Lecture 5.1
The camera matrix P

Thomas Opsahl

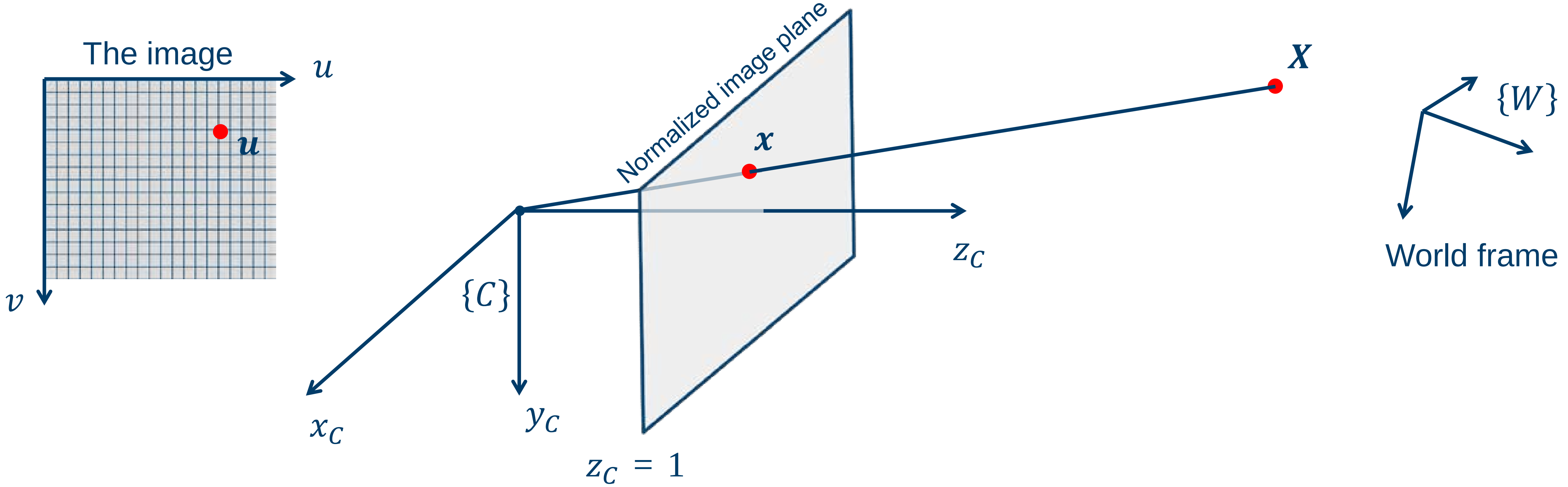
Weekly overview – Single View Geometry

- **The camera matrix P**
  - Repetition on perspective cameras
  - Dissecting the general camera matrix P
  - Vanishing points
  - Vanishing lines
  - Estimating P from known world-image correspondences

- **Pose from known 3D points**
  - Decompose P to K, R and t
  - PnP

- **Camera calibration**
  - Zhangs method briefly
  - Calibration in Matlab
  - Undistortion

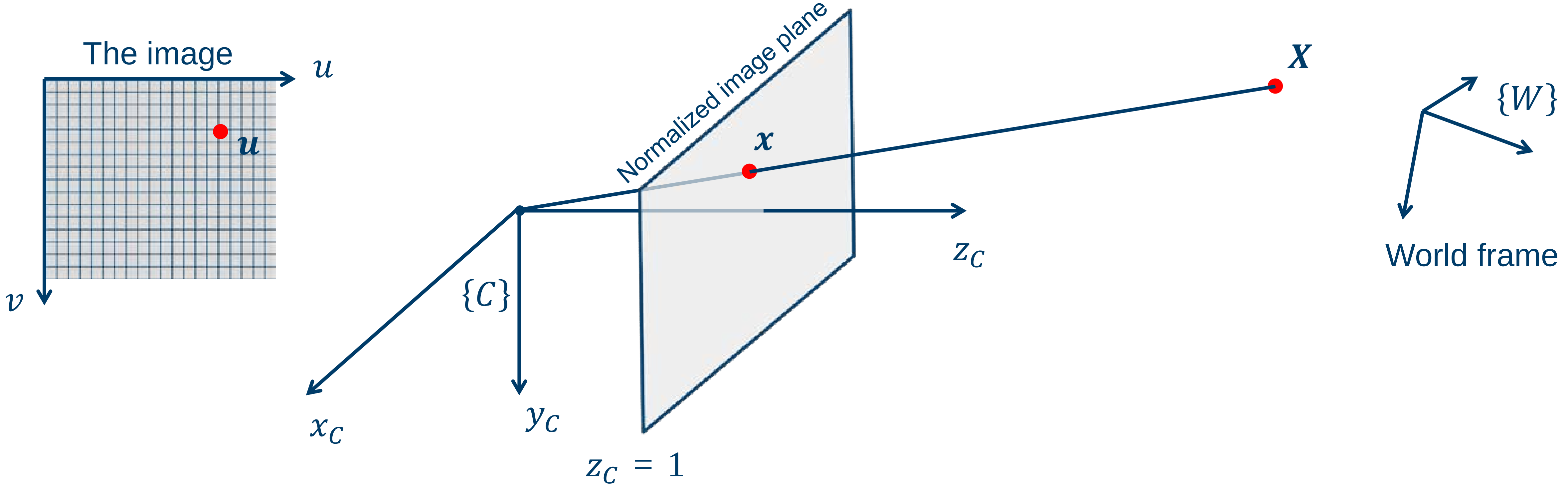
Projective cameras



- The perspective camera model describes the correspondence between points in the world and points in the image

$$\tilde{\boldsymbol{u}} = P^W \tilde{\boldsymbol{X}}$$

Projective cameras



Pinhole camera

$$P = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & \boldsymbol{t} \end{bmatrix}$$

Finite projective camera

$$P = \begin{bmatrix} f_u & s & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & \boldsymbol{t} \end{bmatrix}$$

General projective camera

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}$$

with $\mathrm{rank} = 3$ and 11 dof

The camera matrix P

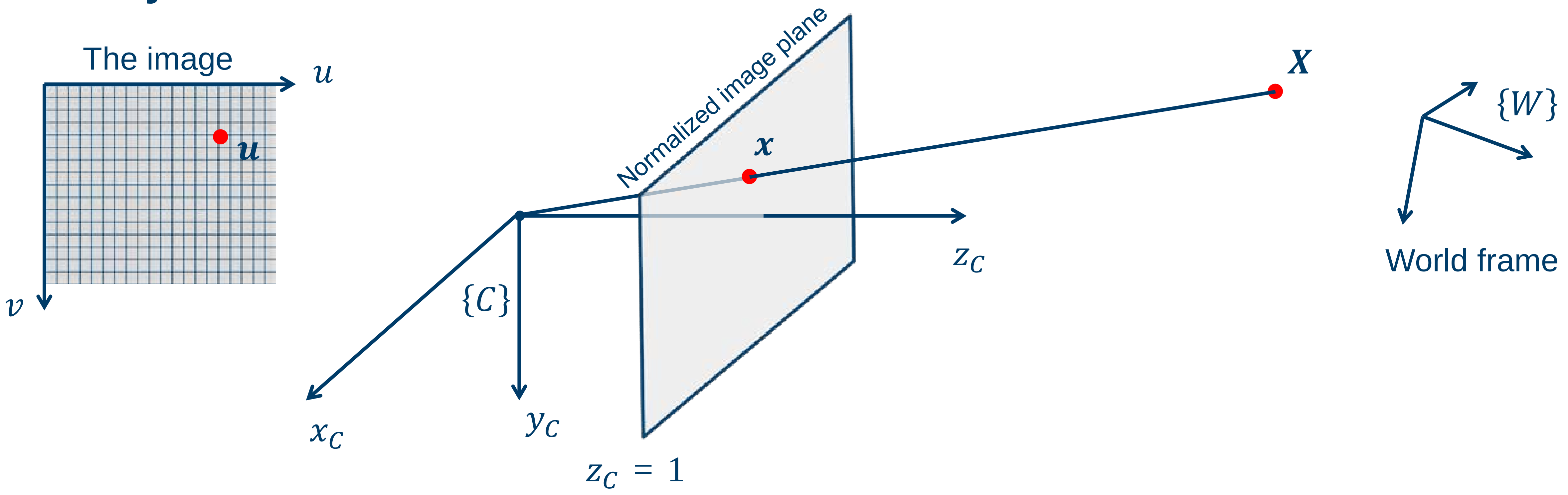
- From our lecture on the perspective camera model, we know that the camera matrix contains geometrical information about the image, the camera-frame and the world-frame

- For finite cameras, $P = K[R \quad \boldsymbol{t}]$ we recall that
  - The camera calibration matrix K describes the relationship between normalized image coordinates and image coordinates
  - The rotation matrix R describes how the world-frame $\{W\}$ is rotated/oriented relative to the camera-frame $\{C\}$ and is a change-of-basis-matrix for vectors (${}^{C}R_{W}\colon \{W\} \rightarrow \{C\}$)

$$\begin{bmatrix} {}^{C}\boldsymbol{w}_1 & {}^{C}\boldsymbol{w}_2 & {}^{C}\boldsymbol{w}_3 \end{bmatrix} = {}^{C}R_{W} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} {}^{W}\boldsymbol{c}_1^{T} \\ {}^{W}\boldsymbol{c}_2^{T} \\ {}^{W}\boldsymbol{c}_3^{T} \end{bmatrix}$$

  - The vector $\boldsymbol{t}$ describes how the world-frame is translated/positioned relative to the camera-frame
  - The matrix ${}^{C}\xi_{W} = [R \quad \boldsymbol{t}]$ represents the pose of the world-frame relative to the camera-frame and transforms points from $\{W\}$ to $\{C\}$

Projective cameras



- What geometric information can we extract from a general camera matrix?

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}$$

Dissecting the camera matrix

- First some practical notations

- Rows and columns

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} = [\boldsymbol{p}_1 \quad \boldsymbol{p}_2 \quad \boldsymbol{p}_3 \quad \boldsymbol{p}_4] = \begin{bmatrix} \boldsymbol{P}^{1T} \\ \boldsymbol{P}^{2T} \\ \boldsymbol{P}^{3T} \end{bmatrix}$$

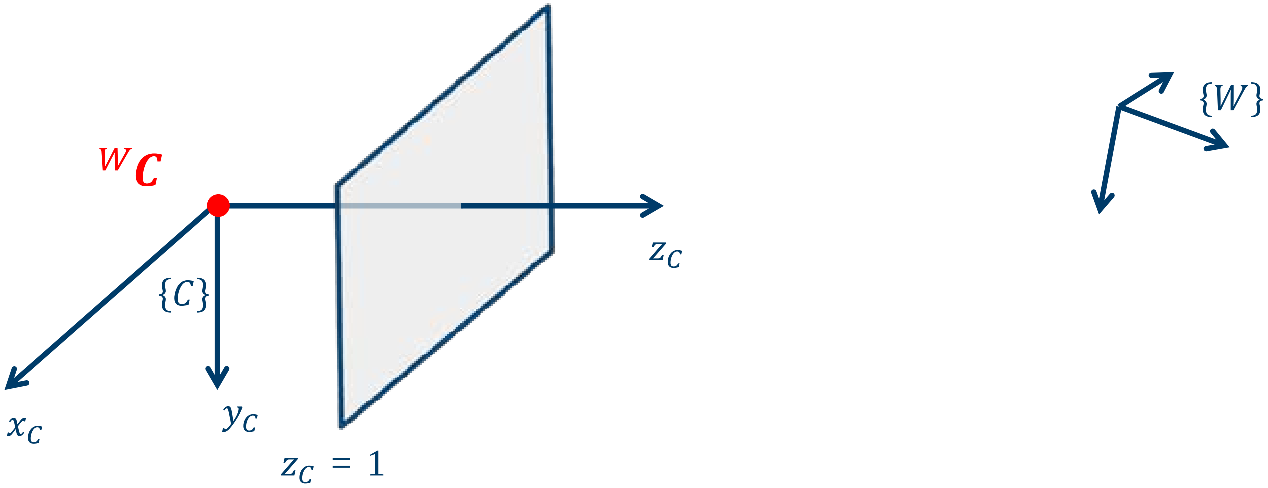
- Left 3×3 matrix in P

$$M = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \Longrightarrow P = [M \quad \boldsymbol{p}_4]$$

  – $det(M) \neq 0 \Longrightarrow \text{Finite projective camera}$
  – $det(M) = 0 \Longrightarrow \text{Camera center at infinity}$

NOTE
In the following slides $\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3, \boldsymbol{p}_4, \boldsymbol{P}^1, \boldsymbol{P}^2$ and $\boldsymbol{P}^3$ are homogeneous vectors, $\boldsymbol{p}_i \in \mathbb{P}^2$ while $\boldsymbol{P}^j \in \mathbb{P}^3$

Dissecting the camera matrix



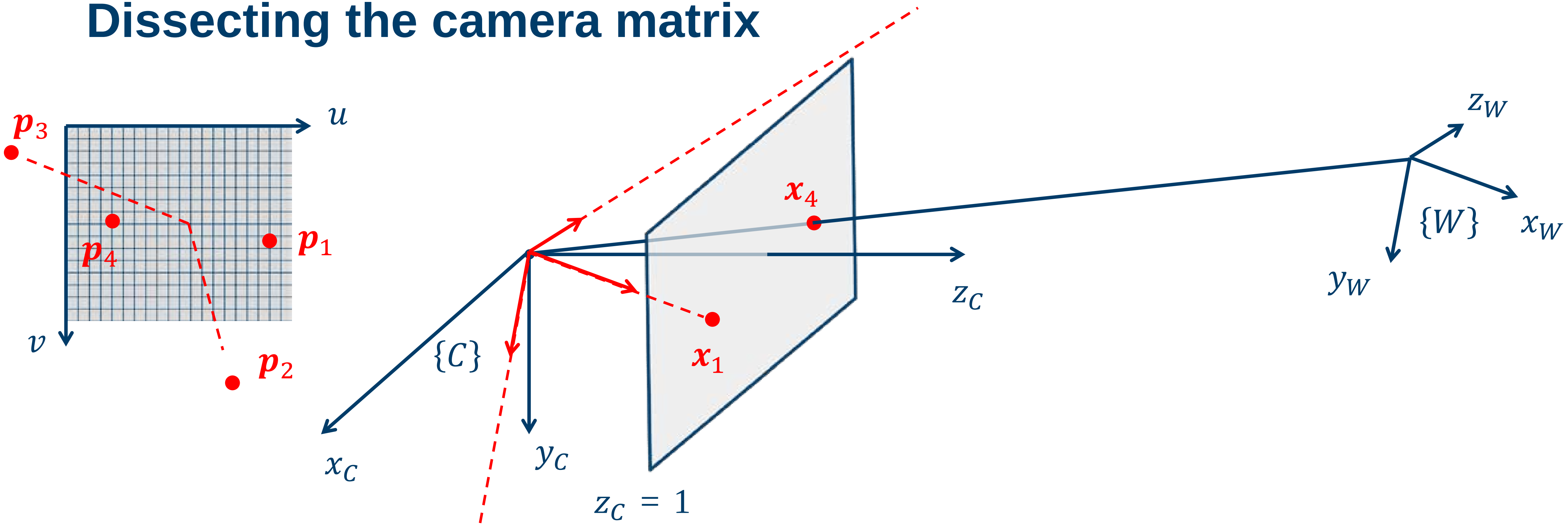
- The **camera center** $^{W}\boldsymbol{C}$ is given by the 1-dimensional right null-space of P
  - Finite camera

$$^{W}\tilde{\boldsymbol{C}} = \begin{bmatrix} -M^{-1}\boldsymbol{p}_4 \\ 1 \end{bmatrix} \Rightarrow {}^{W}\boldsymbol{C} = -M^{-1}\boldsymbol{p}_4$$

  - Camera at infinity

$$^{W}\tilde{\boldsymbol{C}} = \begin{bmatrix} \boldsymbol{d} \\ 0 \end{bmatrix} \text{ where } \boldsymbol{d} \in null(M), i.e. M\boldsymbol{d} = 0$$

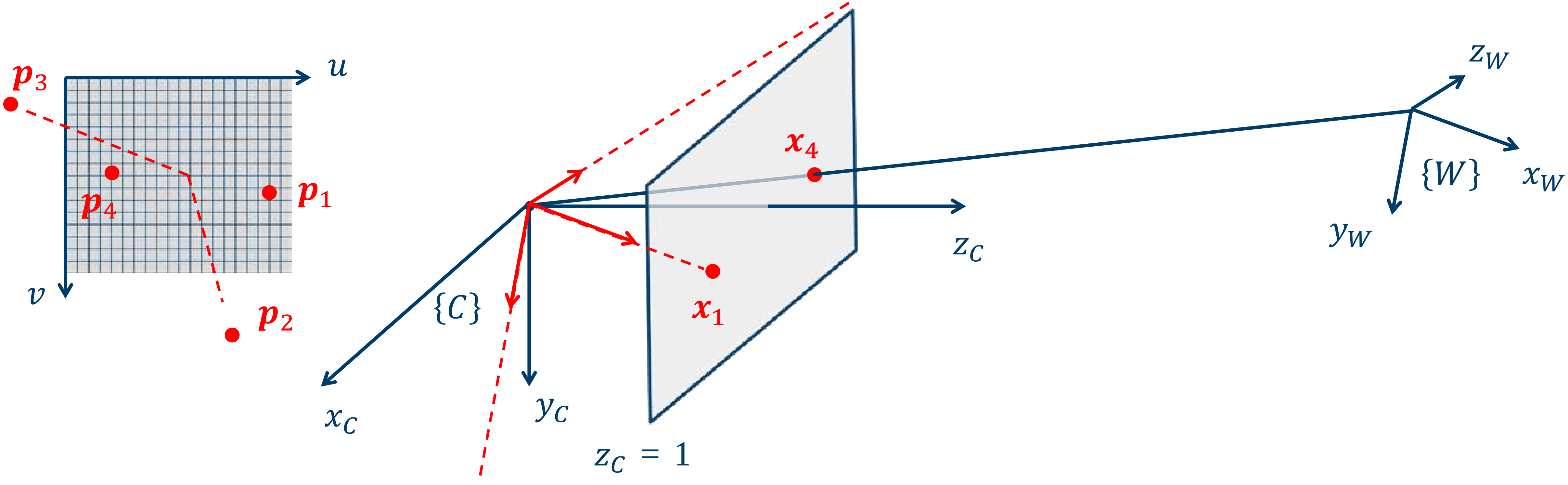
Dissecting the camera matrix



- The **column vectors** $\boldsymbol{p}_1$, $\boldsymbol{p}_2$ and $\boldsymbol{p}_3$ are the vanishing points in the image corresponding to the X-, Y- and Z-axis of $\{W\}$ respectively
  - If a coordinate axis is parallel to the xy-plane of $\{C\}$, the corresponding column vector will an infinite point

- The **column vector** $\boldsymbol{p}_4$ is the image of the origin of $\{W\}$

Dissecting the camera matrix



- Observe

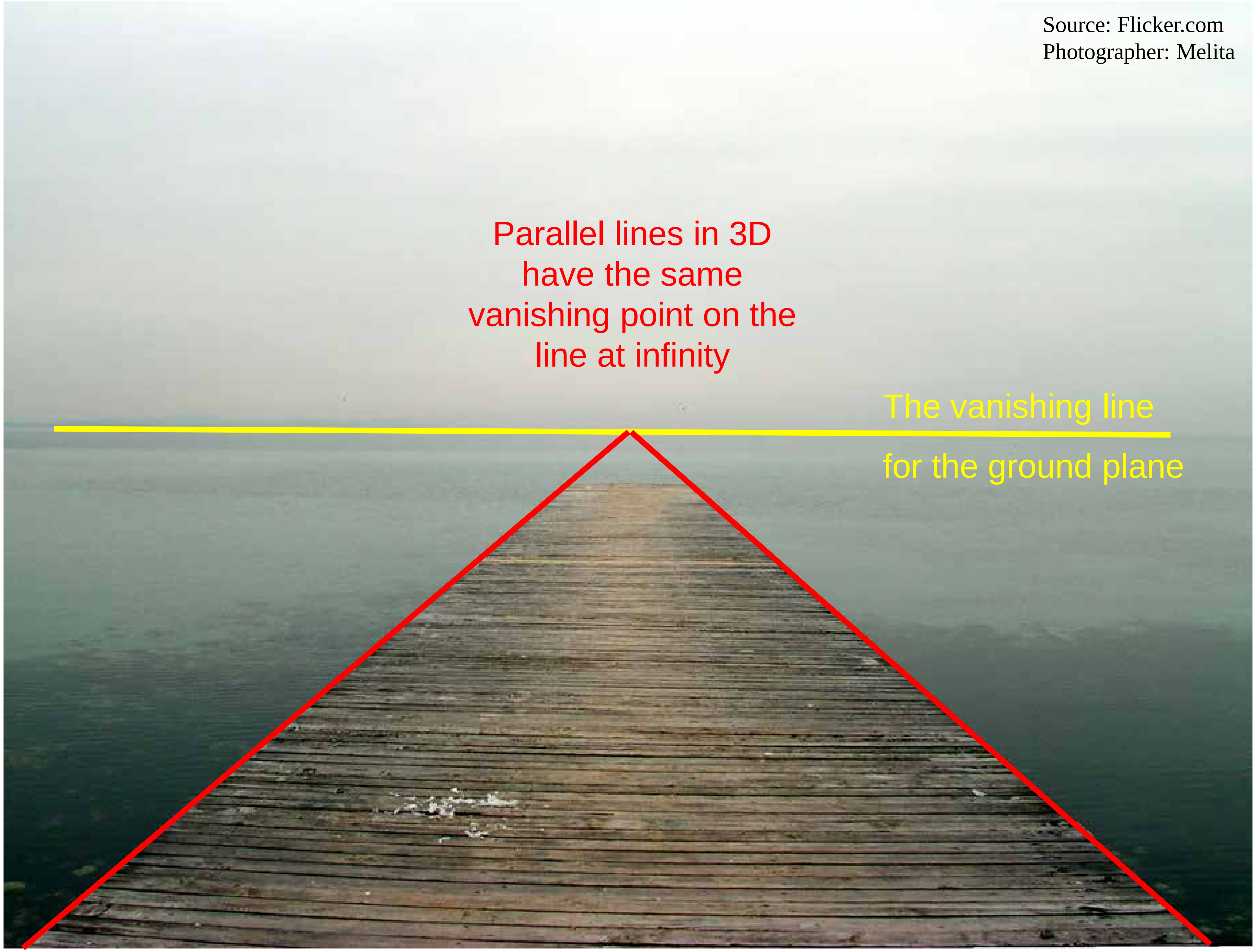
Points at infinity:
Homogeneous representation of the directions of the three coordinate axes of $\{W\}$

$$\boldsymbol{p}_1 = P\begin{bmatrix}1\\0\\0\\0\end{bmatrix} \quad \boldsymbol{p}_2 = P\begin{bmatrix}0\\1\\0\\0\end{bmatrix} \quad \boldsymbol{p}_3 = P\begin{bmatrix}0\\0\\1\\0\end{bmatrix} \quad \boldsymbol{p}_4 = P\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

Finite point:
Homogeneous representation of $\{W\}$'s origin


Source: Flicker.com
Photographer: Melita


Source: Flicker.com
Photographer: Melita
This is where the ground plane intersects the plane at infinity in $\mathbb{P}^3$
The vanishing line
for the ground plane


Source: Flicker.com
Photographer: Melita
Parallel lines in 3D
have the same
vanishing point on the
line at infinity
The vanishing line
for the ground plane

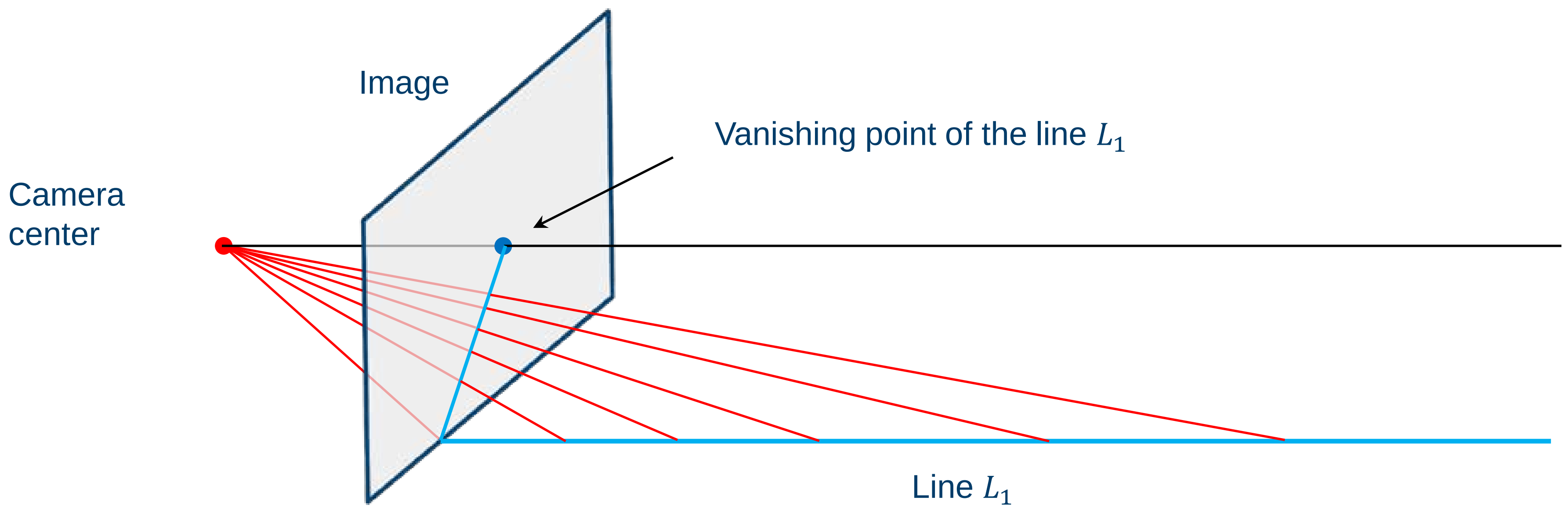
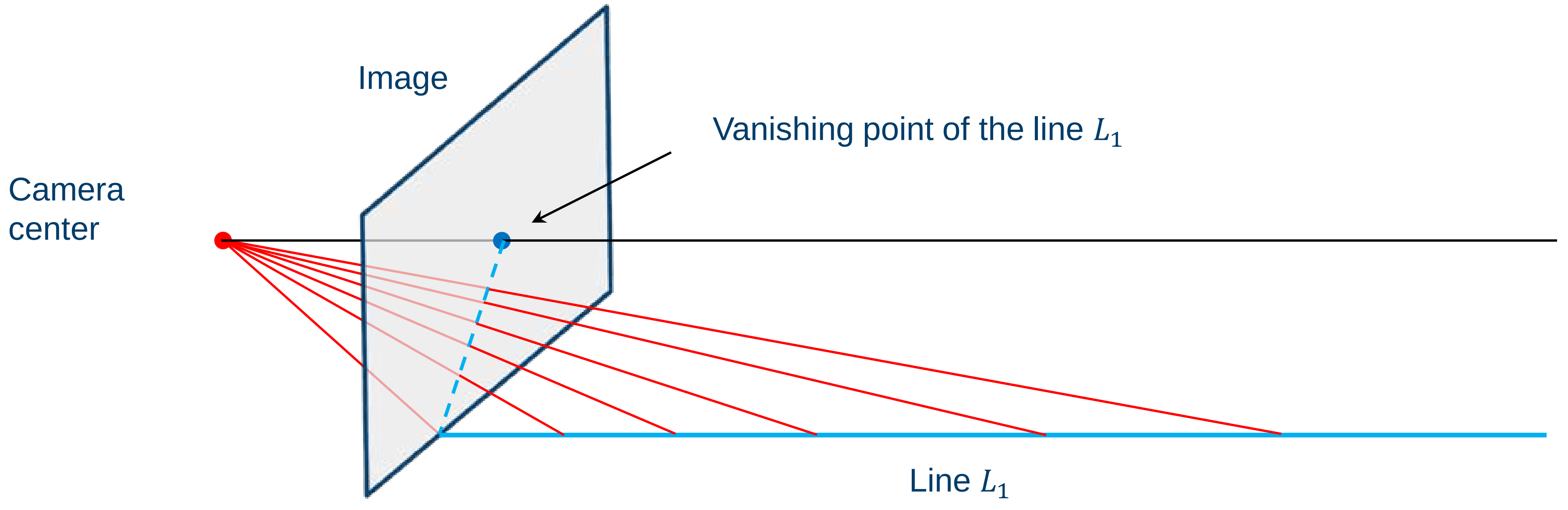
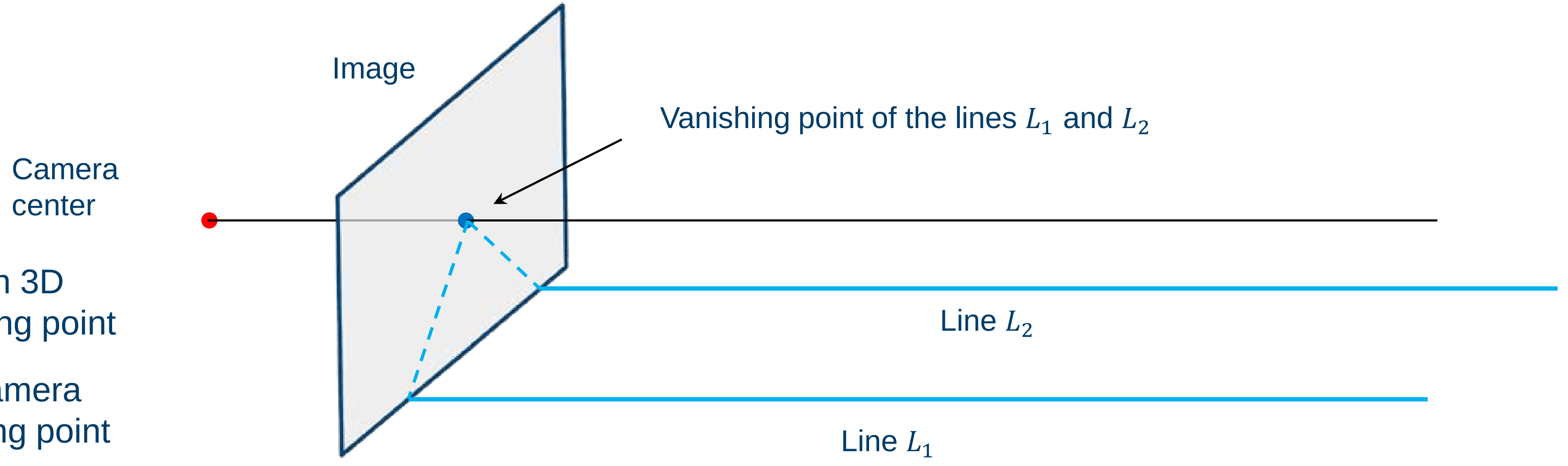


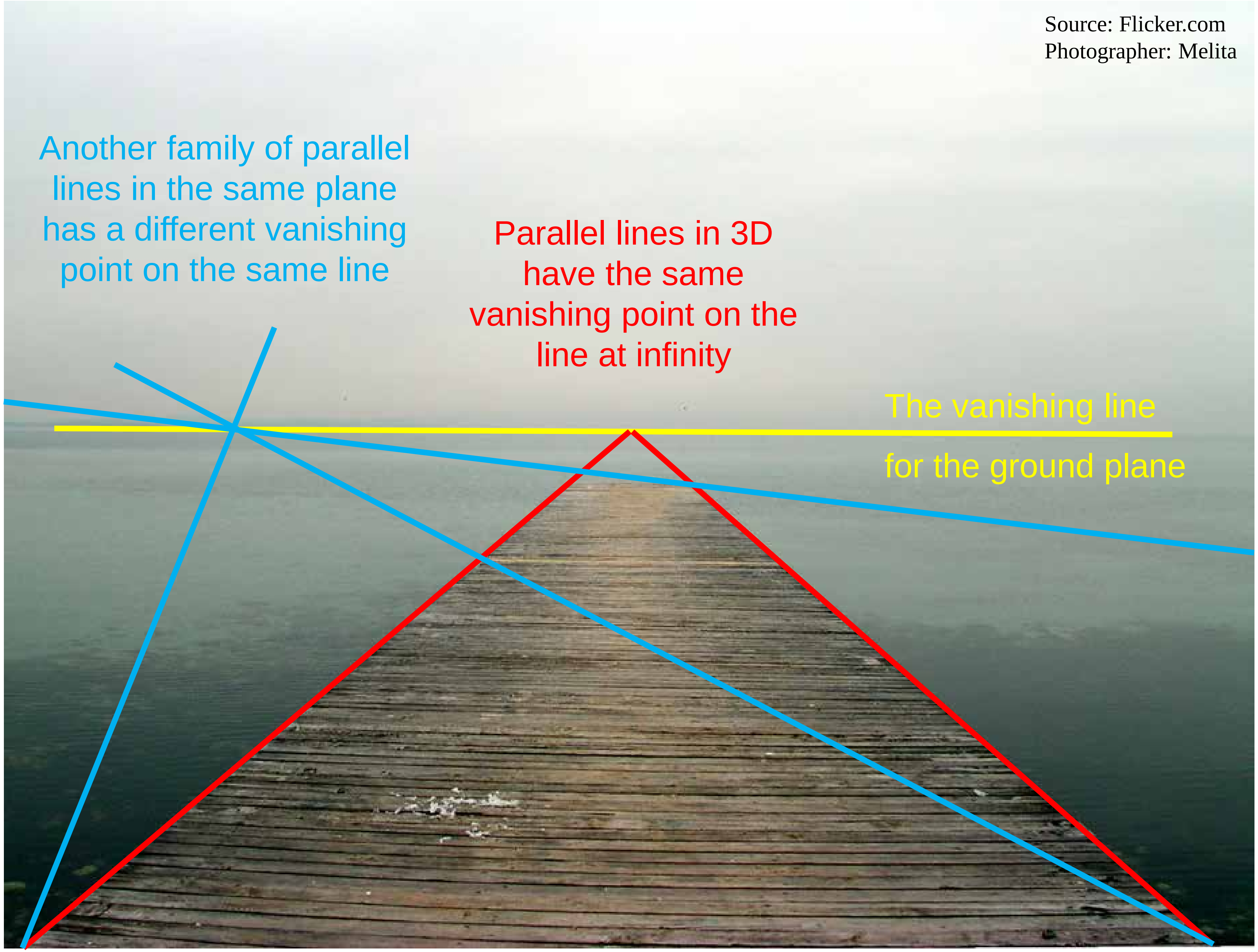
Image
Vanishing point of the line L_1
Camera
center
Line L_1

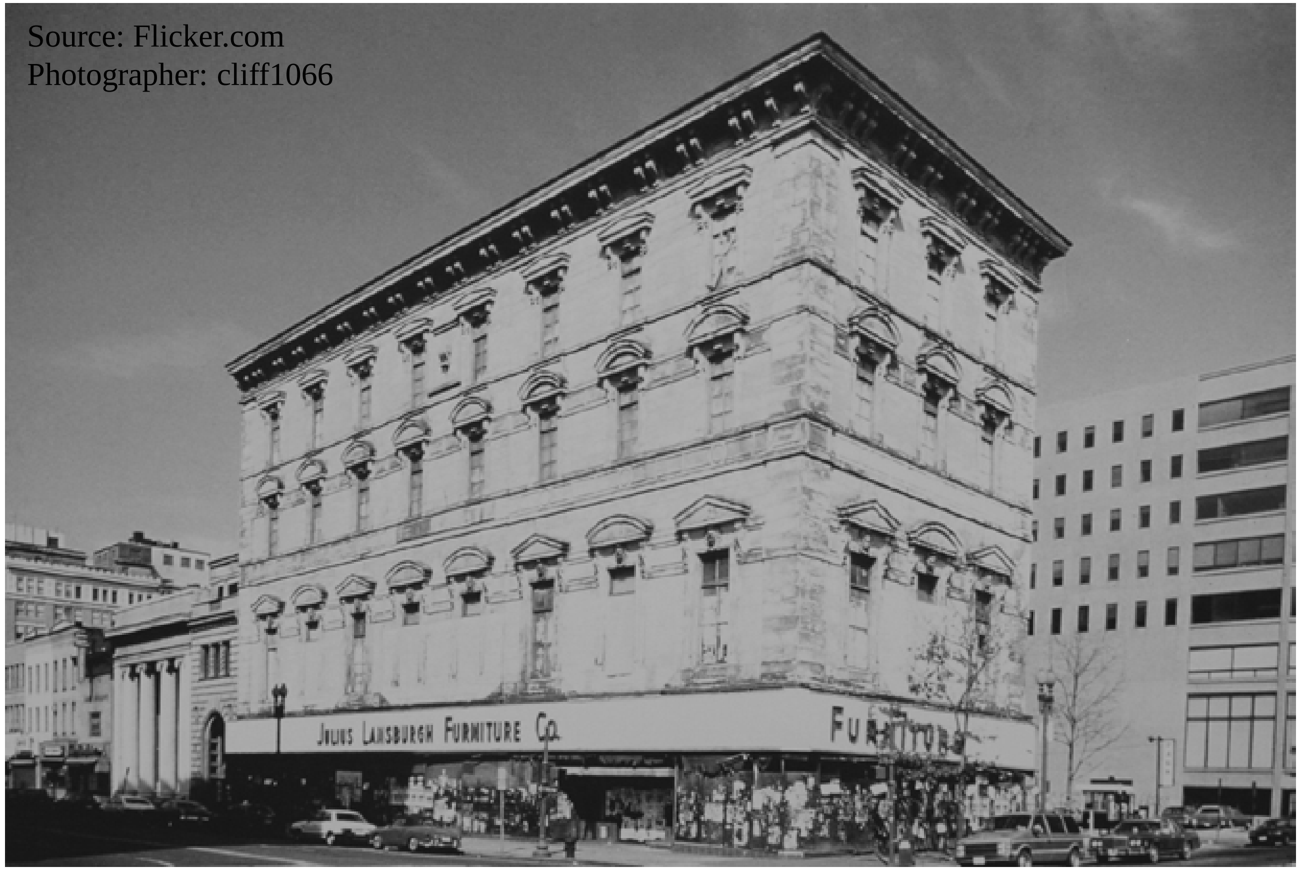




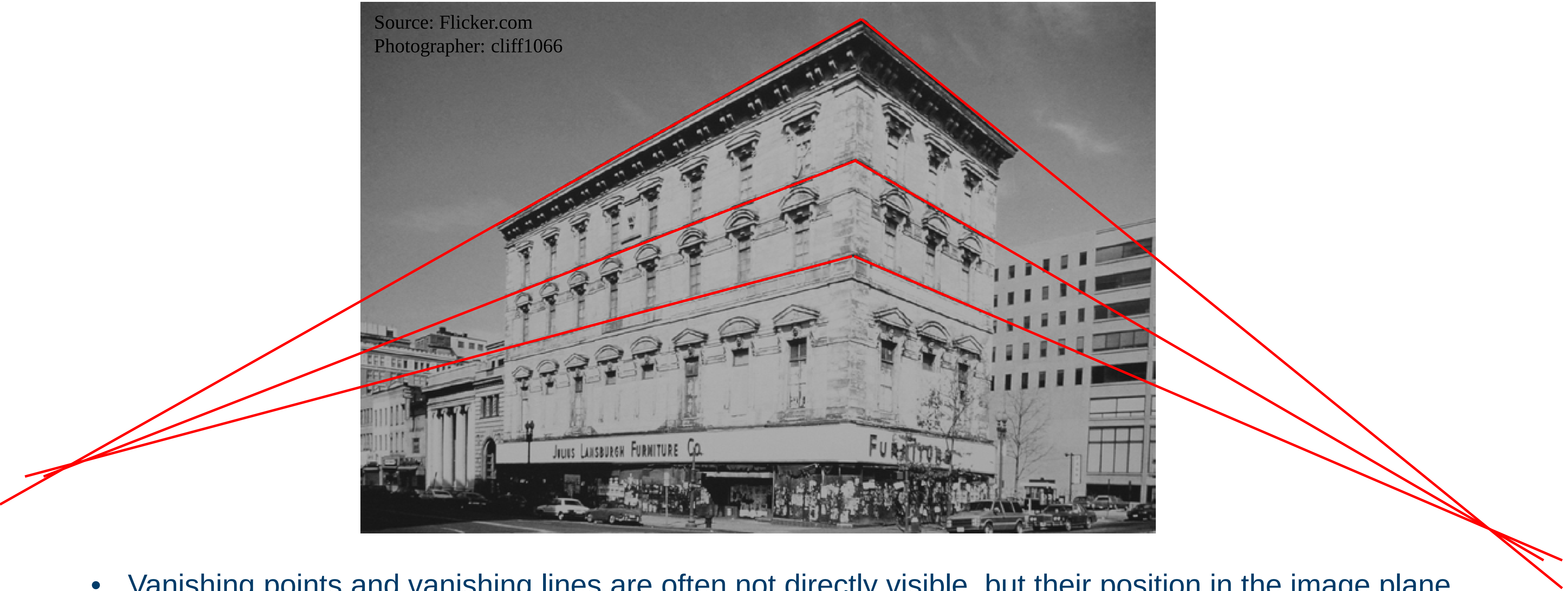
Any two parallel lines in 3D have the same vanishing point

The line through the camera center and the vanishing point is parallel to the lines

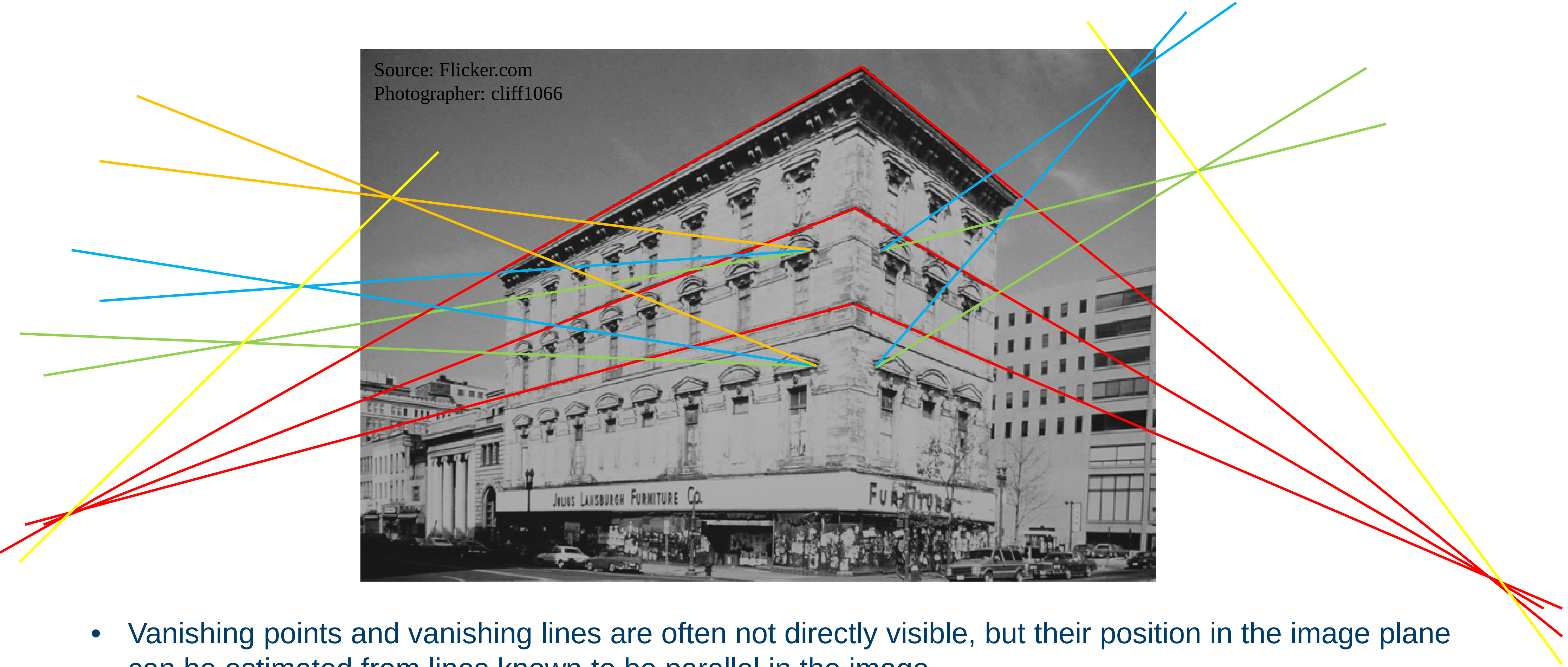

Source: Flicker.com
Photographer: Melita
Another family of parallel lines in the same plane has a different vanishing point on the same line
Parallel lines in 3D have the same vanishing point on the line at infinity
The vanishing line
for the ground plane



- Vanishing points and vanishing lines are often not directly visible, but can be estimated from lines known to be parallel in the image



- Vanishing points and vanishing lines are often not directly visible, but their position in the image plane can be estimated from lines known to be parallel in the image

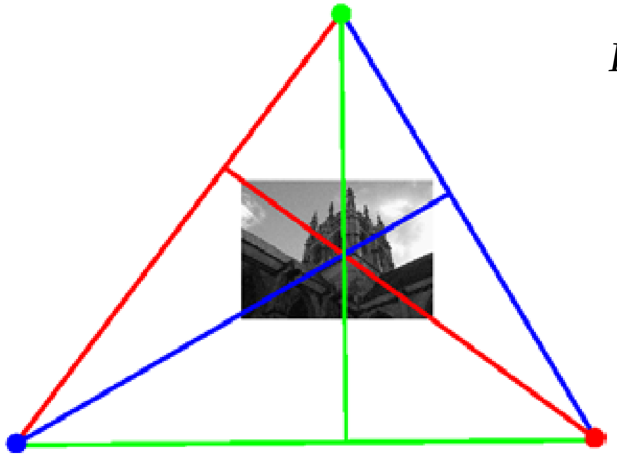


- Vanishing points and vanishing lines are often not directly visible, but their position in the image plane can be estimated from lines known to be parallel in the image

- These quantities are useful in several ways

Camera calibration

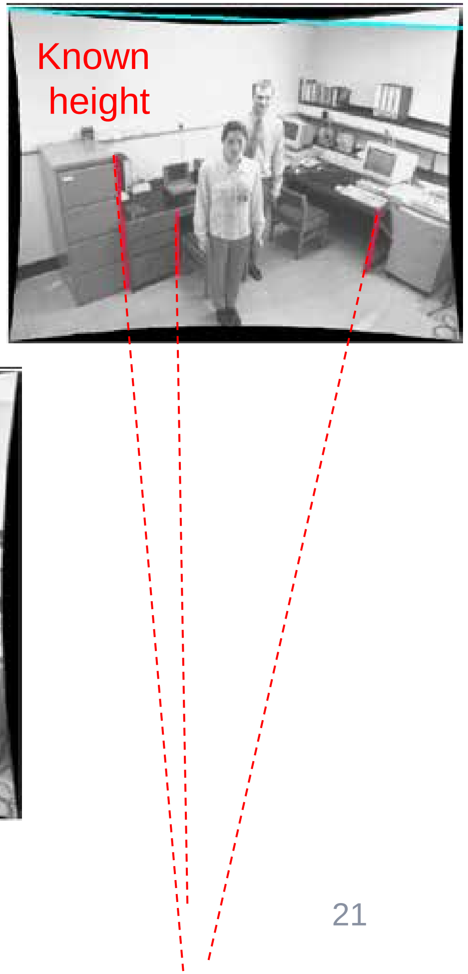
For instance it is possible to calibrate a camera from only three orthogonal vanishing points (if we also assume zero skew and square pixels)



$$K = \begin{bmatrix} 1163 & 0 & 548 \\ 0 & 1163 & 404 \\ 0 & 0 & 1 \end{bmatrix}$$

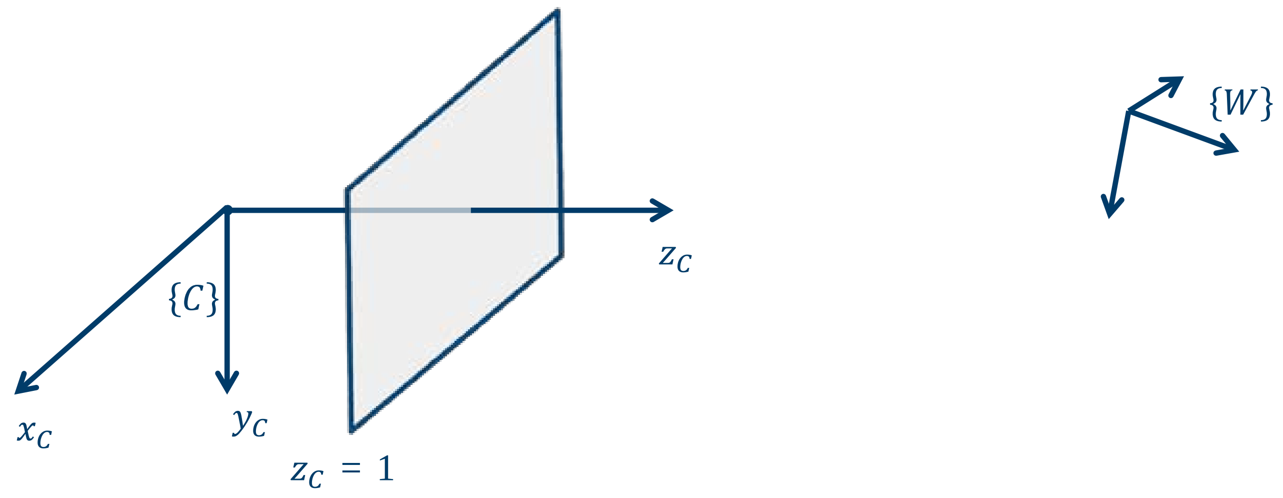
Measurements from a single image

Knowing the vanishing line of a plane, its vertical vanishing point and the size of vertical line segment, it is possible to determine the size of other vertical segments



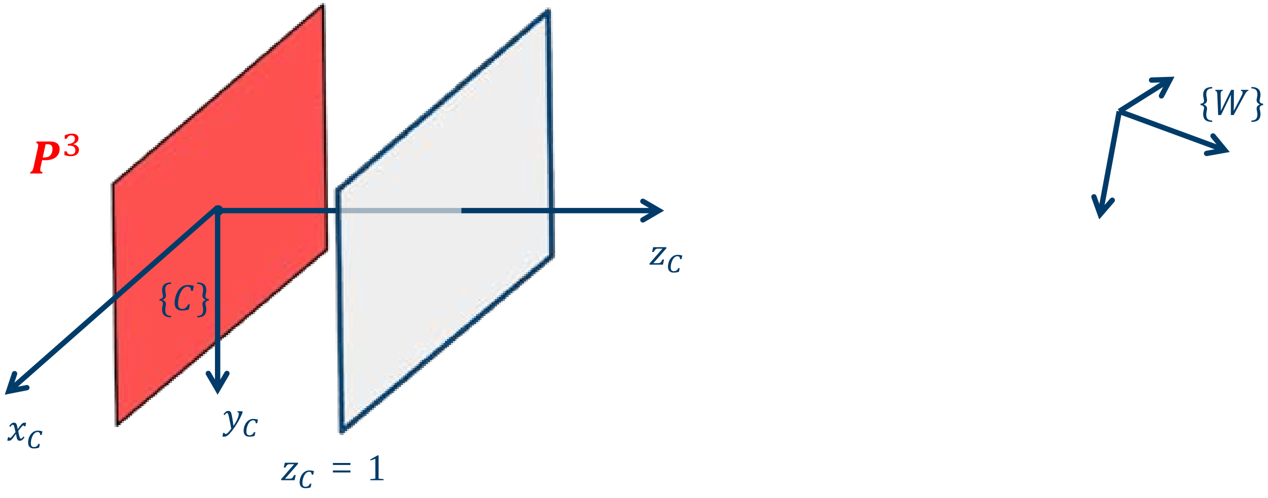
Images: http://www.robots.ox.ac.uk/~vgg/hzbook/

Dissecting the camera matrix



- The **row vectors** $\boldsymbol{P}^1$, $\boldsymbol{P}^2$ and $\boldsymbol{P}^3$ are homogeneous 4-vectors that can be interpreted as planes in $\{W\}$
  - Recall that in $\mathbb{P}^3$ points and planes are dual objects in the same way that points and lines are dual in $\mathbb{P}^2$

Dissecting the camera matrix

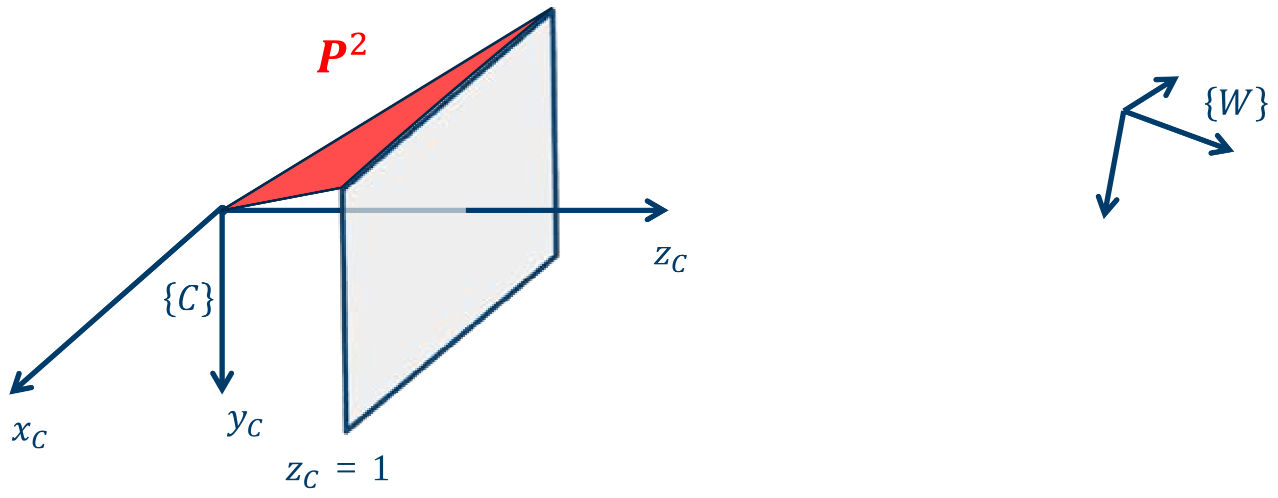


- The **row vectors** $\boldsymbol{P}^1$, $\boldsymbol{P}^2$ and $\boldsymbol{P}^3$ are homogeneous 4-vectors that can be interpreted as planes in $\{W\}$
  - Recall that in $\mathbb{P}^3$ points and planes are dual objects in the same way that points and lines are dual in $\mathbb{P}^2$

- $\boldsymbol{P}^3$ represents the xy-plane of $\{C\}$ in $\{W\}$ coordinates

  Observe that the xy-plane of $\{C\}$ consists of points ${}^W\boldsymbol{X}$ that are imaged on to the line at infinity of the image, i.e. $P\,{}^W\widetilde{\boldsymbol{X}} = [u, v, 0]^T$ which is equivalent with the homogeneous plane equation $\boldsymbol{P}^3\,{}^W\widetilde{\boldsymbol{X}} = 0$

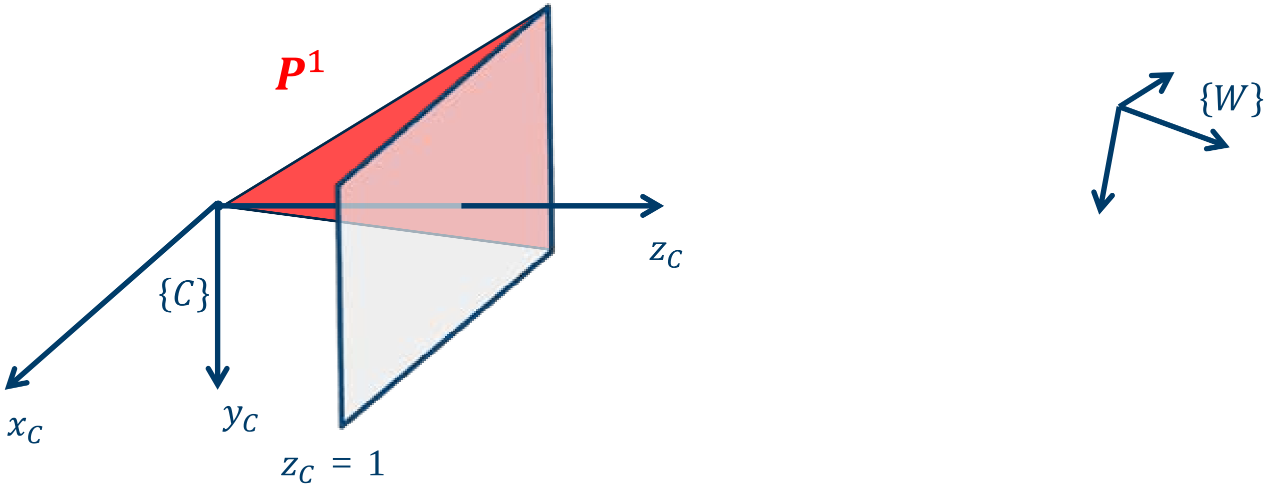
Dissecting the camera matrix



- The **row vectors** $\boldsymbol{P}^1$, $\boldsymbol{P}^2$ and $\boldsymbol{P}^3$ are homogeneous 4-vectors that can be interpreted as planes in $\{W\}$
  – Recall that in $\mathbb{P}^3$ points and planes are dual objects in the same way that points and lines are dual in $\mathbb{P}^2$

- $\boldsymbol{P}^3$ represents the xy-plane of $\{C\}$ in $\{W\}$ coordinates
- $\boldsymbol{P}^2$ represents the plane in $\{W\}$ that projects onto the u-axis of the image
  Observe that the points ${}^W\boldsymbol{X}$ that maps to the u-axis in the image must satisfy $P\,{}^W\widetilde{\boldsymbol{X}} = [u, 0, w]^T$ which is equivalent with the homogeneous plane equation $\boldsymbol{P}^{2\,W}\widetilde{\boldsymbol{X}} = 0$

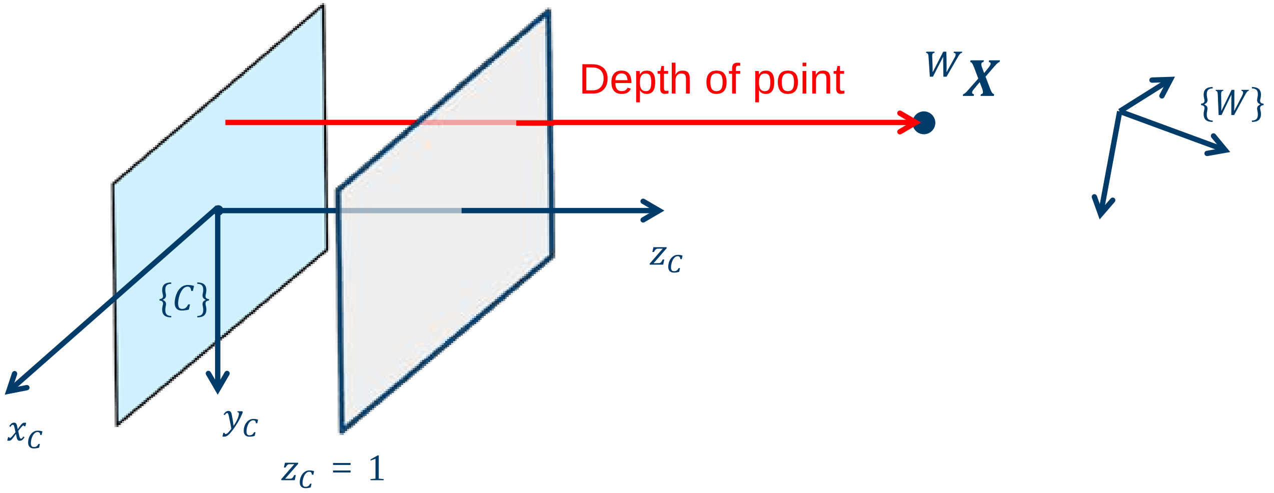
Dissecting the camera matrix



- The **row vectors** $\boldsymbol{P}^1$, $\boldsymbol{P}^2$ and $\boldsymbol{P}^3$ are homogeneous 4-vectors that can be interpreted as planes in $\{W\}$
  - Recall that in $\mathbb{P}^3$ points and planes are dual objects in the same way that points and lines are dual in $\mathbb{P}^2$

- $\boldsymbol{P}^3$ represents the xy-plane of $\{C\}$ in $\{W\}$ coordinates
- $\boldsymbol{P}^2$ represents the plane in $\{W\}$ that projects onto the u-axis of the image
- $\boldsymbol{P}^1$ represents the plane in $\{W\}$ that projects onto the v-axis of the image

Dissecting the camera matrix



- The projective camera model $\tilde{\boldsymbol{u}} = P\,{}^{W}\tilde{\boldsymbol{X}}$ does not require world points ${}^{W}\boldsymbol{X}$ to be in front of the camera
  - Points behind the camera will also project onto the image plane along a line through the projective center

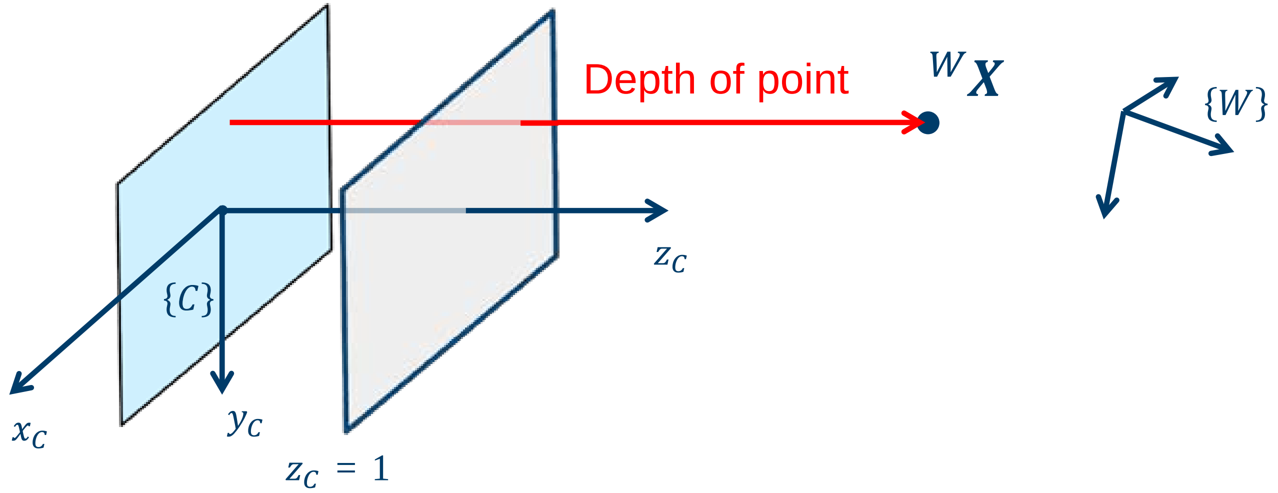
- The depth of point formula allow us to easily check whether or not a point ${}^{W}\boldsymbol{X}$ is in front of the camera,

$$depth({}^{W}\boldsymbol{X}; P) = \frac{w \cdot sign\left(det(M)\right)}{W\|\boldsymbol{m}^3\|}$$

where $\boldsymbol{m}^3$ is the 3'rd row of M and

$$P\begin{bmatrix}X & Y & Z & W\end{bmatrix}^T = w\begin{bmatrix}u & v & 1\end{bmatrix}^T$$

Dissecting the camera matrix



- The projective camera model $\tilde{\boldsymbol{u}} = P\,^W\tilde{\boldsymbol{X}}$ does not require world points $^W\boldsymbol{X}$ to be in front of the camera
  - Points behind the camera will also project onto the image plane along a line through the projective center

- The depth of point formula allow us to easily check whether or not a point $^W\boldsymbol{X}$ is in front of the camera,

$$depth(^W\boldsymbol{X}; P) = \frac{w \cdot sign(det(M))}{W\|\boldsymbol{m}^3\|}$$

where $\boldsymbol{m}^3$ is the 3'rd row of M and

$$P[X \quad Y \quad Z \quad W]^T = w[u \quad v \quad 1]^T$$

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ \underbrace{p_{31} \quad p_{32} \quad p_{33}}_{=\,\boldsymbol{m}^3} & & & p_{34} \end{bmatrix}$$

Dissecting the camera matrix

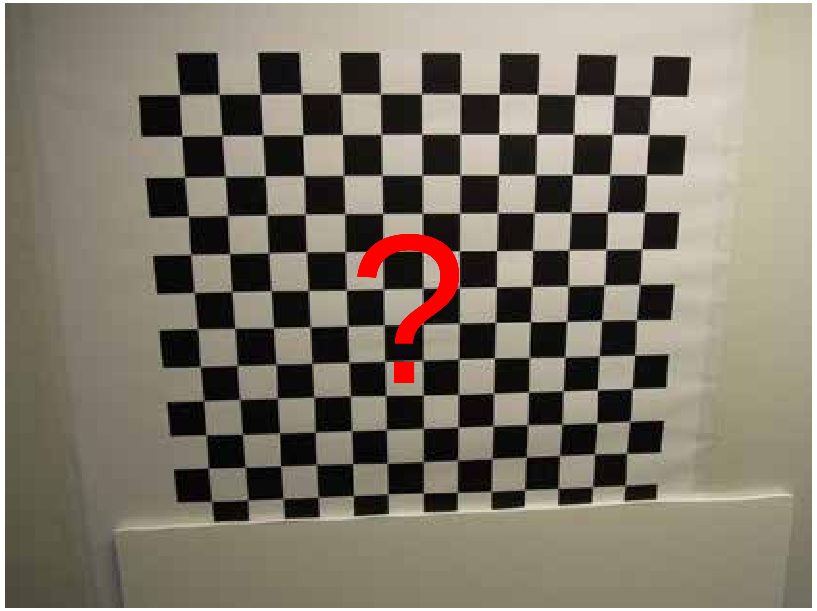
- So the general camera matrix P allow us to extract several geometrical bits of information about the camera

- Given the camera matrix P for a finite camera, it is even possible to retrieve K, R and $\boldsymbol{t}$
  - Another lecture

Estimating the camera matrix

- Apart from constructing it from K, R and $\boldsymbol{t}$, how can we find the camera matrix P for a given camera?

- Can we do it from an image???



Estimating the camera matrix

- Apart from constructing it from K, R and $\boldsymbol{t}$, how can we find the camera matrix P for a given camera?

- Can we do it from an image???
  - Yes, but we need to know the 3D coordinates of at least 6 points in the world and their corresponding 2D image coordinates
  - All 3D points can not be coplanar
  - Given such correspondences ${}^{W}\boldsymbol{X}_{\boldsymbol{i}} \leftrightarrow \boldsymbol{u}_{\boldsymbol{i}}$, we can estimate P from the equation $\tilde{\boldsymbol{u}} = P\,{}^{W}\tilde{\boldsymbol{X}}$

- Solving this problem with respect to P is very similar to how we estimated the homography matrix between two images

Estimating the camera matrix P

- We separate the known quantities u, v, X, Y, Z from the unknown quantities $p_{11}, \dots, p_{34}$

$$\tilde{\boldsymbol{u}} = P\tilde{\boldsymbol{X}}$$

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

$$\begin{bmatrix} X & Y & Z & 1 & 0 & 0 & 0 & 0 & -uX & -uY & -uZ & -u \\ 0 & 0 & 0 & 0 & X & Y & Z & 1 & -vX & -vY & -vZ & -v \end{bmatrix} \boldsymbol{p} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

- Where

$$\boldsymbol{p} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} & p_{21} & p_{22} & p_{23} & p_{24} & p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}^T$$

- Each correspondence ${}^{W}\boldsymbol{X}_{\boldsymbol{i}} \leftrightarrow \boldsymbol{u}_{\boldsymbol{i}}$ contribute with two equations(rows)

Estimating the camera matrix P

- Hence from several 3D-2D correspondences we get a linear system of equations much like how we estimated homographies

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -u_1X_1 & -u_1Y_1 & -u_1Z_1 & -u_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -v_1X_1 & -v_1Y_1 & -v_1Z_1 & -v_1 \\ X_2 & Y_2 & Z_2 & 1 & 0 & 0 & 0 & 0 & -u_2X_2 & -u_2Y_2 & -u_2Z_2 & -u_2 \\ 0 & 0 & 0 & 0 & X_2 & Y_2 & Z_2 & 1 & -v_2X_2 & -v_2Y_2 & -v_2Z_2 & -v_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \boldsymbol{p} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \end{bmatrix}$$

$$A\boldsymbol{p} = \boldsymbol{0}$$

- So we get the basic DLT algorithm for estimating P from 3D-2D correspondences
  1. Build matrix A (from at least 6 point-correspondences)
  2. Obtain SVD of A: $A = UDV^T$
  3. The non-trivial solution to $A\boldsymbol{p} = \mathbf{0}$ corresponds to the right singular vector with the smallest singular value Typically the last column of V
  4. Reconstruct the camera matrix P from $\boldsymbol{p}$

Estimating the camera matrix P

- As usual, the DLT algorithm performs better if we work on normalized points
  - 2D image points should be normalized so that their centroid is at the origin and their root-mean-square distance from the origin is $\sqrt{2}$
  - 3D reference points should be normalized so that their centroid is at the origin and their RMS distance from the origin is $\sqrt{3}$

- Normalized DLT algorithm for estimating P from 3D-2D correspondences
  1. Normalize the 2D and 3D point sets with the transforms T_1 and T_2
  2. Estimate the matrix $\hat{P}$ from normalized points using the basic DLT algorithm
     Ideally one should refine the estimate by minimizing the reprojection error $\sum_i d\left(\tilde{\boldsymbol{u}}_i, \tilde{P}\tilde{\boldsymbol{X}}_i\right)^2$ iteratively
  3. Denormalize $P = {T_1}^{-1}\hat{P}T_2$

Summary

- The camera matrix *P*
  - Contains useful geometrical information
  - Vanishing points
  - Vanishing lines
  - Can be estimated from 3D-2D correspondences

- Additional reading
  - Szeliski: 6.2.1, 6.3.3

- Optional reading
  - *Single View Metrology* by A. Criminisi, I. Reid and A. Zisserman