between projective space (e.g., P^2 to P^2)
Mathematical Formula	p′=K[R|T]P	p′=H·p
Input Space	R^3 (can also be P^3)	P^n (typically P^2 in this class)
Output Space	R^2 (can also be P^2)	P^n (typically P^2 in this class)
Applications	Image formulation, 3D rendering	Image registration, planar transformation, texture mapping

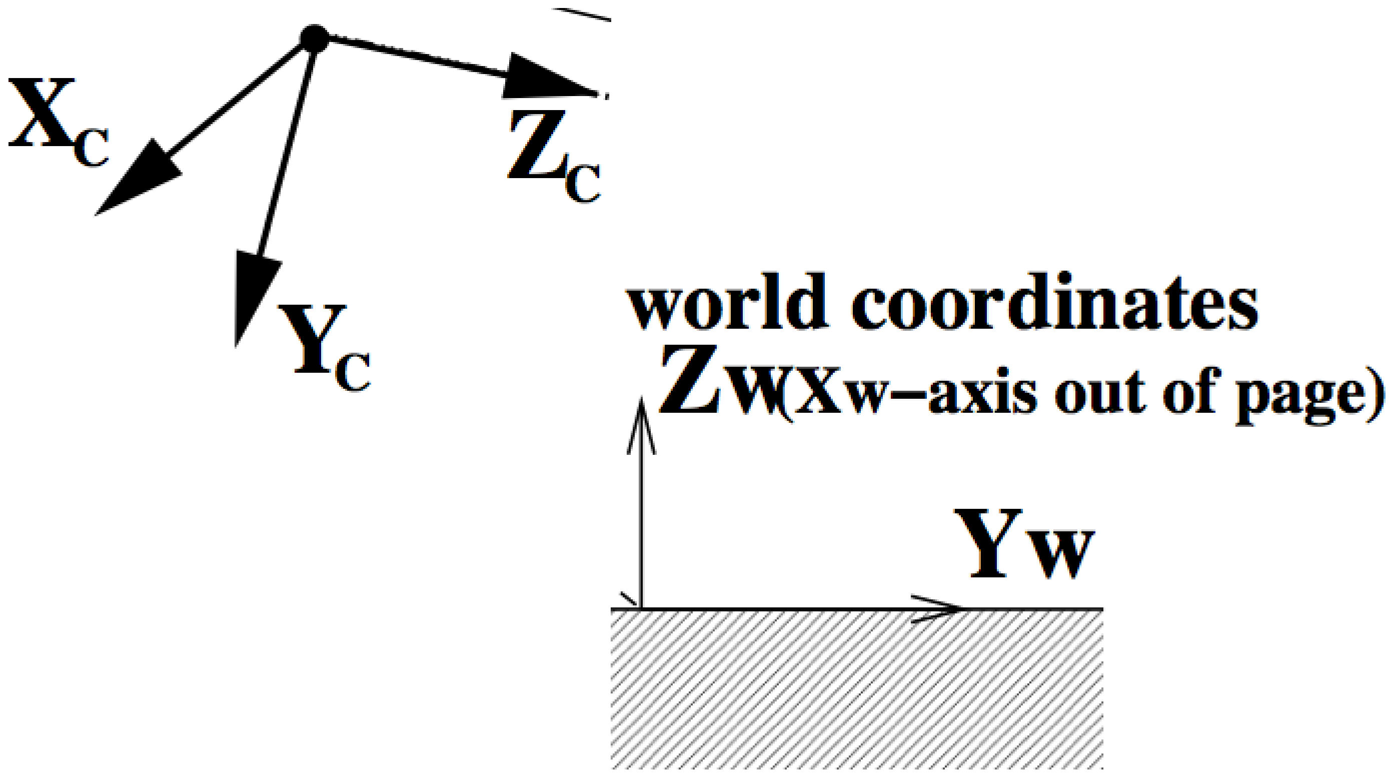
Recap: Perspective Projection -> Homography

- Can we show that the perspective camera projection from $\mathbb{P}^3 \to \mathbb{P}^2$ of a plane in the world is in fact a homography in $\mathbb{P}^2$ (i.e., projective transformation from $\mathbb{P}^2 \to \mathbb{P}^2$) when the world plane coordinates are expressed in $\mathbb{P}^2$?

- Remember:

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \sim K_{3\times3}[R_{3\times3}|\boldsymbol{t}_{3\times1}]\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Recap: Assume world plane $Z_w = 0$



Recap: Pose From Homography

Computing the homography can tell us how the camera (and therefore, e.g. a robot attached to the camera) is oriented w.r.t. to a world plane! (assuming known K)

Q: Where do you get $\boldsymbol{r}_3$ from though? A: $\boldsymbol{r}_3 = \boldsymbol{r}_1 \times \boldsymbol{r}_2$

Recall the projection from world to camera

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} \sim K \begin{pmatrix} r_1 & r_2 & r_3 & T \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix}$$

and assume that all points in the world lie in the ground plane $Z = 0$.

Then the transformation reads

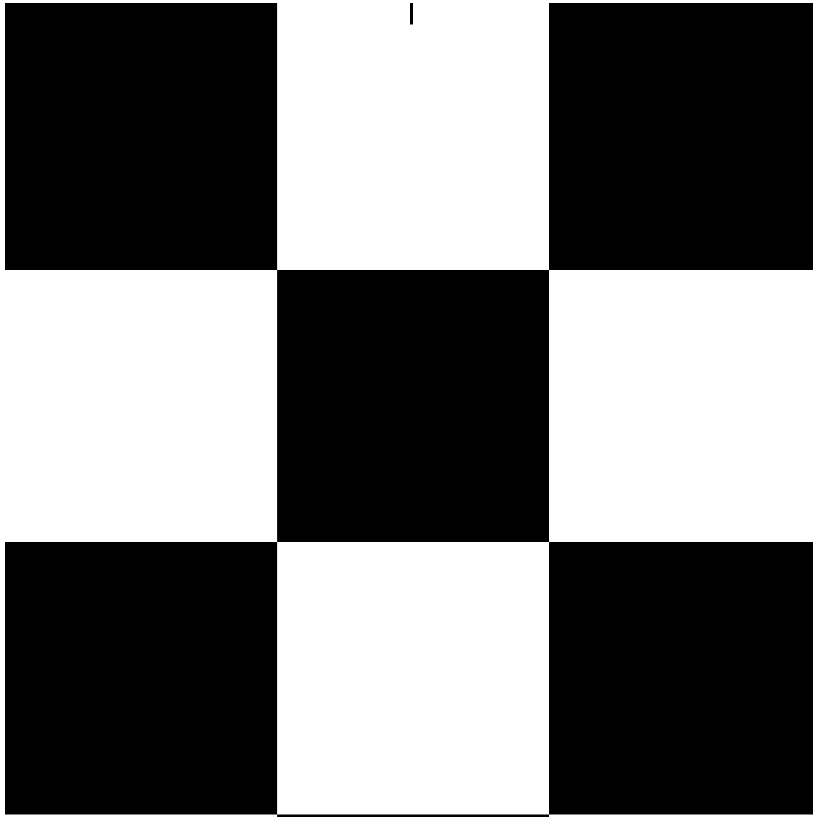
$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} \sim \underbrace{K \begin{pmatrix} r_1 & r_2 & T \end{pmatrix}}_{\text{The planar homography } H: \mathbb{P}^2 \rightarrow \mathbb{P}^2} \begin{pmatrix} X \\ Y \\ W \end{pmatrix}$$

The planar homography
$H: \mathbb{P}^2 \rightarrow \mathbb{P}^2$

Computing Homographies From 4 Point Correspondences

"4-point collineation"

How can we compute the projective transformation between a known pattern and its projection?



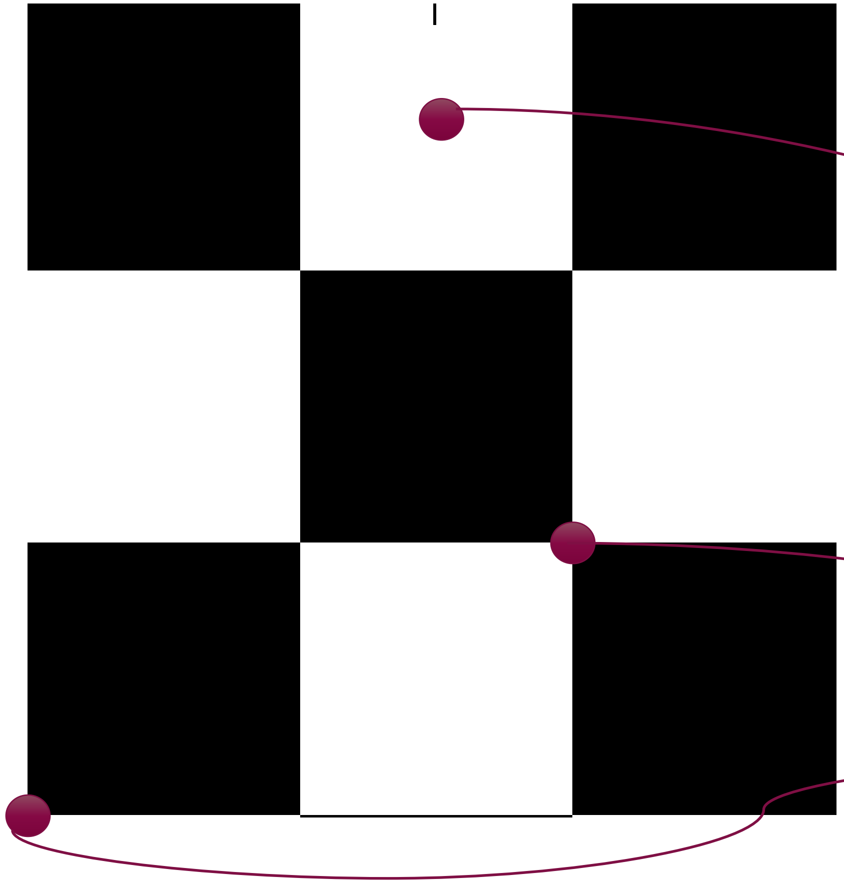
Floor tiles measured in [m]

and

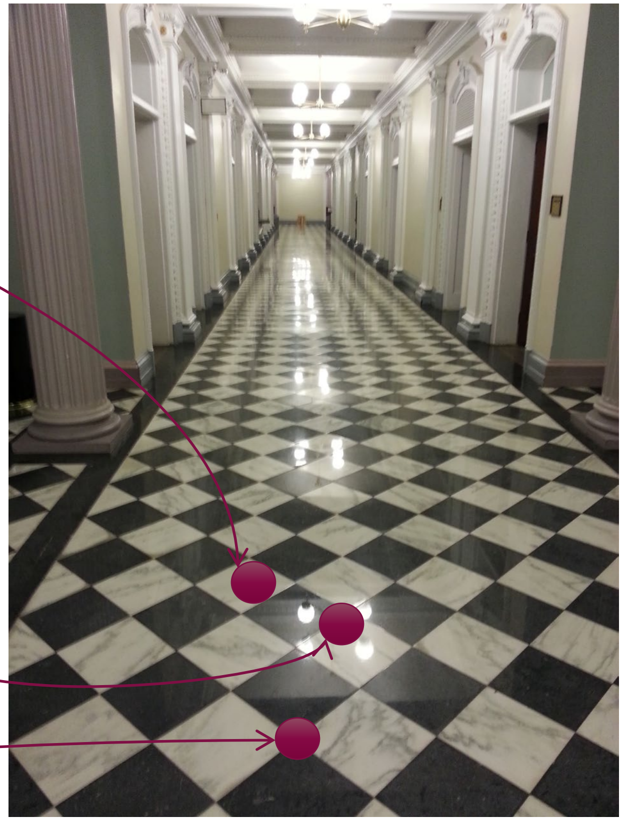
Points in pixel coordinates

The result of such a transformation would map any point in one plane to the corresponding point in the other

“correspondences”



and



Floor tiles measured in [m]

Points in pixel coordinates

Recap:

How many unknowns are in a projective transformation H ?

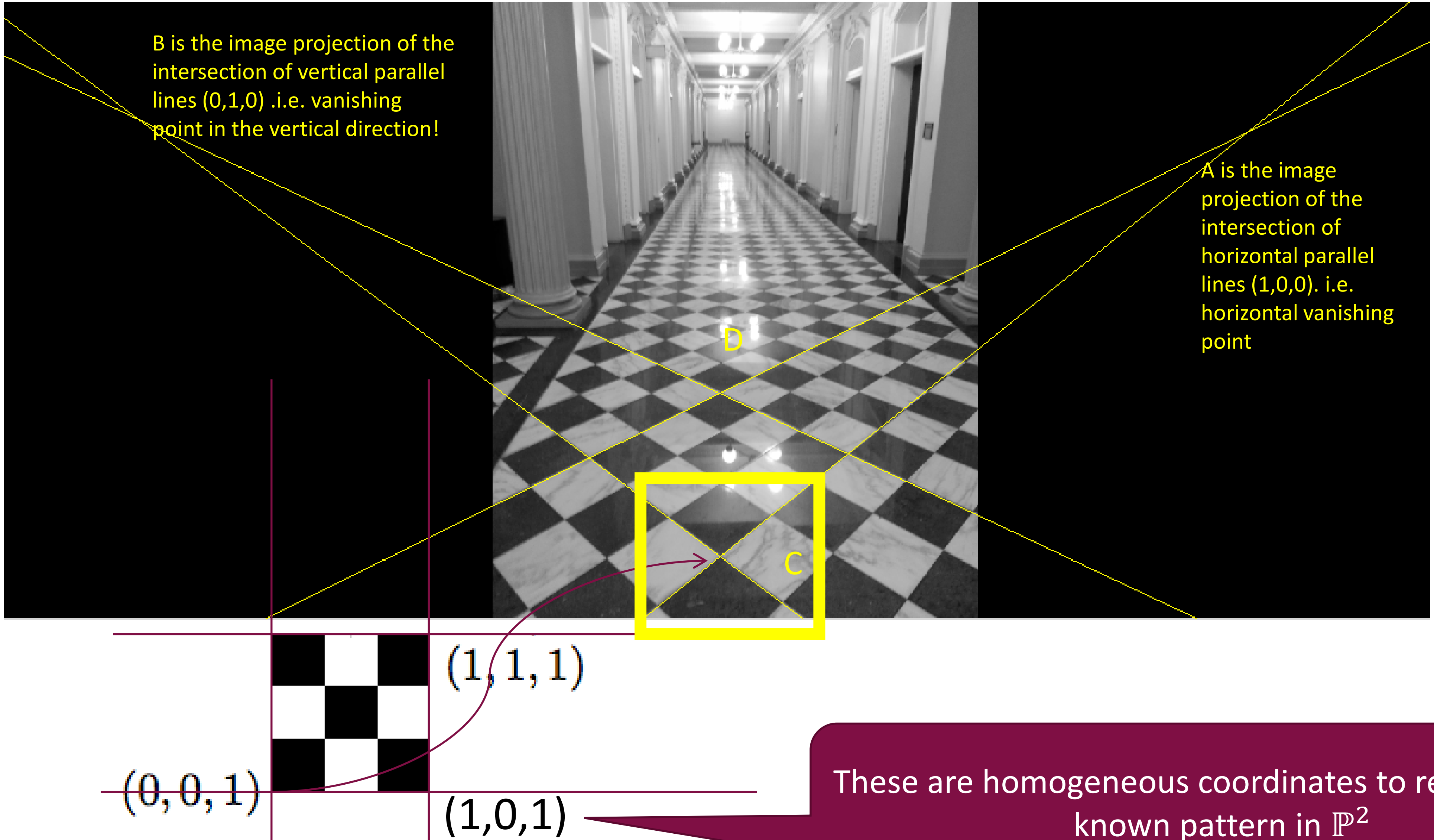
$(\mathbb{P}^2 \to \mathbb{P}^2)$

A projective transformation μH is the same as H since they map to projectively equivalent points:

$$\mu\lambda p' = \mu \; Hp$$

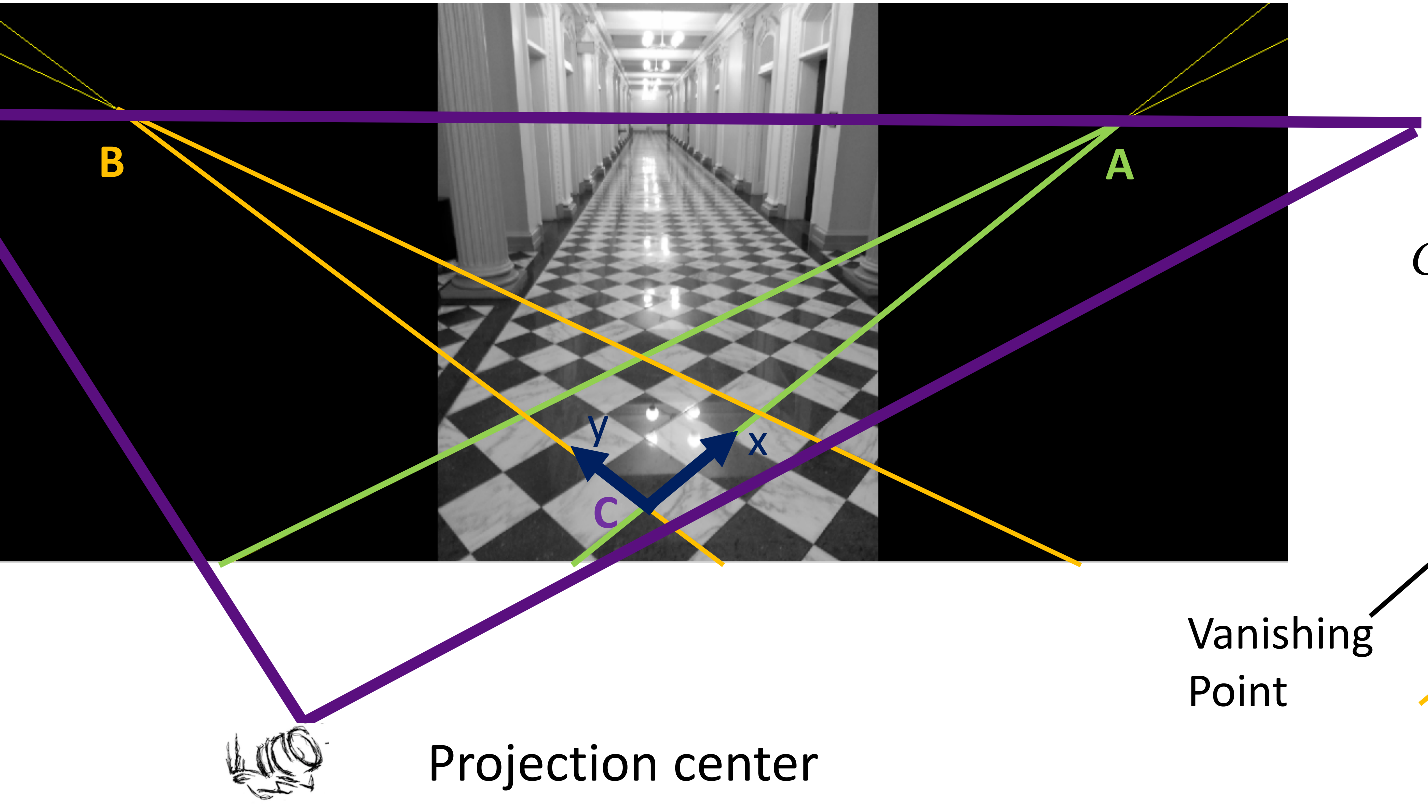
We will be able to determine a projective transformation only up to a scale factor. Hence the 3x3 invertible matrix H will have only EIGHT independent unknowns.

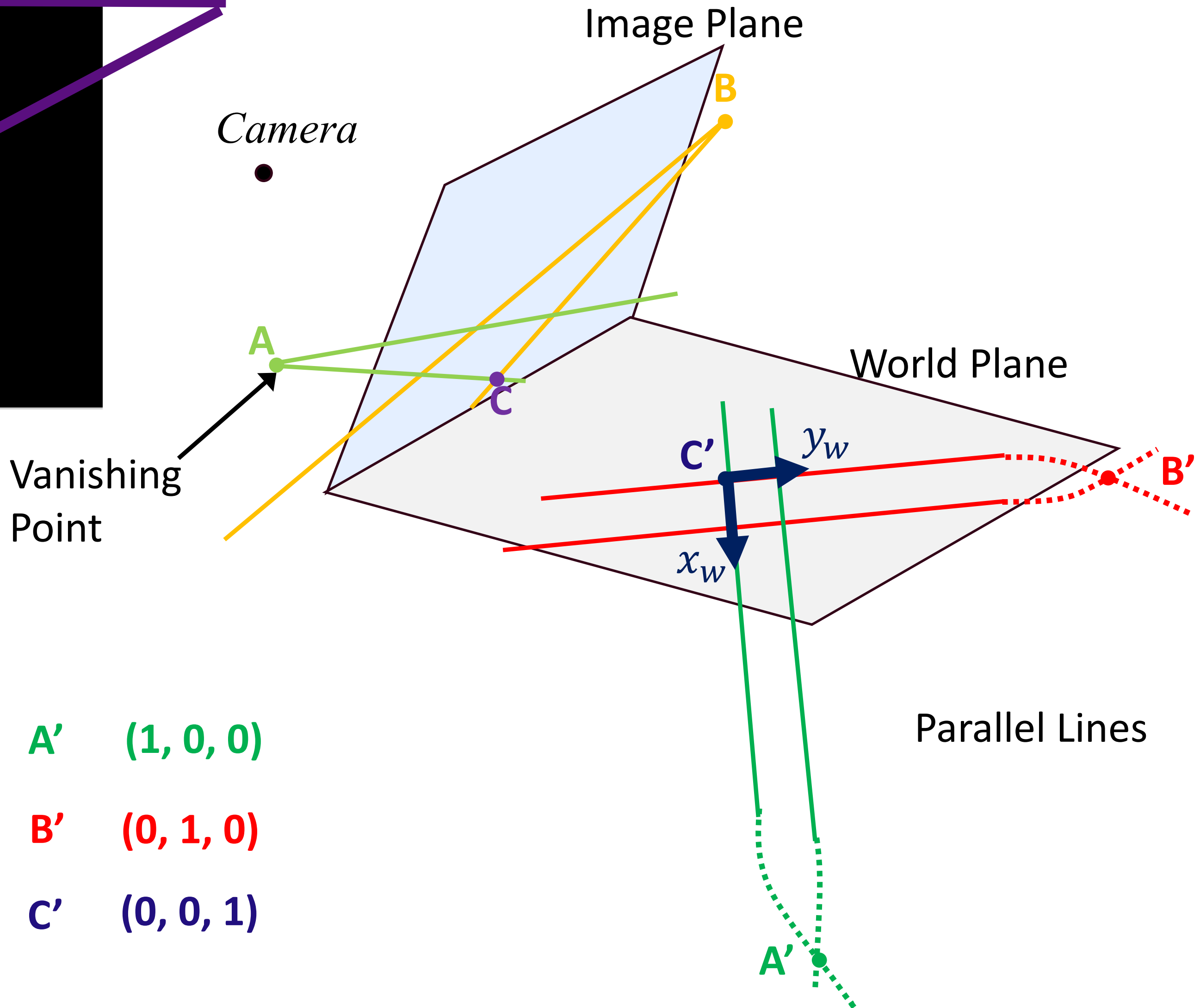
How can we compute the projective transformation between a known pattern and its projection?



These are homogeneous coordinates to represent the known pattern in $\mathbb{P}^2$

Cont.





Correspondences:
($P^2 \rightarrow P^2$)

A	a	$\longleftrightarrow$	A'	(1, 0, 0)
B	b	$\longleftrightarrow$	B'	(0, 1, 0)
C	c	$\longleftrightarrow$	C'	(0, 0, 1)

Assume that a mapping H maps the three points $(1,0,0)$, $(0,1,0)$, and $(0,0,1)$ to the non-collinear points A,B,C

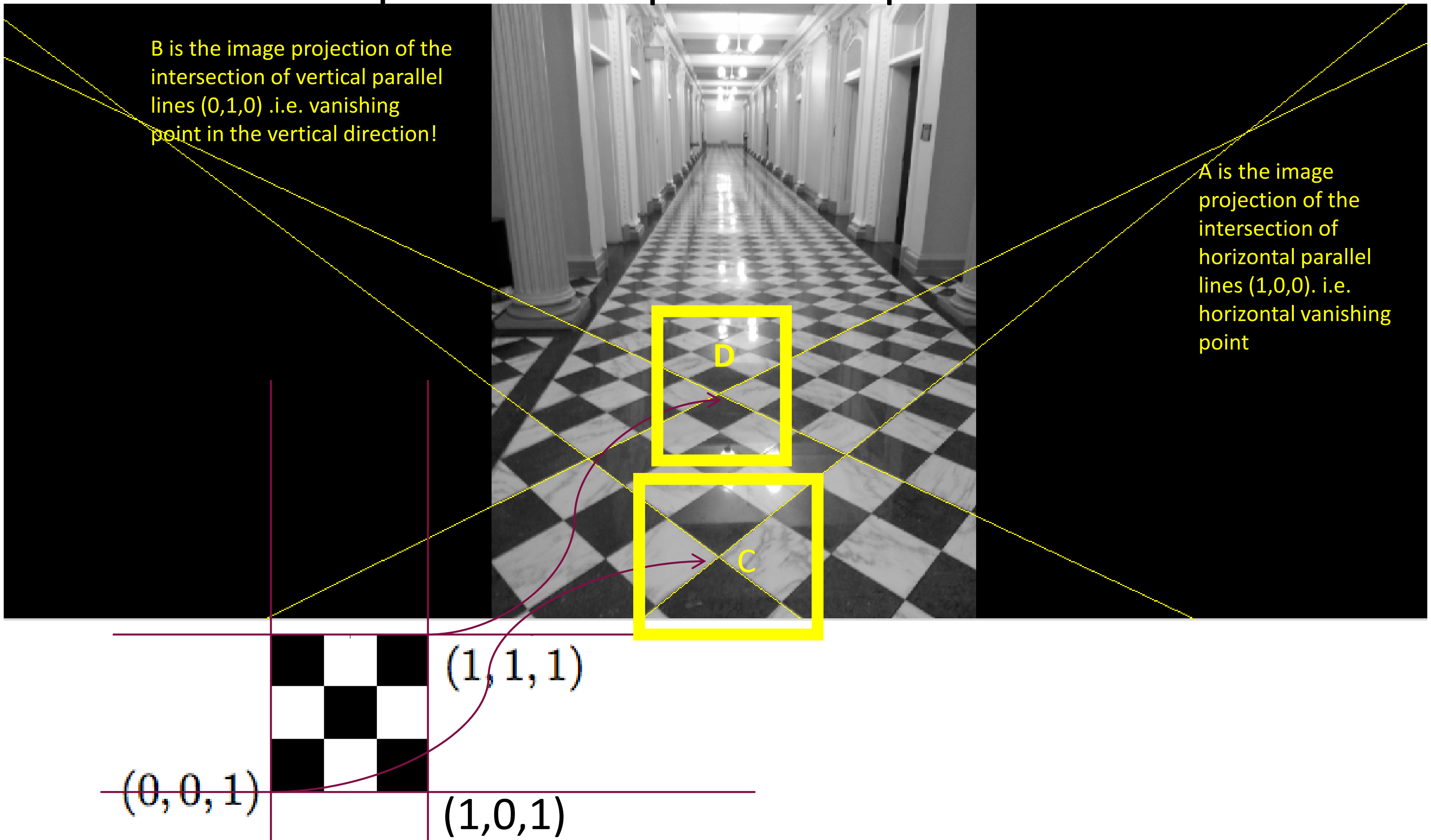
with coordinate vectors a, b and $c \in \mathbb{P}^2$. Then the following is a possible projective transformation:

$$\begin{array}{ccc} A & B & C \\ \end{array} \qquad \qquad \begin{array}{ccc} A' & B' & C' \end{array}$$

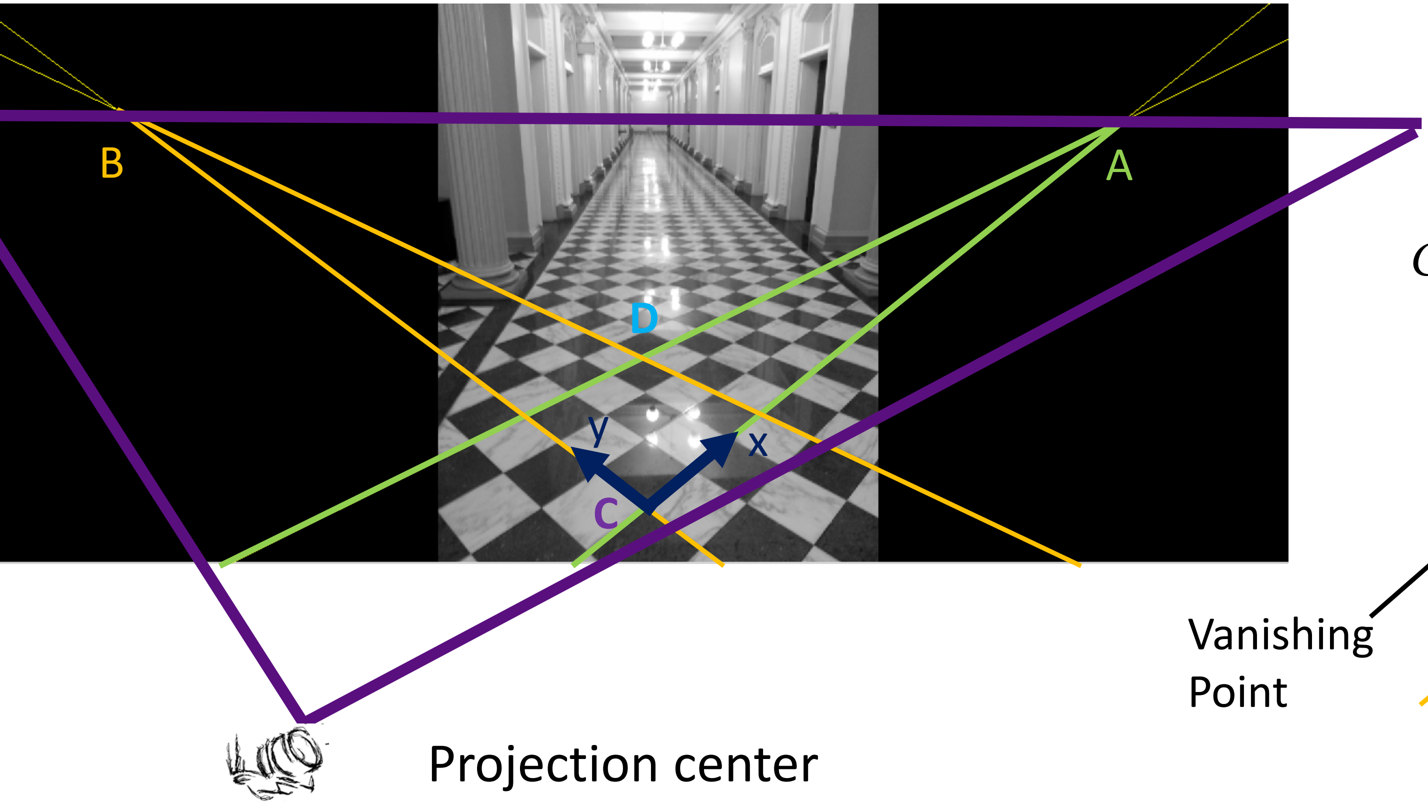
$$\begin{pmatrix} \boxed{a} & \boxed{b} & \boxed{c} \end{pmatrix} \sim H_{3\times 3} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

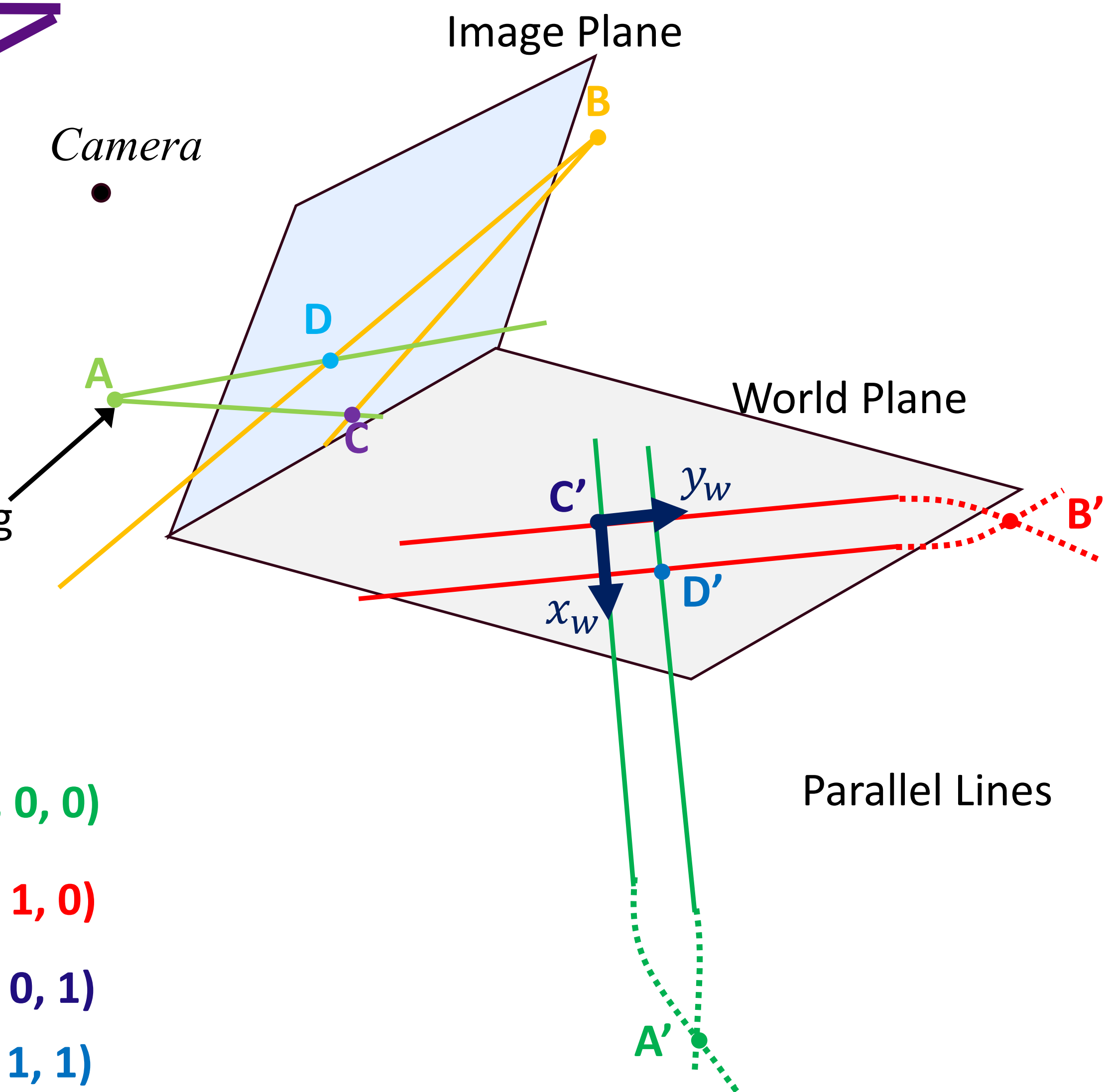
Solution: Introduce a 4th point correspondence D

Note: makes sense, because after all, H has 8 degrees of freedom, and each 2D point correspondence pins down 2DOF.



Cont.





Correspondences:
($P^2 \rightarrow P^2$)

A	a	⟷	A'	(1, 0, 0)
B	b	⟷	B'	(0, 1, 0)
C	c	⟷	C'	(0, 0, 1)
D	d	⟷	D'	(1, 1, 1)

Let us assume that the same H maps $(1,1,1)$ to the point d. Then, the following should hold:

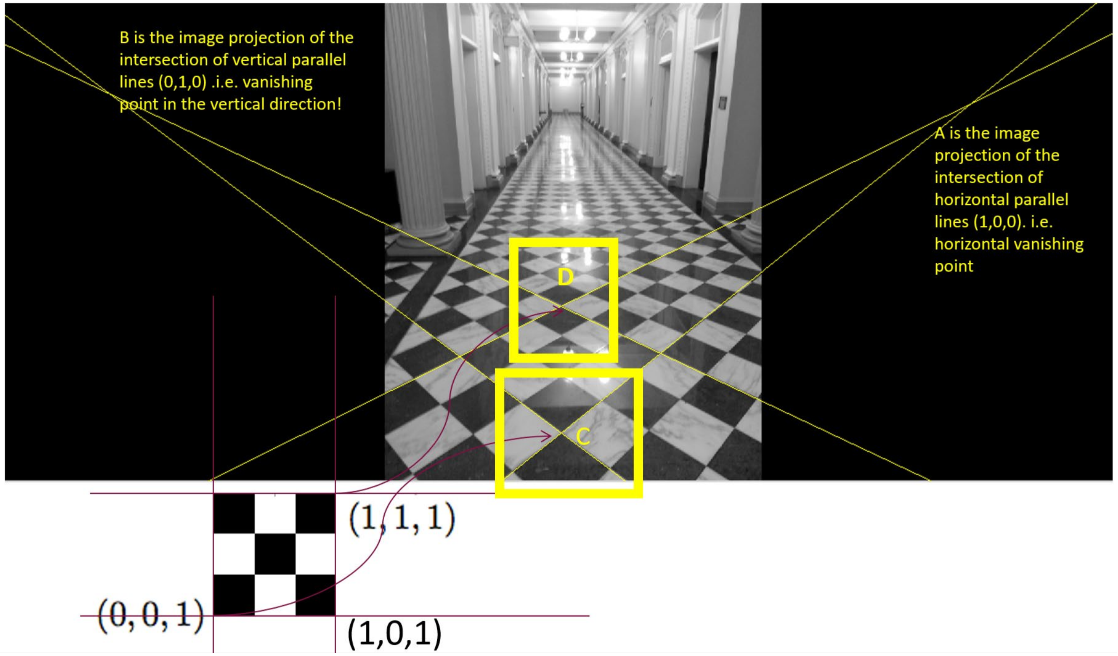
$$d \sim \underbrace{\begin{pmatrix} \alpha a & \beta b & \gamma c \end{pmatrix}}_{H_{3\times 3}} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},$$

hence

$$d \sim \alpha a + \beta b + \gamma c. \quad \text{or} \quad \lambda d = \alpha a + \beta b + \gamma c$$

Because a, b, c are not collinear, there exist unique $\alpha/\lambda, \beta/\lambda, \gamma/\lambda$ for writing this linear combination.

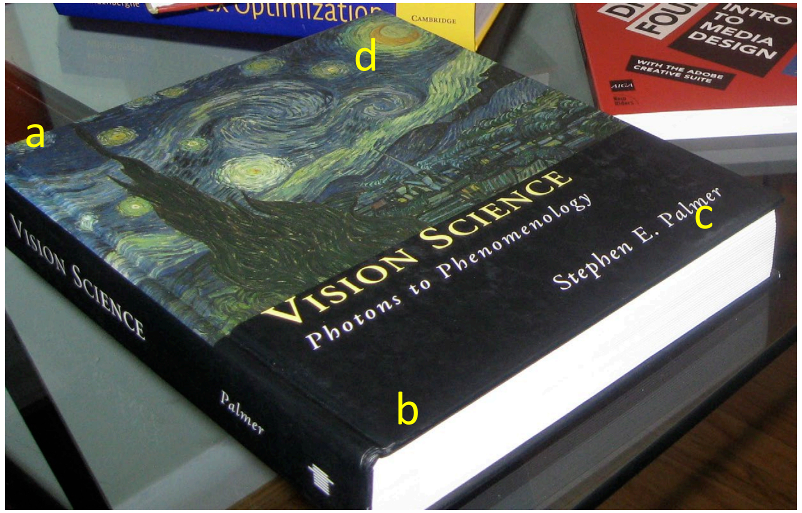
Four points, no three of them collinear, suffice to unambiguously recover a homography



Choosing the points to be the horizontal and vertical vanishing points (1,0,0), (0,1,0) plus origin (0,0,1) and the diagonal (1,1,1) is particularly “nice” especially if you have a square to start from, but really, any four non-collinear points will do.

(coming up next)

What happens when the original set of points is not a square?



c

Find projective transformation mapping $(a, b, c, d) \rightarrow (a', b', c', d')$:

To determine this mapping we go through the four canonical points.

We find the mapping from $(1,0,0), etc$ to (a,b,c,d) and we call it T:

$$a \sim T(1,0,0)^T, etc$$

We find the mapping from $(1,0,0), etc$ to (a',b',c',d') and we call it T':

$$a' \sim T'(1,0,0)^T, etc$$

Then, back-substituing $(1,0,0)^T \sim T^{-1}a, etc$ we obtain that

$$a' = T'T^{-1}a, etc$$

To determine this mapping we go through the four canonical points.

We find the mapping from $(1,0,0), etc$ to (a,b,c,d) and we call it T:

$$a \sim T(1,0,0)^T, etc$$

We find the mapping from $(1,0,0), etc$ to (a',b',c',d') and we call it T':

$$a' \sim T'(1,0,0)^T, etc$$

Then, back-substituing $(1,0,0)^T \sim T^{-1}a, etc$ we obtain that

$$a' = T'T^{-1}a, etc$$

To determine this mapping we go through the four canonical points.

We find the mapping from $(1,0,0), etc$ to (a,b,c,d) and we call it T:

$$a \sim T(1,0,0)^T, etc$$

We find the mapping from $(1,0,0), etc$ to (a',b',c',d') and we call it T':

$$a' \sim T'(1,0,0)^T, etc$$

Then, back-substituing $(1,0,0)^T \sim T^{-1}a, etc$ we obtain that

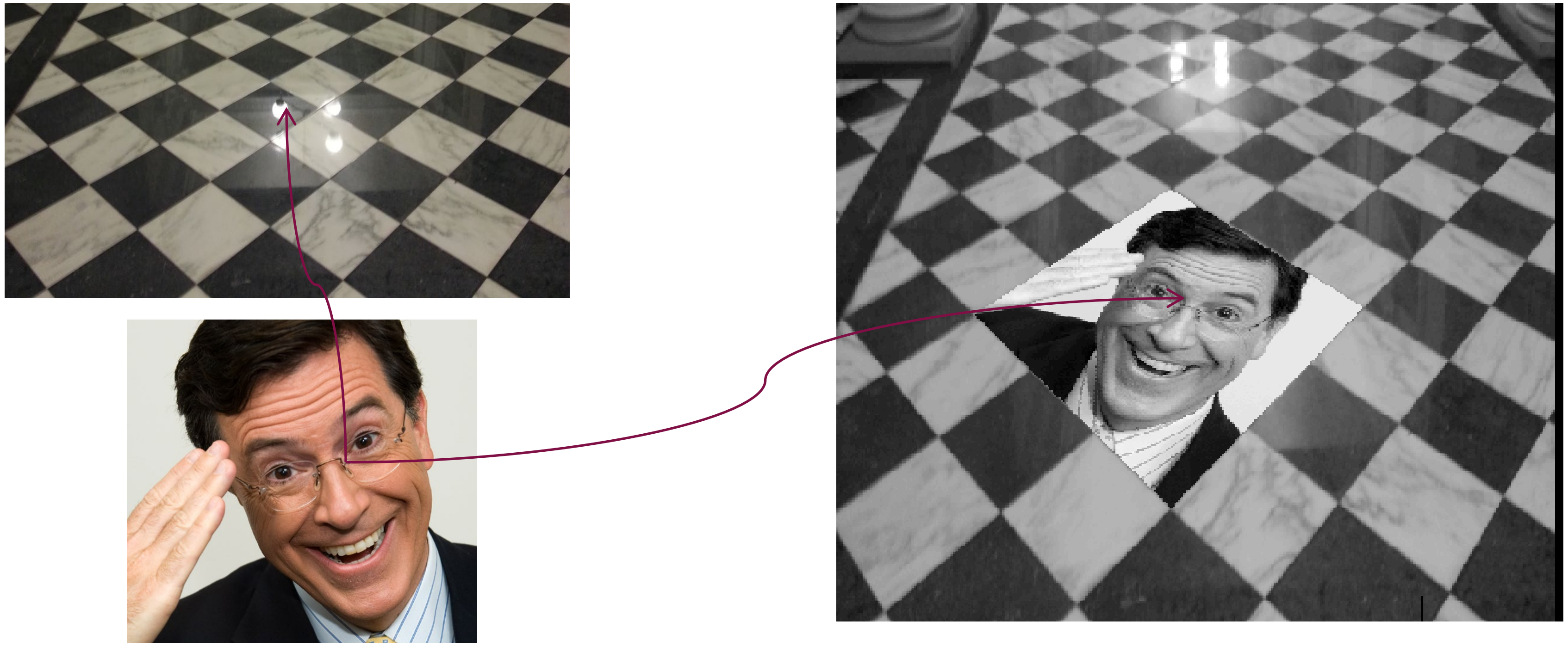
$$a' = T'T^{-1}a, etc$$

This amounts to computing homographies from a made-up square, then inverting one of them and composing!

Homography -> Virtual Billboards

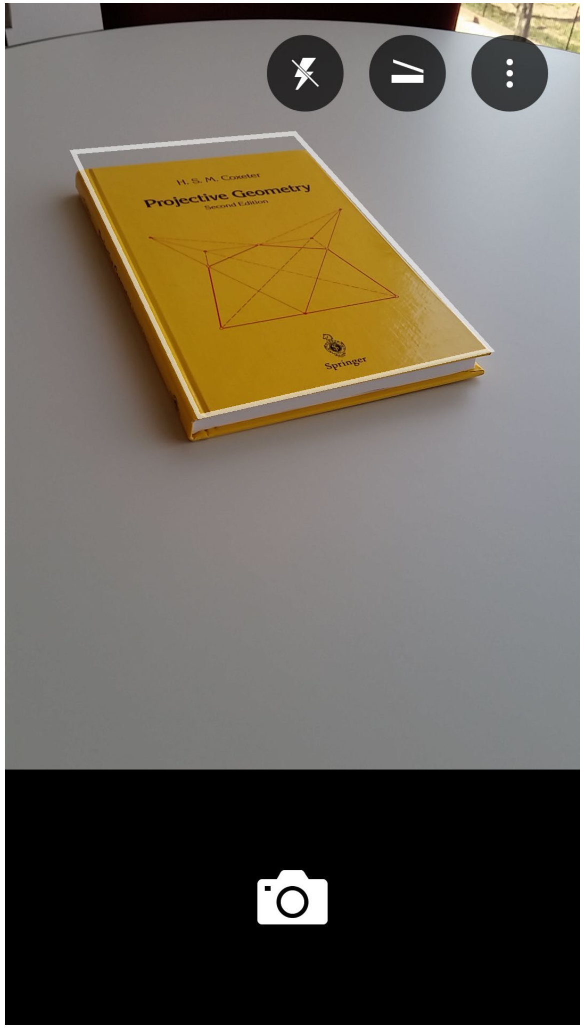
- For virtual billboards, we just treat the desired billboard pattern as the world pattern.
- Recall that the homography is a mapping from world to image pixel coordinates. So, just the homography can directly find the image pixel coordinate corresponding to every image in the world pattern.

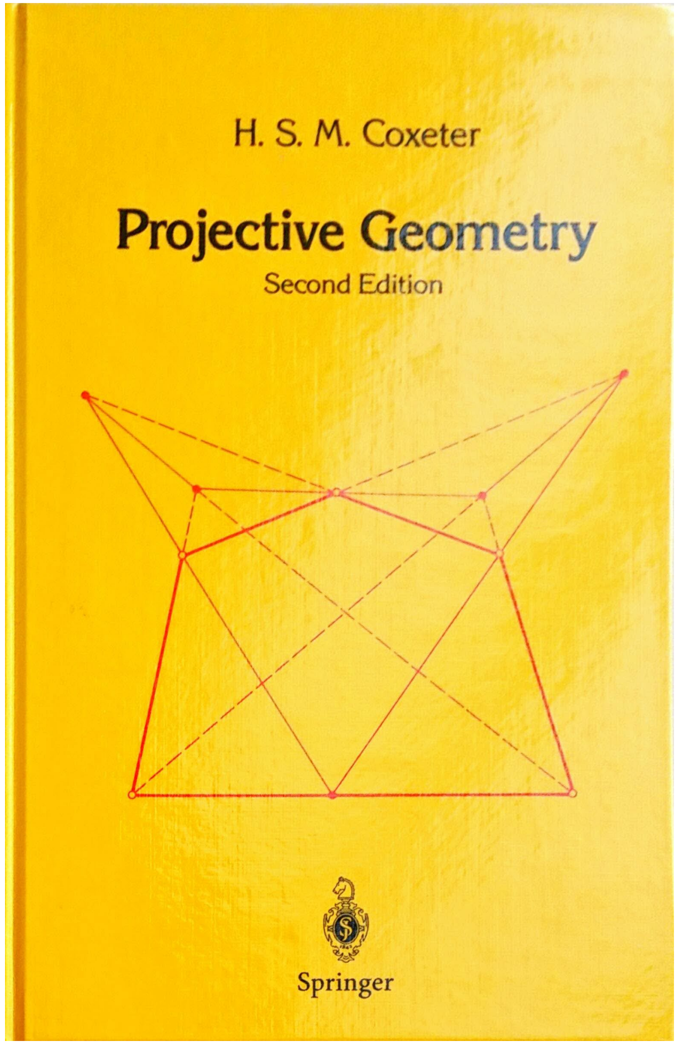
(Q: is this actually what we need for virtual billboards? A: not quite.)



Microsoft Office Lens App

Office Lens


H. S. M. Coxeter
Projective Geometry
Second Edition
Springer


H. S. M. Coxeter
Projective Geometry
Second Edition
Springer

Computing homographies with more correspondences

$$\mathbf{x}' \sim H\mathbf{x}$$

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

$$\lambda x' = h_{11}x + h_{12}y + h_{13}$$

$$\lambda y' = h_{21}x + h_{22}y + h_{23}$$

$$\lambda = h_{31}x + h_{32}y + h_{33}$$

Converting to a linear system of equations

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}$$

$$y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}$$

$$-h_{11}x - h_{12}y - h_{13} + h_{31}xx' + h_{32}yx' + h_{33}x' = 0$$
$$-h_{21}x - h_{22}y - h_{23} + h_{31}xy' + h_{32}yy' + h_{33}y' = 0$$

$$\begin{pmatrix} a_x \\ a_y \end{pmatrix} h = 0$$

Two linear equations for each point correspondence!

$$a_x = \begin{pmatrix} -x & -y & -1 & 0 & 0 & 0 & xx' & yx' & x' \end{pmatrix}$$
$$a_y = \begin{pmatrix} 0 & 0 & 0 & -x & -y & -1 & xy' & yy' & y' \end{pmatrix}$$
$$h = \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{21} & h_{22} & h_{23} & h_{31} & h_{32} & h_{33} \end{pmatrix}^T$$

Computing homographies with more correspondences

Our matrix H has 8 degrees of freedom, and so, as each point gives 2 sets of equations, we will need 4 points to solve for h uniquely. So, given four points (such as the corners provided for this assignment), we can generate vectors a_x and a_y for each, and concatenate them together:

$$A = \begin{pmatrix} a_{x,1} \\ a_{y_1} \\ \vdots \\ a_{x,n} \\ a_{y,n} \end{pmatrix}_{2n \times 9}$$

$$Ah = 0$$

When $n > 4$, due to noise in measurements, there may not be an $h \neq 0$ that produces exactly $Ah = 0$.

Instead, we find the vector s.t. $||Ah||_2$ is smallest.

This involves computing the SVD (singular value decomposition) of A, and setting h to be the “smallest” right singular vector, with smallest singular value.
(more info on SVD etc. soon)

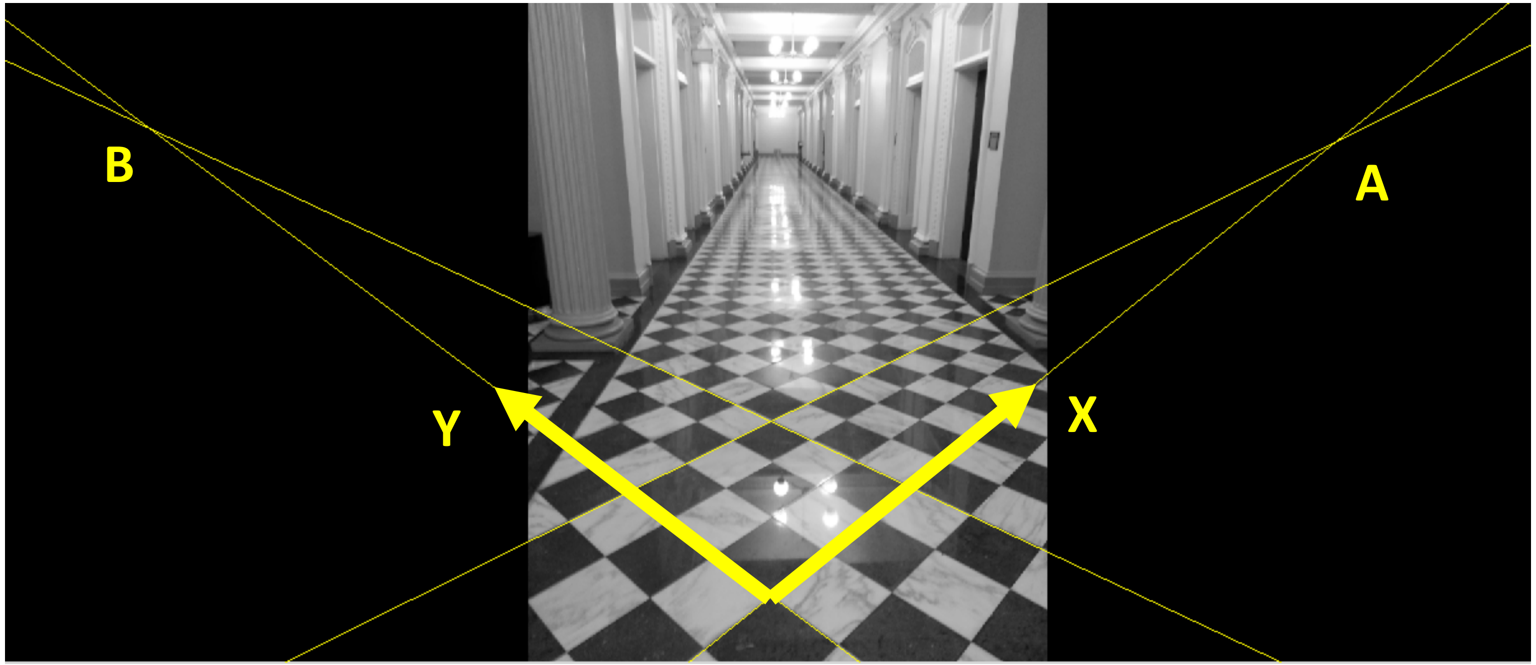
Using homographies to measure lengths!

Motivation: making measurements using H

- Homographies (i.e. World Plane -> Image homographies) allow us to measure lengths in the world plane of interest.

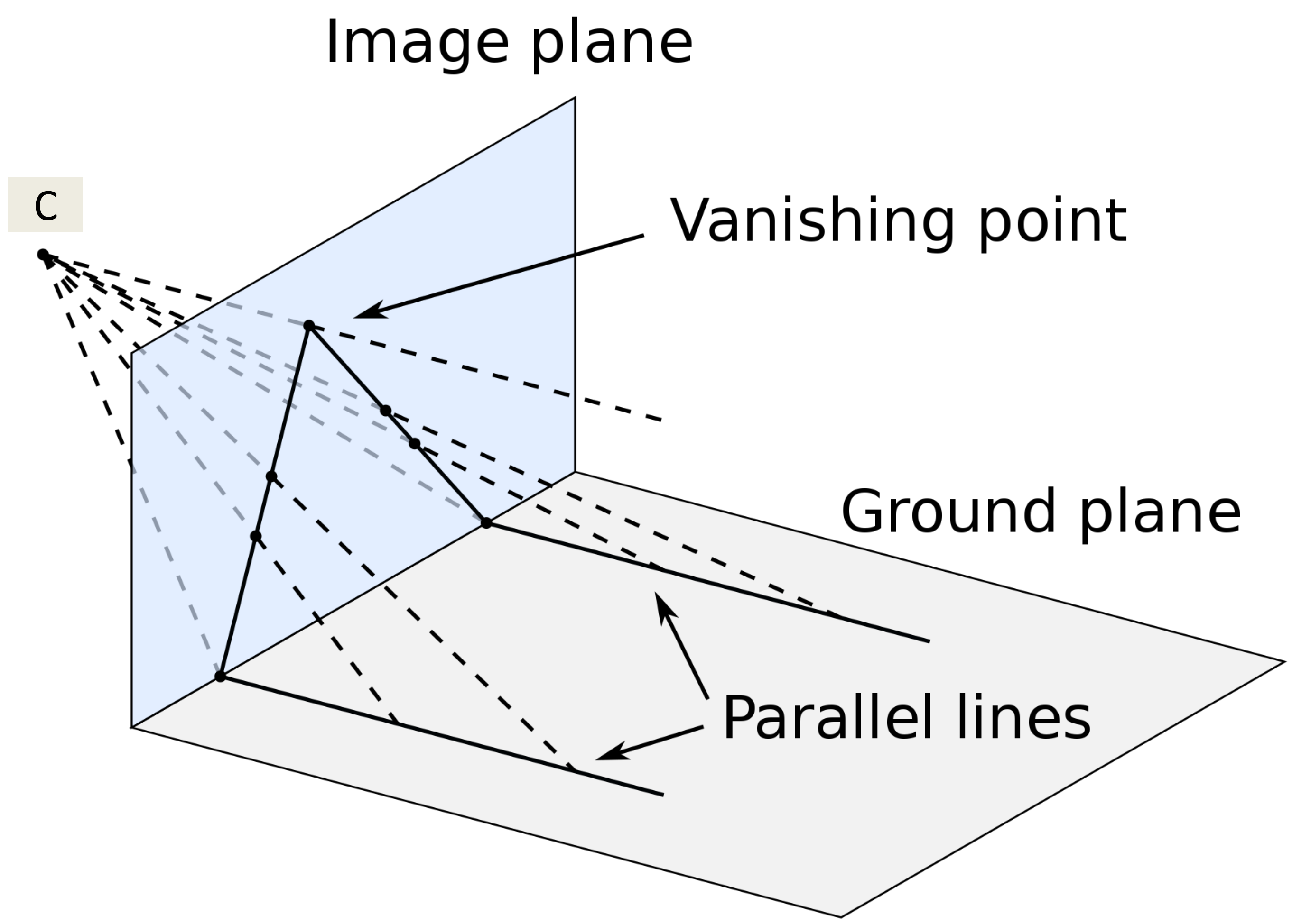
Recall: homography columns are vanishing points

If $H = \begin{pmatrix} h_1 & h_2 & h_3 \end{pmatrix}$ then $h_1 \sim A$ and $h_2 \sim B$.



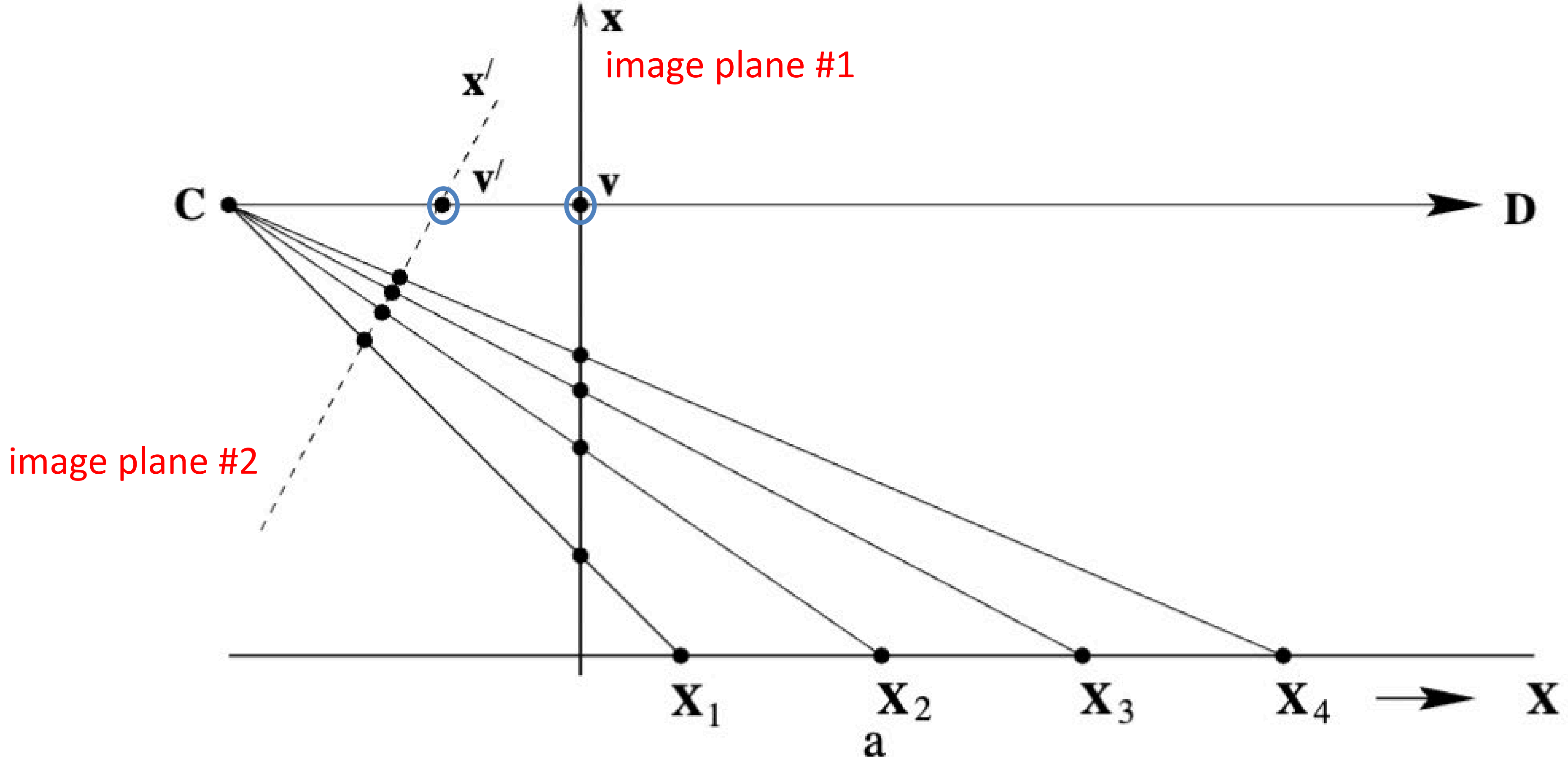
So the first two columns are two Orthogonal vanishing points

Recall: Vanishing Point



ZHCh8

The line connecting the camera origin and the vanishing point is parallel to all lines that share the same direction and converge at the vanishing point.

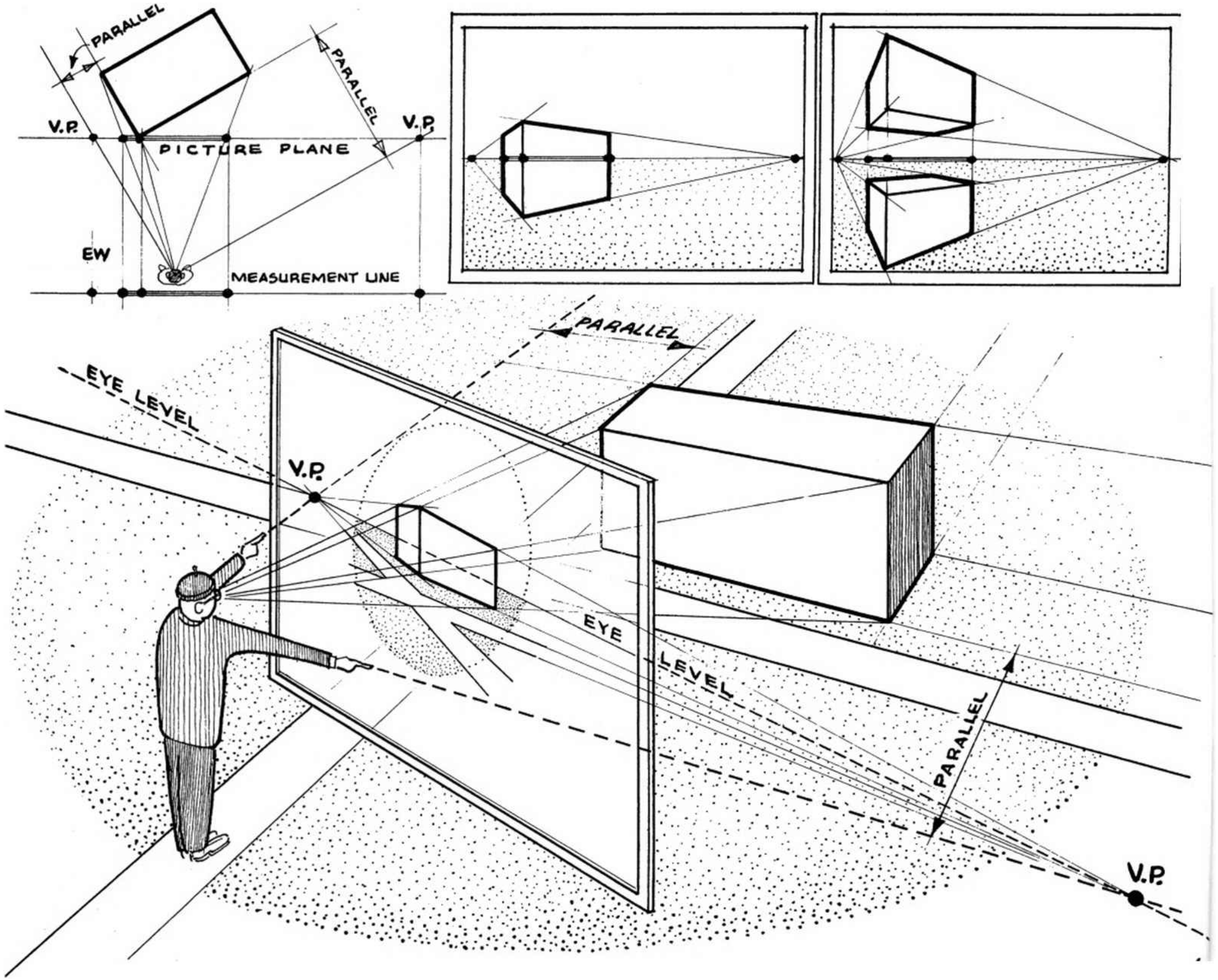


How Artists Find Vanishing Points

Find VP of a world line by:

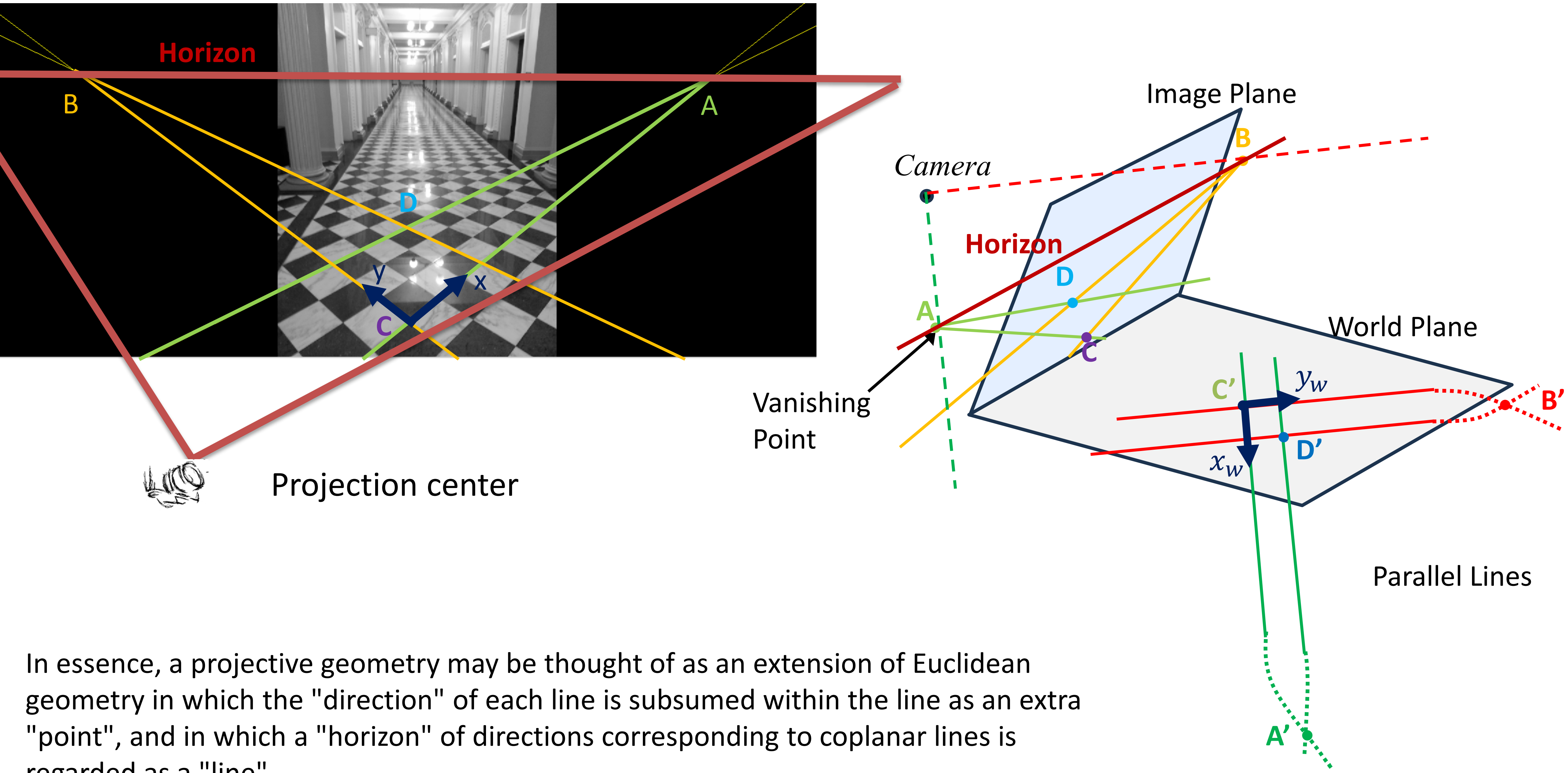
- Standing at "camera center".
- Holding arm out parallel to the world line.
- Noting its intersection with the "canvas" or image plane. i.e. the arm represents the light ray.

"Vanishing rays of a world line" (camera rays through the VP) are just rays parallel to that line, passing through the camera center.



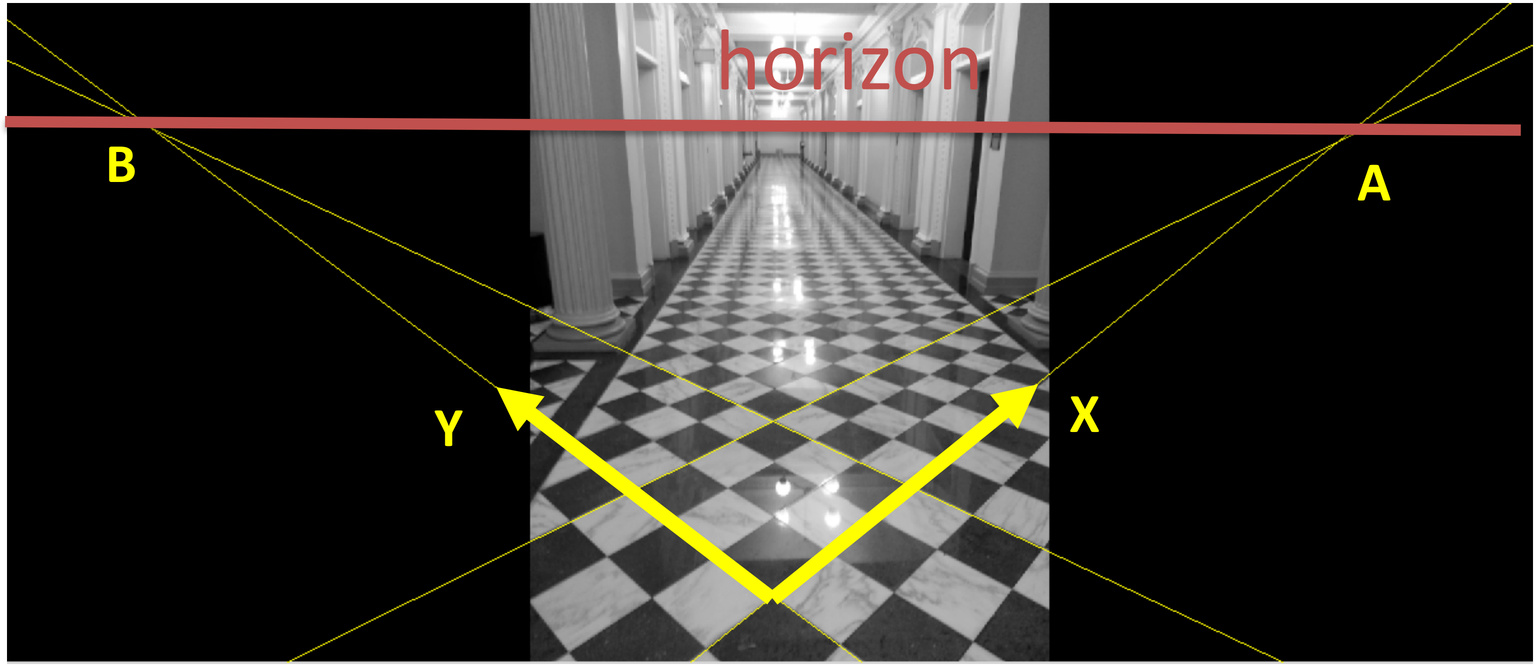
http://www.joshuanava.biz/perspective/in-other-words-the-observer-simply-points-in-the-same-direction-as-the-lines-in-order-to-find-their-vanishing-point.html

Horizon



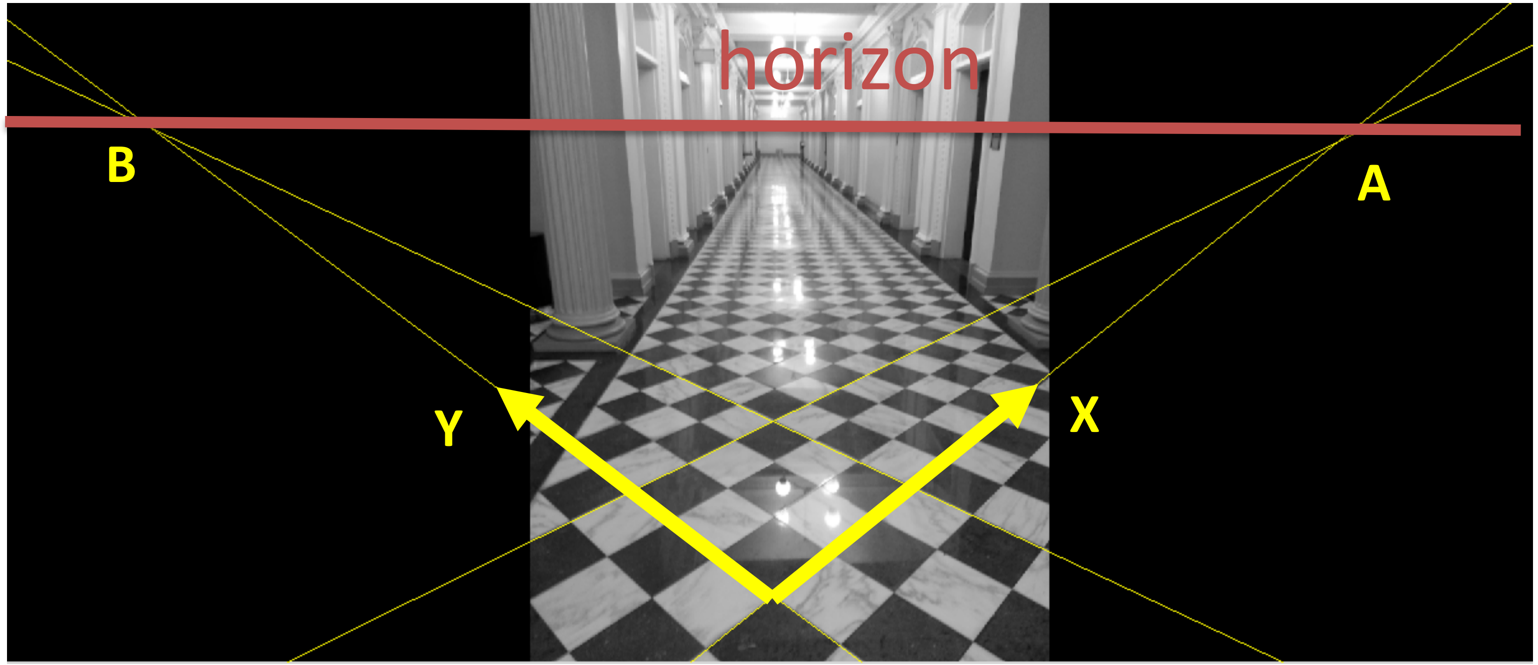
In essence, a projective geometry may be thought of as an extension of Euclidean geometry in which the "direction" of each line is subsumed within the line as an extra "point", and in which a "horizon" of directions corresponding to coplanar lines is regarded as a "line"

If we connect two vanishing points, we obtain the "horizon"!



Side note: with our assumption of the world plane as being the "XY" plane, and following the common convention that xy plane is horizontal, and z is vertical, this indeed maps to our normal notions of a "horizon".

Equation of horizon: $(h_1 \times h_2)^T \begin{pmatrix} x \\ y \\ z \end{pmatrix} = 0$



We will encounter another way to derive this equation of the horizon very soon.

Projecting the line at infinity to compute the horizon

Points at infinity in the world plane look like $(X, Y, W = 0)^T$

The "line" connecting them is $W = 0$, the "line at infinity". The image of this line is the horizon, which contains all vanishing points. Expressed in world plane $\mathbb{P}^2$, this line's coefficients are $(0,0,1)^T$.

So if we could find the projection of this line, we could find the horizon

Deriving the equation of a horizon in another way

Q: We know planar projections H transform points $p \in \mathbb{P}^2$ as $p \rightarrow Hp$. How do they transform lines in $\mathbb{P}^2$?

Projective Transformation of Lines

If H maps a point to Hp, then where does a line l map to?

Line equation in original plane

$$l^T p = 0$$

Line equation in image plane where any point $p' \rightarrow Hp$

$$l^T H^{-1} p' = 0$$

Implies that $l' = H^{-T} l$

Projecting the line at infinity to compute the horizon

Points at infinity in the world plane look like $(X, Y, W = 0)^T$

The "line" connecting them is $W = 0$, the "line at infinity". The image of this line is the horizon, which contains all vanishing points. Expressed in world plane $\mathbb{P}^2$, this line's coefficients are $(0,0,1)^T$.

So if we could project this line, we could find the horizon

We have just seen projections of lines are $H^{-T}l$, so the horizon is $H^{-T}(0,0,1)^T$.

If $H = [\boldsymbol{h}_1\ \boldsymbol{h}_2\ \boldsymbol{h}_3]$ then H^{-T} is $[h_2 \times h_3\ h_3 \times h_1\ h_1 \times h_2]$, so the horizon line $H^{-T}(0,0,1)^{\mathrm{T}} = h_1 \times h_2$.

This is consistent: the horizon connects the two vanishing points h_1 and h_2.

Using the horizon to orient the camera

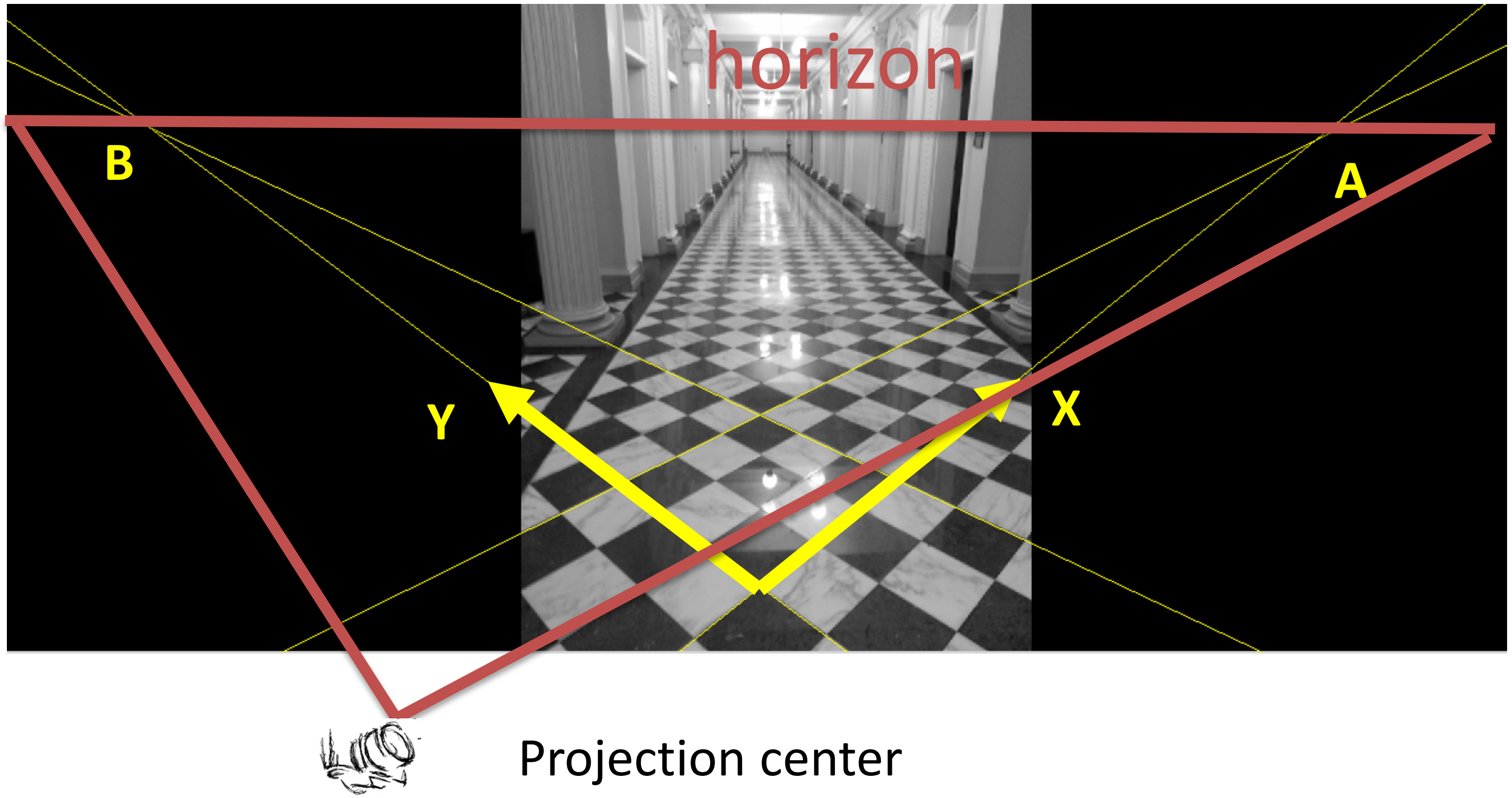
Recap:

Equation of horizon is $h_1 \times h_2$

$$=$$

Normal to the viewing plane through horizon is $h_1 \times h_2$

Plane through camera center and horizon line on image plane = vanishing plane of the world plane

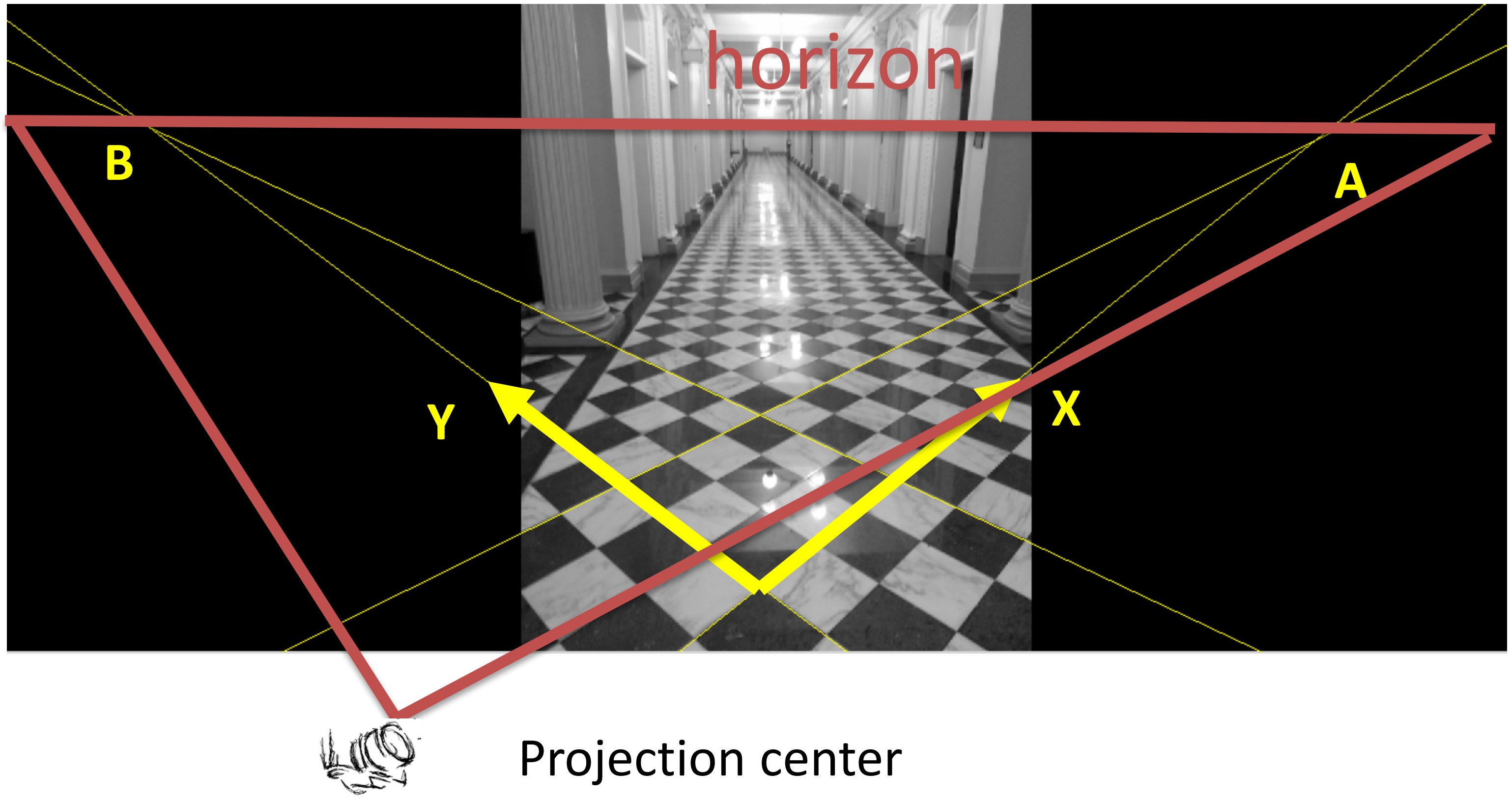


Projection center

Recap:

Vanishing rays/planes through the camera center are parallel to the world lines/planes

So, the horizon plane is parallel to the ground plane
and hence $h_1 \times h_2$ is the normal to the ground plane!



Projection center

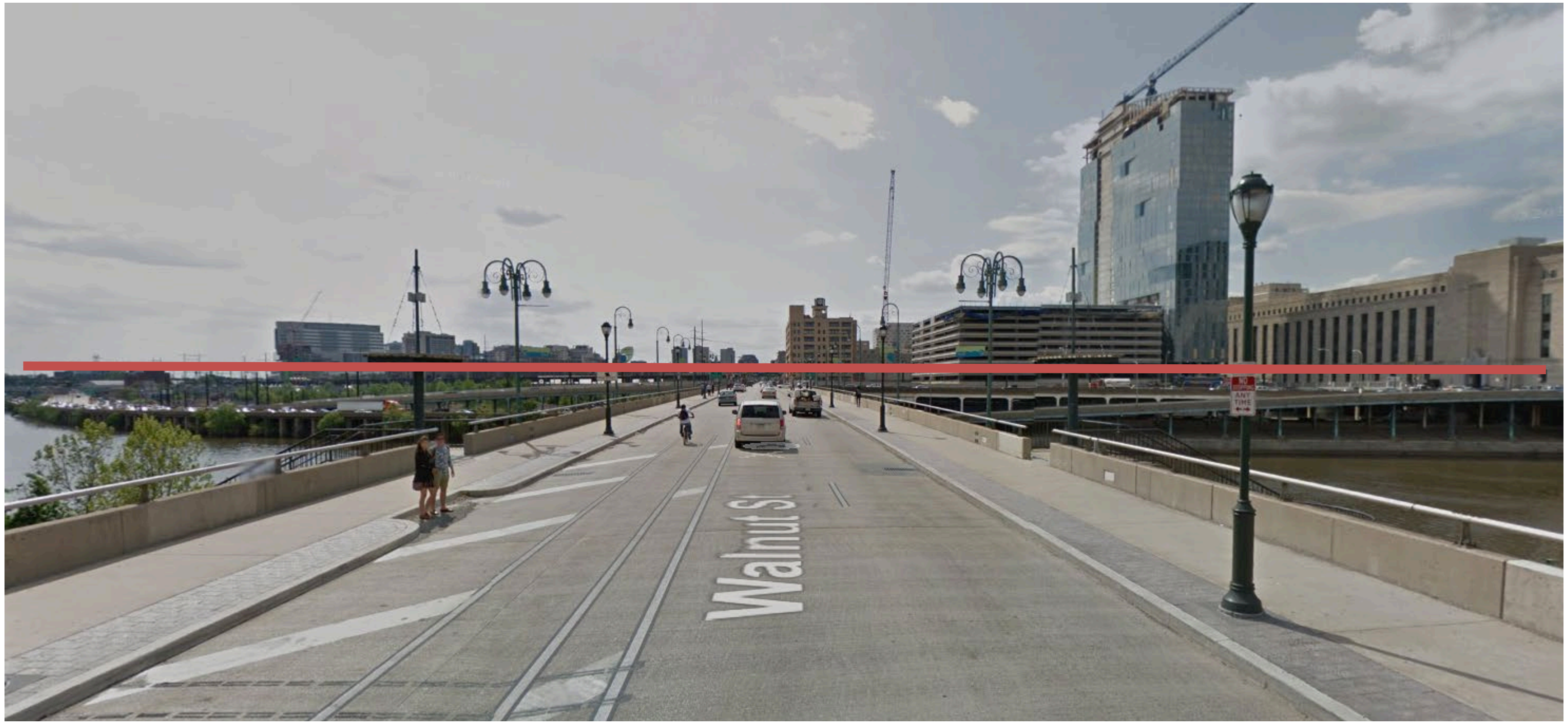
Horizon plane =
Vanishing plane =
Viewing plane

World plane //
vanishing plane

World plane = Ground
plane in this case

Horizon gives complete info about how ground plane is oriented*!

Thumb rule: “If horizon is horizontal & central, camera is correctly vertical & principal axis is parallel to world plane**!”



*caveat: assuming known K

**caveat: assuming that principal axis passes through image center, and sensor axes are horizontal. (usually approximately true)