AMME4710: COMPUTER VISION AND IMAGE PROCESSING WEEK 5

Dr. Mitch Bryson

School of Aerospace, Mechanical and Mechatronic Engineering, University of Sydney

Last Week

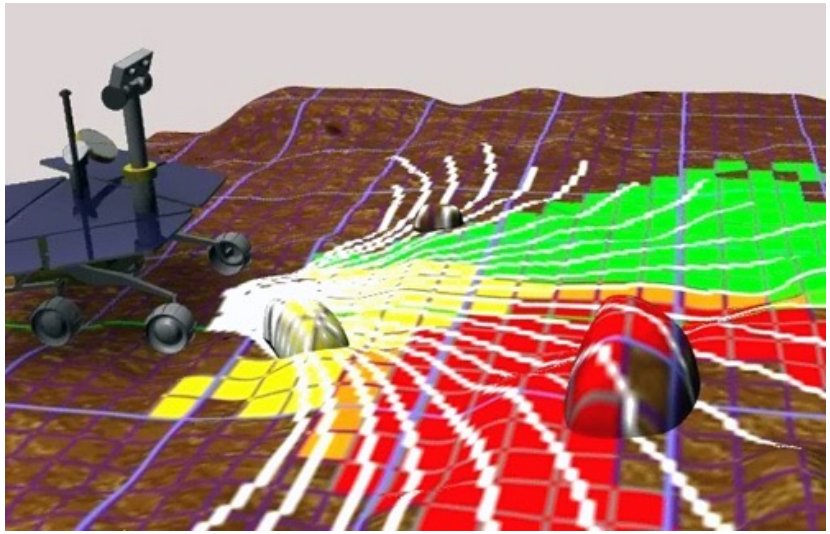
- Shapes, Features and Object Detection
  - Hough transform detectors for lines and circles
  - RANdom SAmple Concensus (RANSAC) for detection and model fitting
  - Image Interest Points, Harris corners, use of SIFT/SURF in feature-based object detection

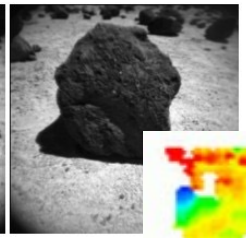
This Week's Lecture

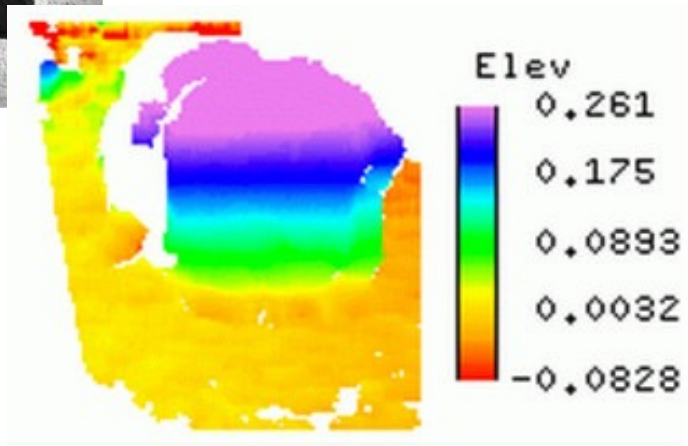
- Introduction to projective geometry and stereo vision

- Learning Objectives:
  - To present and understand models for geometric image formation and the transformation relationships between objects in 3D and their projections in images
  - To explore the fundamentals of stereo vision by taking matching points in two images and using parallax to determine 3D structure

Stereo Vision and 3D Modelling from Images





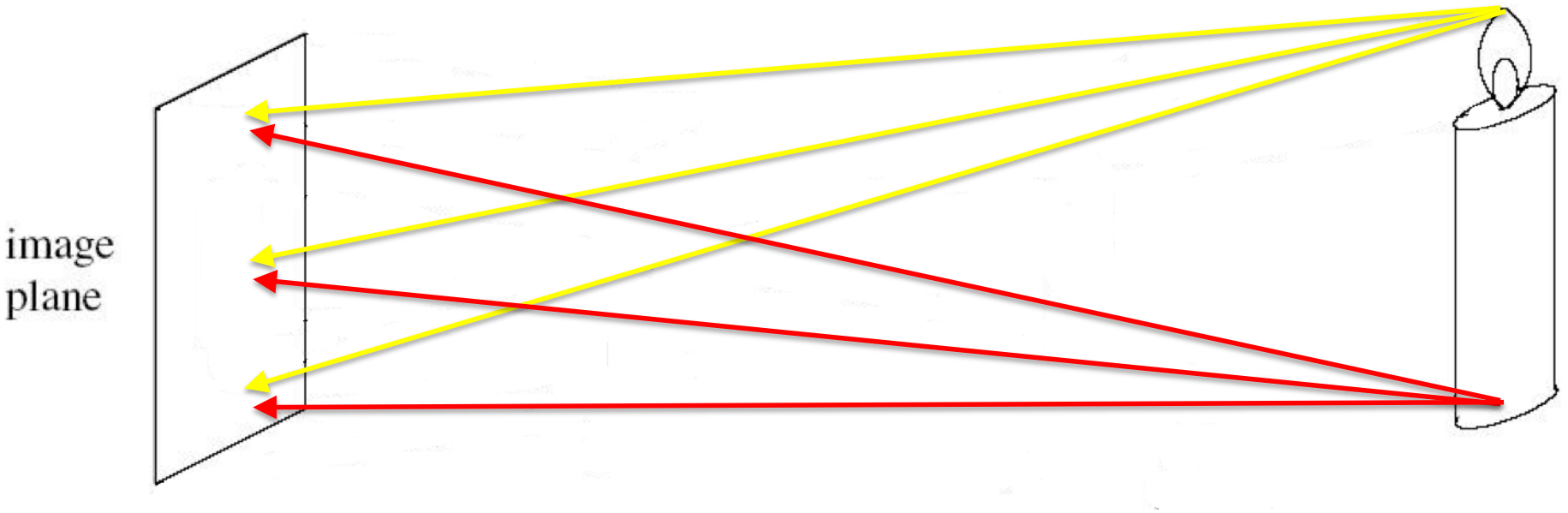


N. Snavely, S.M. Seitz, R. Szeliski, “Modeling the World from Internet Photo Collections”, International Journal of Computer Vision, 2007

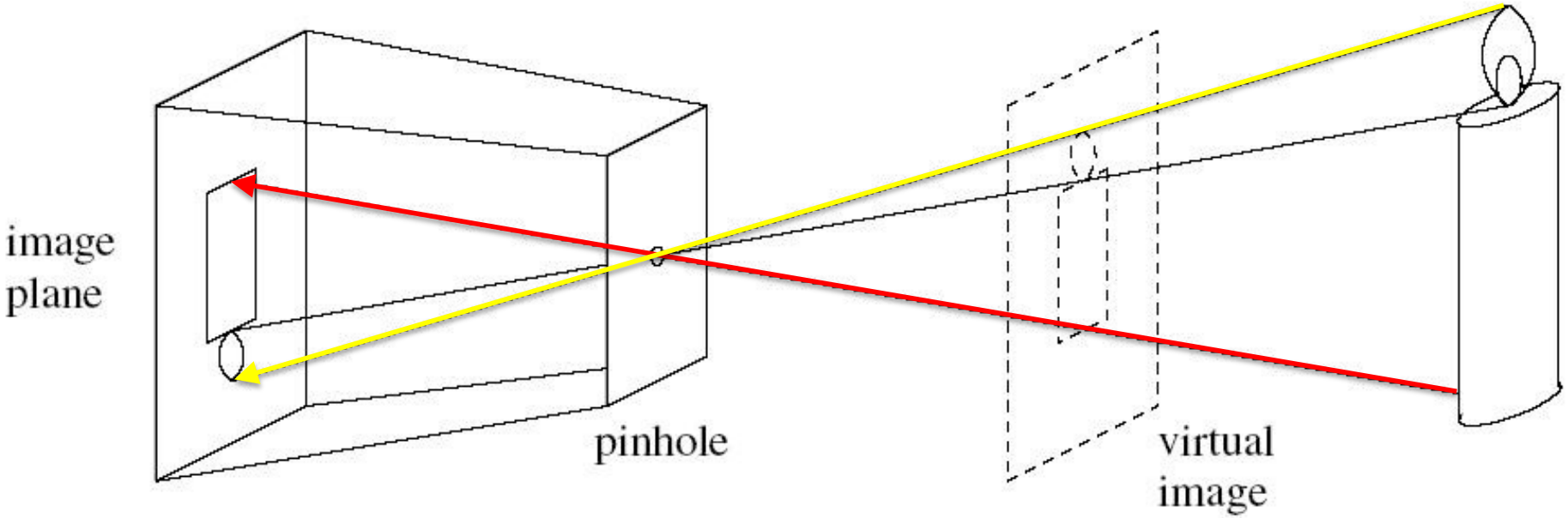
S. Goldberg, M. Maimone and L. Matthies, “Stereo Vision and Rover Navigation Software for Planetary Exploration”, IEEE Aerospace Conference, 2012

Imaging Model

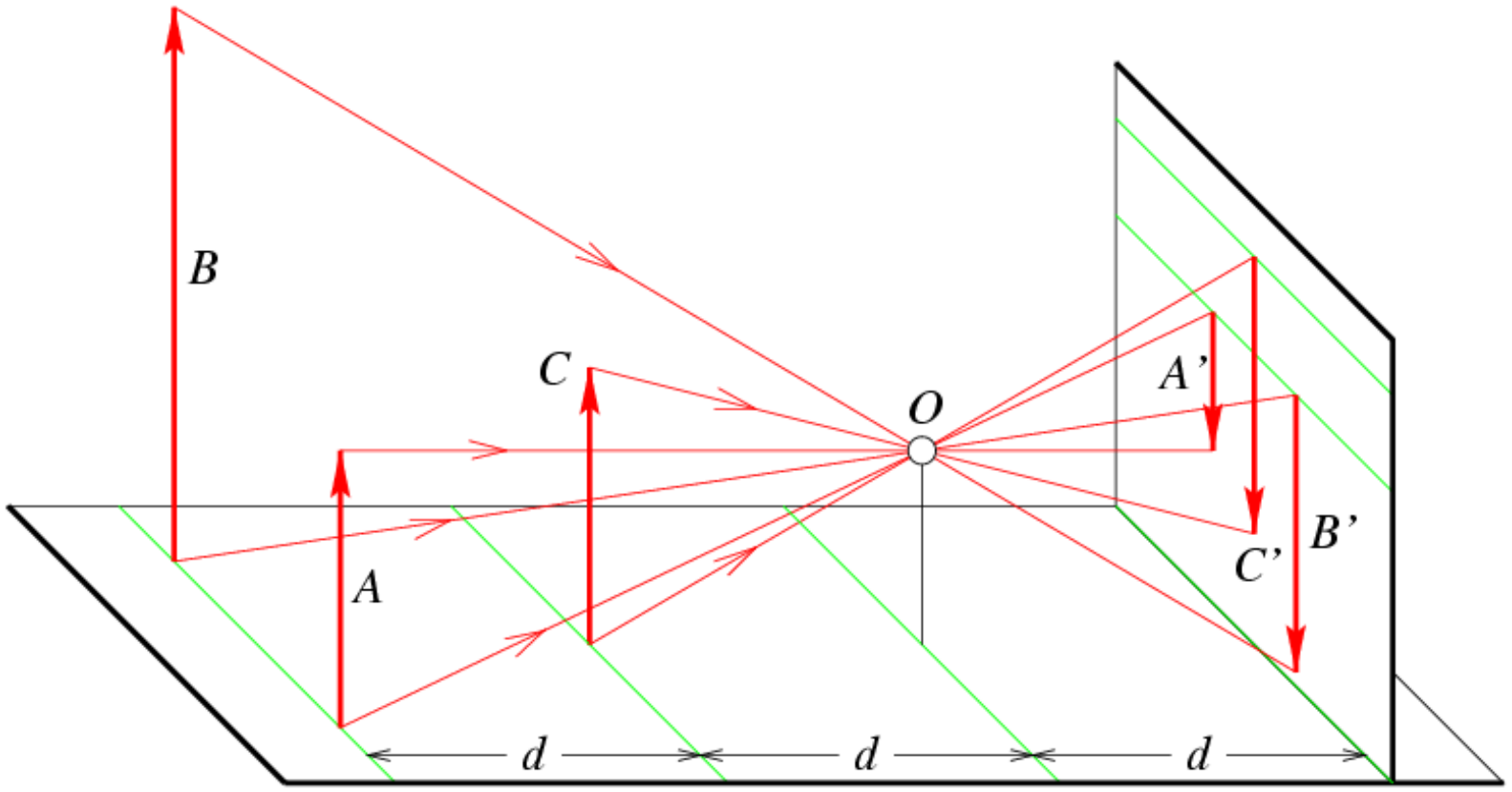
Imaging Model



Pinhole Imaging Model



Perspective Projection



Perspective Projection

Courtesy of Dustin Resch (http://dustinresch.blogspot.com.au/2012/03/grad-school-perspective-assignments.html)

Perspective Projection

Courtesy of Dustin Resch (http://dustinresch.blogspot.com.au/2012/03/grad-school-perspective-assignments.html)

Camera Pinhole Projection Model

- Consider a basic pinhole camera in 3D

Focal point

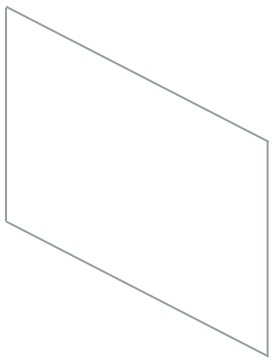
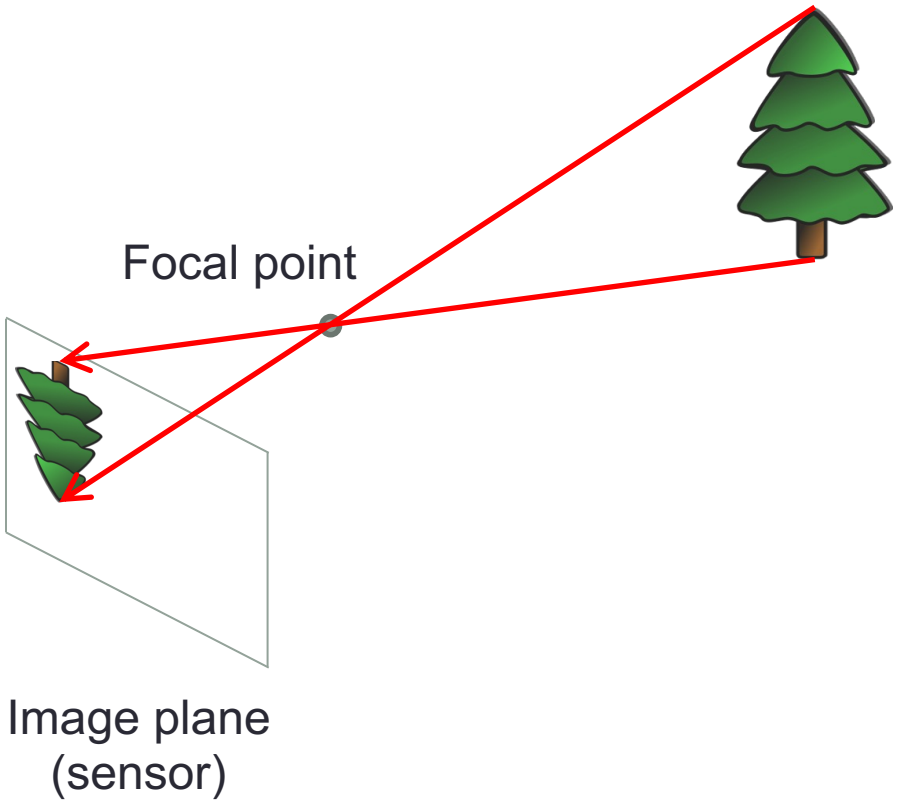


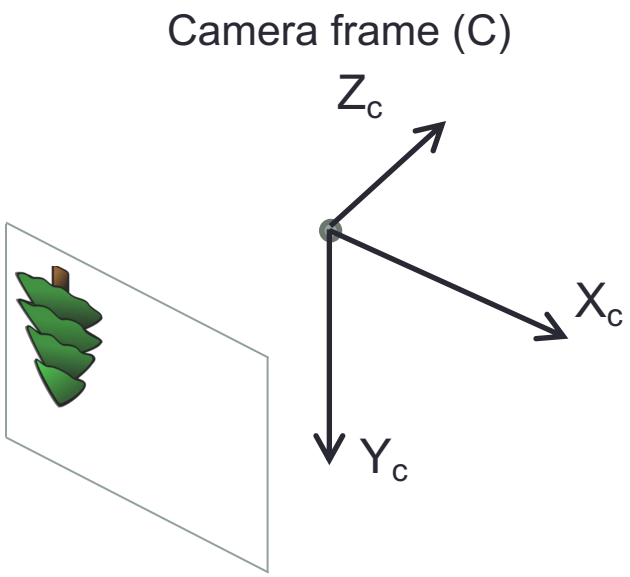
Image plane
(sensor)

Camera Pinhole Projection Model



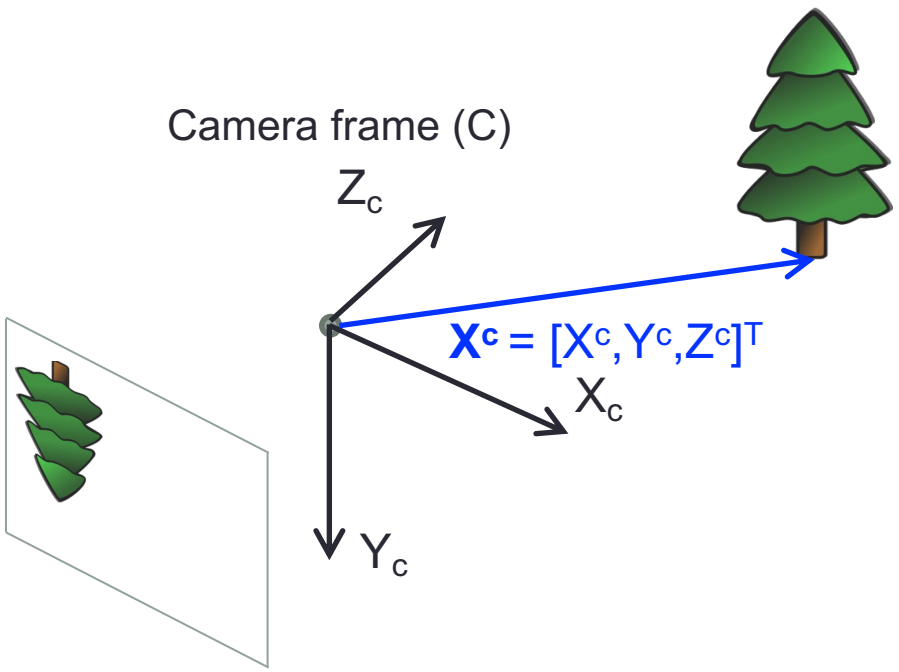
- Consider a basic pinhole camera in 3D

Camera Pinhole Projection Model



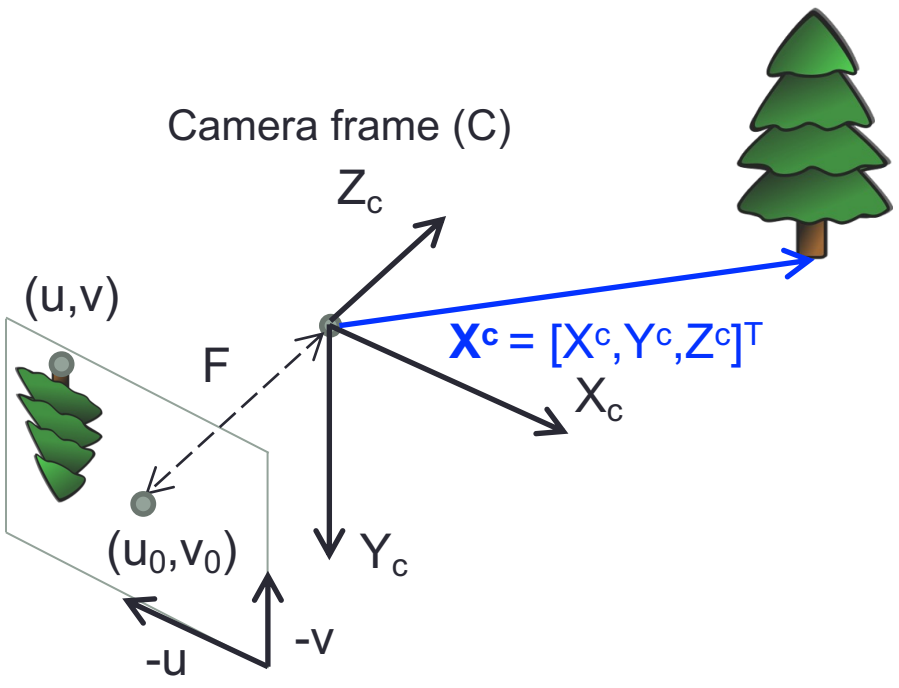
- Consider a basic pinhole camera in 3D
- We define a camera frame C, centered at the focal point with z-axis normal to image plane

Camera Pinhole Projection Model



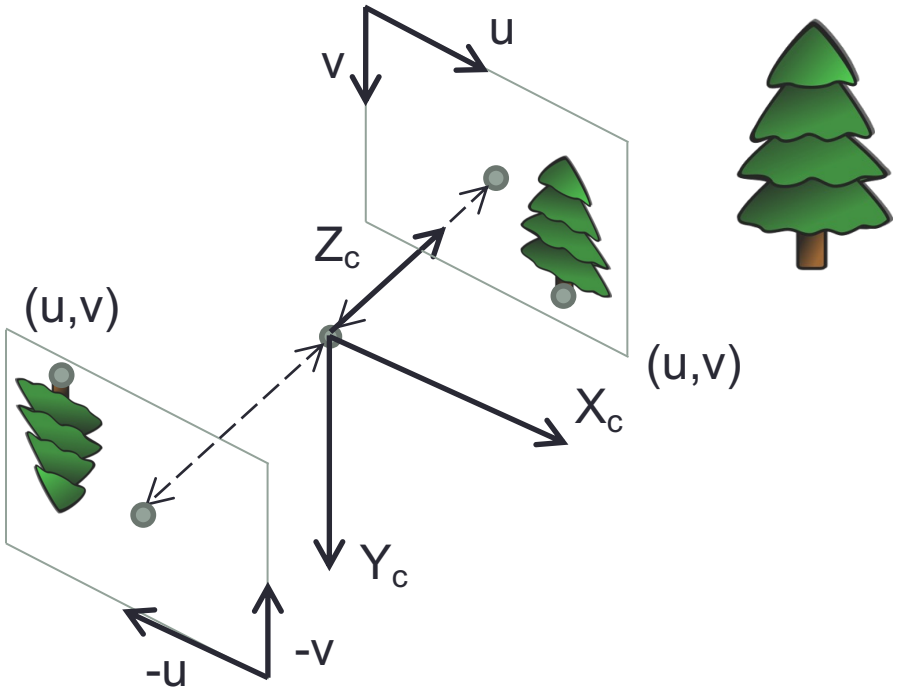
- Consider a basic pinhole camera in 3D
- We define a camera frame C, centered at the focal point with z-axis normal to image plane

Camera Pinhole Projection Model



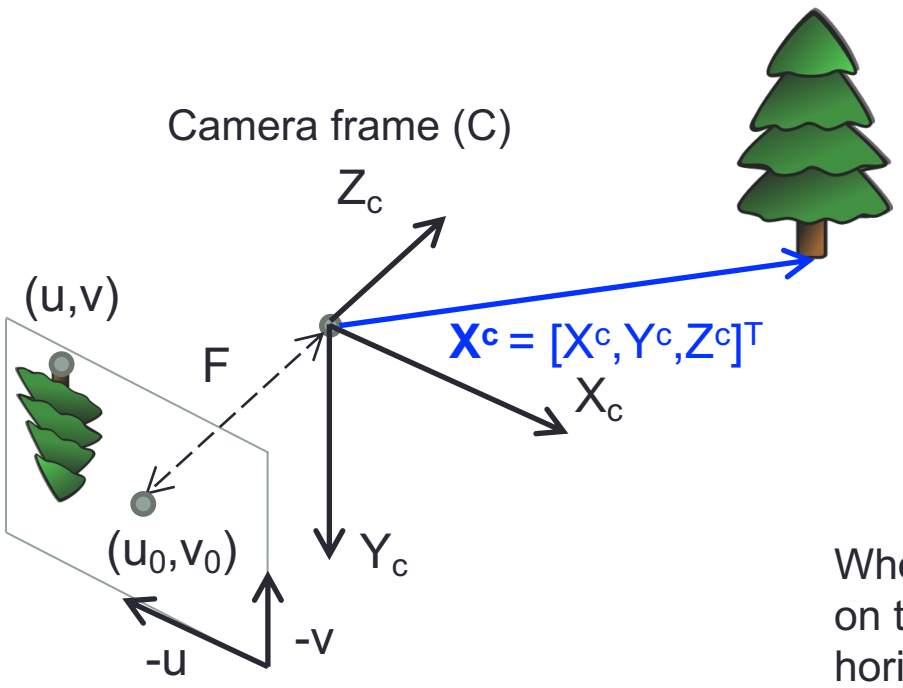
- Consider a basic pinhole camera in 3D
- We define a camera frame C, centered at the focal point with z-axis normal to image plane
- The distance between the image plane and focal point is F and the intersection of the z-axis with the image plane is (u_0,v_0), known as the principle point

Camera Pinhole Projection Model



- Consider a basic pinhole camera in 3D
- We define a camera frame C, centered at the focal point with z-axis normal to image plane
- The distance between the image plane and focal point is F and the intersection of the z-axis with the image plane is (u_0,v_0), known as the principle point
- (u,v) represent horizontal and vertical coordinates on the image plane: most camera "flip" an image, hence the use of inverted coordinates

Camera Pinhole Projection Model

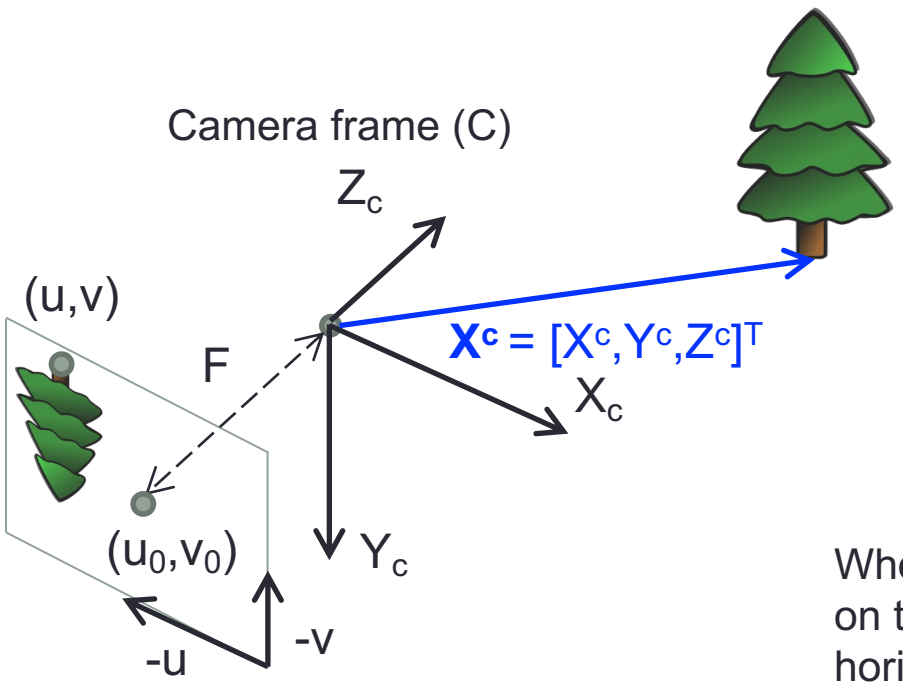


- Coordinates in (u,v) are often referenced in pixels, hence we define the parameters f_x and f_y (also called focal lengths):

$$f_x = \frac{F}{p_x} \qquad f_y = \frac{F}{p_y}$$

Where p_x, p_y are the physical sizes of pixels on the image plane, measured in the horizontal and vertical directions

Camera Pinhole Projection Model



- Coordinates in (u,v) are often referenced in pixels, hence we define the parameters f_x and f_y (also called focal lengths):

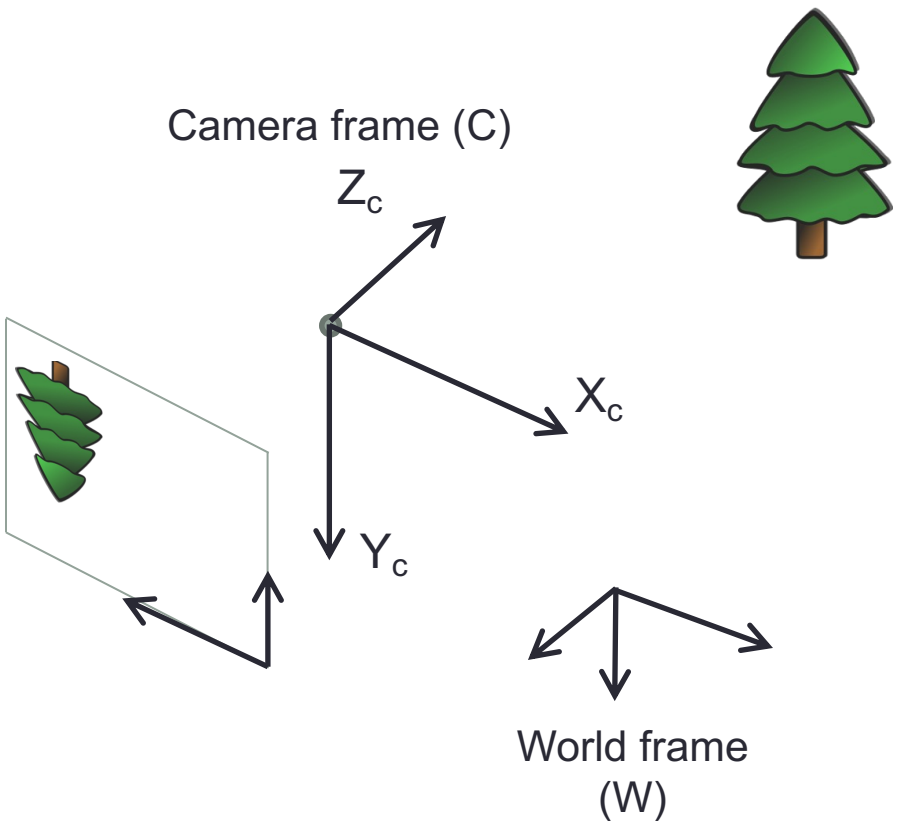
$$f_x = \frac{F}{p_x} \qquad f_y = \frac{F}{p_y}$$

Where p_x, p_y are the physical sizes of pixels on the image plane, measured in the horizontal and vertical directions

- The relationship between pixel coordinates and camera frame coordinates is thus:

$$u = f_x \frac{X_c}{Z_c} + u_0 \qquad v = f_y \frac{Y_c}{Z_c} + v_0$$

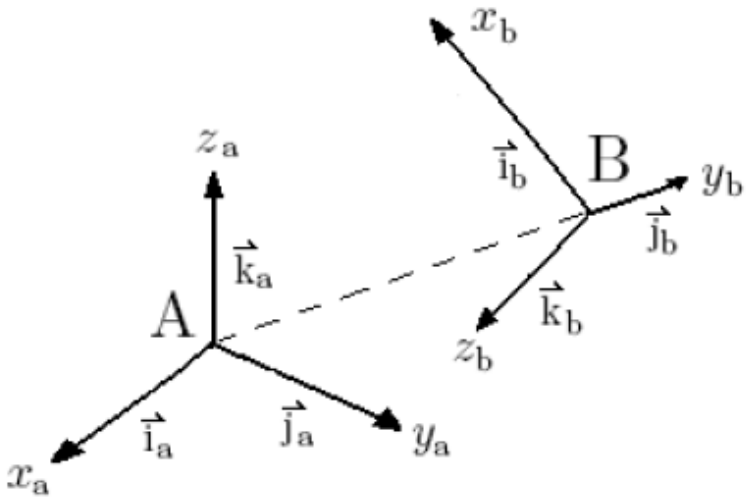
Camera Pinhole Projection Model



- Consider now the relationship between the object's position in camera coordinates and an arbitrarily defined world coordinate system W

Revision: Direction Cosine Matrix (DCM)

- The relative orientation between two reference frames, and hence the transformation of coordinates between the frames can be represented using a Direction Cosine Matrix (DCM), also known as a “rotation matrix”
- Specifically represents the components of the basis vectors of frame B w.r.t the basis vectors of frame A



Two frames of reference A and B

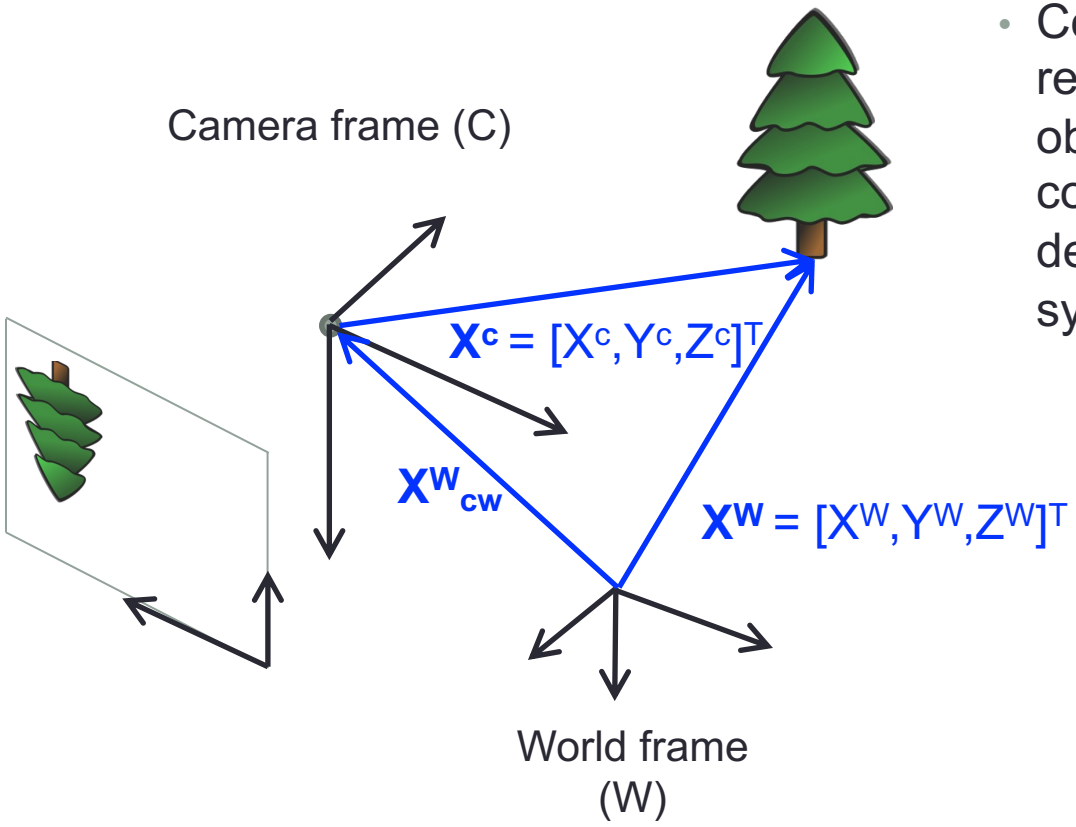
$$\mathbf{x}_b = C_a^b \mathbf{x}_a$$

$$C_a^b = \begin{bmatrix} i_a.i_b & j_a.i_b & k_a.i_b \\ i_a.j_b & j_a.j_b & k_a.j_b \\ i_a.k_b & j_a.k_b & k_a.k_b \end{bmatrix}$$

The DCM is an orthogonal matrix, hence:

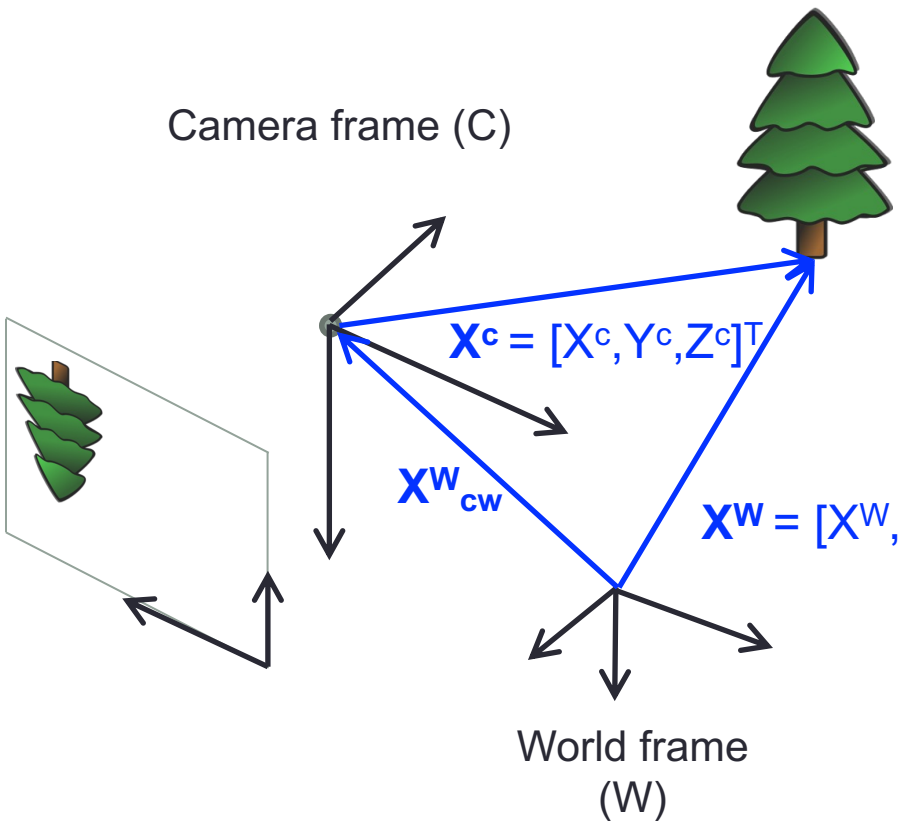
$$C_b^a = (C_a^b)^{-1} = (C_a^b)^T$$

Camera Pinhole Projection Model



- Consider now the relationship between the object's position in camera coordinates and an arbitrarily defined world coordinate system W

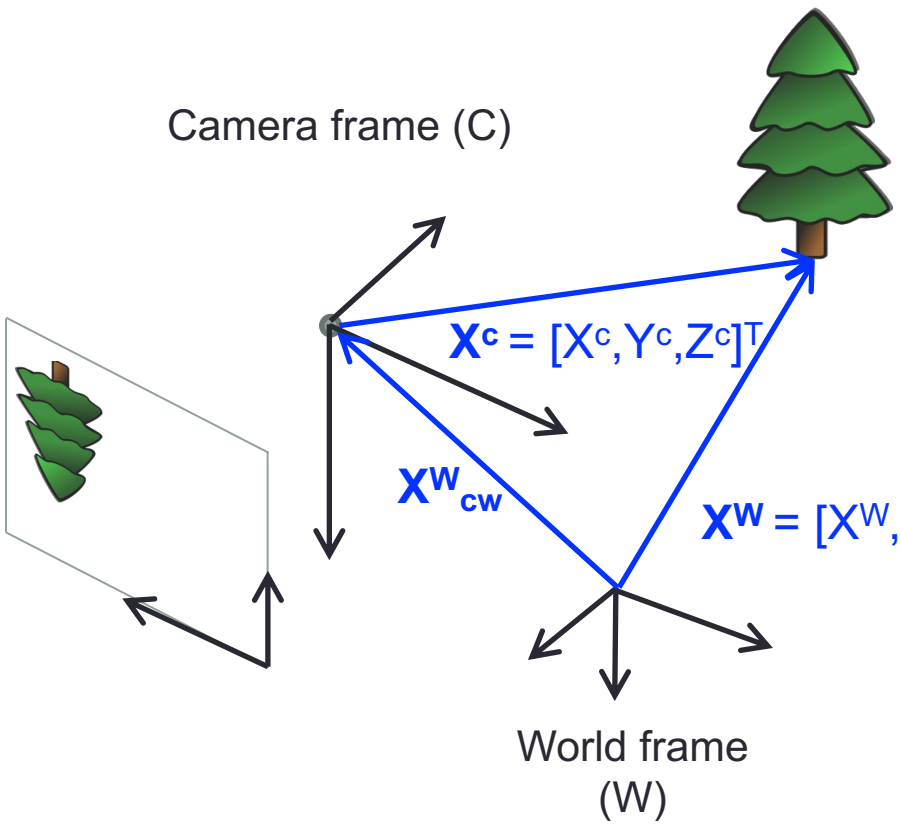
Camera Pinhole Projection Model



- Consider now the relationship between the object's position in camera coordinates and an arbitrarily defined world coordinate system W

$$\mathbf{X}^c = C^c_w(\mathbf{X}^W - \mathbf{X}^W_{cw})$$

Camera Pinhole Projection Model



- Consider now the relationship between the object's position in camera coordinates and an arbitrarily defined world coordinate system W

$$\mathbf{X}^c = C^c_w(\mathbf{X}^W - \mathbf{X}^W_{cw})$$

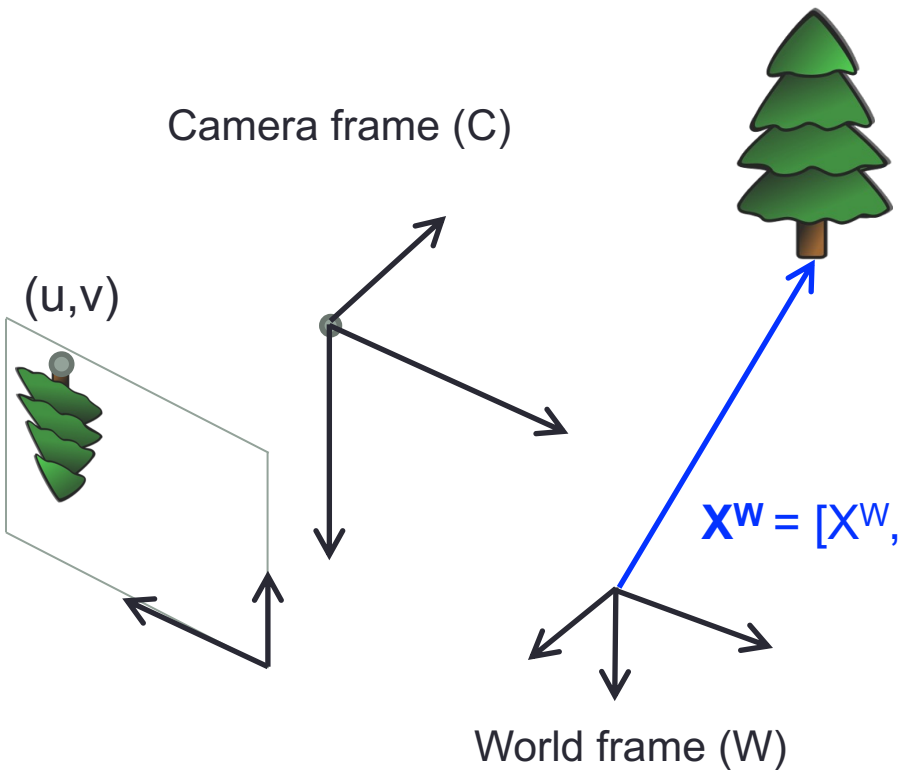
$$\mathbf{X}^c = R\mathbf{X}^W + \mathbf{t}$$

Where $\mathbf{X}^W_{cw}$ is the relative position of the camera w.r.t the world and $C_w^c = R$ is the direction cosine (or rotation) matrix which transforms coordinates from W to C

$$R = C^c_w$$

$$t = -C^c_w \mathbf{X}^W_{cw} = -R\mathbf{X}^W_{cw}$$

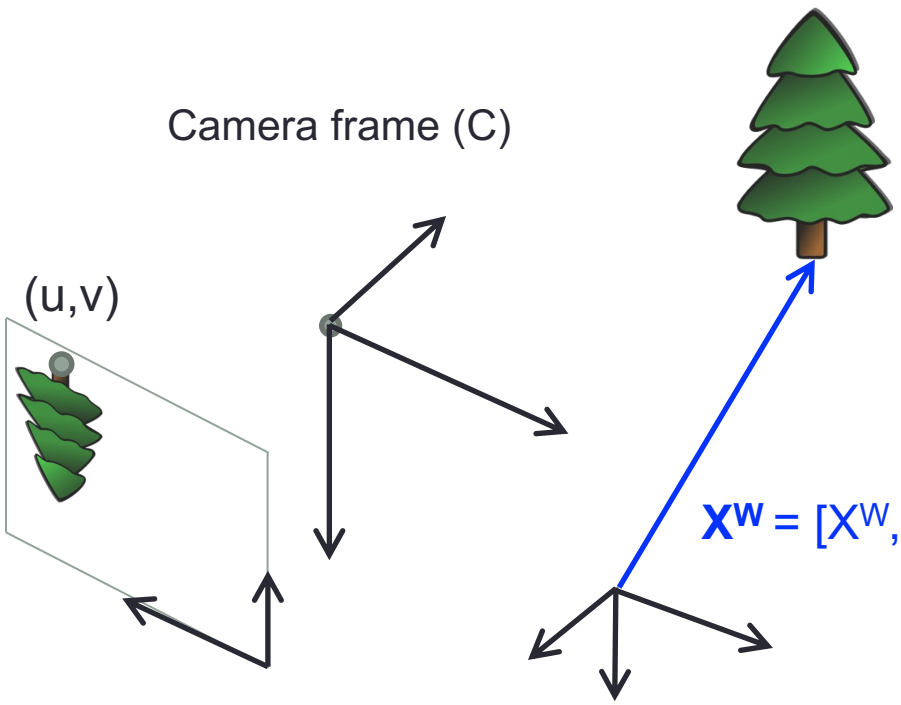
Camera Pinhole Projection Model



- The combined relationships can be represented in matrix form:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} Z^c = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix}$$

Camera Pinhole Projection Model

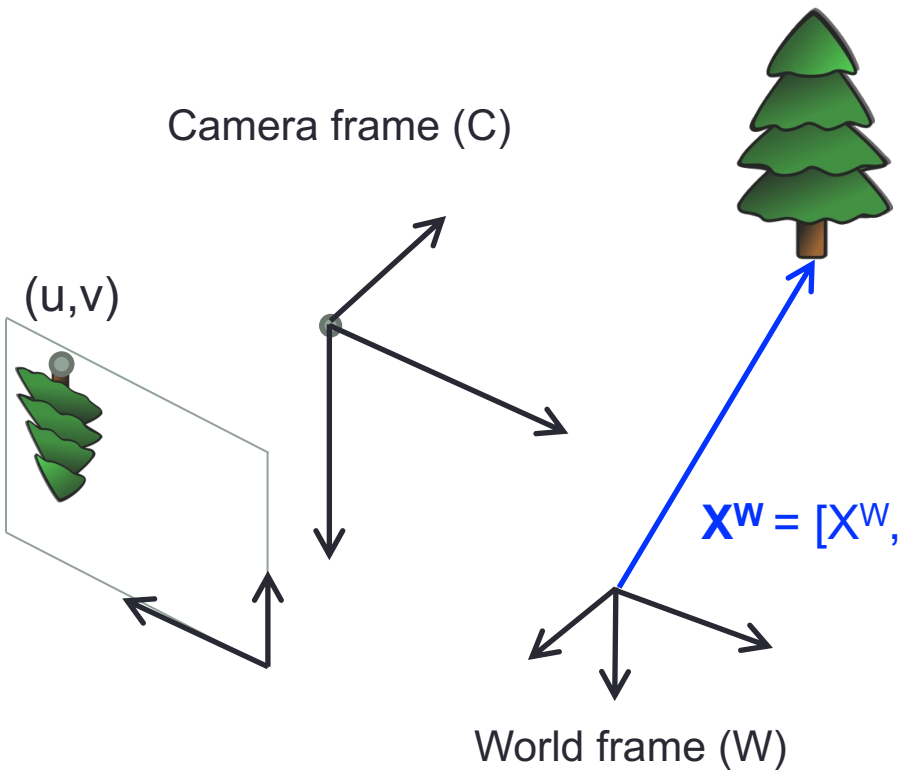


- The combined relationships can be represented in matrix form:

Camera "Intrinsic" parameters

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} Z^c = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix}$$

Camera Pinhole Projection Model

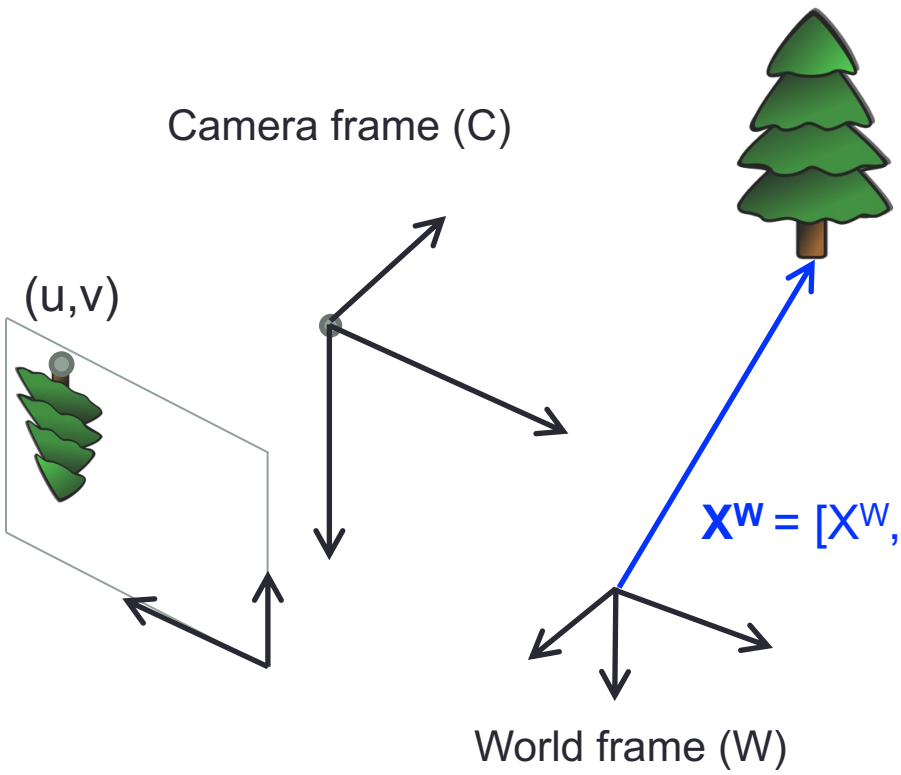


- The combined relationships can be represented in matrix form:

Camera "Extrinsic" parameters

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} Z^c = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix}$$

Camera Pinhole Projection Model



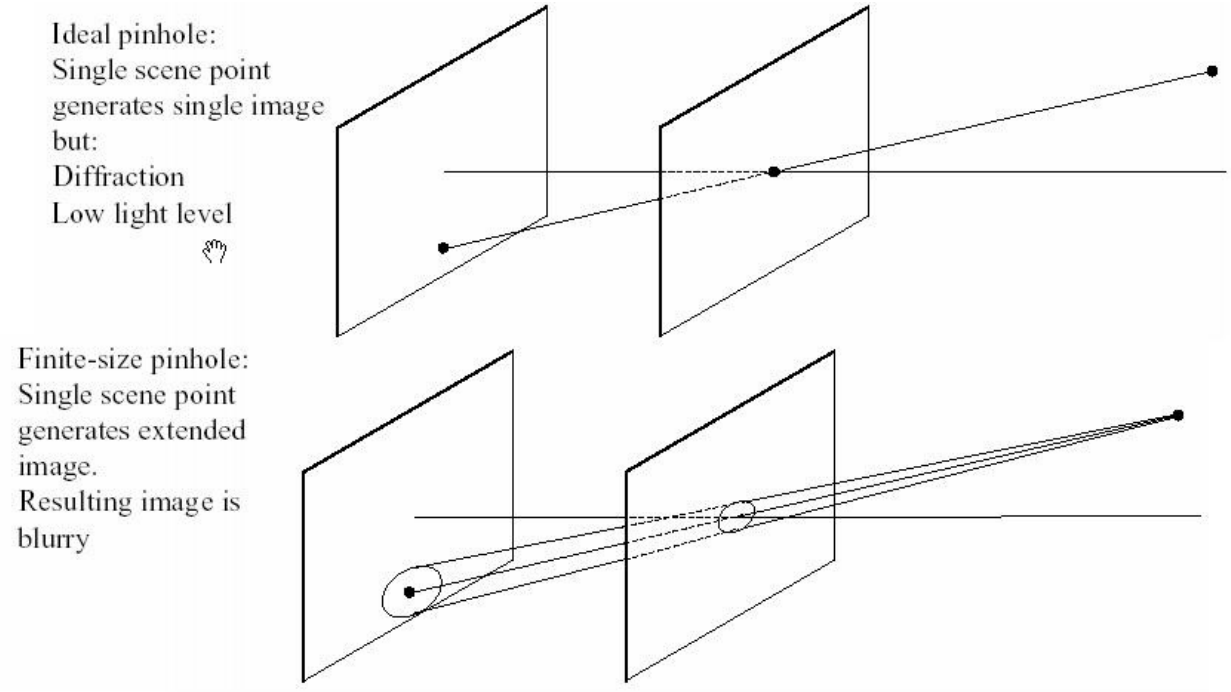
- The combined relationships can be represented in matrix form:

- **K**: Intrinsic Matrix
- **P**: Projection Matrix

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} Z^c = \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix} \qquad \mathbf{K} = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Limitations of the pinhole model

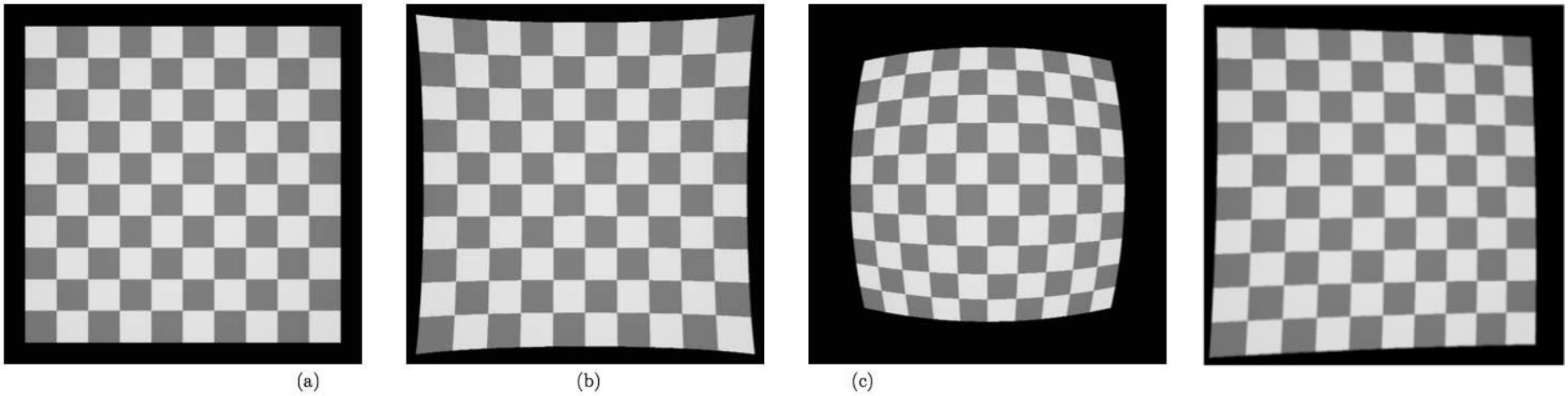
- Apertures have a finite size:
  - combined with lens results in “depth of field”
- Size is typically controllable for cameras:
  - Wide aperture: more light, better exposure (signal to noise) but reduced depth of field
  - Narrow aperture: less light, poorer exposure but larger depth of field



Lens Distortion

- Lenses are used on most real cameras to increase or decrease the focal length
- Most real lenses result in variations in viewing angle depending on the angle at which light enters the lens

No distortion | Pin cushion distortion | Barrel distortion | Tangential distortion



(a) (b) (c)

FIG. 2.3 – *Image de synthèse de la grille sans distorsions (a) avec distorsions radiales en barrilet (b) et en coussinet (c))*

Camera Pinhole Projection Model

- The predominant effects of radial and tangential distortion can be modelled using polynomials of the x, y (normalised image coordinates) and radial positions of objects w.r.t the optical center of the camera:

$$x = \frac{X_c}{Z_c} = \frac{u - u_0}{f_x} \qquad y = \frac{Y_c}{Z_c} = \frac{v - v_0}{f_y} \qquad r^2 = x^2 + y^2$$

Camera Pinhole Projection Model

- The predominant effects of radial and tangential distortion can be modelled using polynomials of the x, y (normalised image coordinates) and radial positions of objects w.r.t the optical center of the camera:

$$x = \frac{X_c}{Z_c} = \frac{u - u_0}{f_x} \qquad y = \frac{Y_c}{Z_c} = \frac{v - v_0}{f_y} \qquad r^2 = x^2 + y^2$$

- The distorted normalised image coordinates can approximated by:

$$x_d = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2k_4 xy + k_5(r^2 + 2x^2)$$
$$y_d = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + k_4(r^2 + 2y^2) + 2k_5 xy$$

Where k_1,k_2,k_3 are radial distortion coefficients and k_4,k_5 are tangential distortion coefficients

Camera Pinhole Projection Model

- The predominant effects of radial and tangential distortion can be modelled using polynomials of the x, y (normalised image coordinates) and radial positions of objects w.r.t the optical center of the camera:

$$x = \frac{X_c}{Z_c} = \frac{u - u_0}{f_x} \qquad y = \frac{Y_c}{Z_c} = \frac{v - v_0}{f_y} \qquad r^2 = x^2 + y^2$$

- The distorted normalised image coordinates can approximated by:

$$x_d = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2k_4 xy + k_5(r^2 + 2x^2)$$
$$y_d = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + k_4(r^2 + 2y^2) + 2k_5 xy$$

Where k_1, k_2, k_3 are radial distortion coefficients and k_4, k_5 are tangential distortion coefficients

Radial distortion parameters

Camera Pinhole Projection Model

- The predominant effects of radial and tangential distortion can be modelled using polynomials of the x, y (normalised image coordinates) and radial positions of objects w.r.t the optical center of the camera:

$$x = \frac{X_c}{Z_c} = \frac{u - u_0}{f_x} \qquad y = \frac{Y_c}{Z_c} = \frac{v - v_0}{f_y} \qquad r^2 = x^2 + y^2$$

- The distorted normalised image coordinates can approximated by:

$$x_d = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2k_4 xy + k_5(r^2 + 2x^2)$$
$$y_d = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + k_4(r^2 + 2y^2) + 2k_5 xy$$

Where k_1,k_2,k_3 are radial distortion coefficients and k_4,k_5 are tangential distortion coefficients

Tangential distortion parameters

Camera Pinhole Projection Model

- The predominant effects of radial and tangential distortion can be modelled using polynomials of the x, y (normalised image coordinates) and radial positions of objects w.r.t the optical center of the camera:

$$x = \frac{X_c}{Z_c} = \frac{u - u_0}{f_x} \qquad y = \frac{Y_c}{Z_c} = \frac{v - v_0}{f_y} \qquad r^2 = x^2 + y^2$$

- The distorted normalised image coordinates can approximated by:

$$x_d = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2k_4 xy + k_5(r^2 + 2x^2)$$
$$y_d = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + k_4(r^2 + 2y^2) + 2k_5 xy$$

Where k_1,k_2,k_3 are radial distortion coefficients and k_4,k_5 are tangential distortion coefficients

- The distorted pixels coordinates (u_d,v_d) are then:

$$u_d = f_x x_d + u_0 \qquad v_d = f_y y_d + v_0$$

Image Undistortion

- If the distortion parameters for a given camera/lens combination are known, then the effects of lens distortion can be removed from an image to arrive at an image which obeys the projective geometry of a pinhole model
- Typically performed using an interpolation:
  - For each pixel (u,v) in the corrected image, get the corresponding coordinates in the distorted image (u_d,v_d)
  - Render the pixel from this location into the new image, while interpolating with nearby pixel values

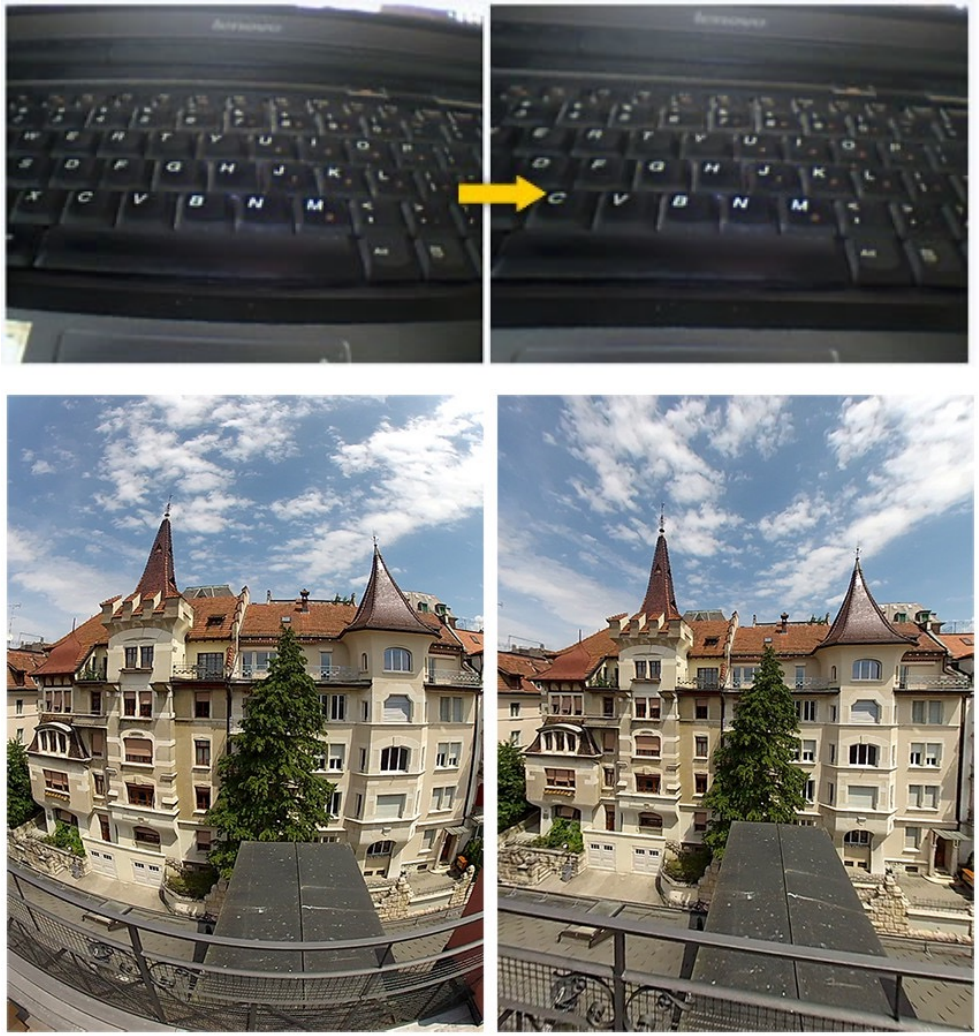


Image Credits: Ya Feng Shih, Utkarsh Sinha

Geometric Image Formation: Other Issues: Global vs. Rolling Shutters

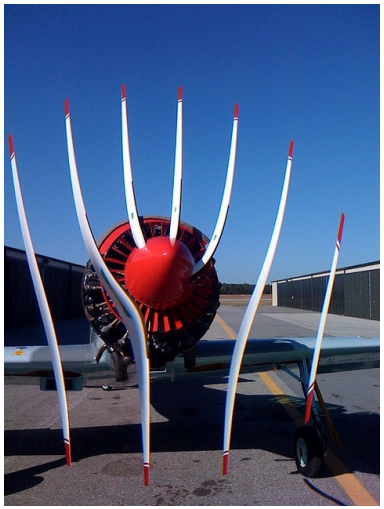
- So far we have considered image geometry captured in a snap shot in time
- For most ”normal” cameras, light is captured across all pixels at the same time: we refer to this system as a **global** shutter

- For real camera (particularly low-cost cameras) light across all pixels is not typically captured simultaneously:
  - Cheap cameras typically use an “electronic” shutter (as opposed to a mechanical shutter)
  - For high-resolution images and low-cost electronics, data can’t be read off the whole image array between exposures: rows of pixels are exposed to light and read-off sequentially

- The resulting imaging system is referred to as a **rolling** shutter system

Geometric Image Formation: Other Issues: Global vs. Rolling Shutters

- Global Shutter: Image captured simultaneously across image sensor
- Rolling Shutter: Image captured sequentially



- For rolling shutter cameras, scene motion or camera motion result in violations to projective geometric models

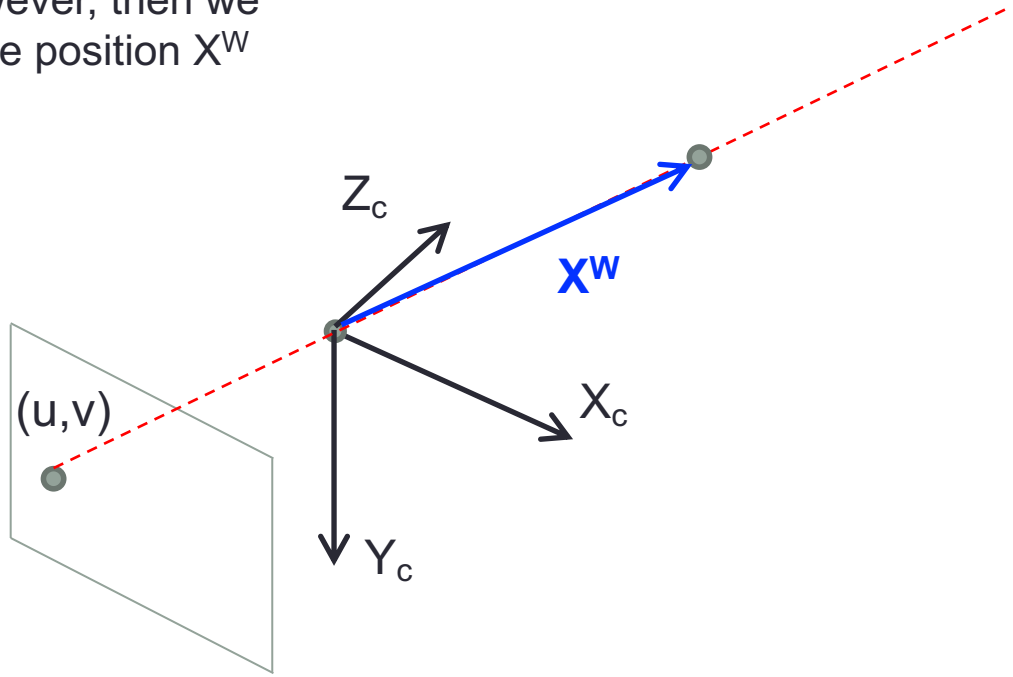
http://m43photo.blogspot.com.au/2010/07/rolling-shutter.html http://jasmcole.com/2014/10/12/rolling-shutters/

5 minute break

Stereo Vision

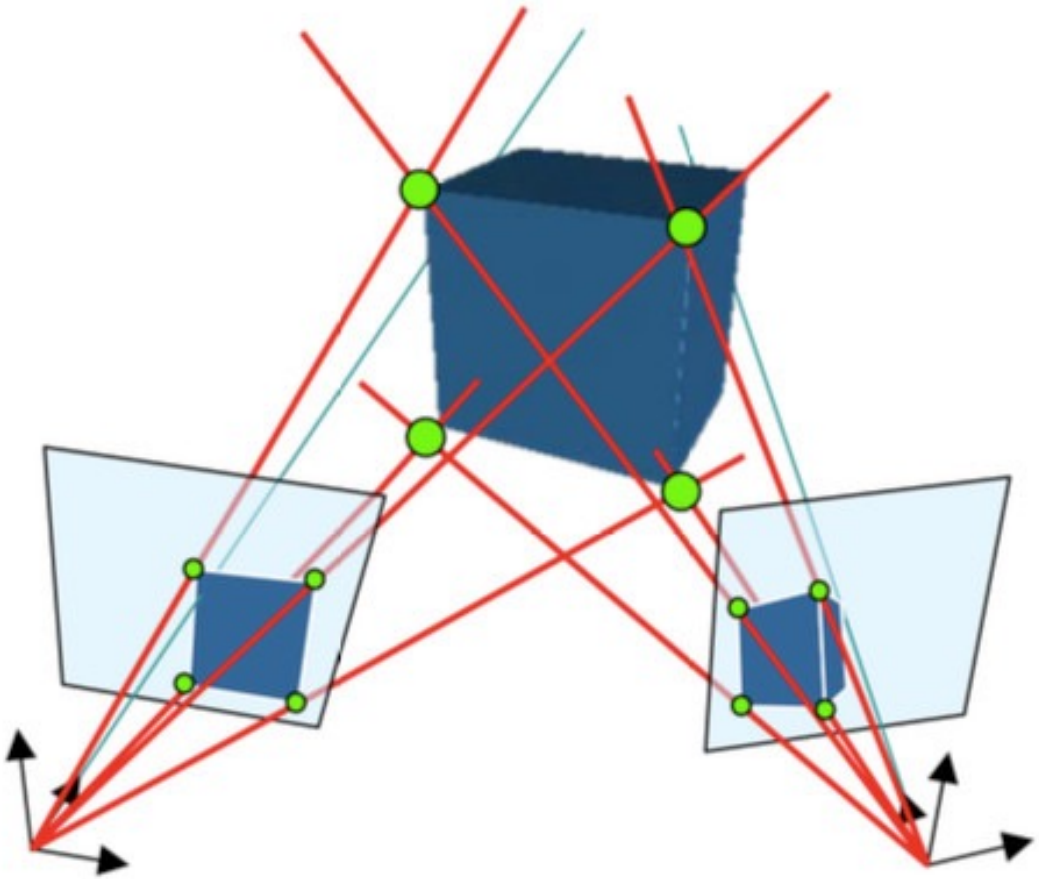
- The pixel location of an object/point in a single image provides insufficient information to compute the world coordinates (in 3D) by inverting our geometric image formation model:
  - The information provided by one view allows us to project a 3D line in space which the point must lie on
  - If we knew the depth of the object in the image Z^c however, then we could solve for the position X^W

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} Z^c = \begin{bmatrix} f_x & 0 & u_0 & 0 \\ 0 & f_y & v_0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X^W \\ Y^W \\ Z^W \\ 1 \end{bmatrix}$$



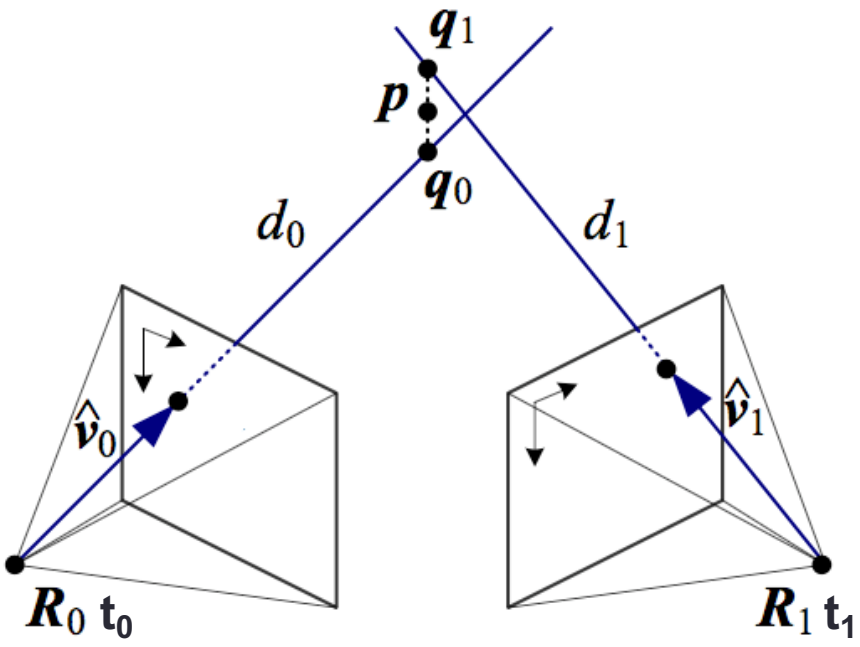
Stereo Vision

- Having information on the pixel location (and hence another line-of-sight) in a second image provides the ability to calculate the features 3D location using stereopsis



Stereo Vision: Triangulation

- Given a second image of the same point from a camera with a different position in space, we can calculate the position of the point in W coordinates by taking the intersection of the 3D lines from each image
- The lines don't always intersect exactly: so the closest point **p** between them is a best guess



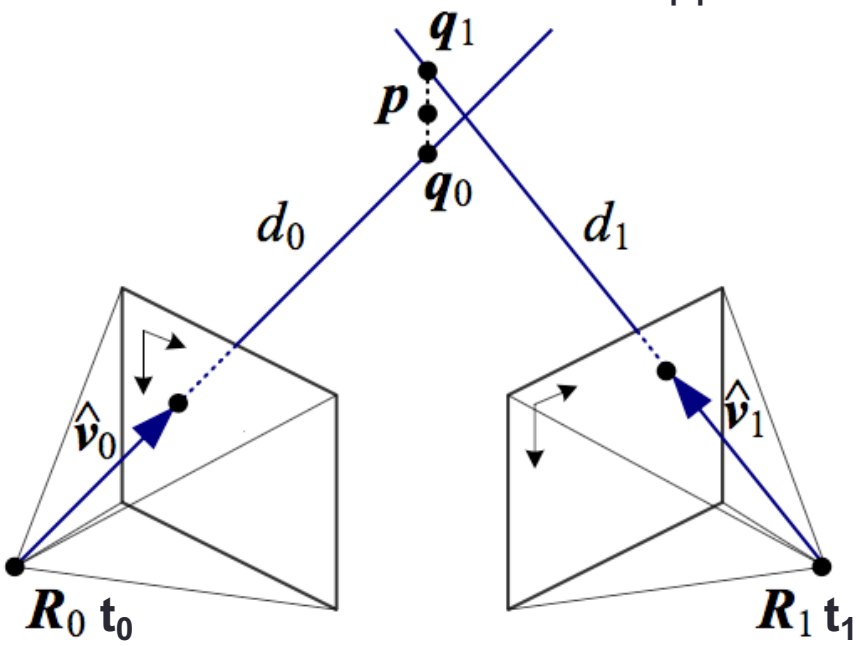
Stereo Vision: Triangulation

- The unit vector of each line-of-sight in world coordinates (v_0 and v_1) are calculated:

$$\mathbf{v}_j^w = \mathbf{R}^T[\frac{u_j - u_0}{r f_x}, \frac{v_j - v_0}{r f_y}, \frac{1}{r}]^T \qquad r = 1 + (\frac{u_j - \hat{u}_0}{\hat{f}_x})^2 + (\frac{v_j - \hat{v}_0}{\hat{f}_y})^2$$

for j = 0,1

- The point p is equal to the averages of the closest approach between each line:



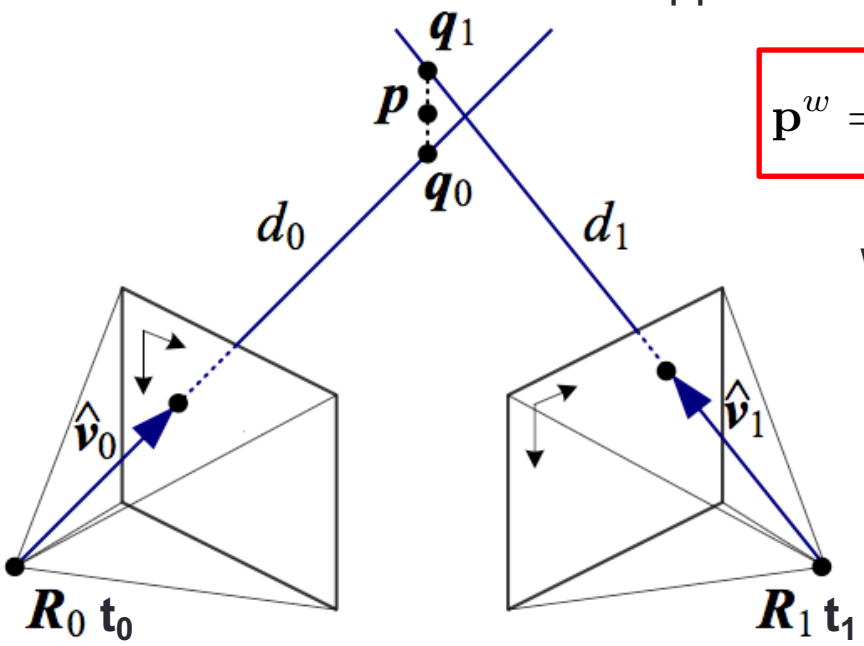
Stereo Vision: Triangulation

- The unit vector of each line-of-sight in world coordinates (v_0 and v_1) are calculated:

$$\mathbf{v}_j^w = \mathbf{R}^T[\frac{u_j - u_0}{r f_x}, \frac{v_j - v_0}{r f_y}, \frac{1}{r}]^T \qquad r = 1 + (\frac{u_j - \hat{u}_0}{\hat{f}_x})^2 + (\frac{v_j - \hat{v}_0}{\hat{f}_y})^2$$

for j = 0,1

- The point p is equal to the averages of the closest approach between each line:



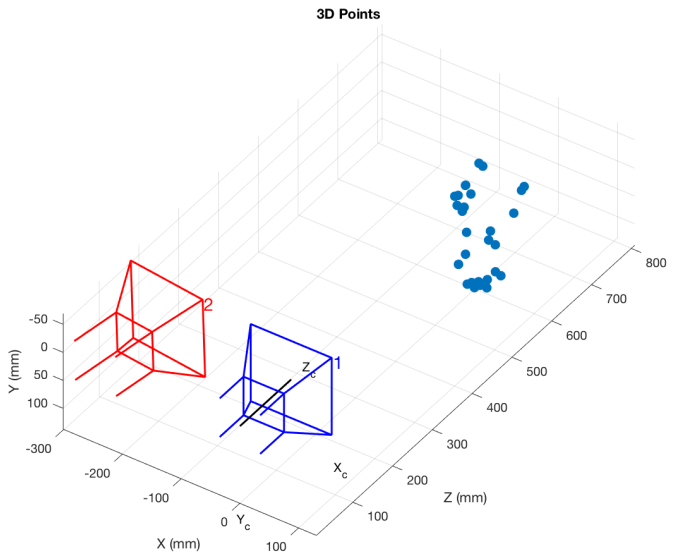
$$\mathbf{p}^w = \frac{1}{2}(-\mathbf{R}_0^T\mathbf{t}_0 + -\mathbf{R}_1^T\mathbf{t}_1 + d_0.\mathbf{v}_0^w + d_1.\mathbf{v}_1^w)$$

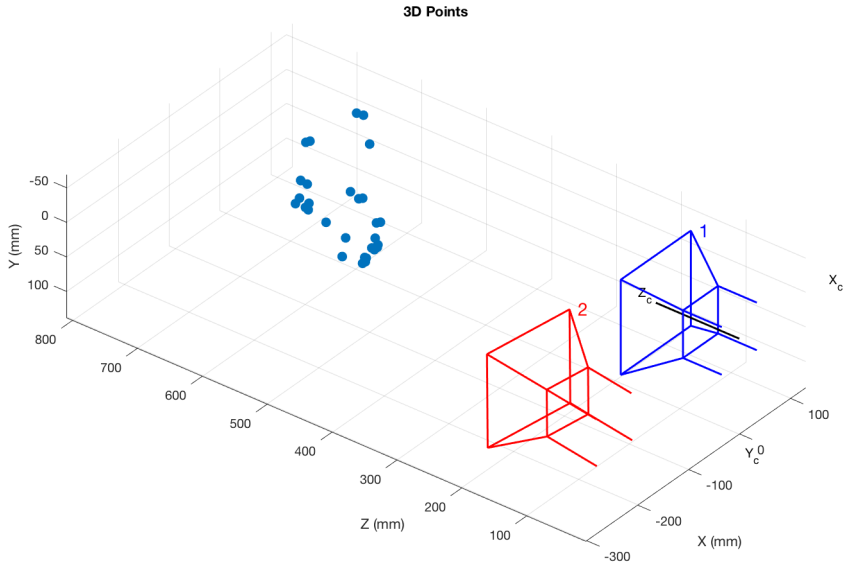
Where:

$$d_0 = \frac{((\mathbf{R}_0^T\mathbf{t}_0 - \mathbf{R}_1^T\mathbf{t}_1) \times \mathbf{v}_1^w) \cdot (\mathbf{v}_0^w \times \mathbf{v}_1^w)}{|\mathbf{v}_0^w \times \mathbf{v}_1^w|^2}$$

$$d_1 = \frac{((\mathbf{R}_1^T\mathbf{t}_1 - \mathbf{R}_0^T\mathbf{t}_0) \times \mathbf{v}_0^w) \cdot (\mathbf{v}_1^w \times \mathbf{v}_0^w)}{|\mathbf{v}_1^w \times \mathbf{v}_0^w|^2}$$

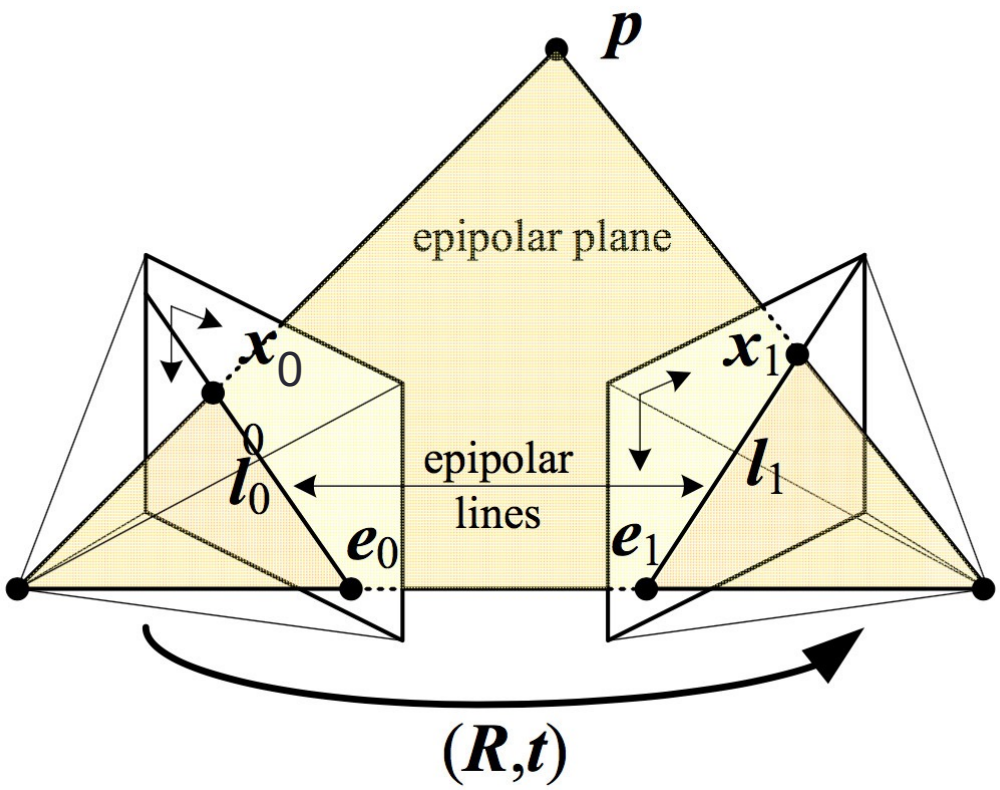
Stereo Vision: Triangulation





Epipolar Geometry

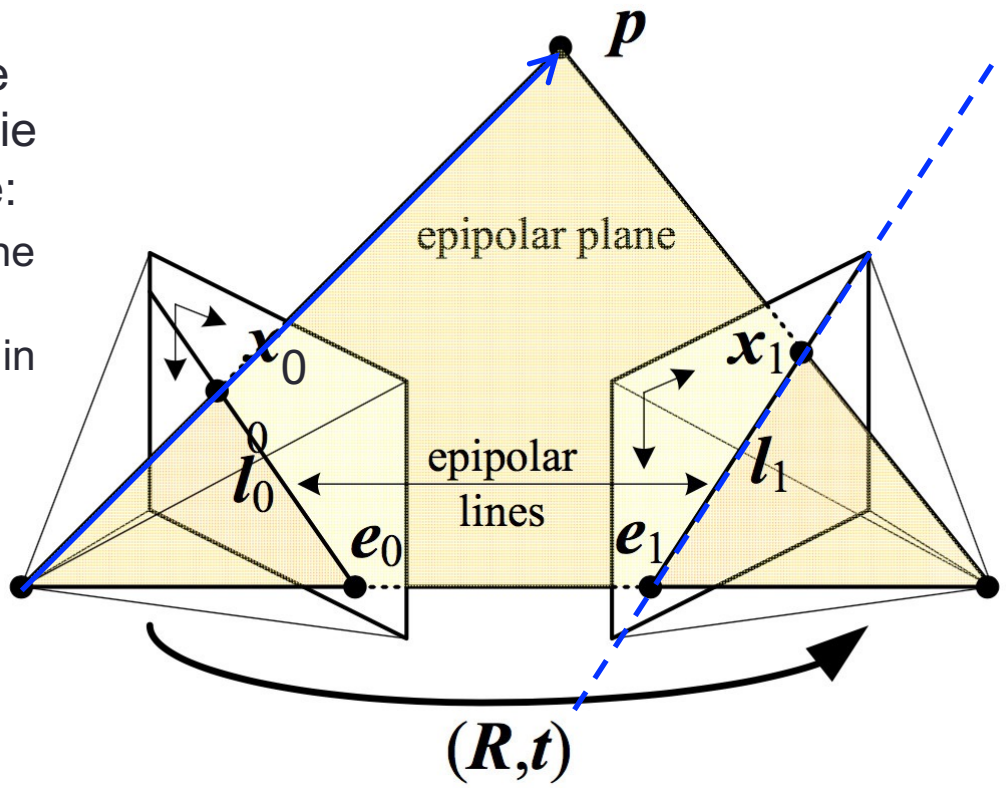
- Consider two views of a point **p**, $\mathbf{x_0} = [x,y,1]^T$ and $\mathbf{x_1} = [x_1,y_1,1]^T$ in normalised image coordinates
- We will define world coordinates to be centered on the first camera such that $\mathbf{R_0} = \mathbf{I}$ and $\mathbf{t_0} = 0$: hence **R** and **t** now represent the parameters of the second camera (1) w.r.t the first (0)



Epipolar Geometry

- Consider two views of a point **p**, $\mathbf{x_0} = [x,y,1]^T$ and $\mathbf{x_1} = [x_1,y_1,1]^T$ in normalised image coordinates
- We will define world coordinates to be centered on the first camera such that $\mathbf{R_0} = \mathbf{I}$ and $\mathbf{t_0} = 0$: hence **R** and **t** now represent the parameters of the second camera (1) w.r.t the first (0)

- For a fixed image location $\mathbf{x_0}$ in the first image, the location of p must lie along a line $\mathbf{l_1}$ in the second image:
  - This line is known as the epipolar line and it runs through the epipole $\mathbf{e_1}$ (position of the first camera's origin in the second camera image plane

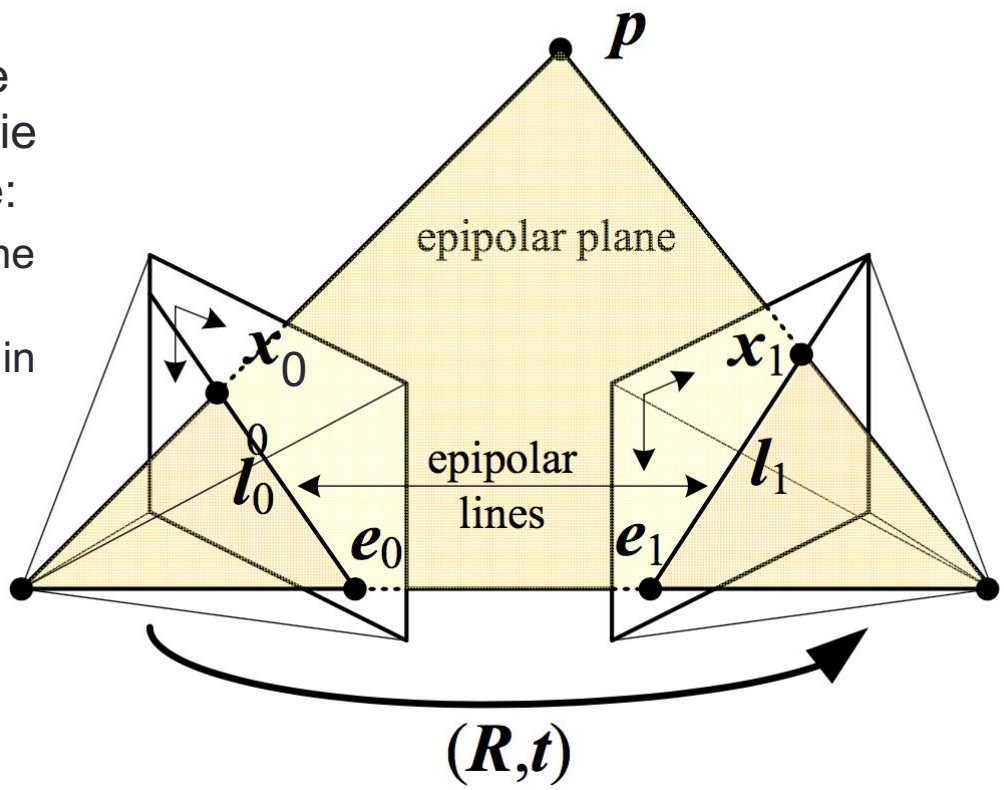


Epipolar Geometry

- Consider two views of a point **p**, $\mathbf{x_0} = [x,y,1]^T$ and $\mathbf{x_1} = [x_1,y_1,1]^T$ in normalised image coordinates
- We will define world coordinates to be centered on the first camera such that $\mathbf{R_0} = \mathbf{I}$ and $\mathbf{t_0} = 0$: hence **R** and **t** now represent the parameters of the second camera (1) w.r.t the first (0)

- For a fixed image location $\mathbf{x_0}$ in the first image, the location of p must lie along a line $\mathbf{l_1}$ in the second image:
  - This line is known as the epipolar line and it runs through the epipole $\mathbf{e_1}$ (position of the first camera's origin in the second camera image plane

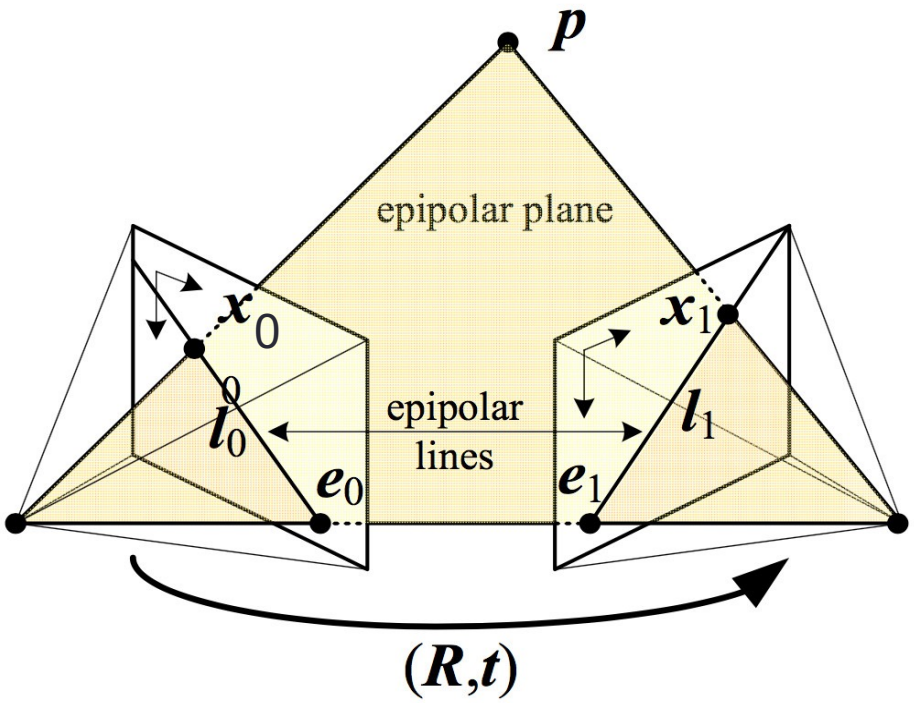
- A similar line is formed in the first image: these lines intersect along a plane that joins the camera centers and the point known as the **epipolar constraint**



Epipolar Geometry

- The projections of the point in each image can be related via:

$$d_1\mathbf{x}_1 = \mathbf{p}_1 = \mathbf{R}\mathbf{p}_0 + \mathbf{t} = \mathbf{R}(d_0\mathbf{x}_0) + \mathbf{t}$$



Epipolar Geometry

- The projections of the point in each image can be related via:

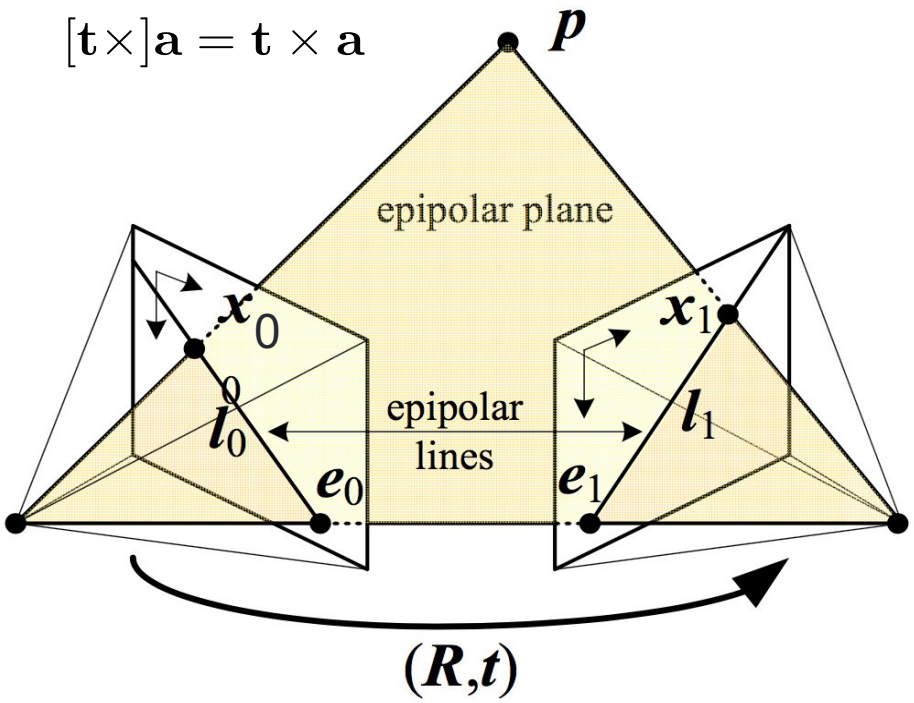
$$d_1\mathbf{x}_1 = \mathbf{p}_1 = \mathbf{R}\mathbf{p}_0 + \mathbf{t} = \mathbf{R}(d_0\mathbf{x}_0) + \mathbf{t}$$

- Cross producting on both sides with **t** gives:

$$d_1(\mathbf{t} \times \mathbf{x}_1) = d_0(\mathbf{t} \times \mathbf{R}\mathbf{x}_0) + \mathbf{t} \times \mathbf{t}$$

$$d_1[\mathbf{t}\times]\mathbf{x}_1 = d_0[\mathbf{t}\times]\mathbf{R}\mathbf{x}_0$$

$$[\mathbf{t}\times]\mathbf{a} = \mathbf{t} \times \mathbf{a}$$



Epipolar Geometry

- The projections of the point in each image can be related via:

$$d_1\mathbf{x}_1 = \mathbf{p}_1 = \mathbf{R}\mathbf{p}_0 + \mathbf{t} = \mathbf{R}(d_0\mathbf{x}_0) + \mathbf{t}$$

- Cross producting on both sides with **t** gives:

$$d_1(\mathbf{t} \times \mathbf{x}_1) = d_0(\mathbf{t} \times \mathbf{R}\mathbf{x}_0) + \mathbf{t} \times \mathbf{t}$$

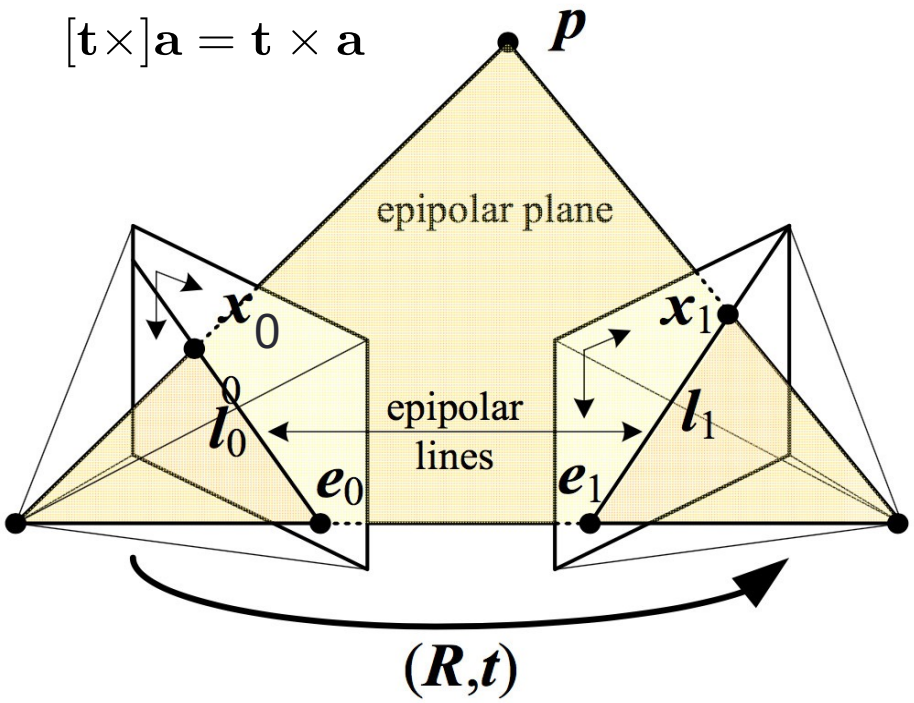
$$d_1[\mathbf{t}\times]\mathbf{x}_1 = d_0[\mathbf{t}\times]\mathbf{R}\mathbf{x}_0$$

- Left multiplying with $\mathbf{x}_1$ gives:

$$d_1\mathbf{x}_1^T[\mathbf{t}\times]\mathbf{x}_1 = d_0\mathbf{x}_1^T[\mathbf{t}\times]\mathbf{R}\mathbf{x}_0$$

$$d_1\mathbf{x}_1^T[\mathbf{t}\times]\mathbf{x}_1 = 0 \qquad \mathbf{x}_1 \cdot (\mathbf{t} \times \mathbf{x}_1) = 0$$

$$[\mathbf{t}\times]\mathbf{a} = \mathbf{t} \times \mathbf{a}$$



Epipolar Geometry

- The projections of the point in each image can be related via:

$$d_1\mathbf{x}_1 = \mathbf{p}_1 = \mathbf{R}\mathbf{p}_0 + \mathbf{t} = \mathbf{R}(d_0\mathbf{x}_0) + \mathbf{t}$$

- Cross producting on both sides with **t** gives:

$$d_1(\mathbf{t} \times \mathbf{x}_1) = d_0(\mathbf{t} \times \mathbf{R}\mathbf{x}_0) + \mathbf{t} \times \mathbf{t}$$

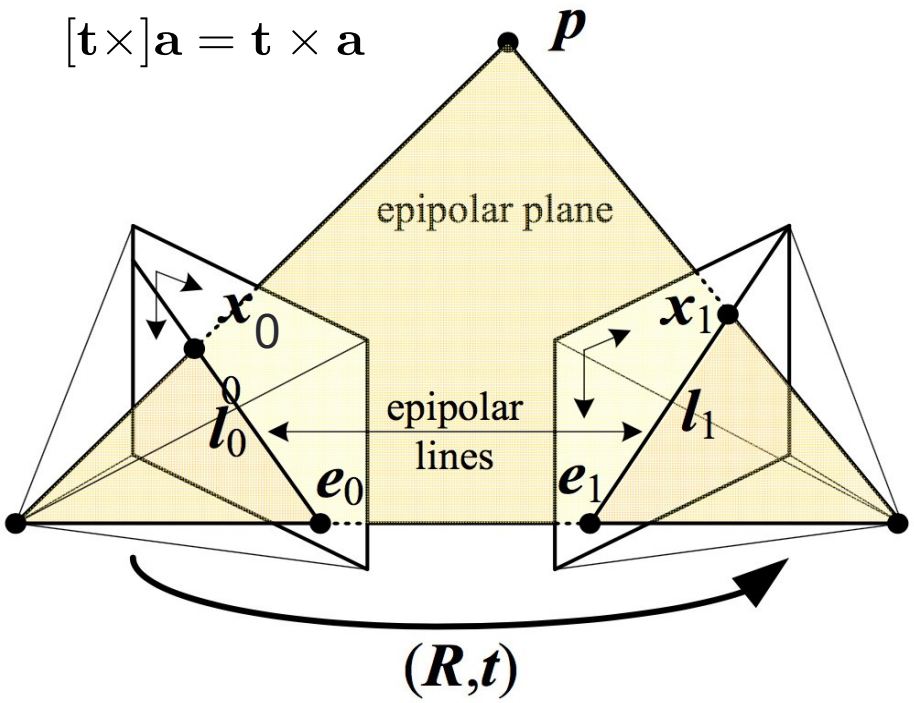
$$d_1[\mathbf{t}\times]\mathbf{x}_1 = d_0[\mathbf{t}\times]\mathbf{R}\mathbf{x}_0$$

$$[\mathbf{t}\times]\mathbf{a} = \mathbf{t} \times \mathbf{a}$$

- Left multiplying with $\mathbf{x}_1$ gives:

$$d_1\mathbf{x}_1^T[\mathbf{t}\times]\mathbf{x}_1 = d_0\mathbf{x}_1^T[\mathbf{t}\times]\mathbf{R}\mathbf{x}_0$$

$$d_1\mathbf{x}_1^T[\mathbf{t}\times]\mathbf{x}_1 = 0 \qquad \mathbf{x}_1 \cdot (\mathbf{t} \times \mathbf{x}_1) = 0$$



$$\mathbf{x}_1^T\mathbf{E}\mathbf{x}_0 = 0$$

$$\mathbf{E} = [\mathbf{t}\times]\mathbf{R}$$

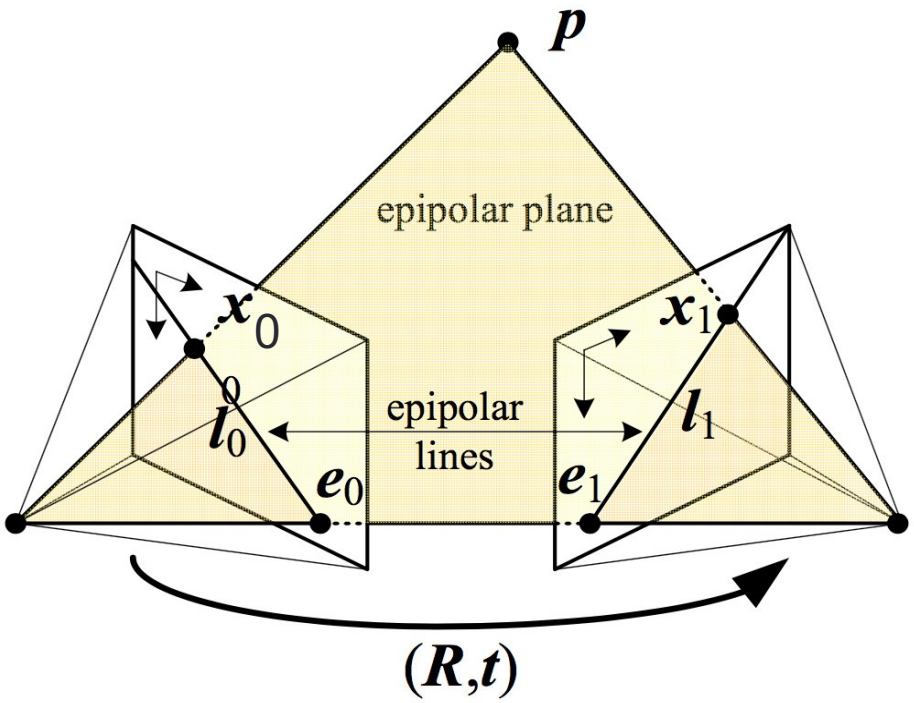
Where **E** is known as the **essential matrix**

Epipolar Geometry

- The essential matrix **E** relates all pairs of normalised image coordinates in each image

$$\mathbf{x}_1^T \mathbf{E} \mathbf{x}_0 = 0$$

$$\mathbf{E} = [\mathbf{t}\times]\mathbf{R}$$



Epipolar Geometry

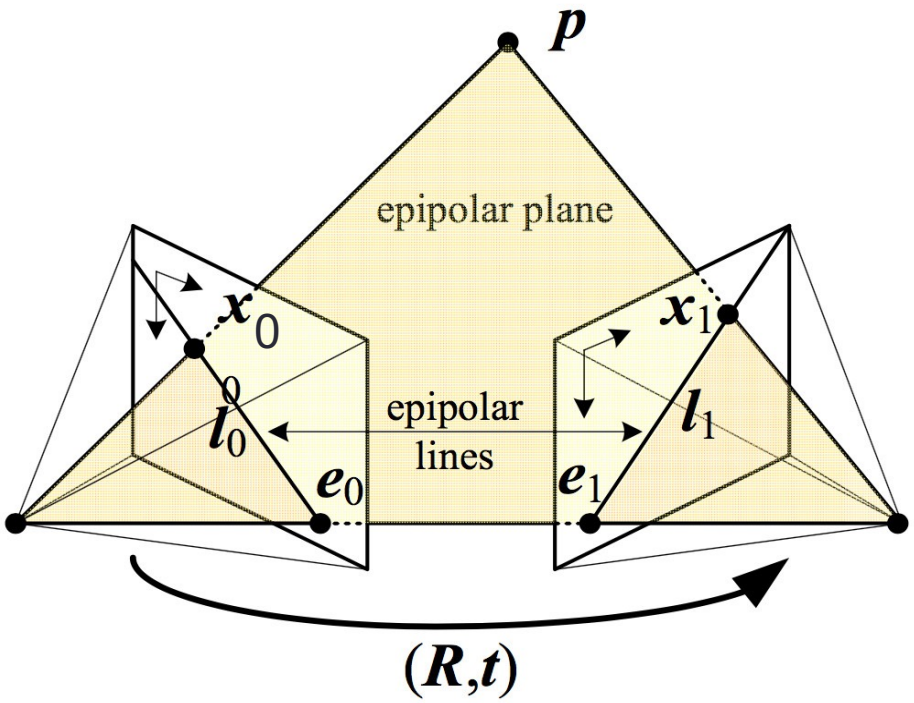
- The essential matrix **E** relates all pairs of normalised image coordinates in each image

$$\mathbf{x}_1^T \mathbf{E} \mathbf{x}_0 = 0$$

$$\mathbf{E} = [\mathbf{t}\times]\mathbf{R}$$

- We can also formulate a similar relationship by using un-normalised (pixel coordinates) which holds even in cases for which we do not know the intrinsic parameters of each camera:

$$[u, v, 1]^T = \mathbf{K}\mathbf{x}$$



Epipolar Geometry

- The essential matrix **E** relates all pairs of normalised image coordinates in each image

$$\mathbf{x}_1^T \mathbf{E} \mathbf{x}_0 = 0$$

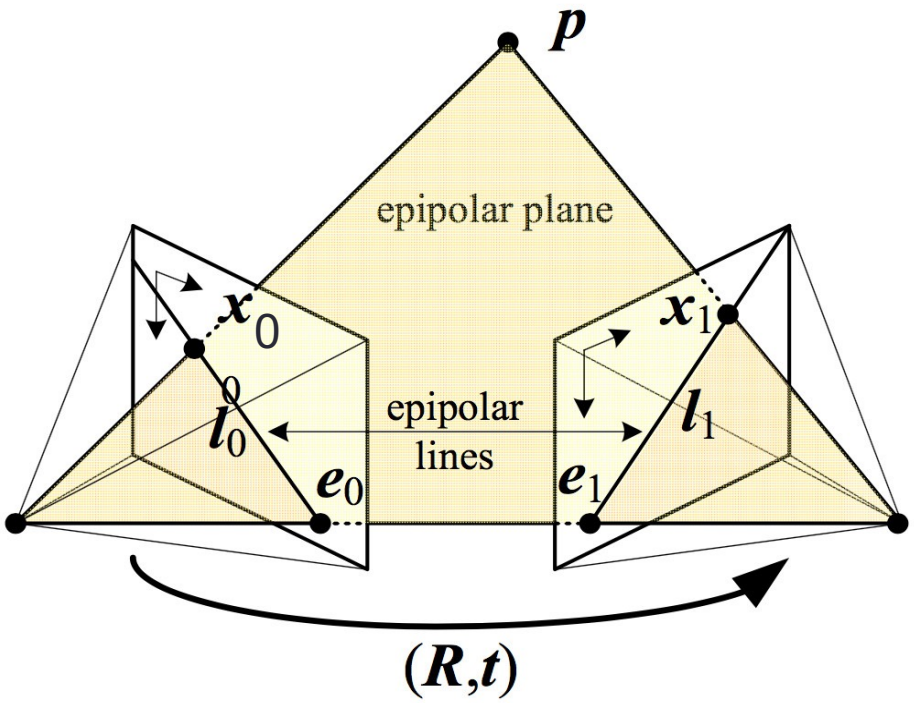
$$\mathbf{E} = [\mathbf{t}\times]\mathbf{R}$$

- We can also formulate a similar relationship by using un-normalised (pixel coordinates) which holds even in cases for which we do not know the intrinsic parameters of each camera:

$$[u, v, 1]^T = \mathbf{K}\mathbf{x}$$

$$[u, v, 1]_1 \mathbf{F} [u, v, 1]_0^T = 0$$

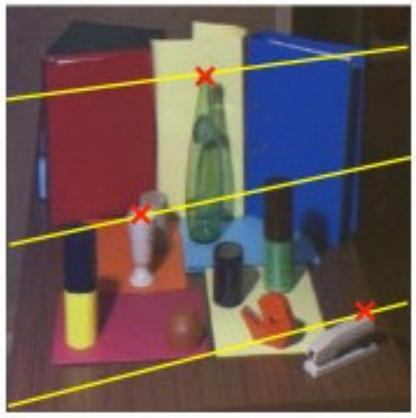
$$\mathbf{F} = \mathbf{K}_1^{-T} \mathbf{E} \mathbf{K}_0^{-1}$$



Where **F** is known as the **fundamental matrix**

Using epipolar geometry to assist in finding image correspondences

- Knowledge of the epipolar geometry between two images simplifies the search for correspondences from 2D to a 1D search along epipolar lines

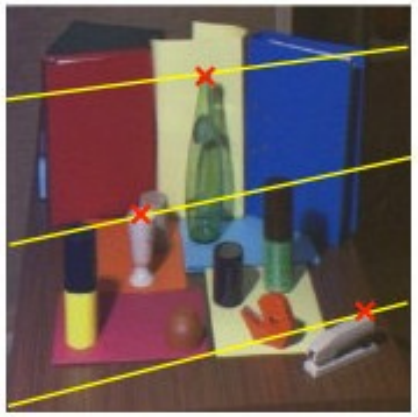


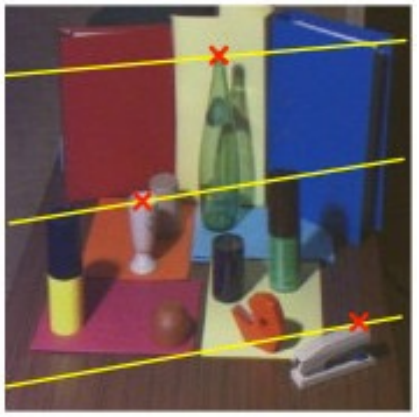
Using epipolar geometry to assist in finding image correspondences

- Knowledge of the epipolar geometry between two images simplifies the search for correspondences from 2D to a 1D search along epipolar lines

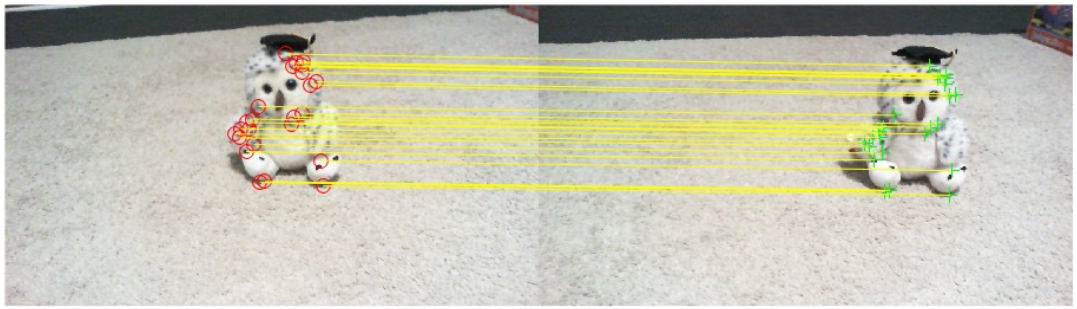
- If the parameters of a stereo image pair is known the fundamental matrix **F** can be calculated and used to determine the inliers (based on geometry) in a candidate set of matches (for example from feature descriptor matching) by finding all points that are less than a small distance **δ** from the constraint:

$$|[u, v, 1]_1 \mathbf{F} [u, v, 1]_0^T| < \delta$$





Original SURF Correspondences



Inliers based on epipolar geometry

Using epipolar geometry to assist in finding image correspondences

- A similar approach can also be employed to determine inliers when no previous knowledge of the stereo parameters is available
- It is possible to estimate the value of the fundamental matrix F given nine corresponding sets of points by solving for the homogenous equations formed by the epipolar constraint:

$$u_{j0}v_{j1}f_{10} + v_{j0}v_{j1}f_{11} + v_{j1}f_{12} + u_{j0}u_{j1}f_{00} + v_{j0}u_{j1}f_{01} + u_{j1}f_{02} + u_{j0}f_{20} + v_{j0}f_{21} + f22 = 0$$

Where j = 1,2, … 9

Using epipolar geometry to assist in finding image correspondences

- A similar approach can also be employed to determine inliers when no previous knowledge of the stereo parameters is available
- It is possible to estimate the value of the fundamental matrix F given nine corresponding sets of points by solving for the homogenous equations formed by the epipolar constraint:

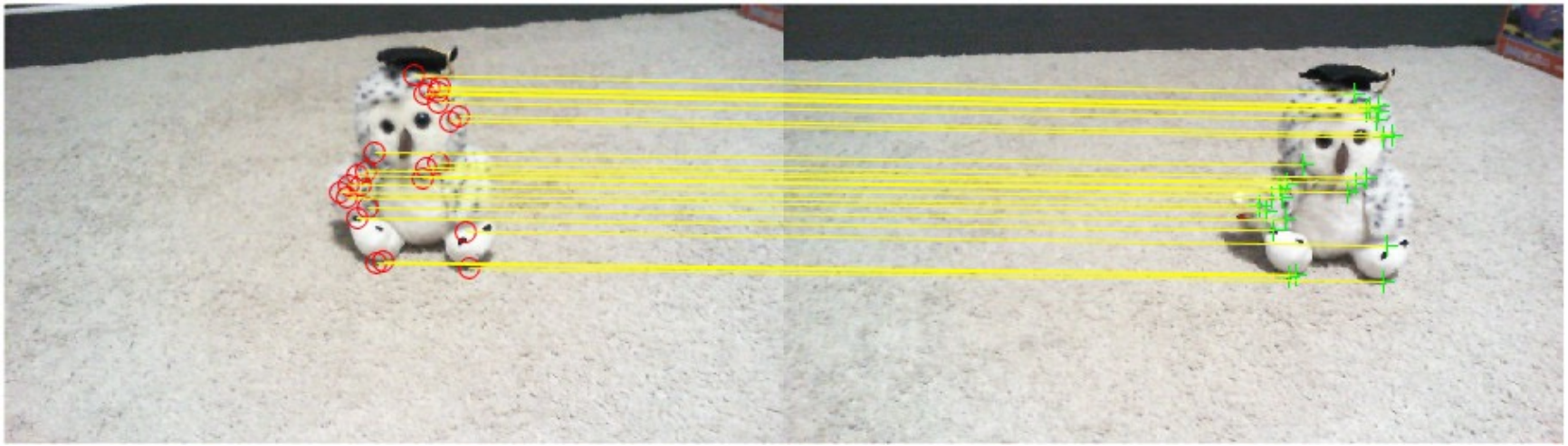
$$u_{j0}v_{j1}f_{10} + v_{j0}v_{j1}f_{11} + v_{j1}f_{12} + u_{j0}u_{j1}f_{00} + v_{j0}u_{j1}f_{01} + u_{j1}f_{02} + u_{j0}f_{20} + v_{j0}f_{21} + f22 = 0$$

Where j = 1,2, … 9

- Given a candidate set of matches (more than nine) for which some matches are expected to contain outliers, the fundamental matrix and candidate inliers can be estimated using a RANSAC model fitting algorithm:
  - Sample a random set of nine points and use these to estimate F
  - Check for the number of remaining correspondences that meet the epipolar constraint for this estimated F
  - Repeat a statistically-appropriate number of times while recording F (and inlier set) which was largest

Using epipolar geometry to assist in finding image correspondences

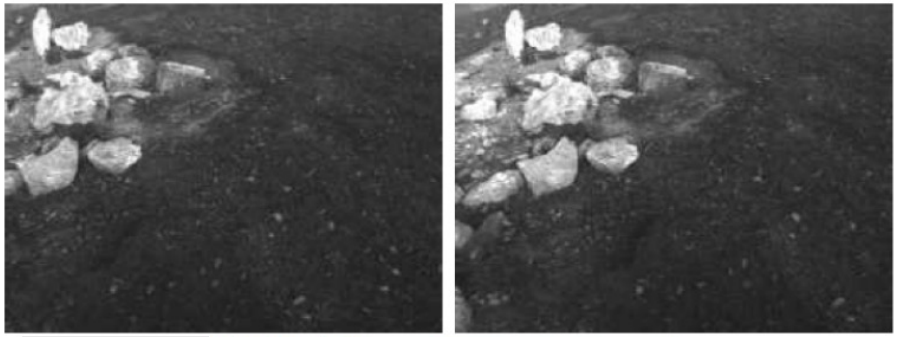
Original SURF Correspondences

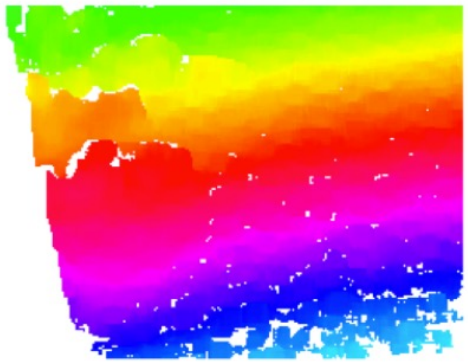


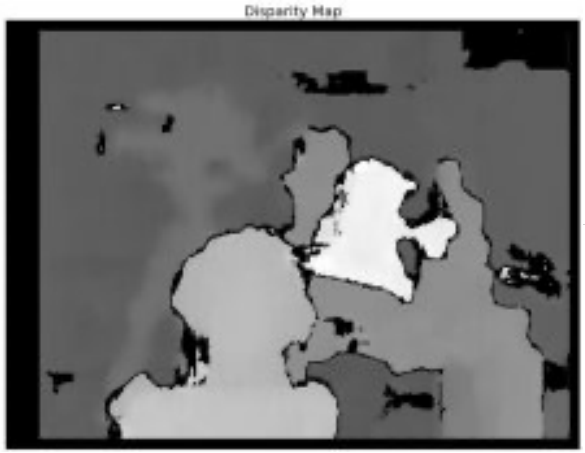
Inliers based on RANSAC estimation of fundamental matrix

Dense Stereo

- As opposed to using sparse points, dense stereo techniques are those that attempt to compute the 3D depth for each and every pixel in one of the two camera images
- Dense stereo relies on being able to find correspondences between images across the whole image space:
  - If searching in 2D the problem quickly becomes intractable for high-resolution images

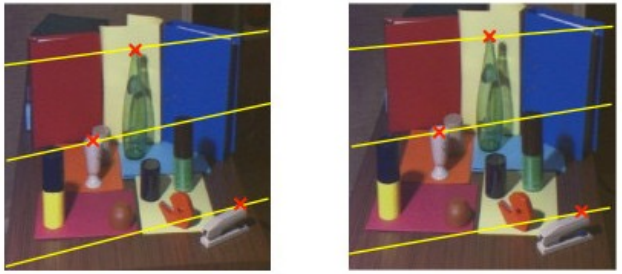




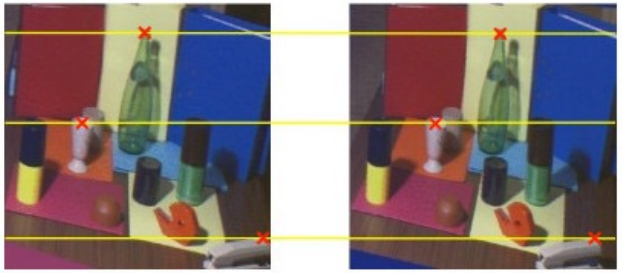


Stereo Rectification

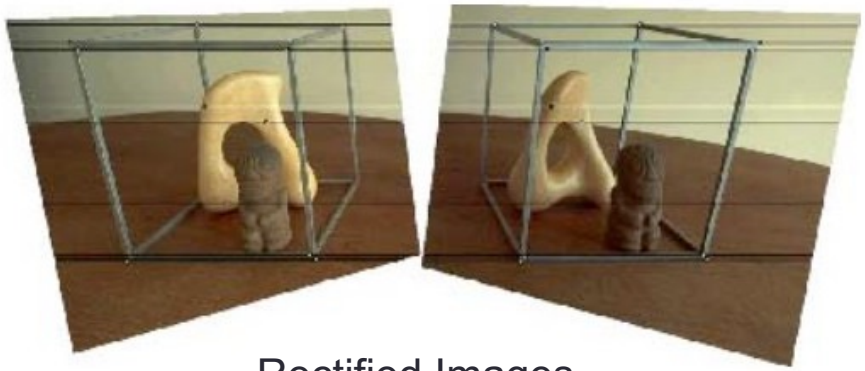
- Stereo rectification is the process of applying projective transformation to each image such that epipolar lines run horizontally in the image
  - Searching occurs therefore in blocks in a horizontal direction across the image, which is more easily implemented than searching along a non-horizontal line



- The essential or fundamental matrix can be used to designate these projective transformations



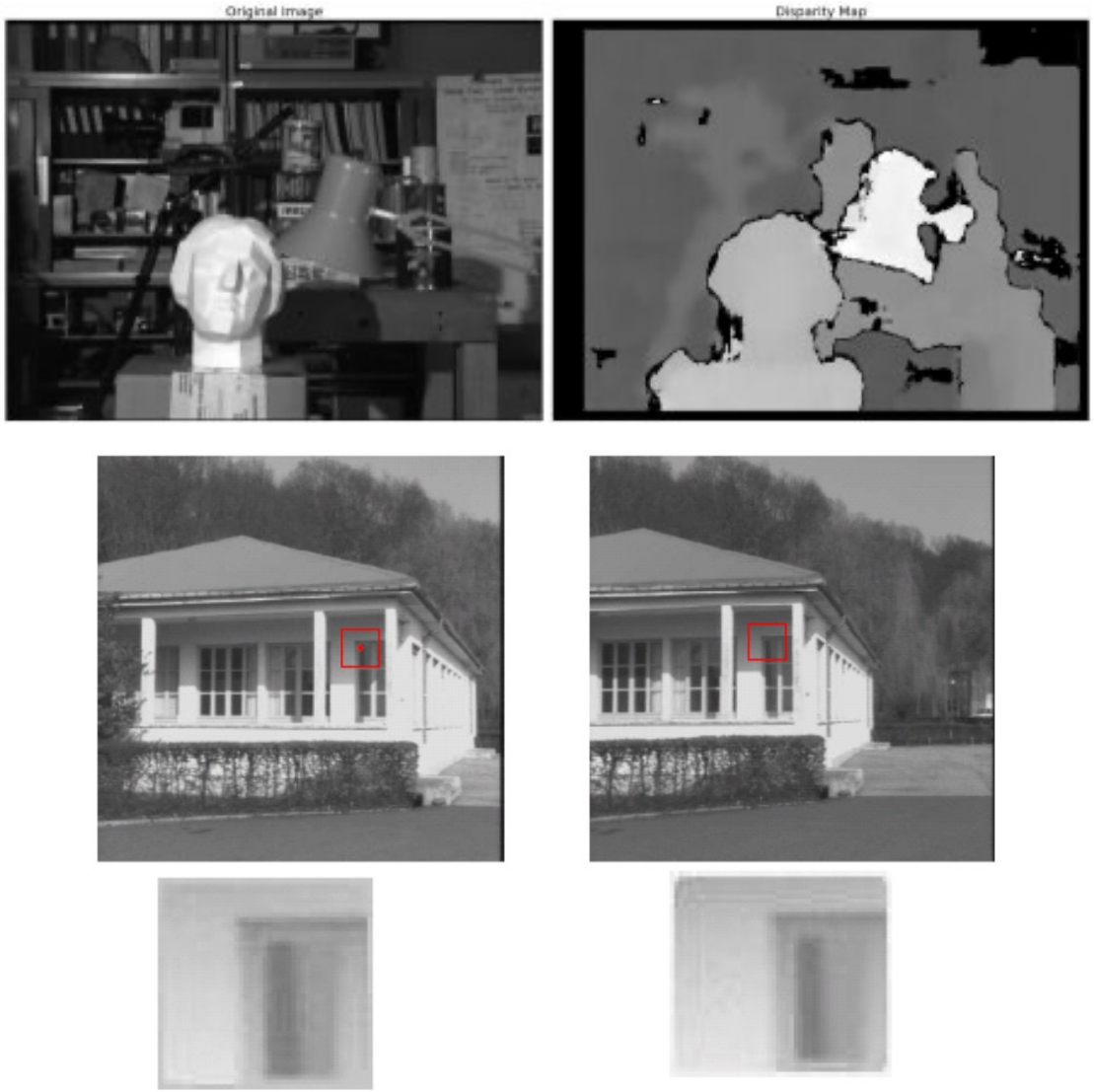
Original Images



Rectified Images

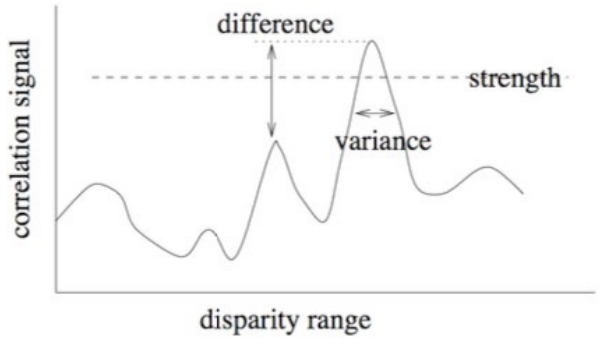
Dense Stereo Triangulation

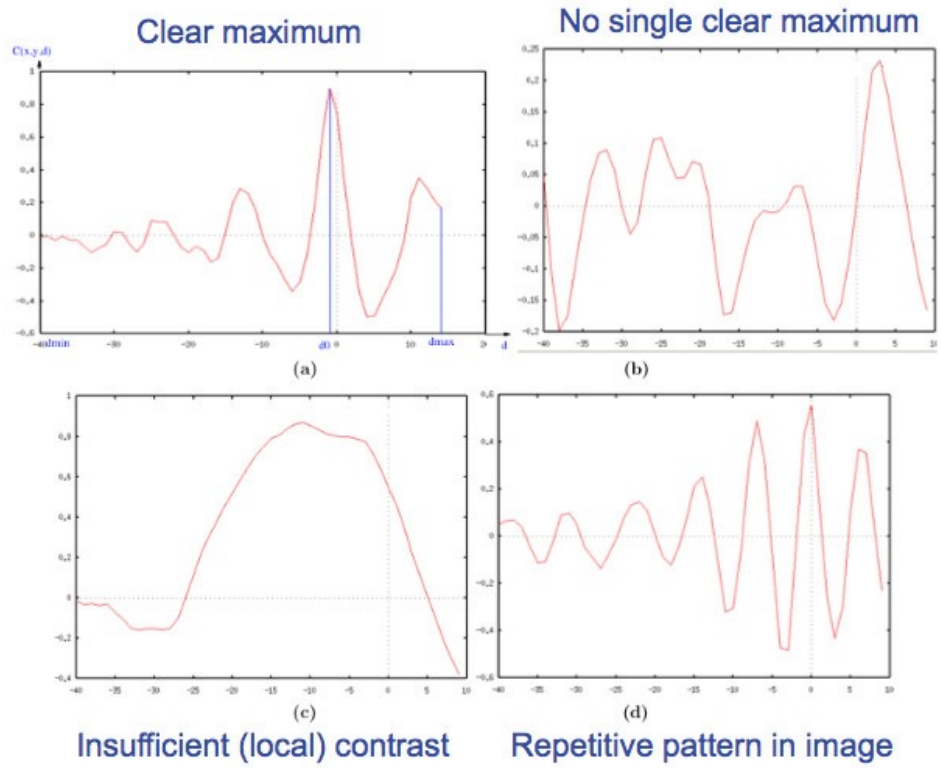
- Once images have been rectified, the disparity (horizontal pixel distance to corresponding point in other image) is calculated for each pixel:
  - Because of rectification, disparity is inversely proportionate to the depth of the object in the scene
- Various methods exist for estimating disparity: a common approach is to score matches across a line based on correlation or inversely proportional to Sum of Square Differences (SSD)



Dense Stereo Triangulation

- Strong sharp points of correlation across the line typically correspond to matching points in the image
- Several effects in images can result in poor disparity estimates including sharp changing in lighting, occluding edges in images and multiple repeating patterns

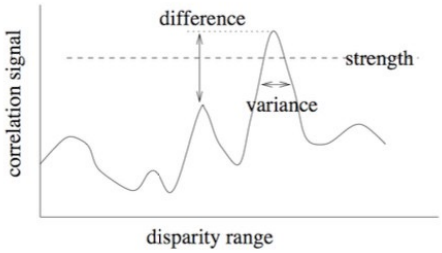




Dense Stereo Triangulation

- Strong sharp points of correlation across the line typically correspond to matching points in the image

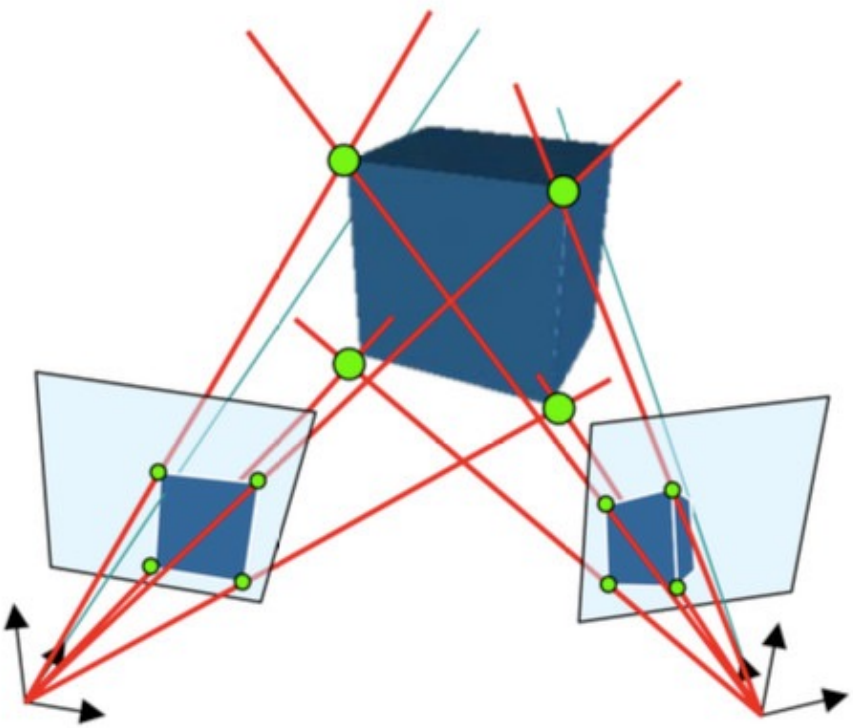
- Several effects in images can result in poor disparity estimates including sharp changing in lighting, occluding edges in images and multiple repeating patterns



Stereo Correspondence Ambiguity

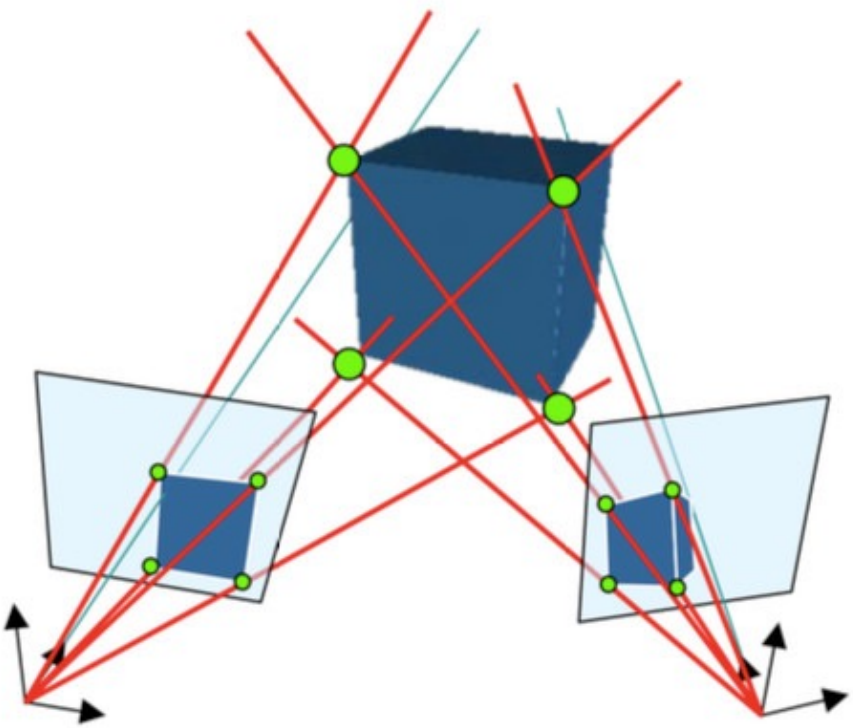
Camera Geometric Calibration

- Geometric calibration is the process of determining the intrinsic and extrinsic parameters of cameras using multiple images of known 3D world points
- Given a series of N points with known 3D coordinates and a series of M views (different camera poses) of the set of N points the intrinsic and extrinsic parameters can be estimated via a least squares minimisation of the pixel re-projection errors
  - Given that each of the N features can be correctly identified (and located in each image) it becomes possible to estimate the intrinsic camera parameters (including distortion terms) simultaneously with the camera orientations and positions relative to the 3D worlds coordinate system



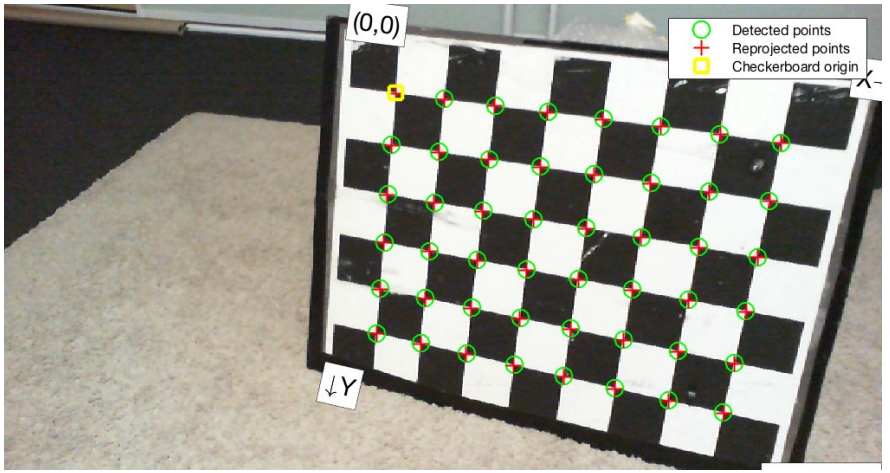
Camera Geometric Calibration

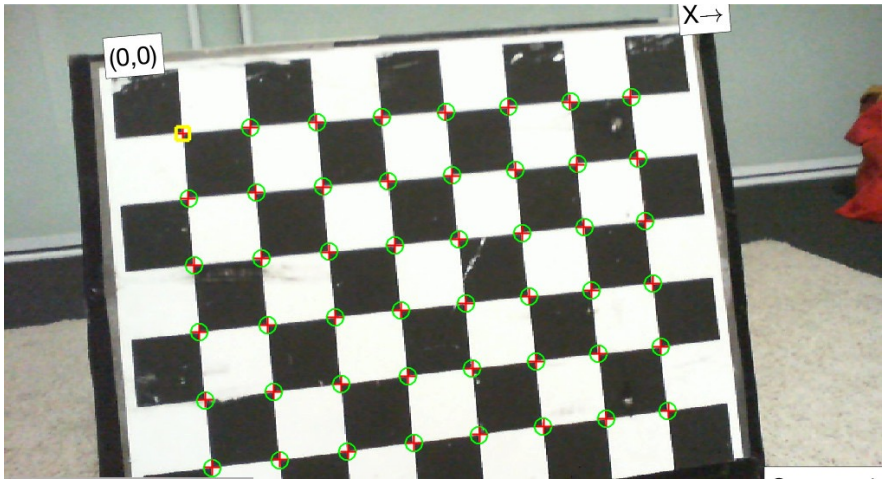
- Geometric calibration is the process of determining the intrinsic and extrinsic parameters of cameras using multiple images of known 3D world points
- Given a series of N points with known 3D coordinates and a series of M views (different camera poses) of the set of N points the intrinsic and extrinsic parameters can be estimated via a least squares minimisation of the pixel re-projection errors
  - Given that each of the N features can be correctly identified (and located in each image) it becomes possible to estimate the intrinsic camera parameters (including distortion terms) simultaneously with the camera orientations and positions relative to the 3D worlds coordinate system



Camera Geometric Calibration

- One common method for camera calibration is via the use of a checkerboard target:
  - Presents easily detectable corner points which can be automatically associated across images based on row-column number on the board
  - Is easy to build accurately and cheaply
- Calibration routines based on multiple images of a checkerboard:
  - MATLAB Camera Calibration Toolbox
  - OpenCV Camera Calibration Functions
  - MATLAB Camera Calibration App

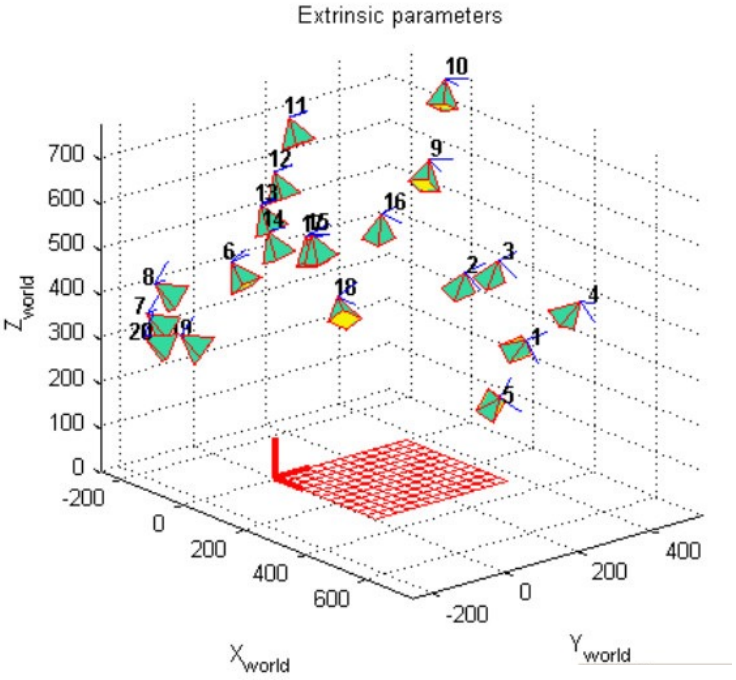


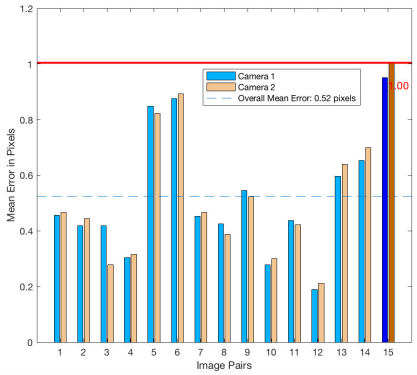


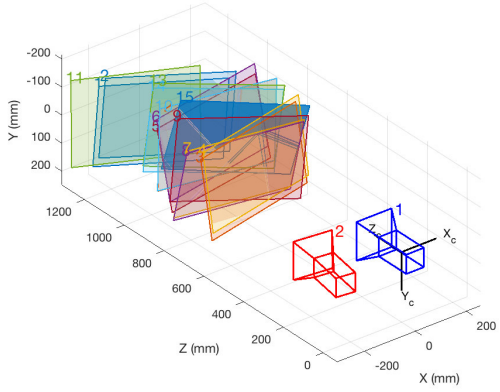
J. Heikkila and O. Silven, “A Four-step Camera Calibration Procedure with Implicit Image Correction”, Computer Vision and Pattern Recognition, 1997

Camera Geometric Calibration

- Images must be obtained from multiple perspectives in order to provide the ability to estimate the view extrinsics (relative rotation and translation w.r.t the checkboard) for each image:
- Once the calibration parameters are estimated, re-projection errors per image can be used to assess if images have been correctly associated or determine regions in the image for which the parametric models for lens distortion are inaccurate
- The same estimation methods can be used for estimating stereo camera extrinsic parameters







J. Heikkila and O. Silven, “A Four-step Camera Calibration Procedure with Implicit Image Correction”, Computer Vision and Pattern Recognition, 1997

Further Reading and Next Week

- References:
  - D. A. Forsyth and J. Ponce, “Computer Vision - A Modern Approach”, Prentice Hall, 2002
  - R. Szeliski, “Computer Vision: Algorithms and Applications”, Springer, 2010

- Next Week:
  - Image Segmentation