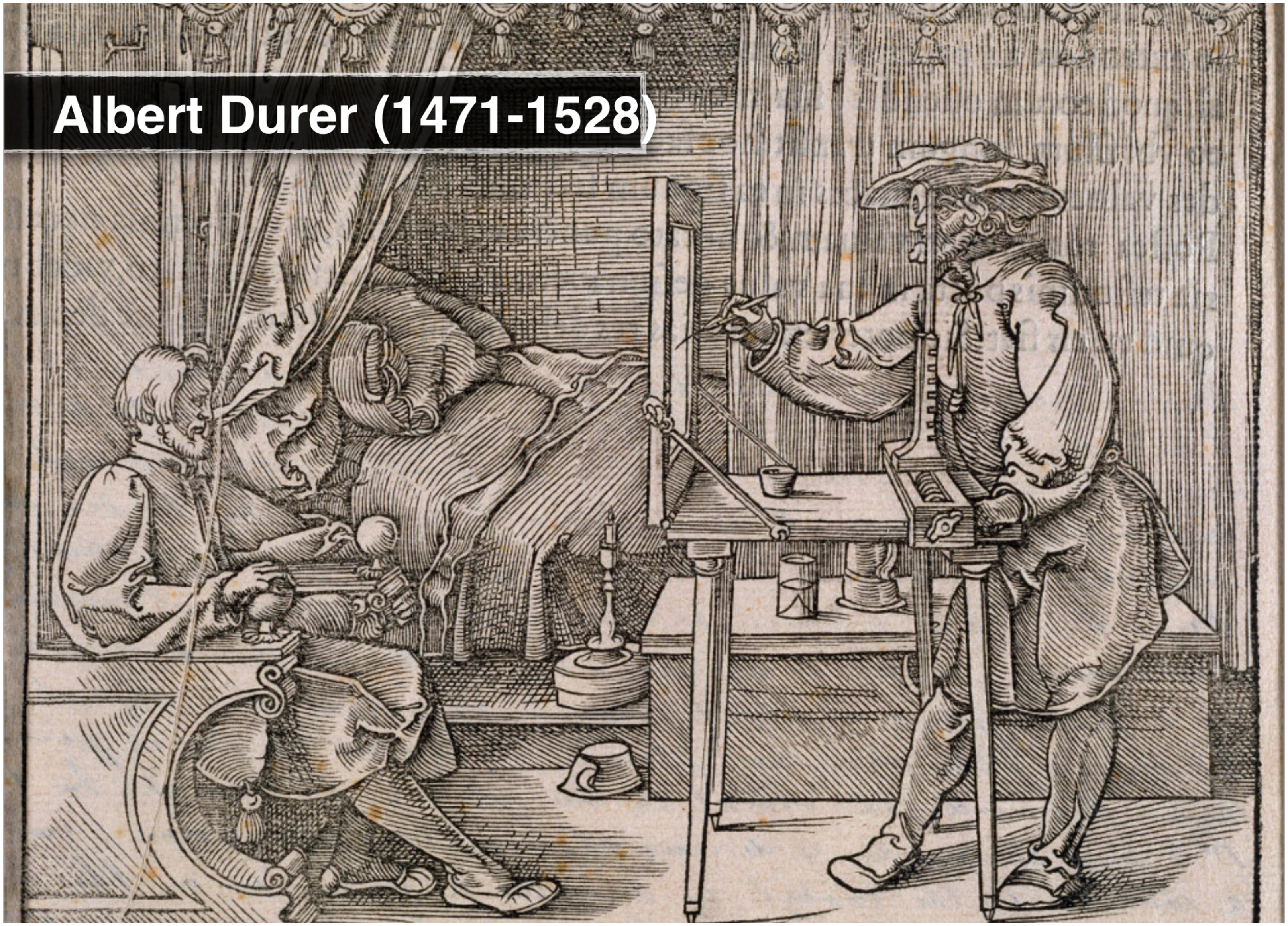
CIS 580 Machine Perception Spring 2021

- Resources:
  - Website (-> ppt pdfs, note pdfs, readings)
  - Canvas (-> Zoom, Recordings, Piazza, Gradescope)
- No Final, Homework: 60%, Midterm 1 (Mar 8): 20%, Midterm 2 (Apr 29): 20%
- The only way to fail this class is to violate honors code (cheat, plagiarize)
  - Zero Tolerance. Think before you act: The chance that one homework or one question in midterm affects your final grade is epsilon.
- Learning Outcomes:
  - Firm knowledge of fundamentals of computer vision: geometry and image processing (plus some Deep Learning approaches in Geometry)
  - Understand challenges, why algorithms work or do not work
  - Perform as a vision engineer in vision and robotics companies
  - 99% of deployed geometric algorithms work without Deep Learning

Perspective Projection

CIS 580 2021

Albert Durer (1471-1528)



History of Perspective Projection

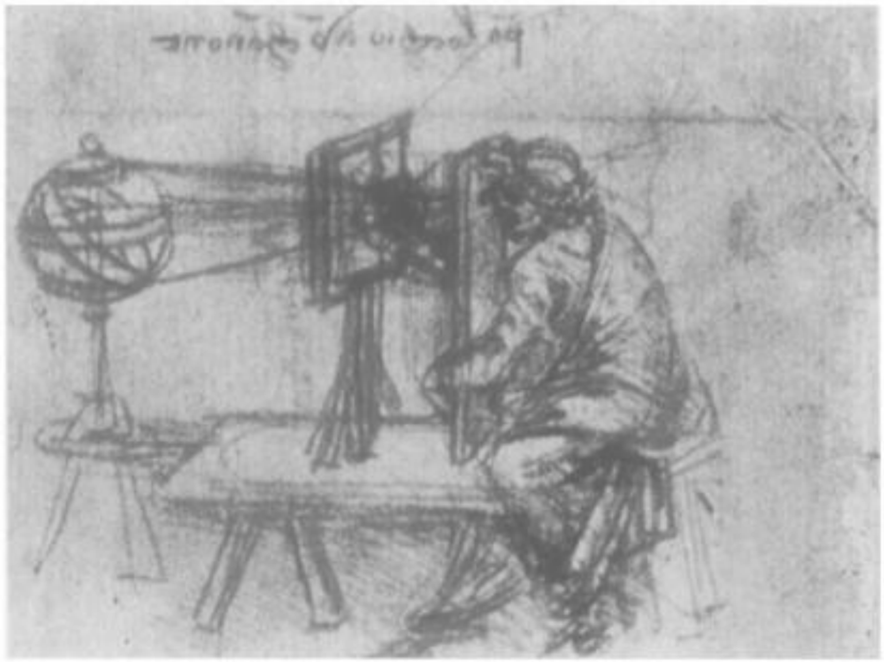


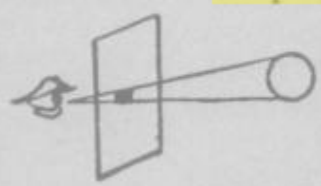
Figure 2.1. Leonardo's technique for making a perspectival drawing of the sphere of the macrocosm (CA 1 ra bis).

Leonardo da Vinci

PARIETE DI VETRO

Demonstration of perspective by means of a vertical glass plane (83–5).

[2]Prospettiva · non è altro · che vedere [3]uno sito dirieto uno vetro piano [4]e ben transparẽte, sulla superfitie del [5]quale · siano · segniate · tutte le cose · che [6]sono da esso · vetro · ĩdirieto: le qua[7]li si possono cõdurre per piramidi [8]al pũto dell' ochio e esse piramidi si [9]tagliano su detto vetro.



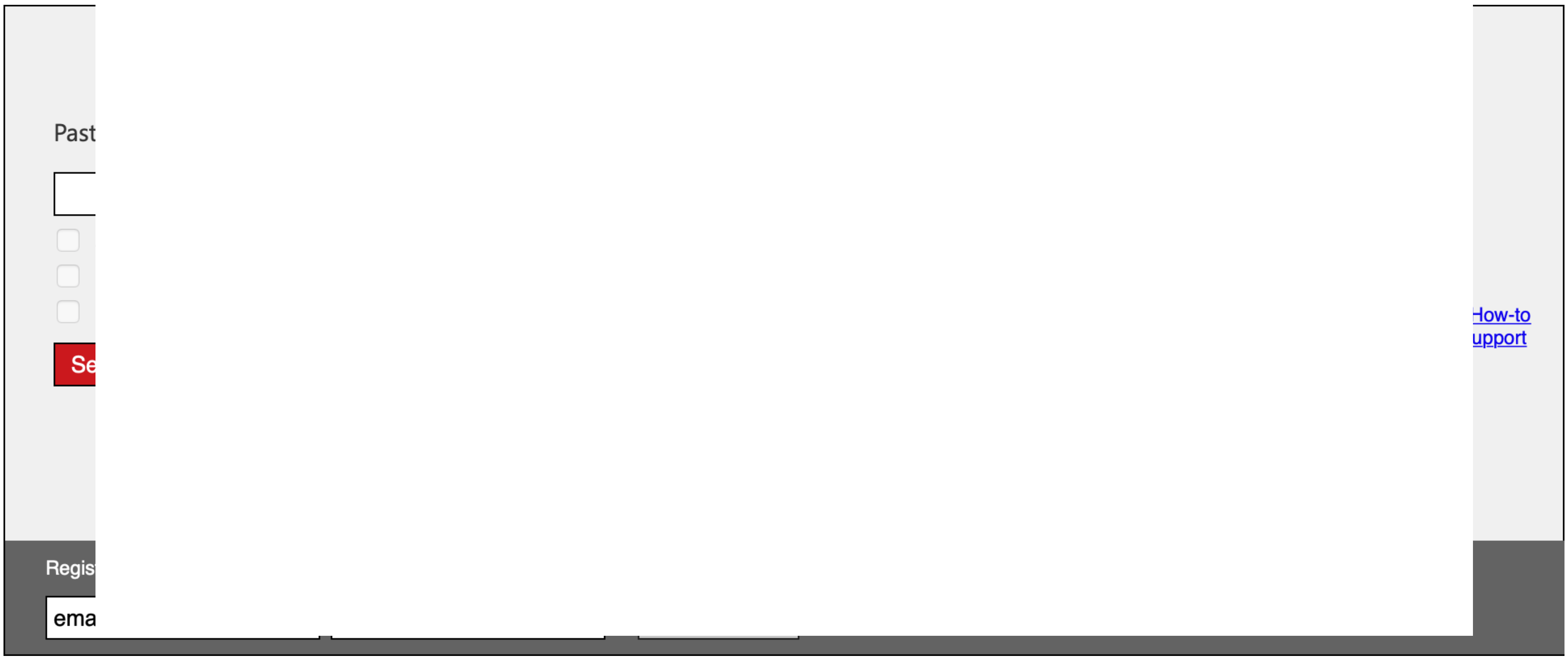
OF THE PLANE OF GLASS

Perspective is nothing else than seeing a place [or objects] behind a pane of glass, quite transparent, on the surface of which the objects behind that glass are to be drawn. These can be traced in pyramids to the point in the eye, and these pyramids are intersected on the glass plane.

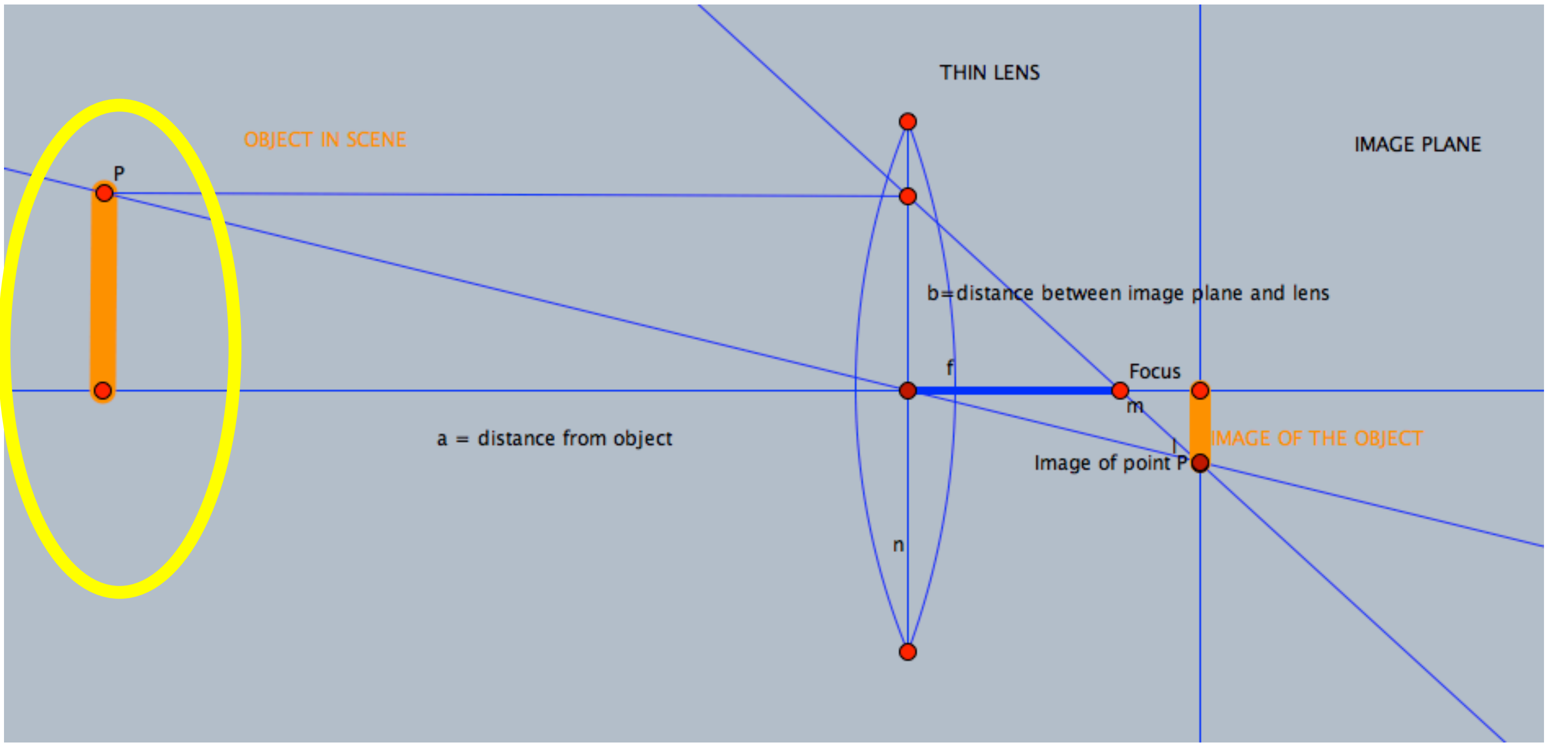
Figure 2.2. The principle of perspective (A 1 v). Figure from J. P. Richter, *The Literary Works of Leonardo da Vinci*, 3d edition (New York, Phaidon, 1970), vol. 1, p. 150.


LMSC

Ames Room

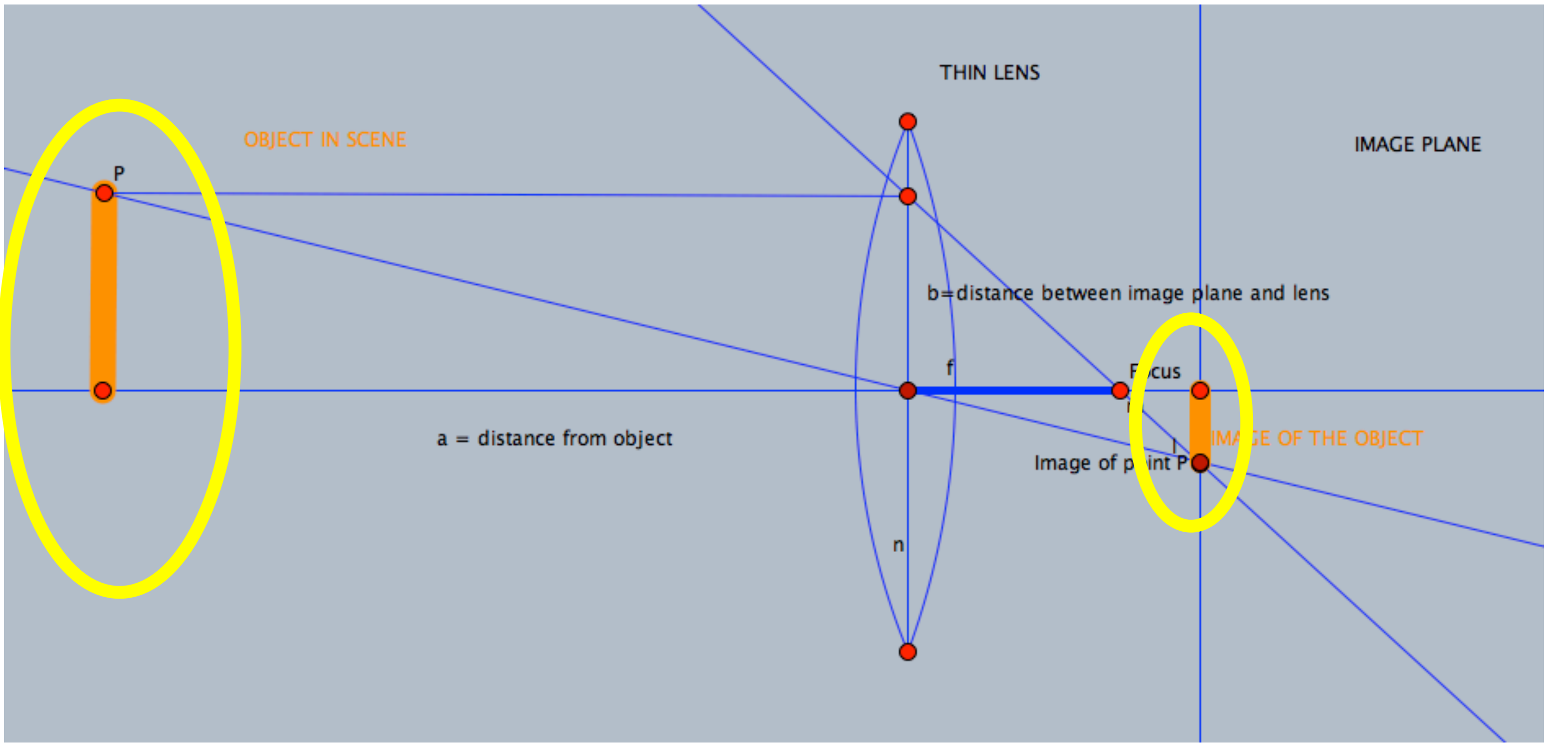


Model of a Thins Lens in today's cameras



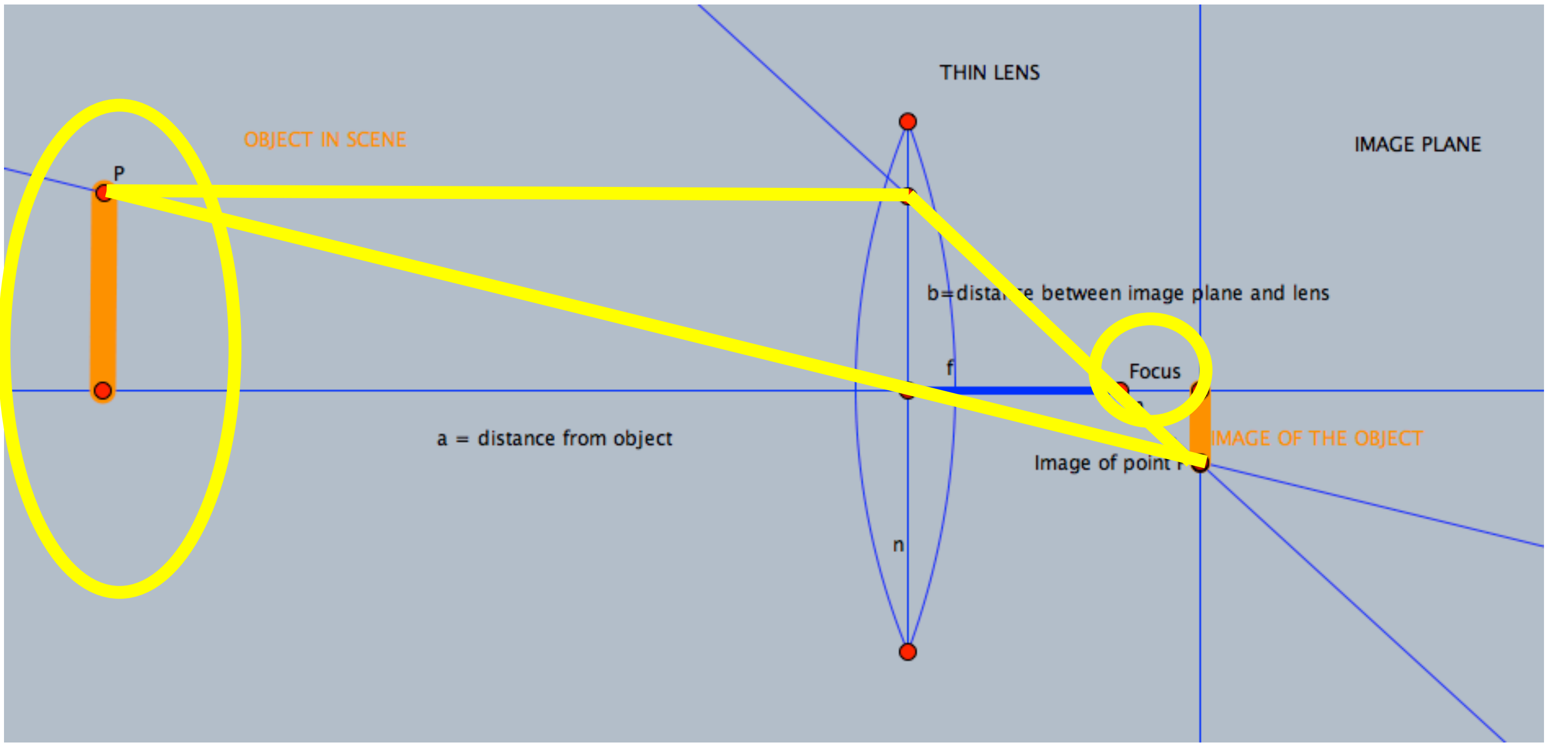
Rays from on object point P converge on a point p on the image plane

How does a thin lens work



Rays from a point P in the scene converge into a point in the image plane

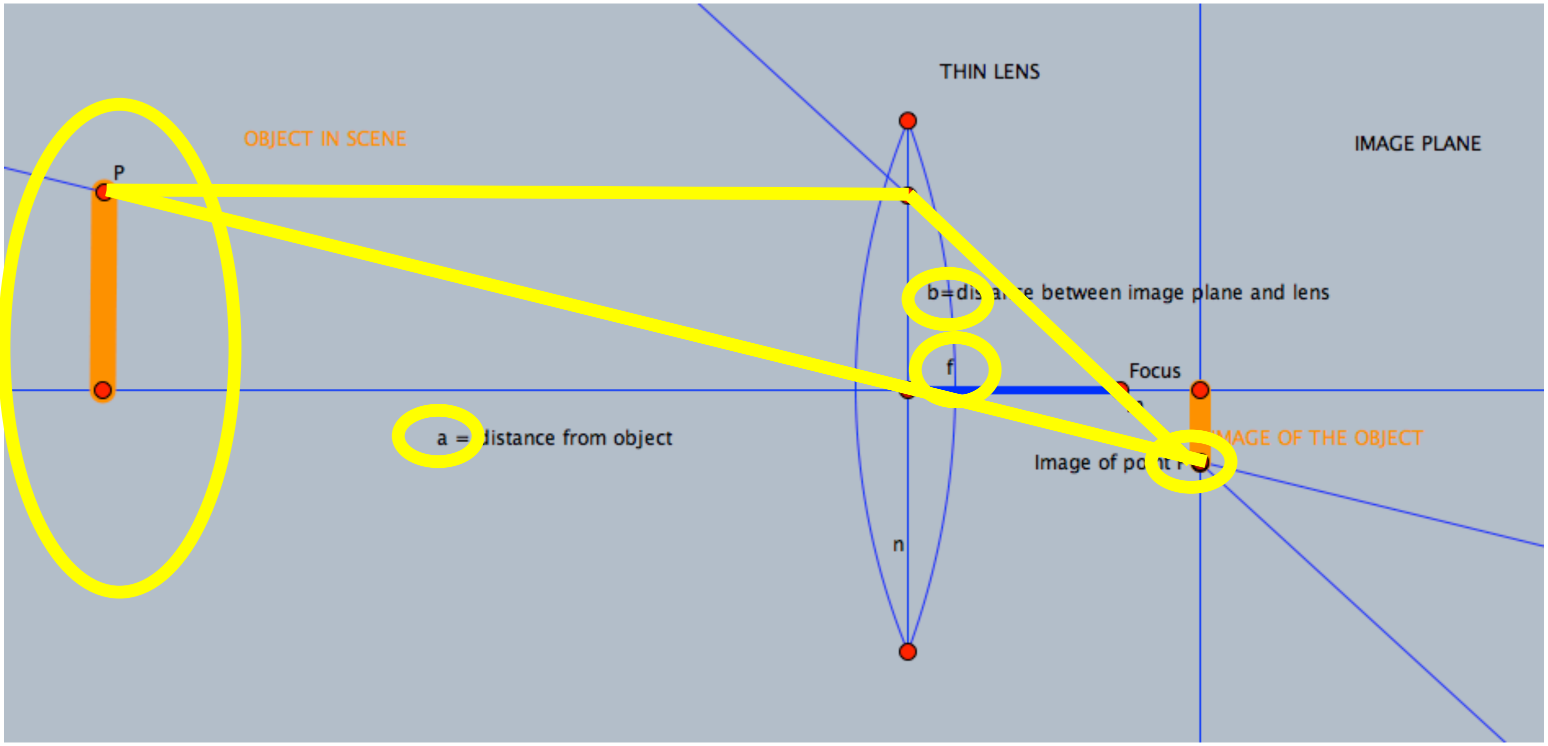
How does a thin lens work



Rays parallel to the optical axis meet the focus after leaving the lens.
Rays through center of the lens do not change direction.

$$\frac{1}{f} = \frac{1}{a} + \frac{1}{b}$$

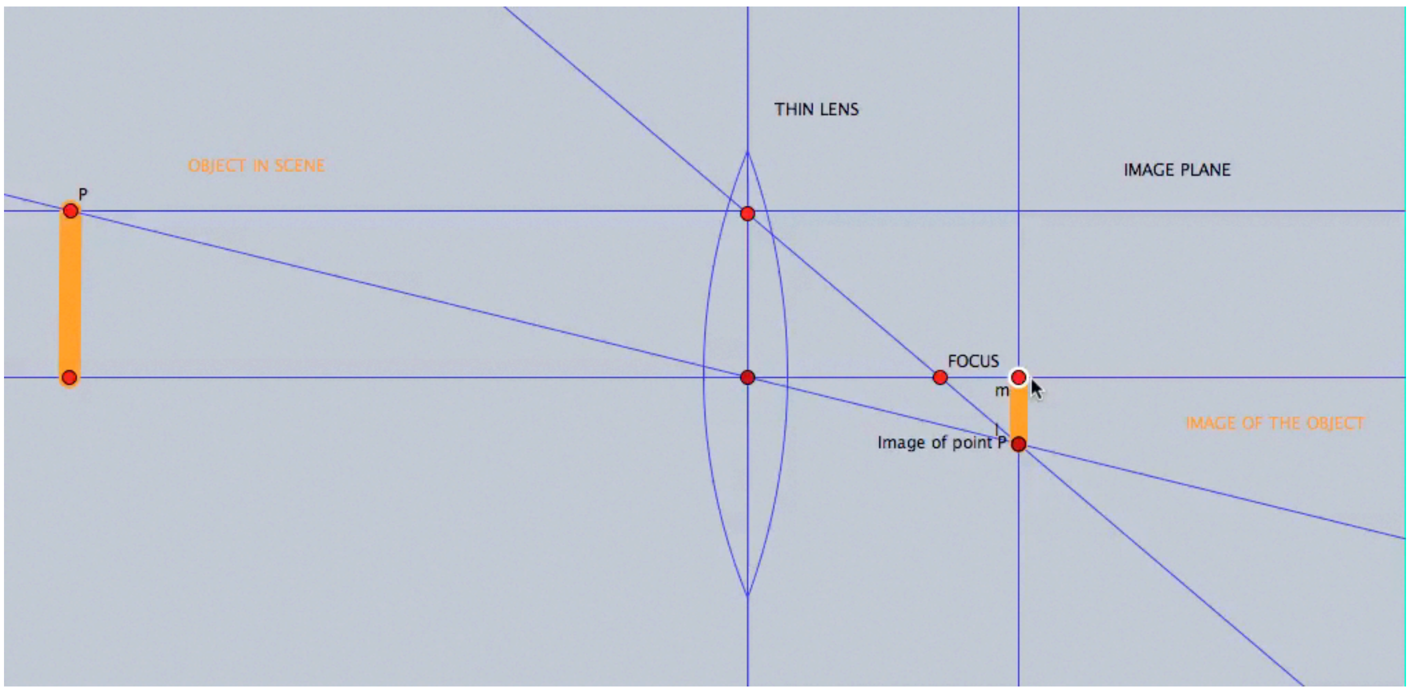
How does a thin lens work



These rays meet at one point if

$$\frac{1}{f} = \frac{1}{a} + \frac{1}{b}$$

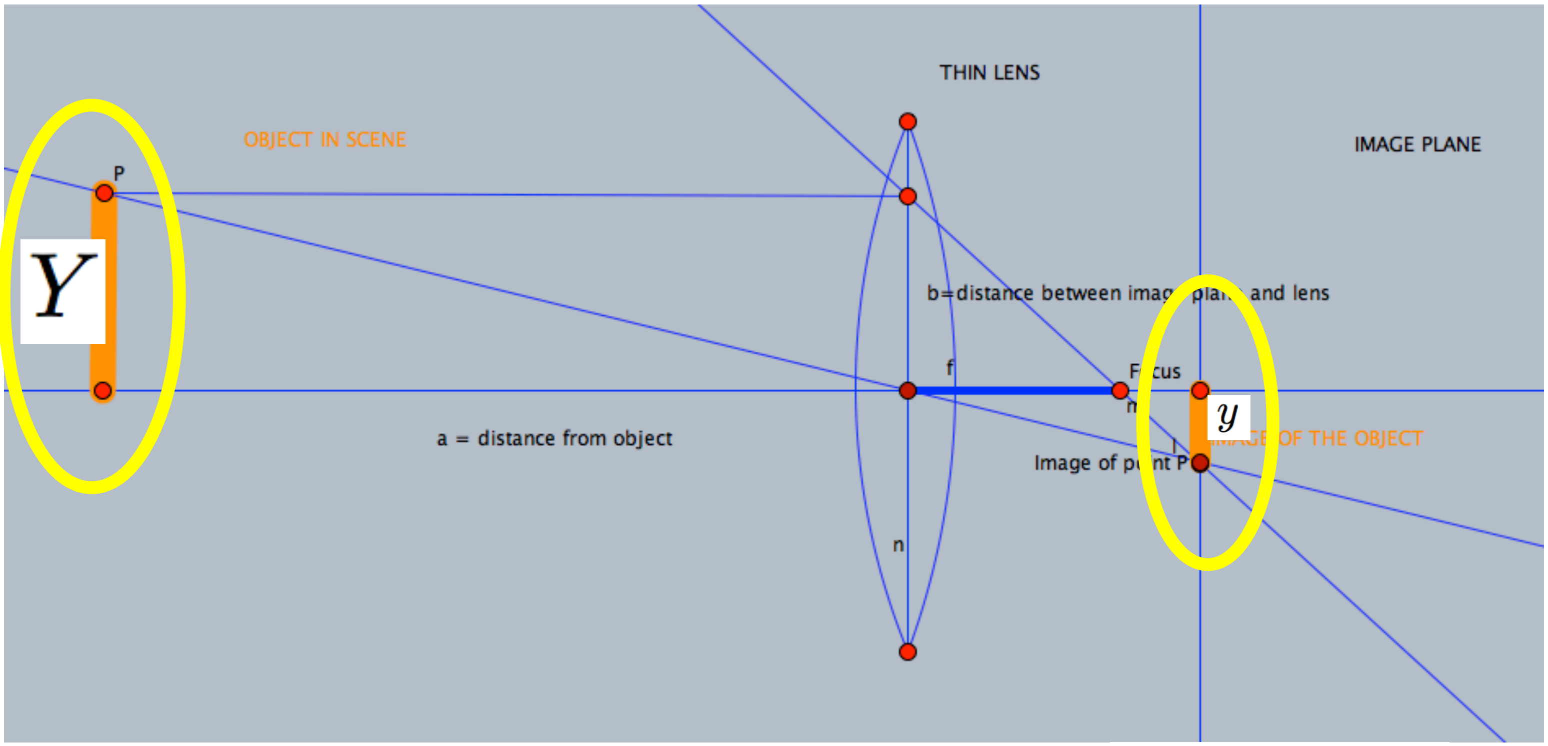
What happens when we move b, the image plane



Moving the image plane is what we call (de-) focusing !
Image starts blurring!

$$\frac{1}{f} \neq \frac{1}{a} + \frac{1}{b}$$

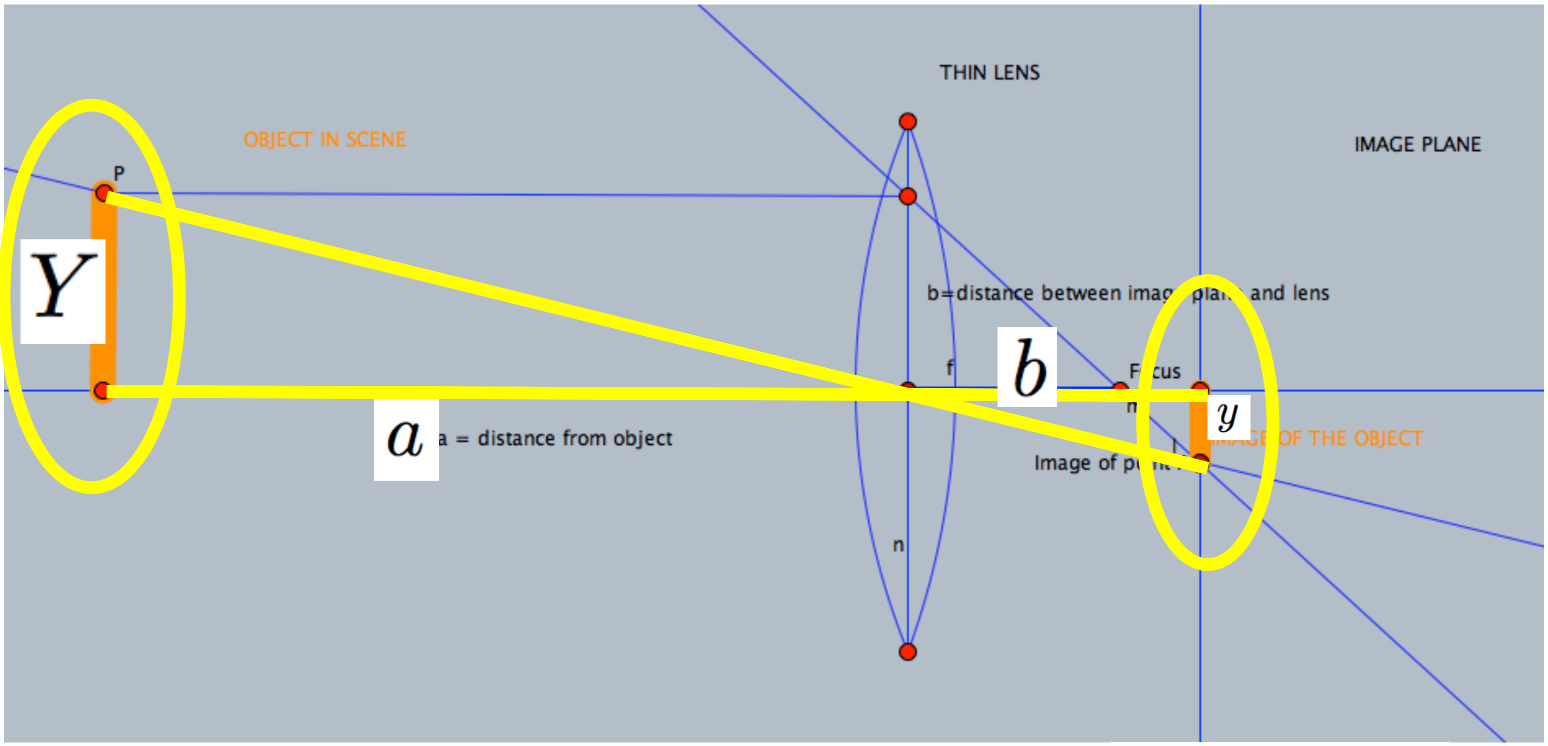
Perspective projection: size of object image



If you look only at the ray going through the center of the lens

$$\frac{Y}{a} = \frac{y}{b}$$

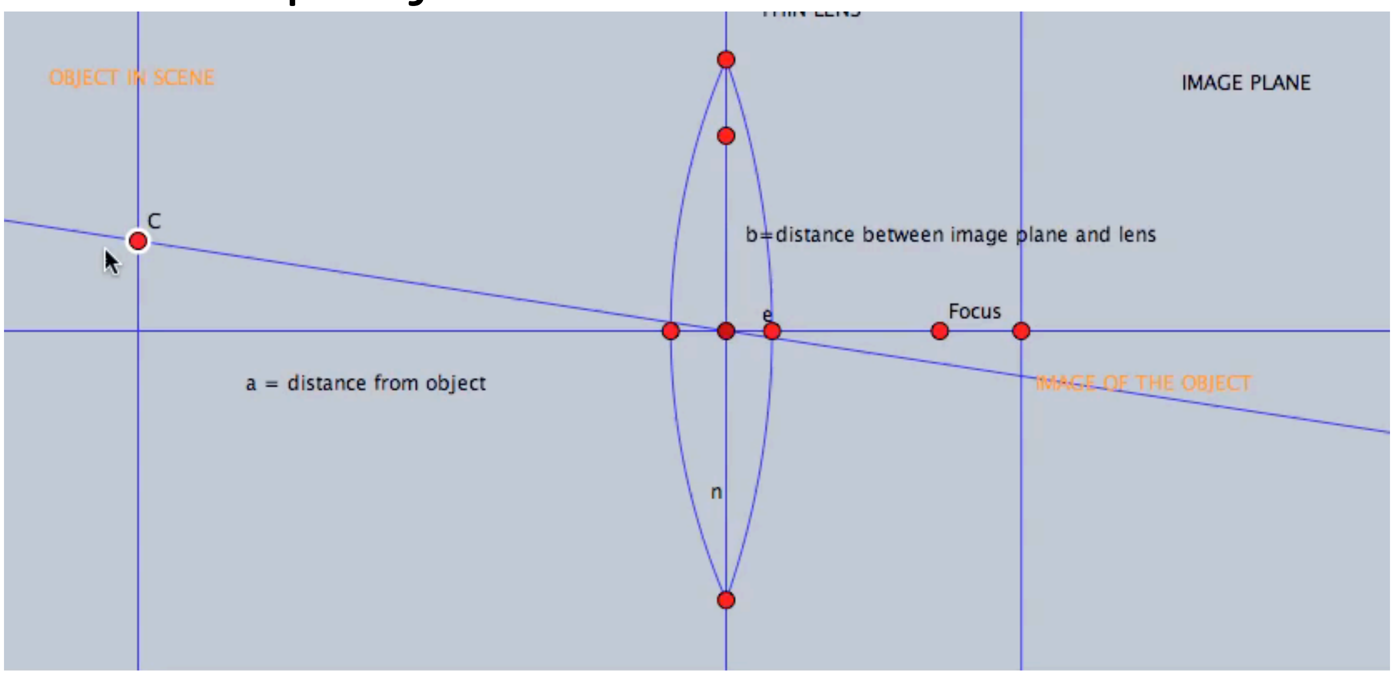
Perspective projection: size of object image



If you look only at the ray going through the center of the lens

$$\frac{Y}{a} = \frac{y}{b}$$

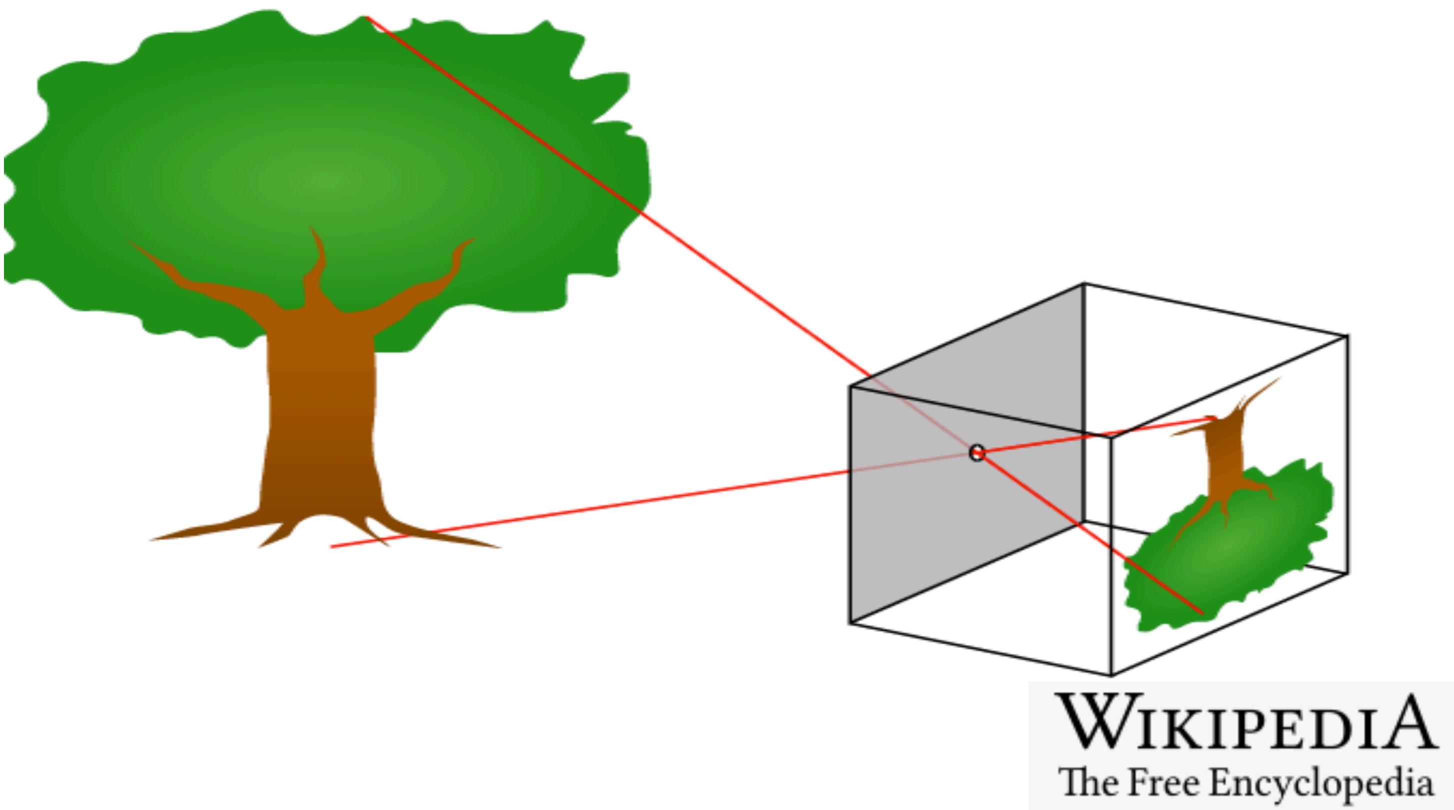
Perspective projection



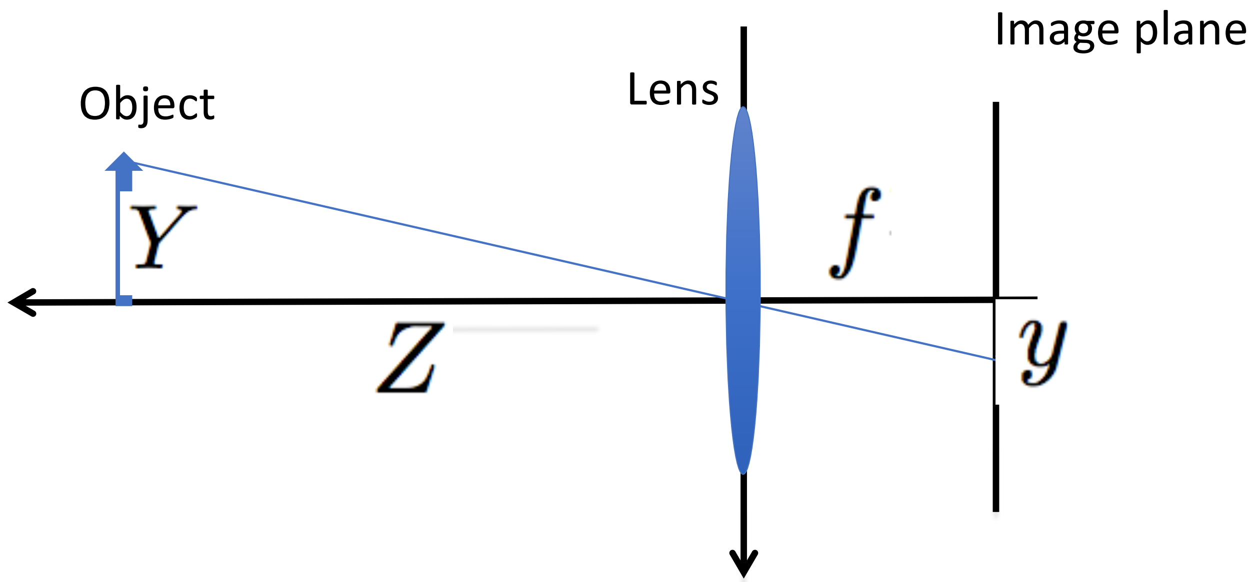
- An object of the same size coming closer results on a larger image
- A point moving on the same ray does not change its image

$$\frac{Y}{a} = \frac{y}{b}$$

Perspective projection = Pinhole Model

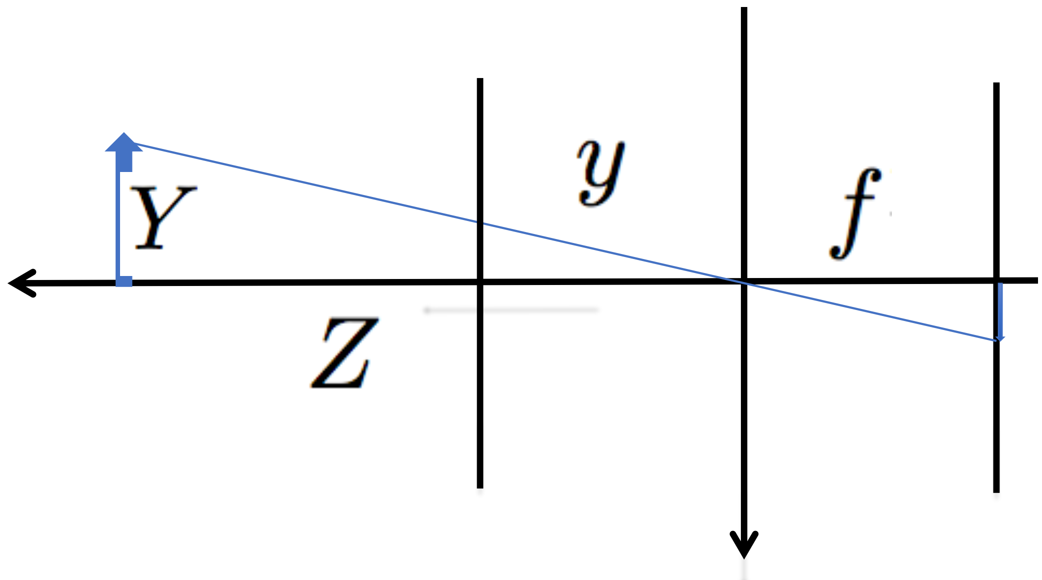


Pinhole model



$$y = -f\frac{Y}{Z}$$

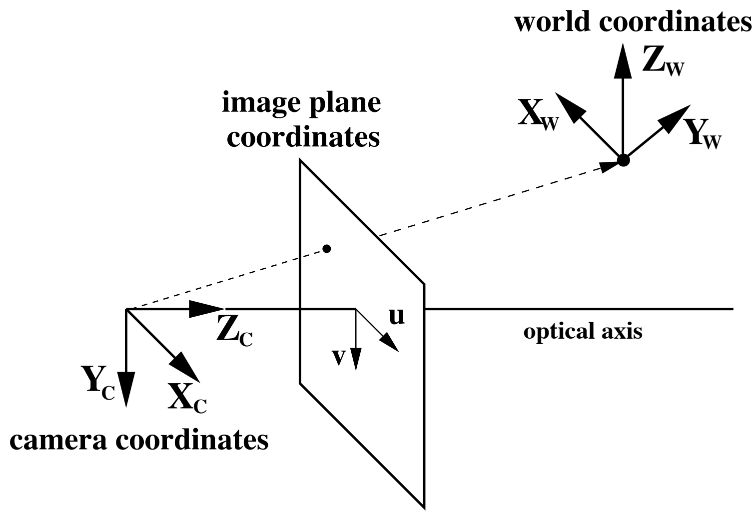
We usually assume that image plane is in front of the lens so that objects appear upright



So we omit the minus sign

$$y = f\frac{Y}{Z}$$

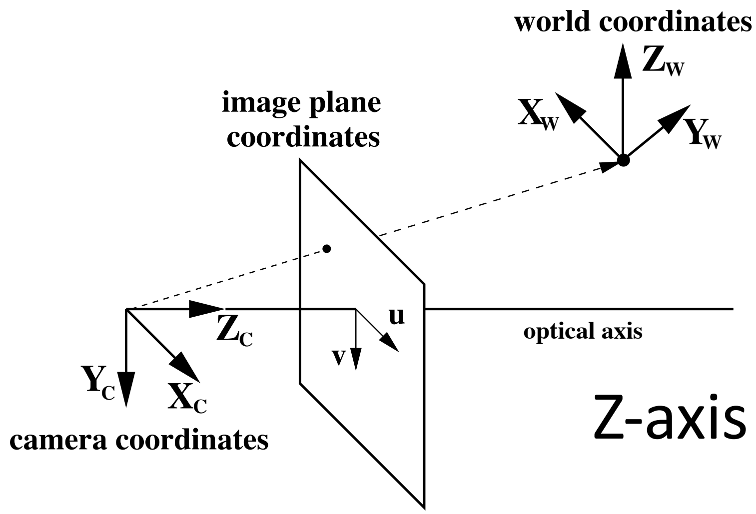
Camera Coordinate System



world coordinate system coordinates (X_w, Y_w, Z_w),

camera coordinate system coordinates (X_c, Y_c, Z_c).

Camera Coordinate System

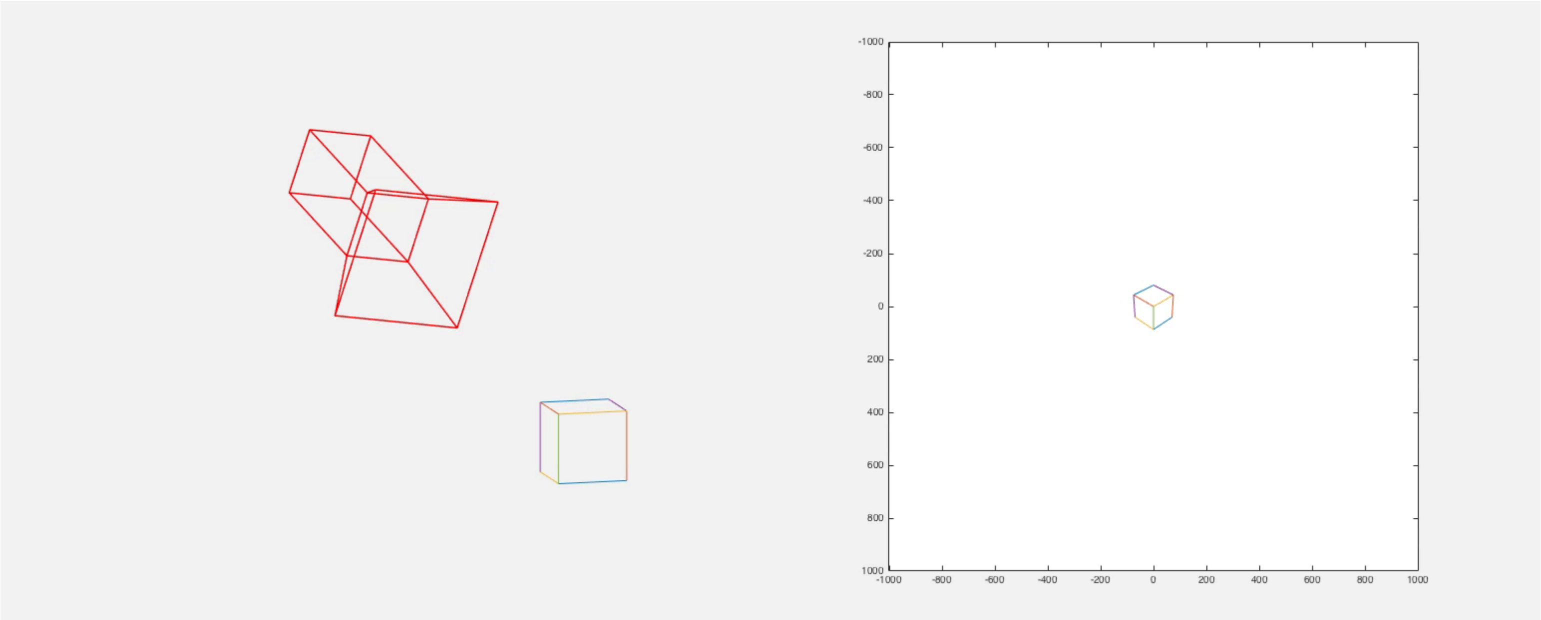


Z-axis is the optical axis

camera coordinate system coordinates (X_c, Y_c, Z_c).

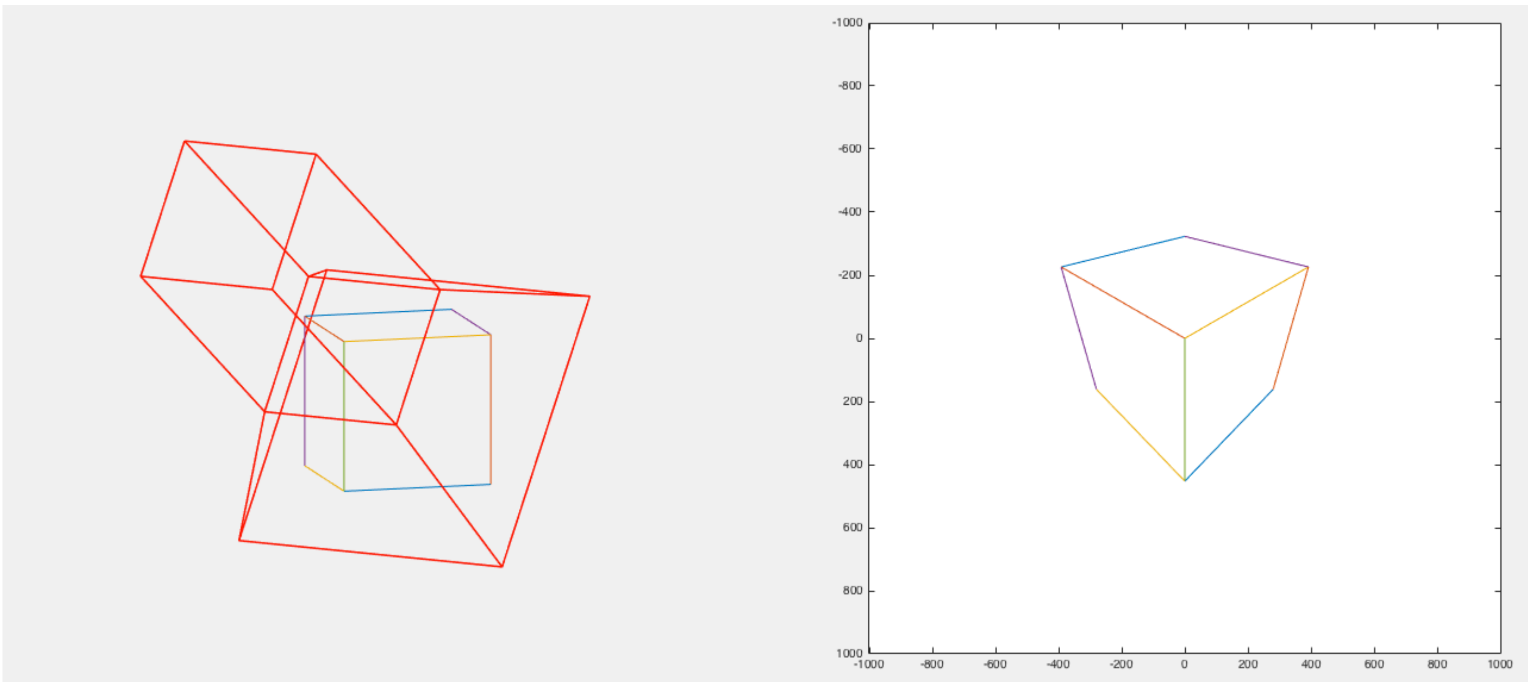
Projection in pixels $$u = f\frac{X_c}{Z_c} \quad v = f\frac{Y_c}{Z_c}\,.$$

Perspective effects



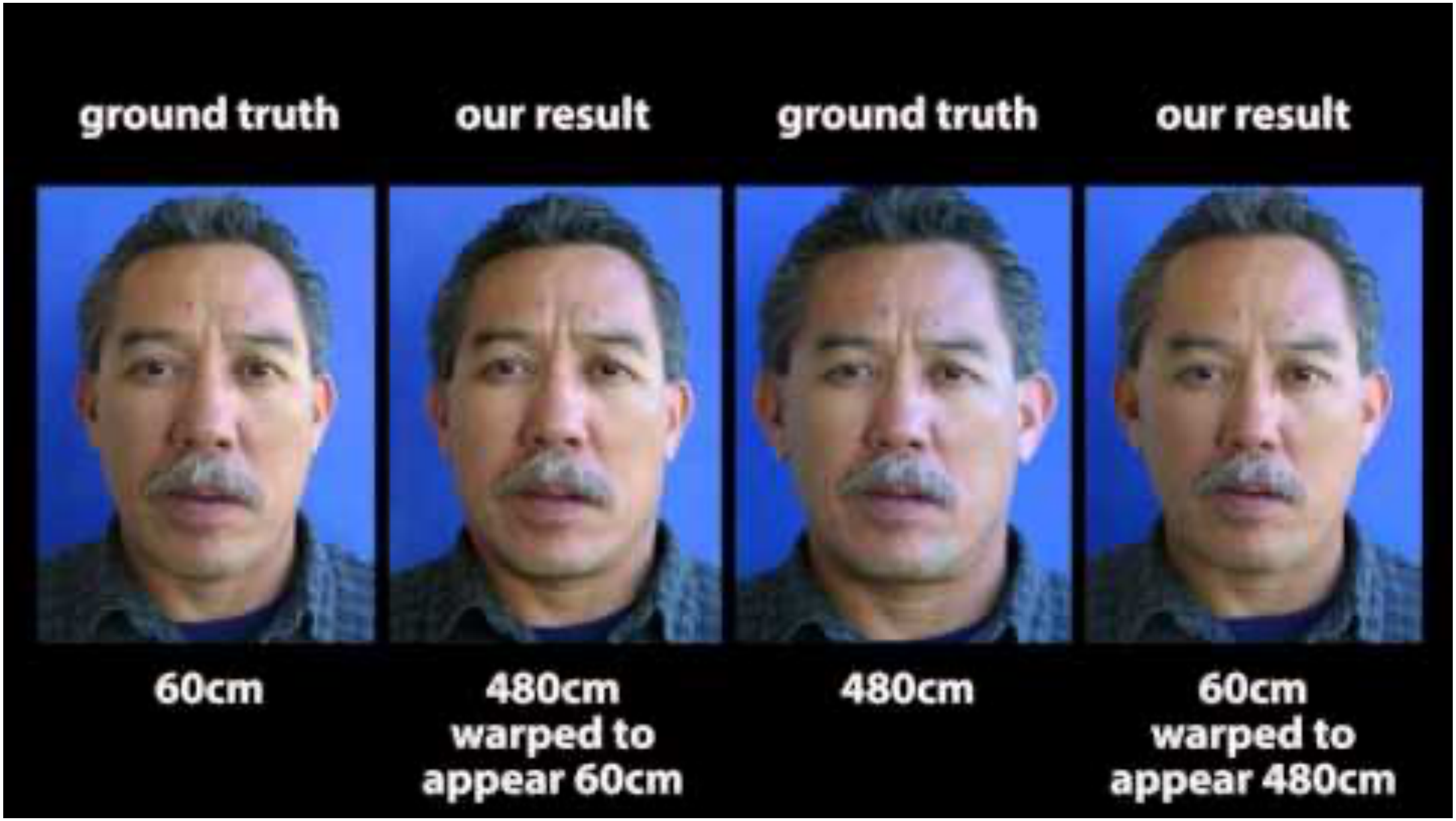
Objects decrease in size with distance

Perspective effects

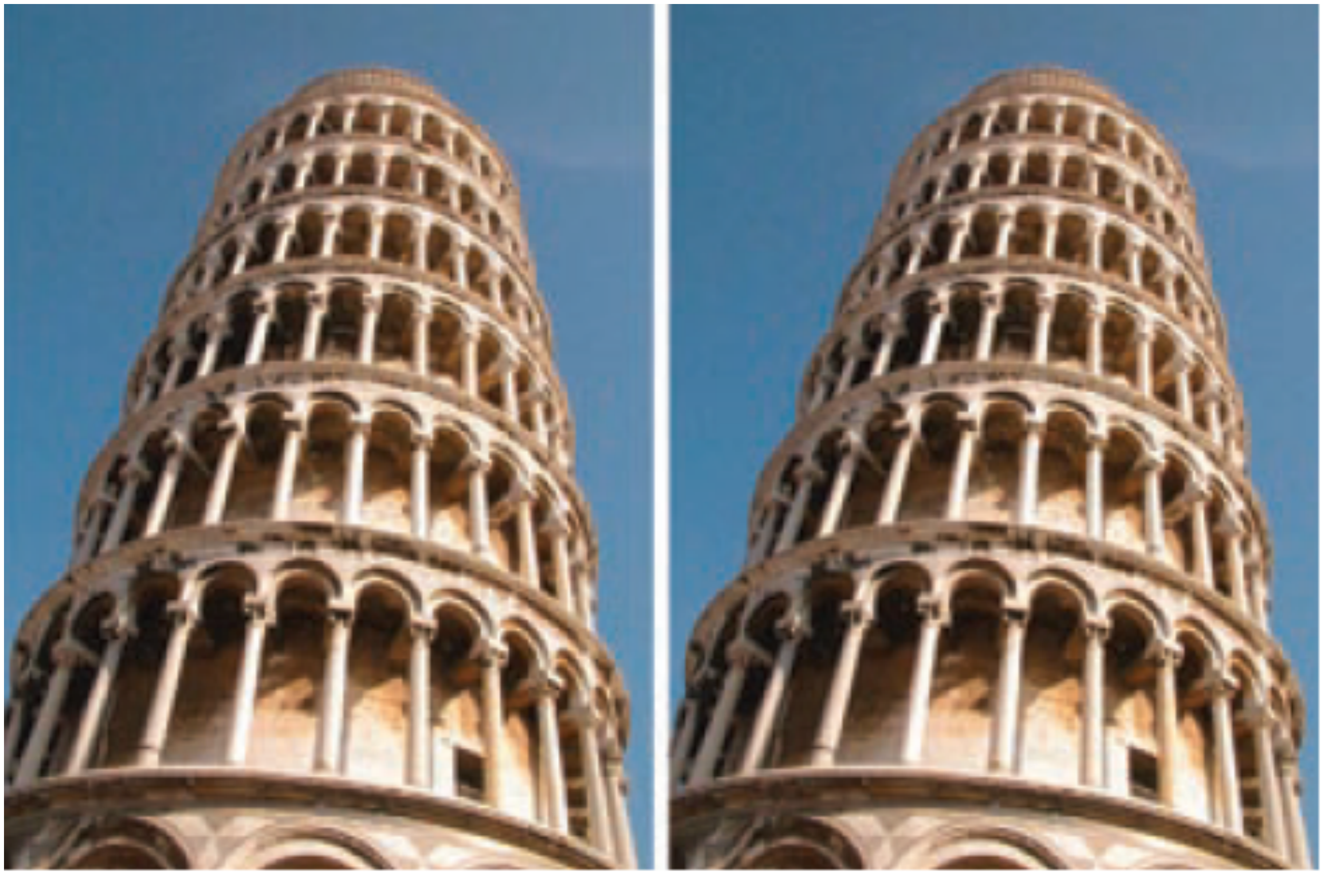


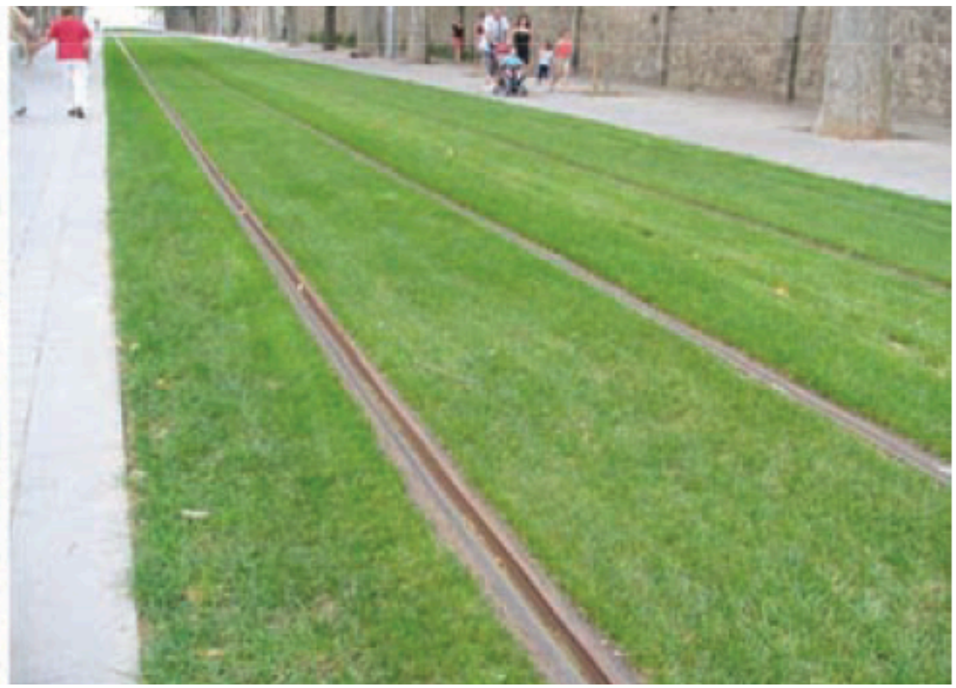
Parallel lines do not remain parallel !

Dolly Zoom Effect


ground truth
our result
ground truth
our result
60cm
480cm
warped to
appear 60cm
480cm
60cm
warped to
appear 480cm

Leaning Tower Illusion



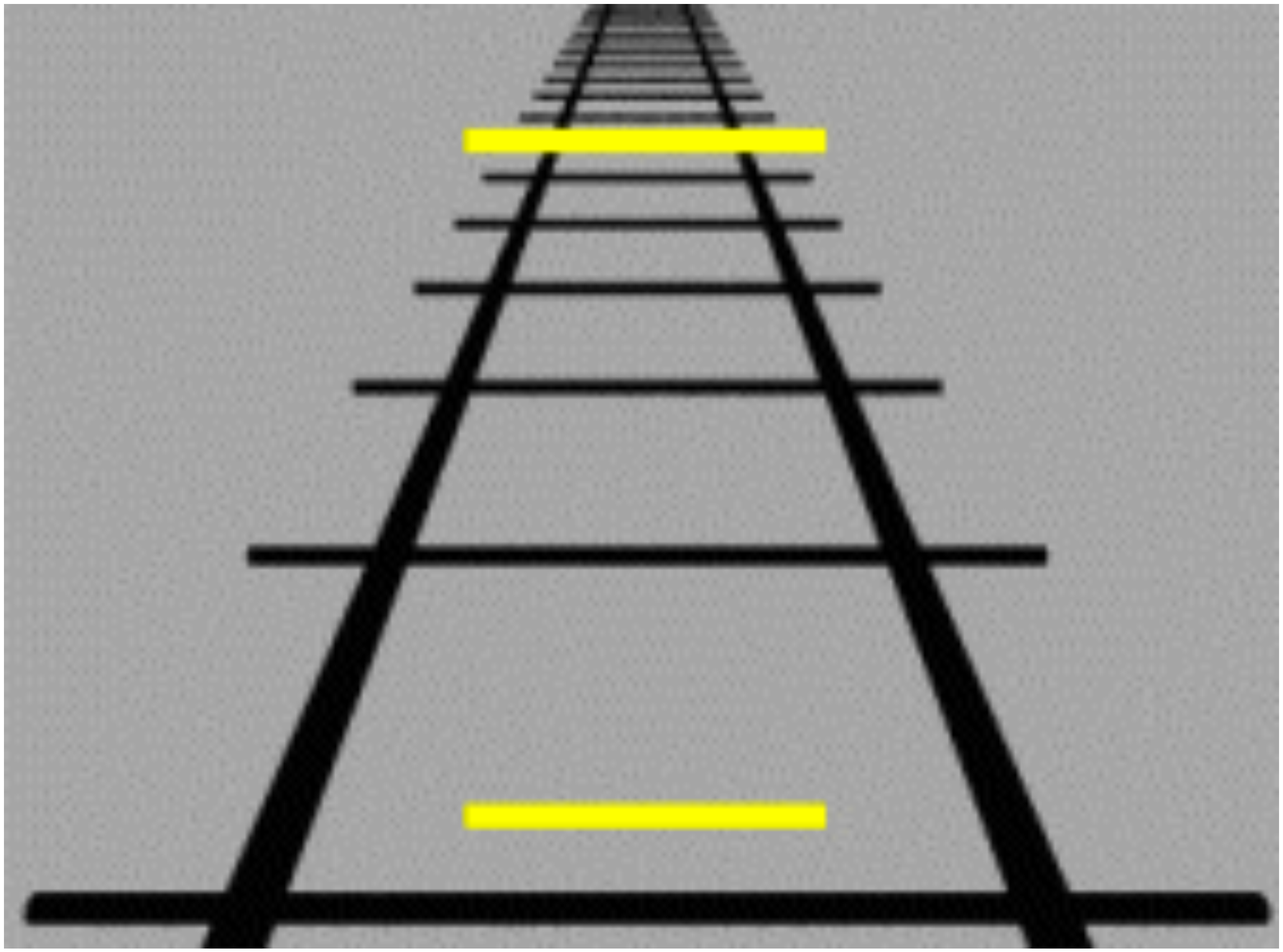


Our visual system expects all parallel lines to intersect at one point !!

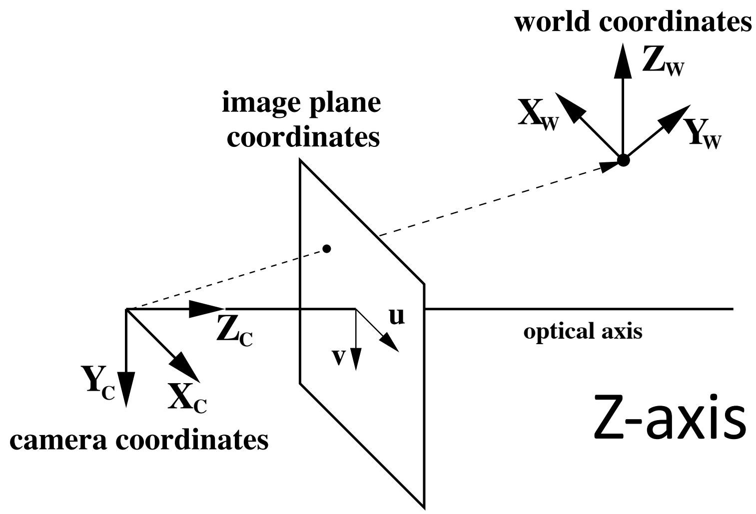
But only when we expect depth !

LACKING DEPTH

The leaning tower illusion does not occur when viewing two leaning Japanese *manga* girls, even though the two cartoon images are tilted. The reason is that the cartoon girls do not appear to recede in depth, so our brain does not expect that they would converge in the distance. This phenomenon demonstrates that the brain applies its depth-perception tool kit only in specific situations.



Camera Coordinate System



Z-axis is the optical axis

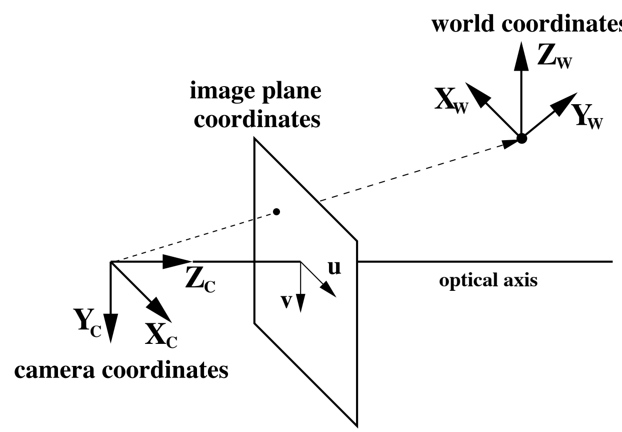
The image plane (u, v) is perpendicular to the optical axis.
Intersection of the image plane with the optical axis is the *image center* (u_o, v_o)

Projection in pixels

$$u = f\frac{X_c}{Z_c} + u_o \quad v = f\frac{Y_c}{Z_c} + v_o.$$

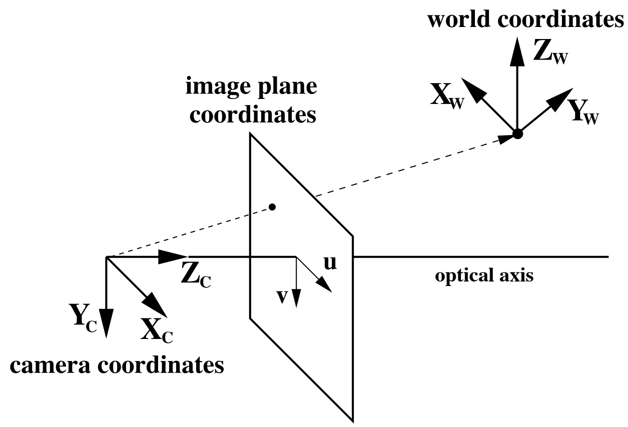
Perspective projection in matrix form

$$\lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & u_o \\ 0 & f & v_o \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix}$$



From camera to world

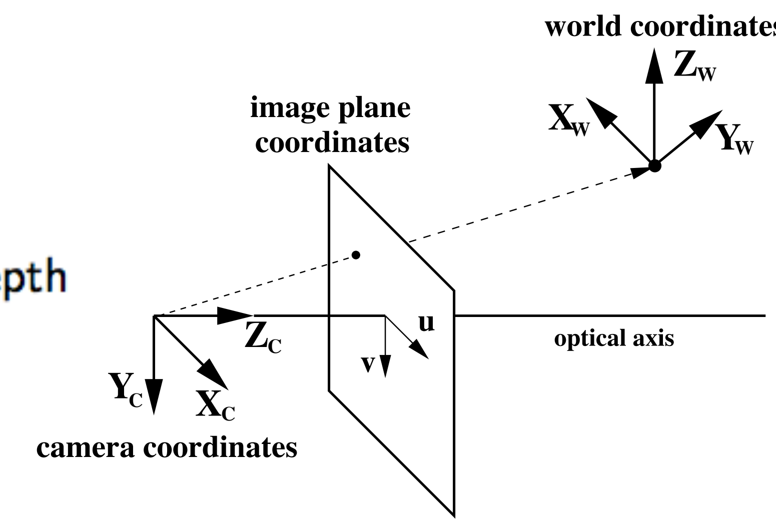
$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$



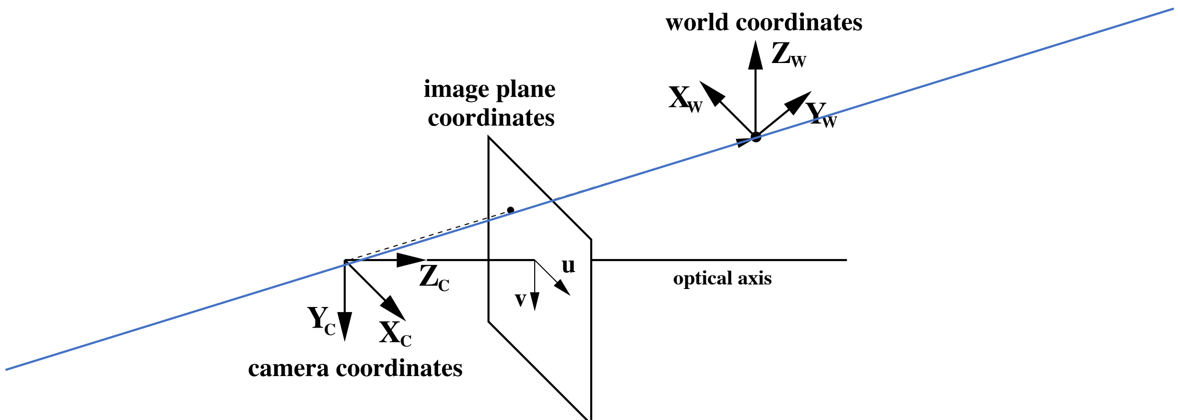
The 3x4 projection matrix P

$$\lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & u_o \\ 0 & f & v_o \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} R & t \end{pmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} = P \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

λ is the unknown depth



The meaning of the projection equation:
It is the equation of a ray in world coordinates going through the camera center

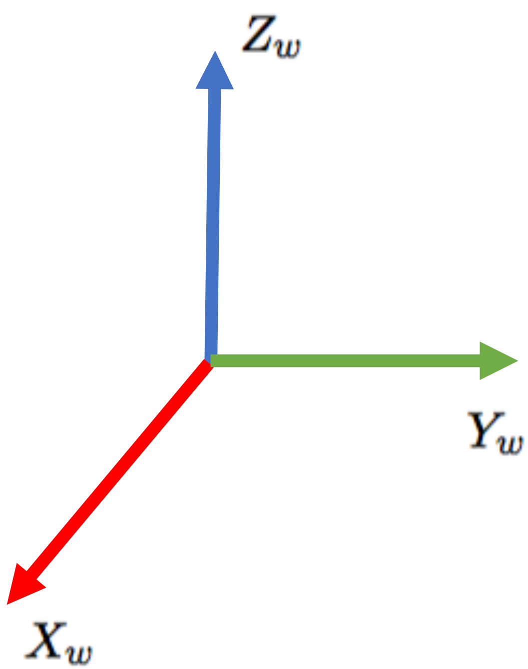


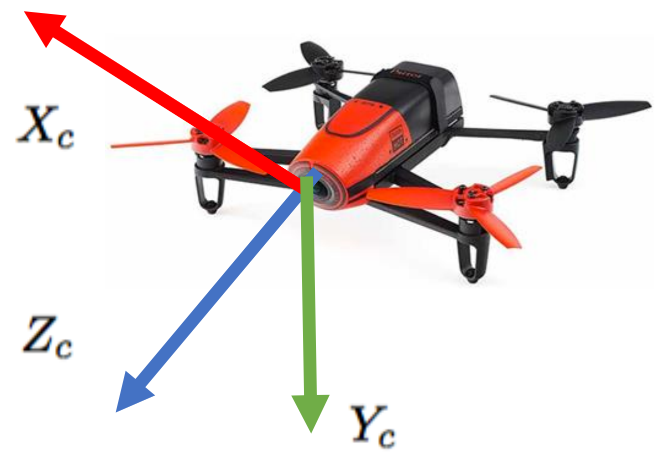
$$\lambda K^{-1}\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = R\begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} + t \Longrightarrow \begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} = \overbrace{\boxed{-R^T T}}^{\text{Camera center}} + \lambda \overbrace{\boxed{R^T K^{-1}\begin{pmatrix} u \\ v \\ 1 \end{pmatrix}}}^{\text{Ray direction}}$$

Rotations and Translations

Kostas Daniilidis

Transformation between camera and world coordinate systems



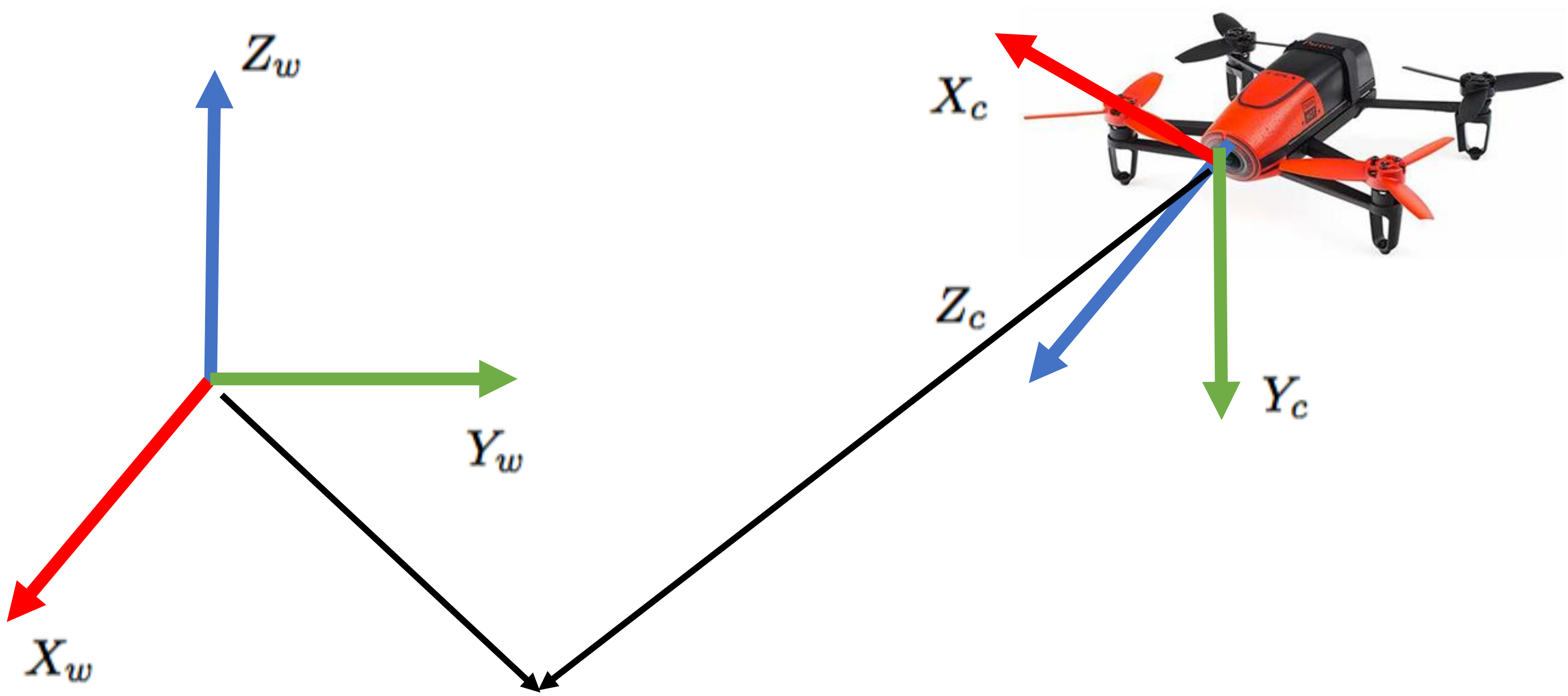


Red for X-Axis
Green for Y-Axis
Blue for Z-Axis

Remember
RGB is XYZ

$$ {}^{c}P = {}^{c}R_w \, {}^{w}P + {}^{c}T_w $$

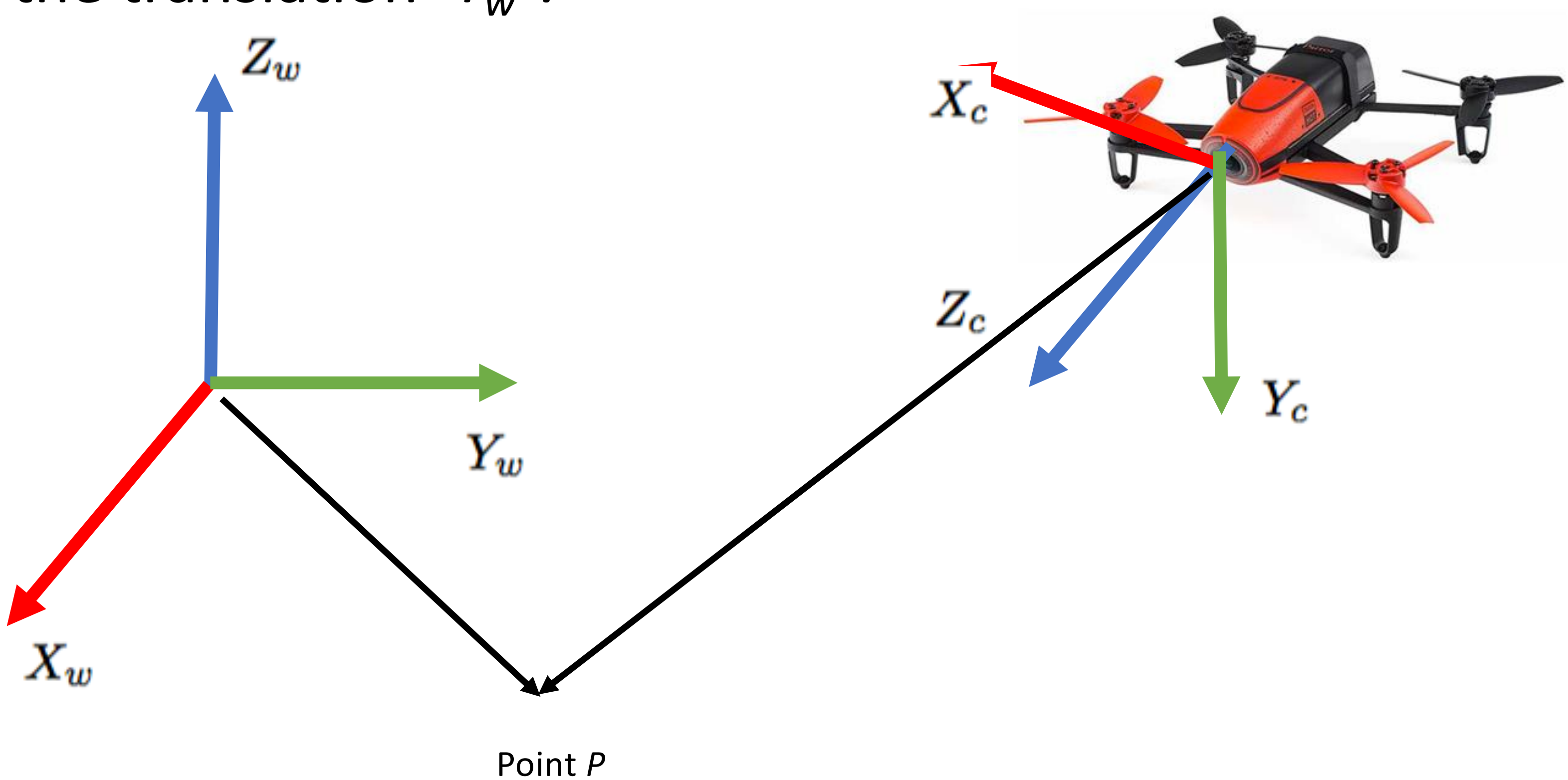
Point *P* can be expressed with respect to “w” or “c” coordinate frames



Point *P*

$$^{c}P = {}^{c}R_{w}\,{}^{w}P + {}^{c}T_{w}$$

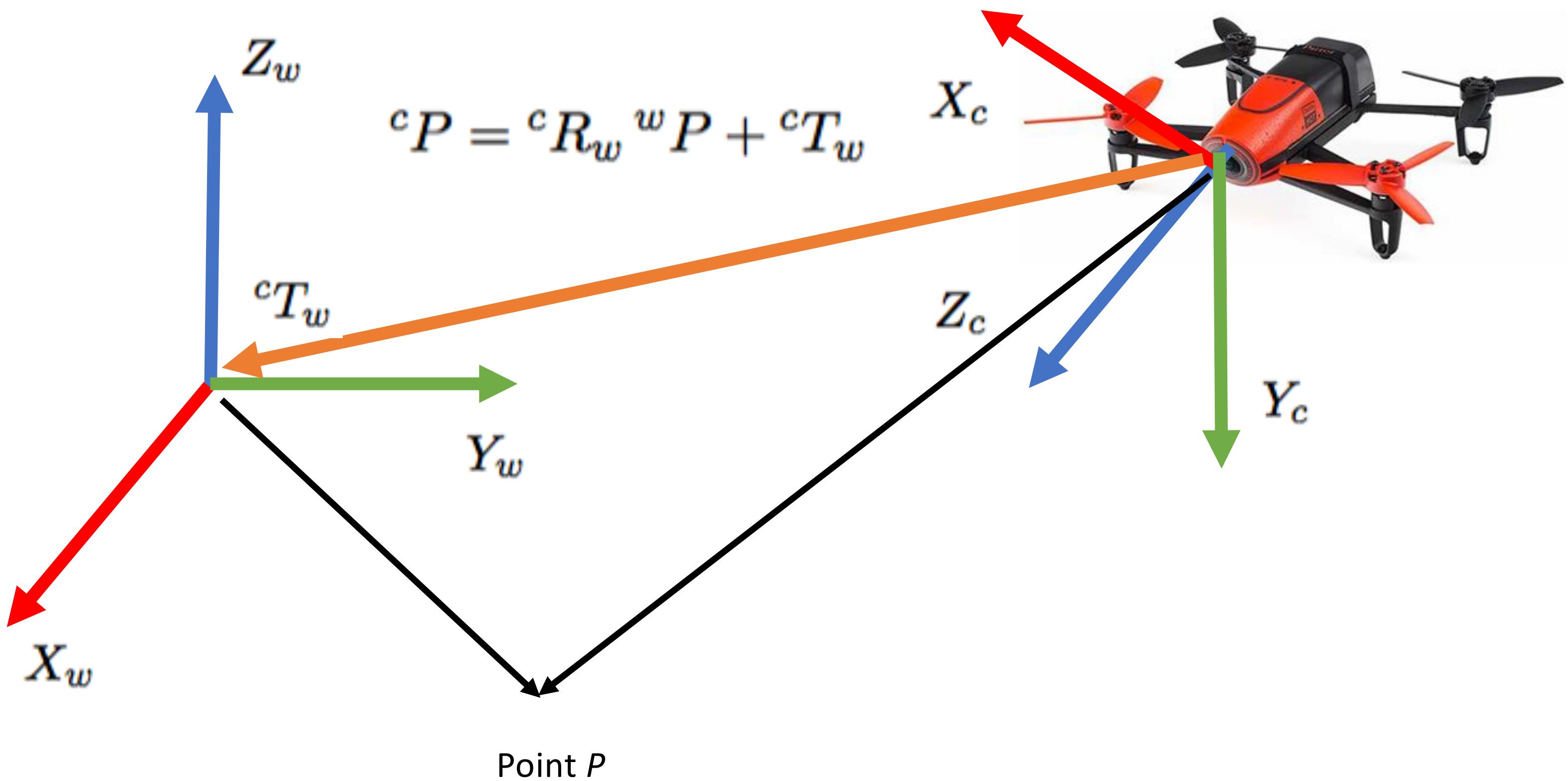
What is the geometric meaning of the rotation ${}^{c}R_{w}$ and the translation ${}^{c}T_{w}$?



What is the geometric meaning of the translation ${}^cT_w$?

This is easy to see if we set ${}^wP$ to zero.

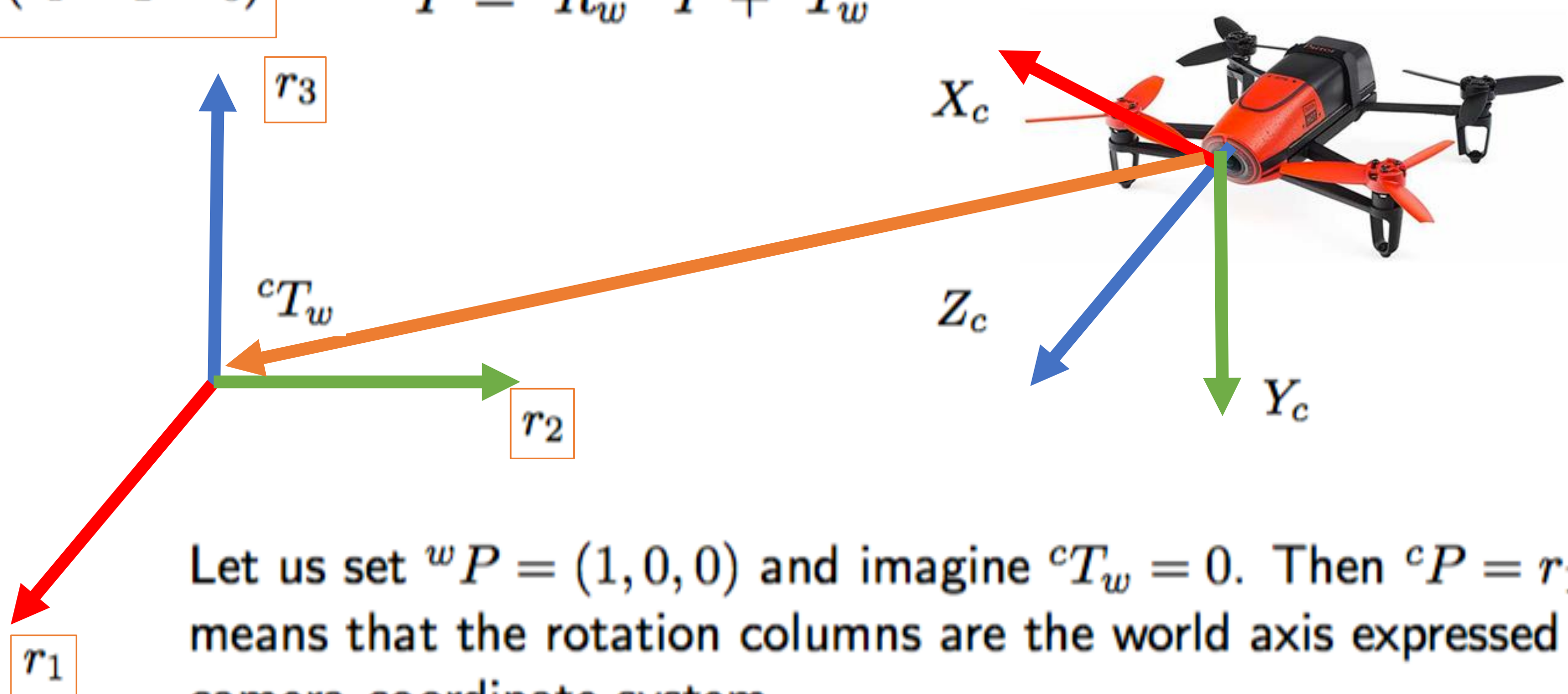
Then, ${}^cP = {}^cR_w\, 0 + {}^cT_w$ is the vector from camera origin to world origin:



What is the geometric meaning of the rotation ${}^{c}R_{w}$?

Let the rotation matrix be written as 3 orthogonal column vectors:

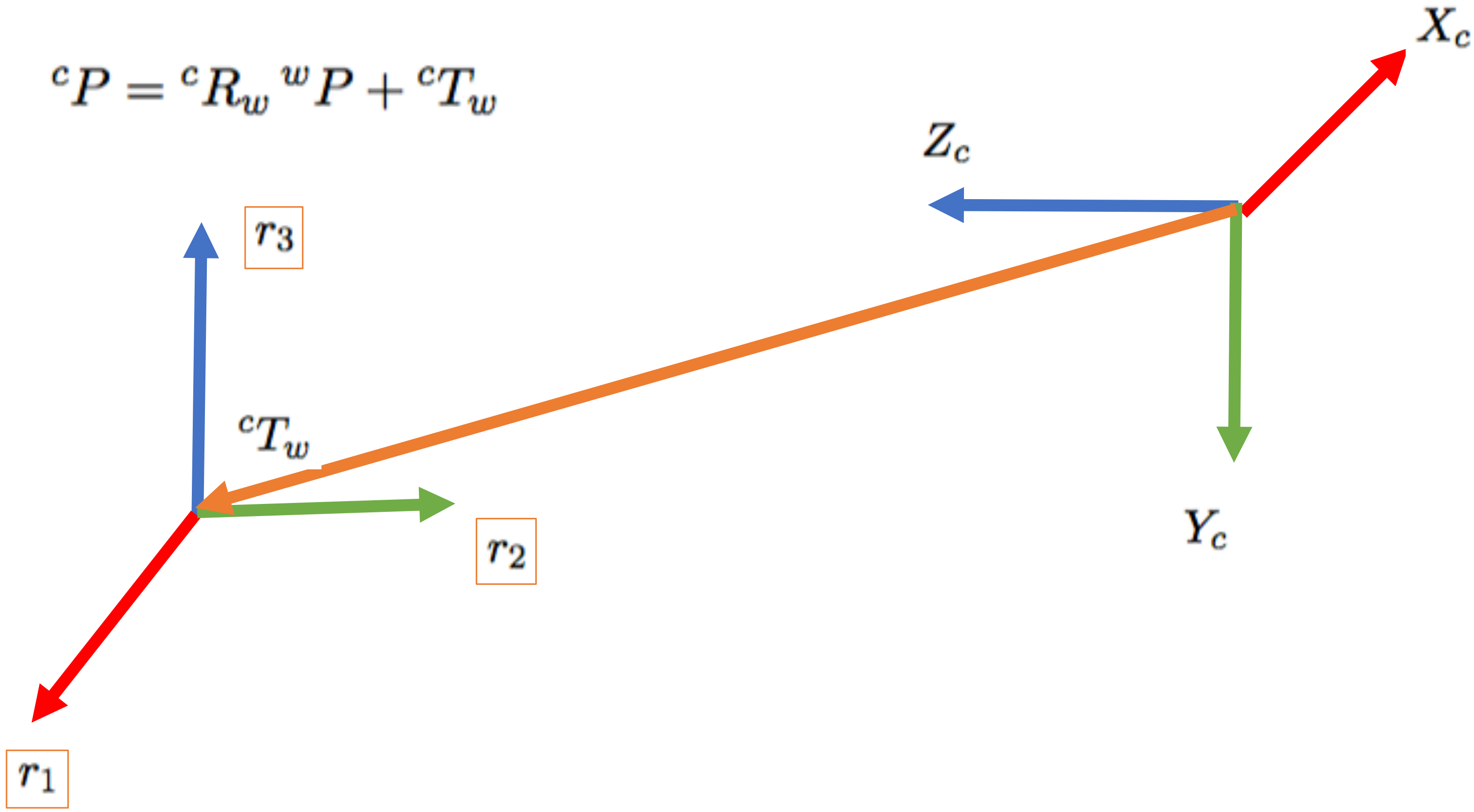
$$\boxed{{}^{c}R_{w} = \begin{pmatrix} r_1 & r_2 & r_3 \end{pmatrix}} \qquad {}^{c}P = {}^{c}R_{w}\, {}^{w}P + {}^{c}T_{w}$$



Let us set ${}^{w}P = (1, 0, 0)$ and imagine ${}^{c}T_{w} = 0$. Then ${}^{c}P = r_1$ which means that the rotation columns are the world axis expressed in the camera coordinate system.

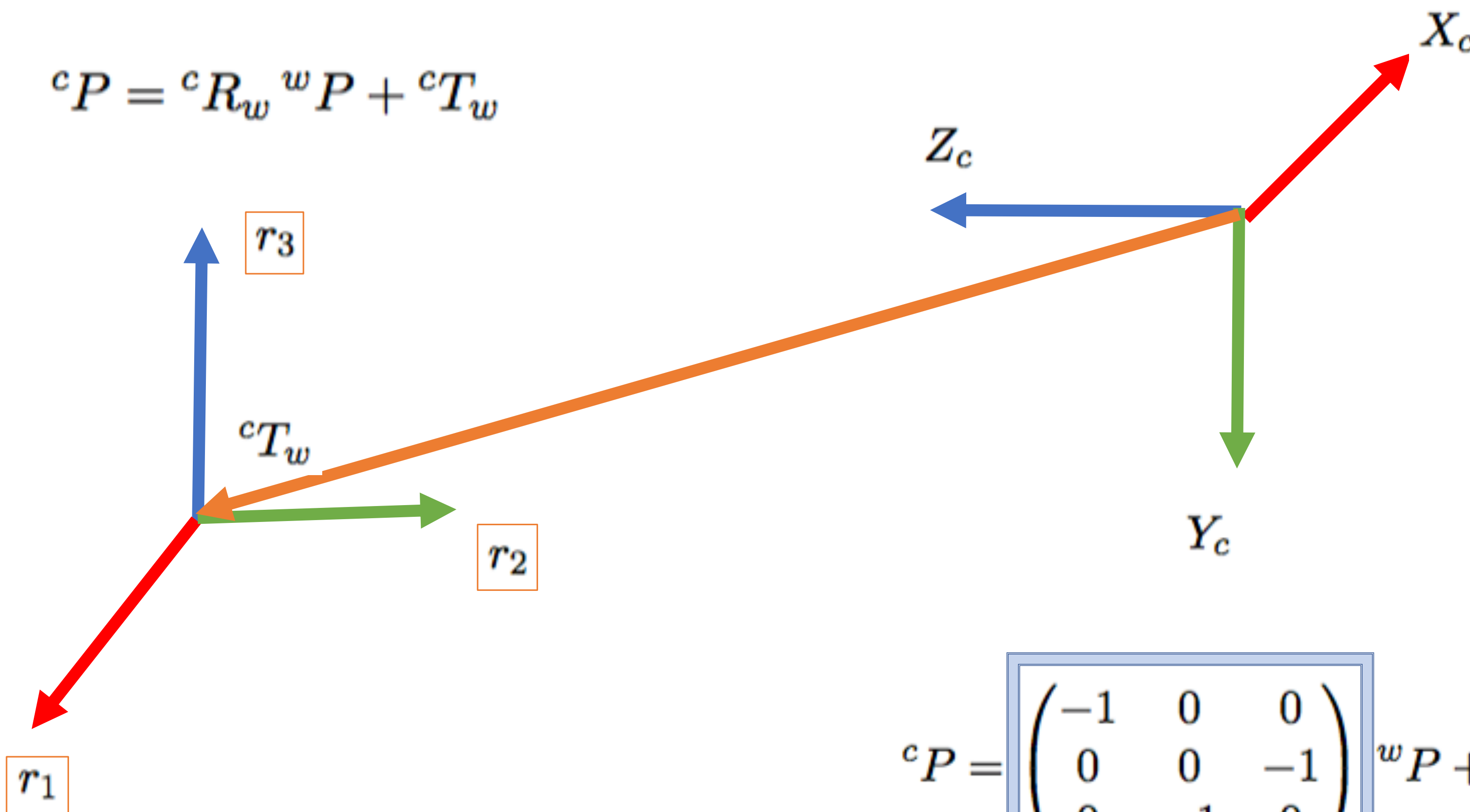
Let us look at the simple example:

$${}^{c}P = {}^{c}R_{w}\,{}^{w}P + {}^{c}T_{w}$$



How does the rotation matrix read?

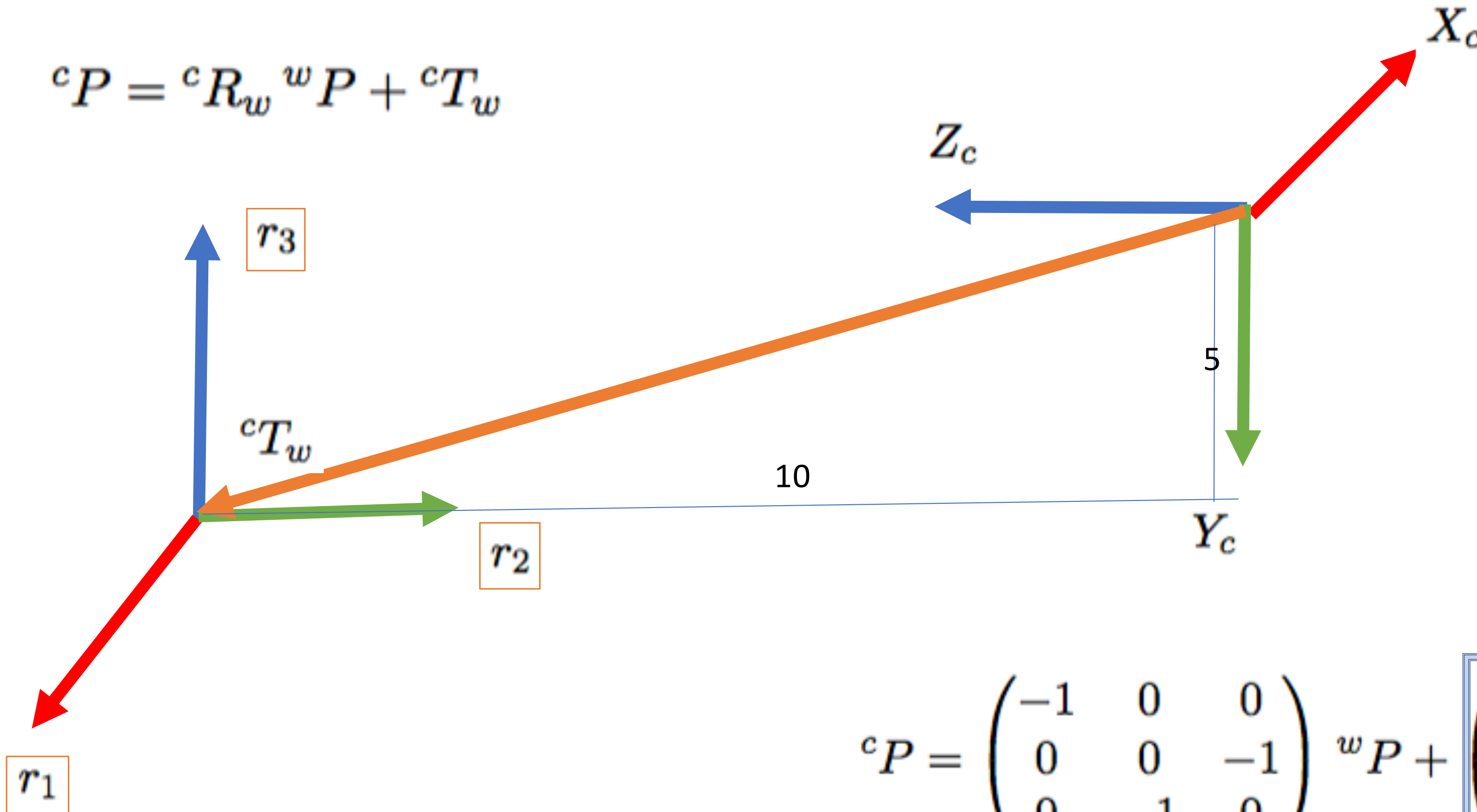
$$^{c}P = {^{c}R_{w}}\,{^{w}P} + {^{c}T_{w}}$$



$$^{c}P = \boxed{\begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}}\,{^{w}P} + \begin{pmatrix} ? \\ ? \\ ? \end{pmatrix}$$

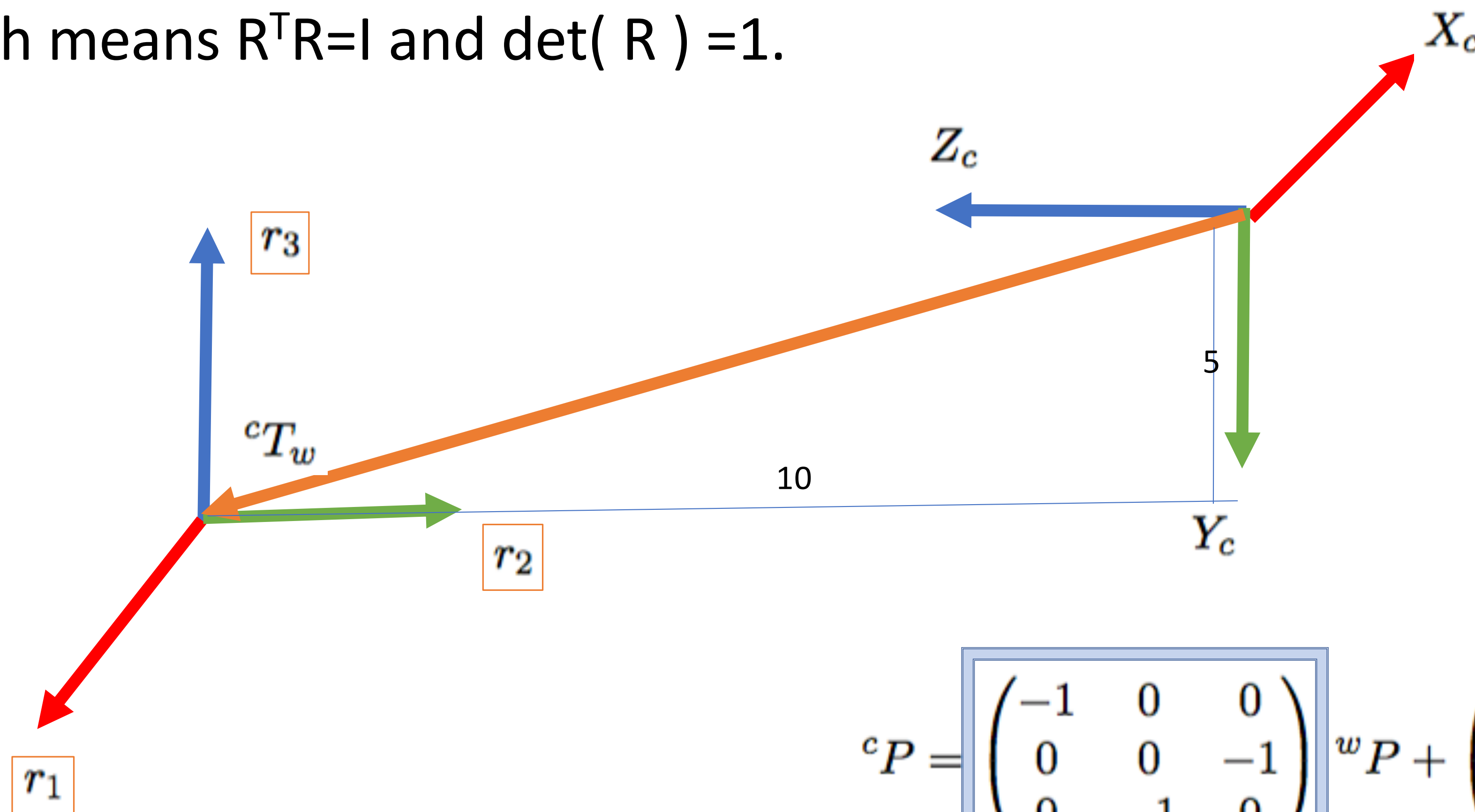
What about the translation:

$$^{c}P = {}^{c}R_{w}\, {}^{w}P + {}^{c}T_{w}$$



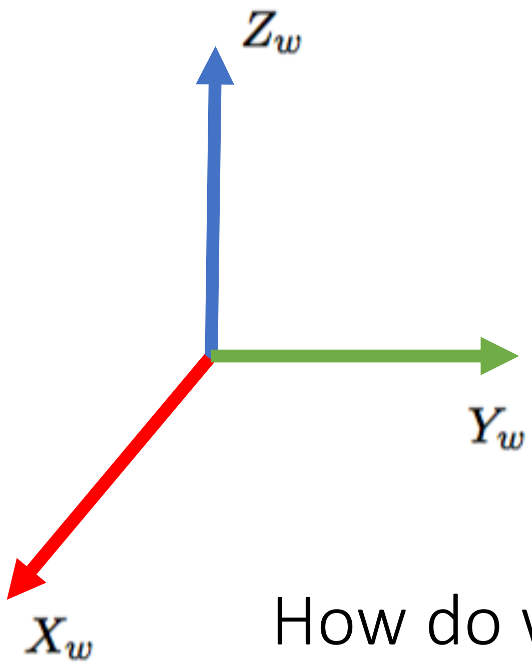
$$^{c}P = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix} {}^{w}P + \boxed{\begin{pmatrix} 0 \\ 5 \\ 10 \end{pmatrix}}$$

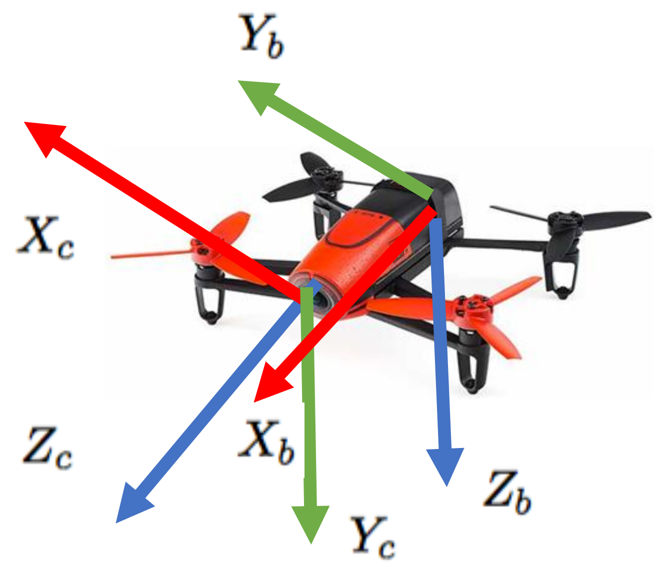
We have to make sure that the 3x3 matrix is a rotation matrix,
Which means $R^T R = I$ and $\det(R) = 1$.



$${}^cP = \boxed{\begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{pmatrix}} {}^wP + \begin{pmatrix} 0 \\ 5 \\ 10 \end{pmatrix}$$

Now imagine one more coordinate frame: a body frame with axes corresponding to roll (X_b), pitch (Y_b), yaw (Z_b) angles.

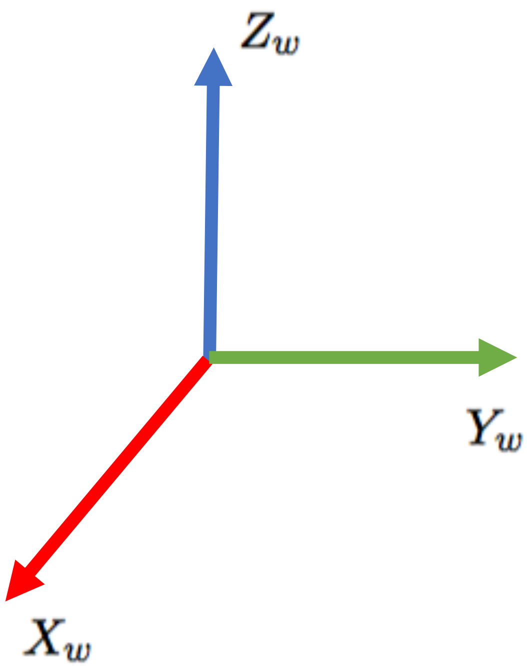


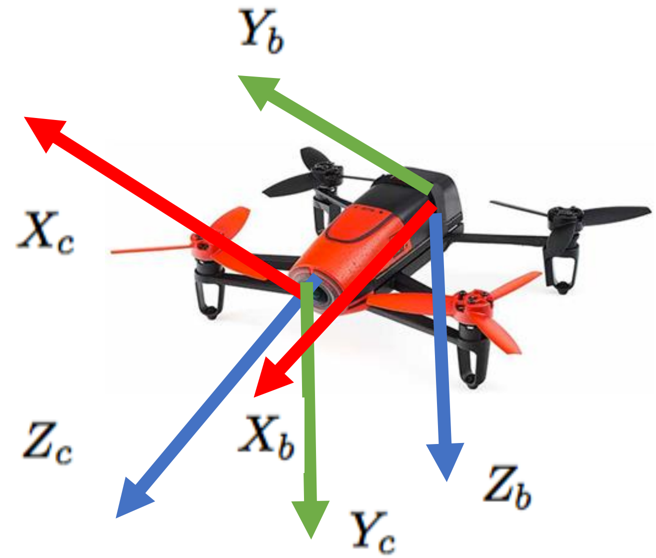


How do we compose transformations?

The easiest way to transform between coordinate systems is to use 4x4 matrices:

$$ {}^{c}M_{w} = \begin{pmatrix} & {}^{c}R_{w} & & {}^{c}T_{w} \\ 0 & 0 & 0 & 1 \end{pmatrix} $$



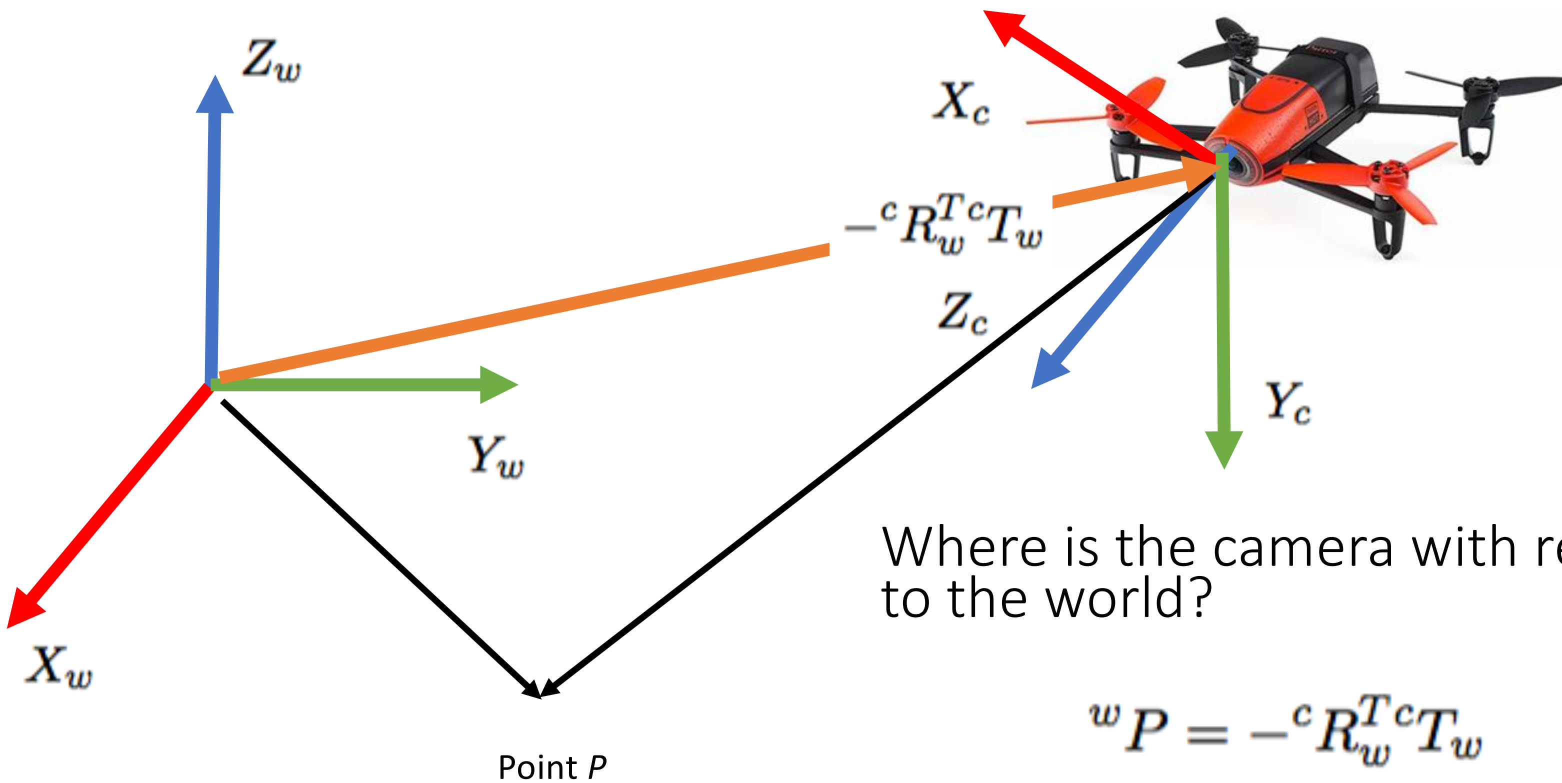


Then we just concatenate the 4x4 matrices

$$ {}^{w}M_{b} = {}^{w}M_{c}\,{}^{c}M_{b} $$

What about the inverse transformation?

$$^{w}M_c = \begin{pmatrix} & {}^{c}R_w^T & & -{}^{c}R_w^T\,{}^{c}T_w \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

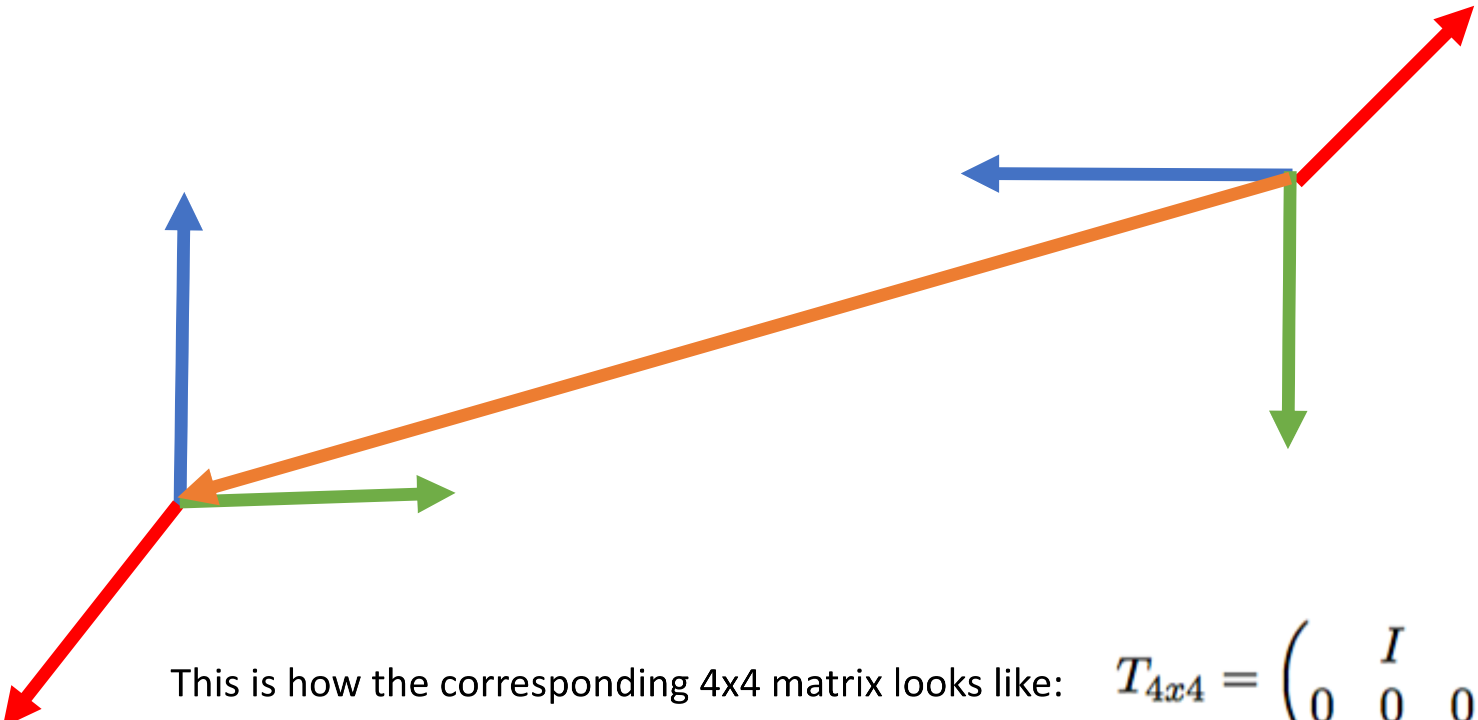


Where is the camera with respect to the world?

$$^{w}P = -{}^{c}R_w^T\,{}^{c}T_w$$

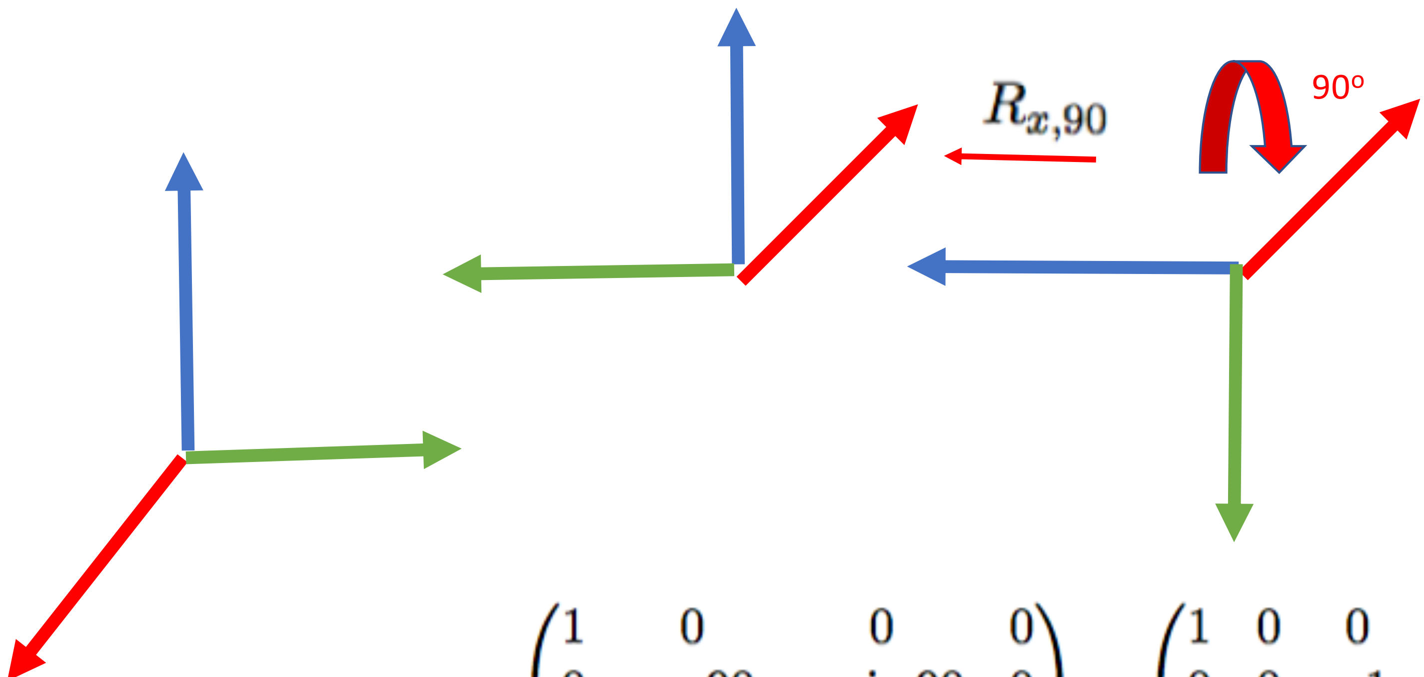
Alternative interpretation as a sequence of motions:

1. The camera frame first translates to the world



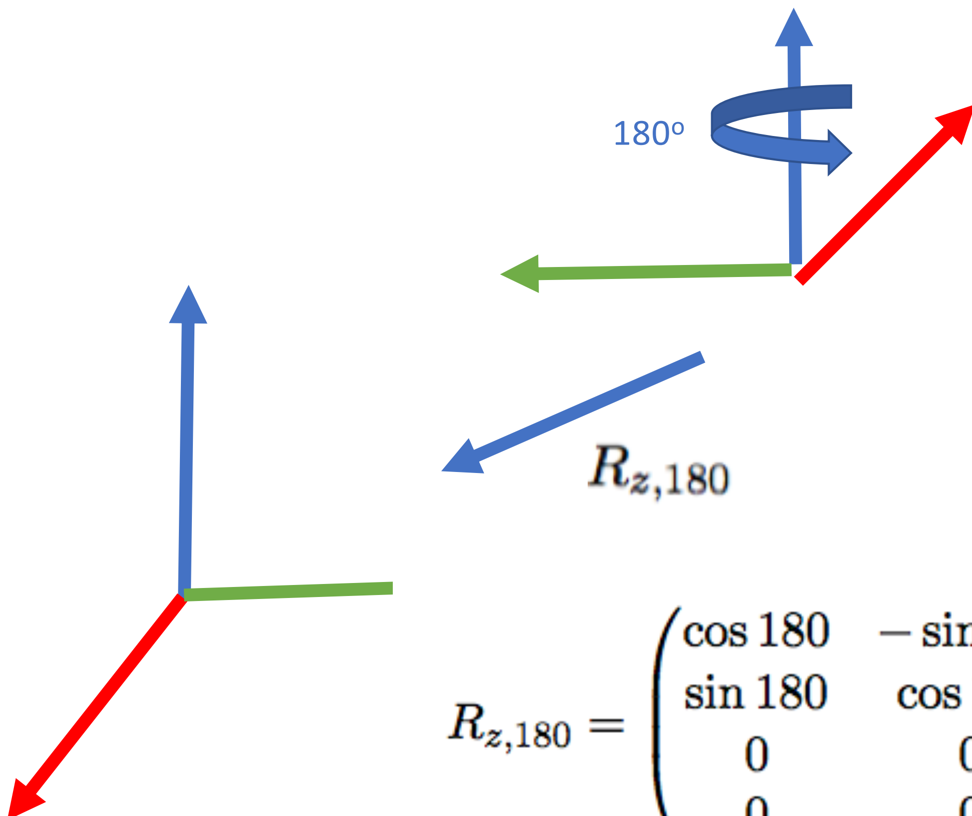
This is how the corresponding 4x4 matrix looks like: $T_{4x4} = \begin{pmatrix} & I & & {}^{c}T_{w} \\ 0 & 0 & 0 & 1 \end{pmatrix}$

2. The camera frame rotates 90 degrees around x



$$R_{x,90} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 90 & -\sin 90 & 0 \\ 0 & \sin 90 & \cos 90 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

A3. The camera frame rotates 180 degrees around z



$$R_{z,180} = \begin{pmatrix} \cos 180 & -\sin 180 & 0 & 0 \\ \sin 180 & \cos 180 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

How do we compose these motions? Golden rule: when we move coordinate frames and we refer to the most recent coordinate frame
we always postmultiply!

$$
\begin{aligned}
{}^{c}M_{w} &= T R_{x,90} R_{z,180} \\
&= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 5 \\ 0 & 0 & 1 & 10 \\ 0 & 0 & 0 & 1 \end{pmatrix}
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 5 \\ 0 & -1 & 0 & 10 \\ 0 & 0 & 0 & 1 \end{pmatrix}
\end{aligned}
$$