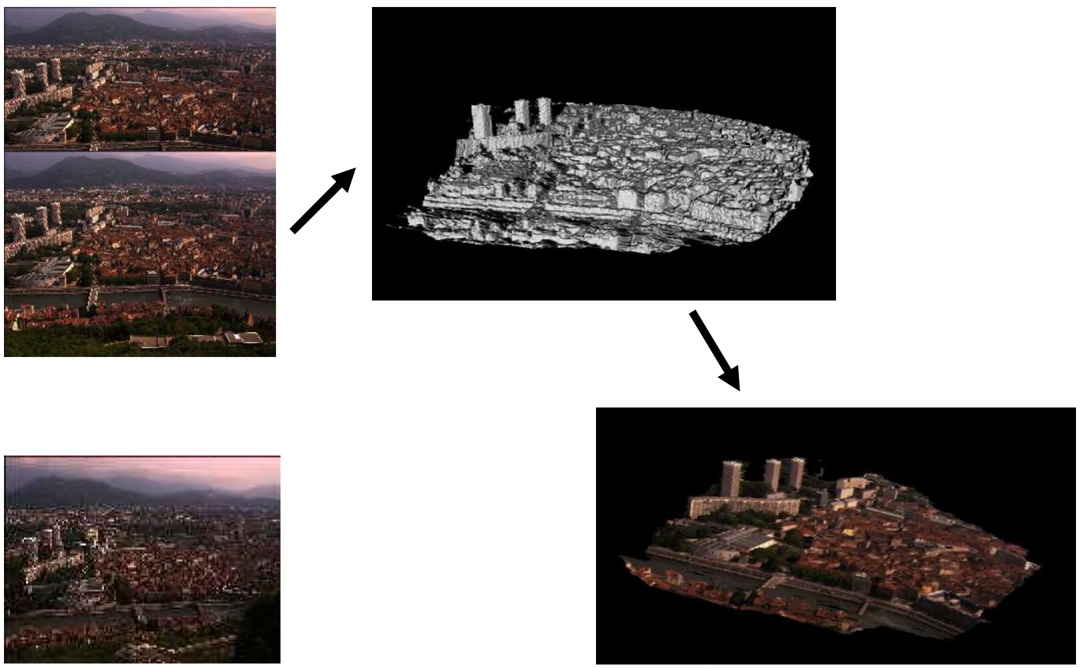
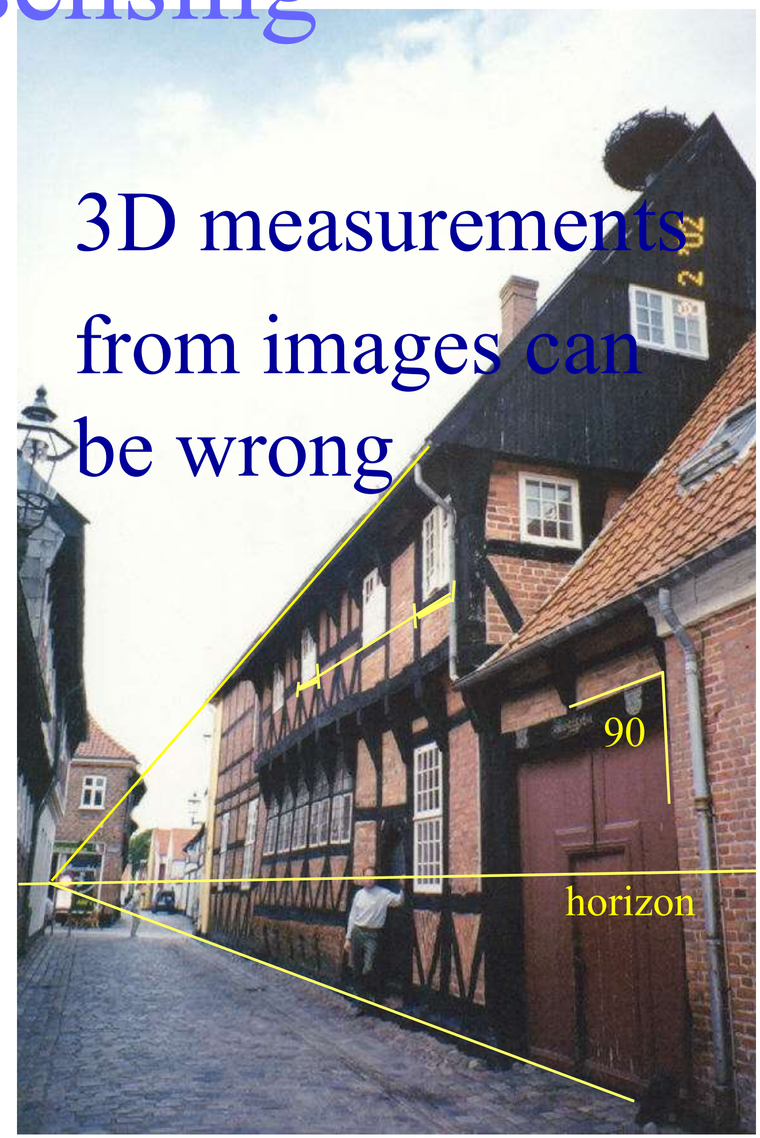
Cmput412
3D vision and sensing



3D modeling from images can be complex

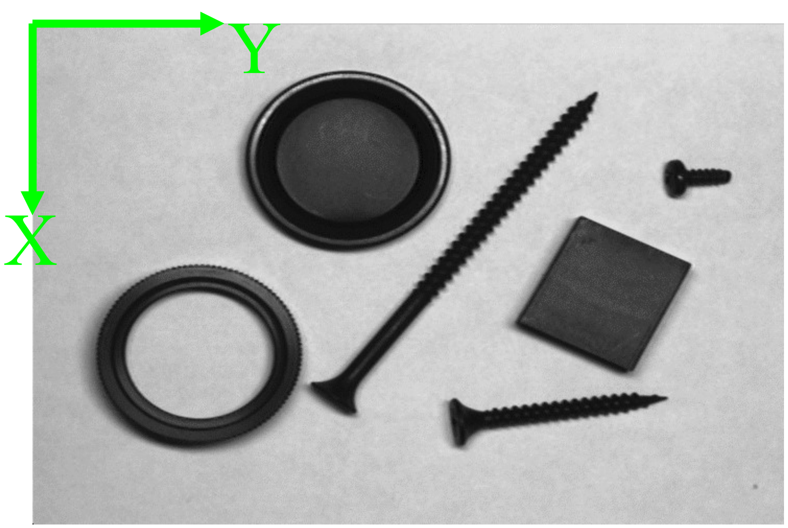


3D measurements from images can be wrong

Previous lectures: 2D machine vision and image processing

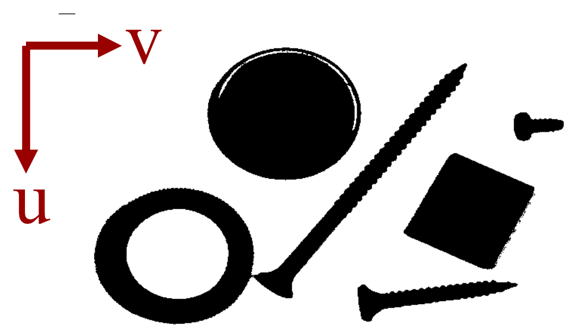
So far 2D vision for measurements on a 2D world plane:

- Usually **overhead camera pointing straight down on a work table**
- Adjust cam position so pixel [u,v] = s[X,Y]. s = scalefactor (pix/mm)
- Pixel coordinates are scaled world coord



Robot scene

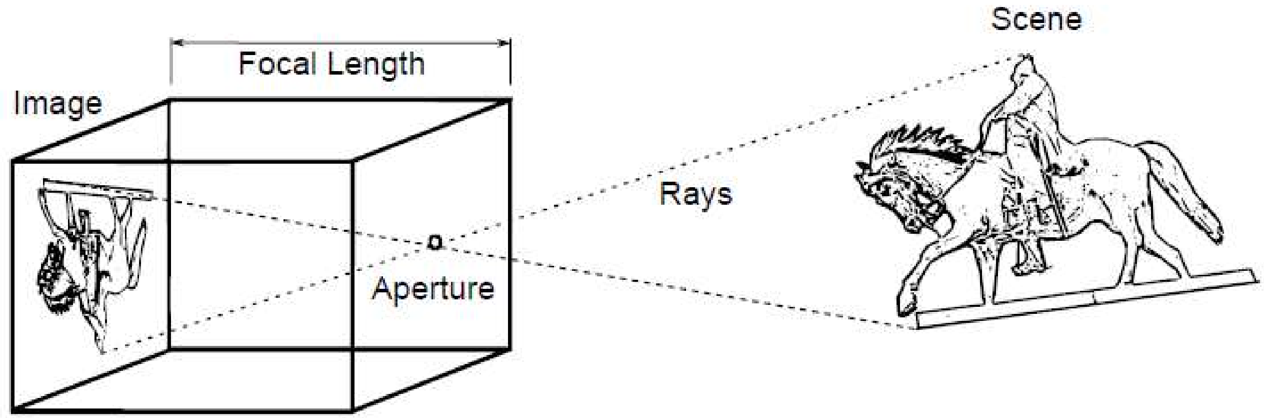
Camera

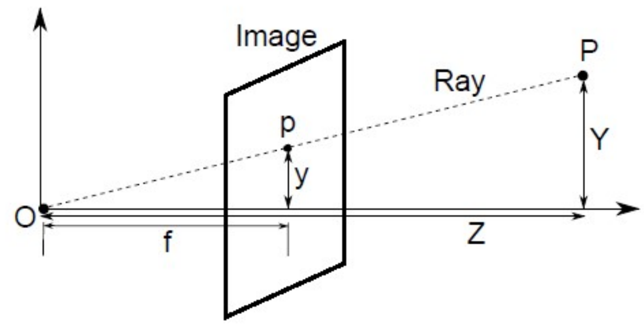


Thresholded image

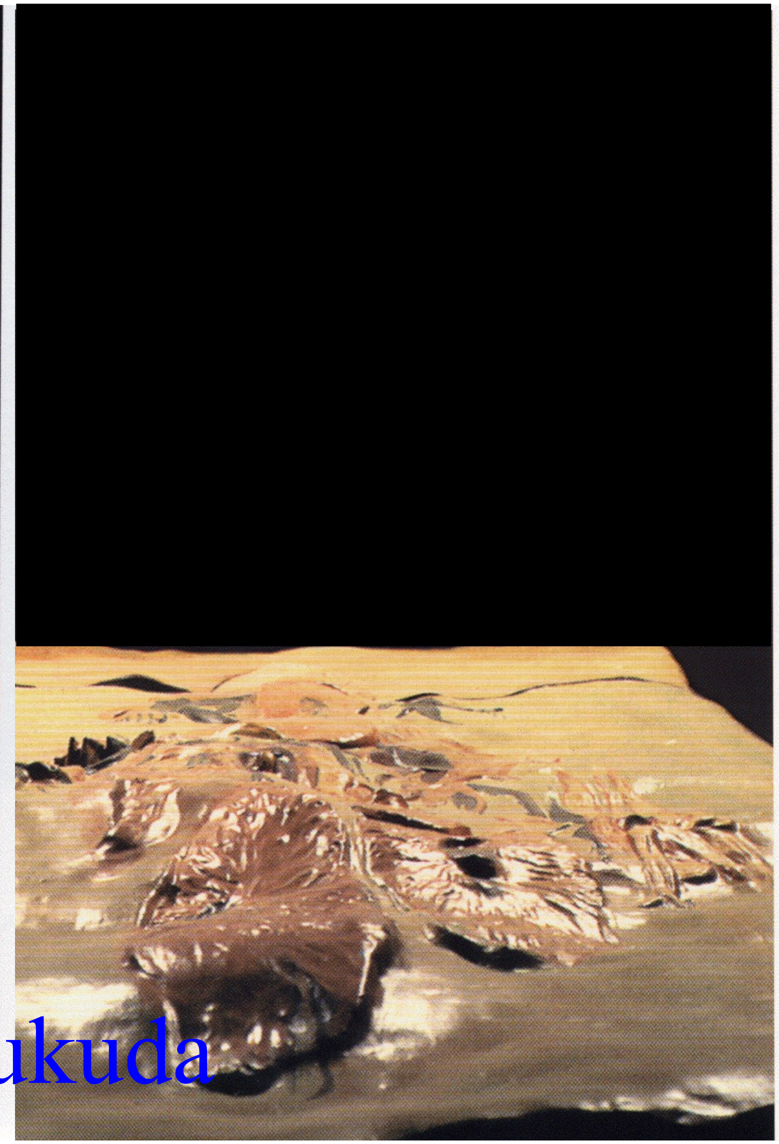
A camera projects the 3D world to 2D images in a complex way

- 3D points project by rays of light that cross the camera's center of projection





A camera projects the 3D world to 2D images in a complex way

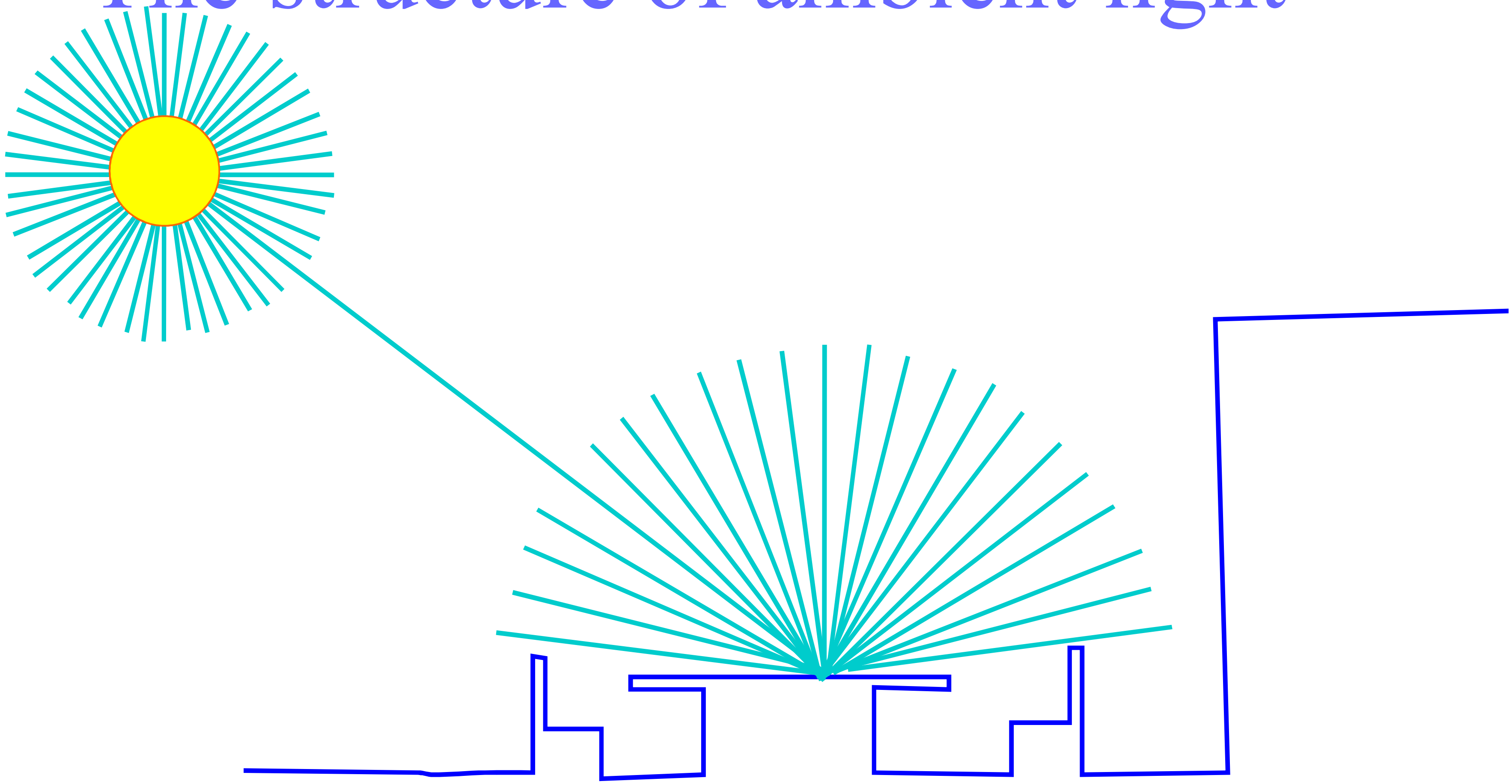


[redacted]

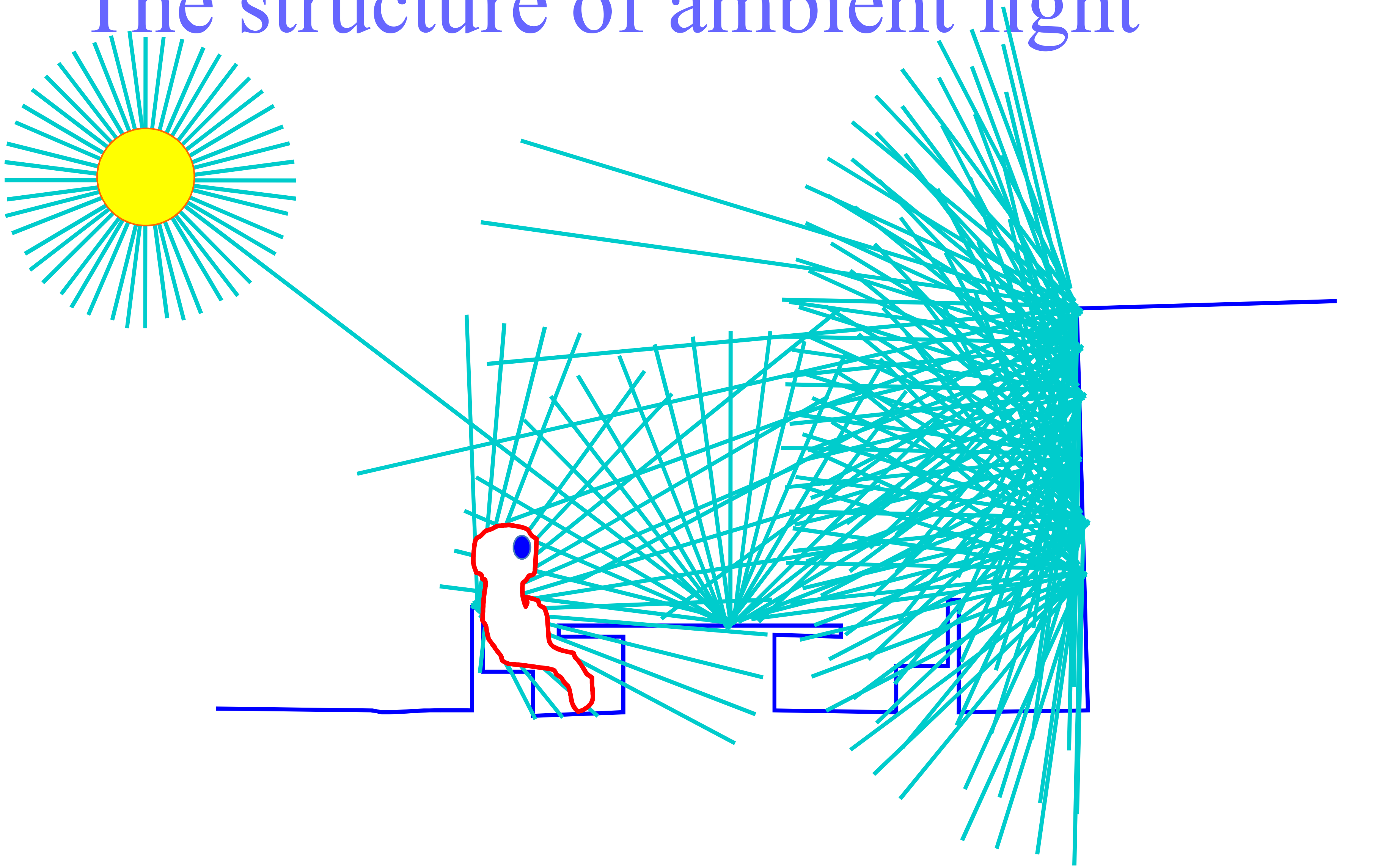
A camera projects the 3D world to 2D images in a complex way

Shigeo Fukuda

The structure of ambient light

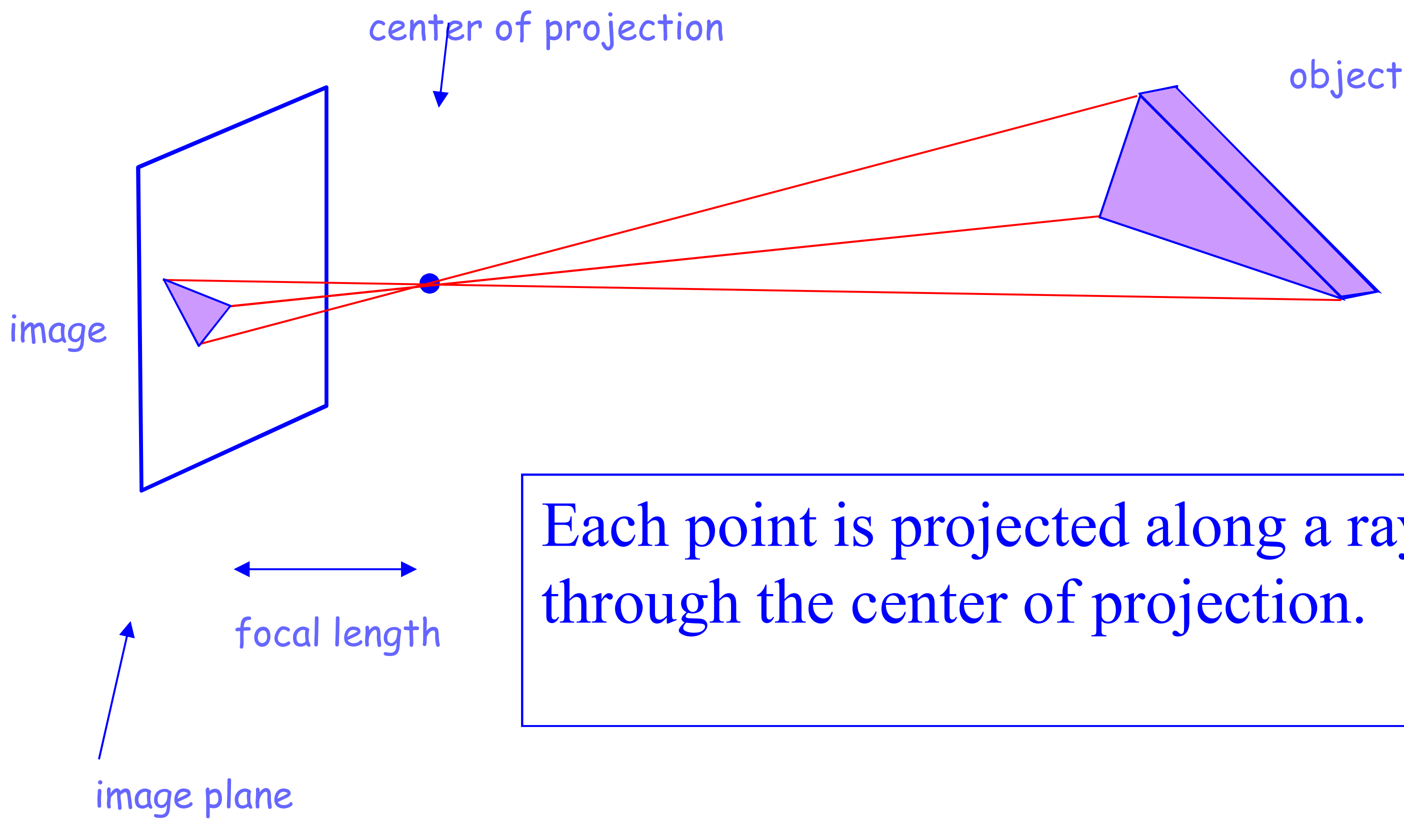


The structure of ambient light



We focus on Camera Geometry

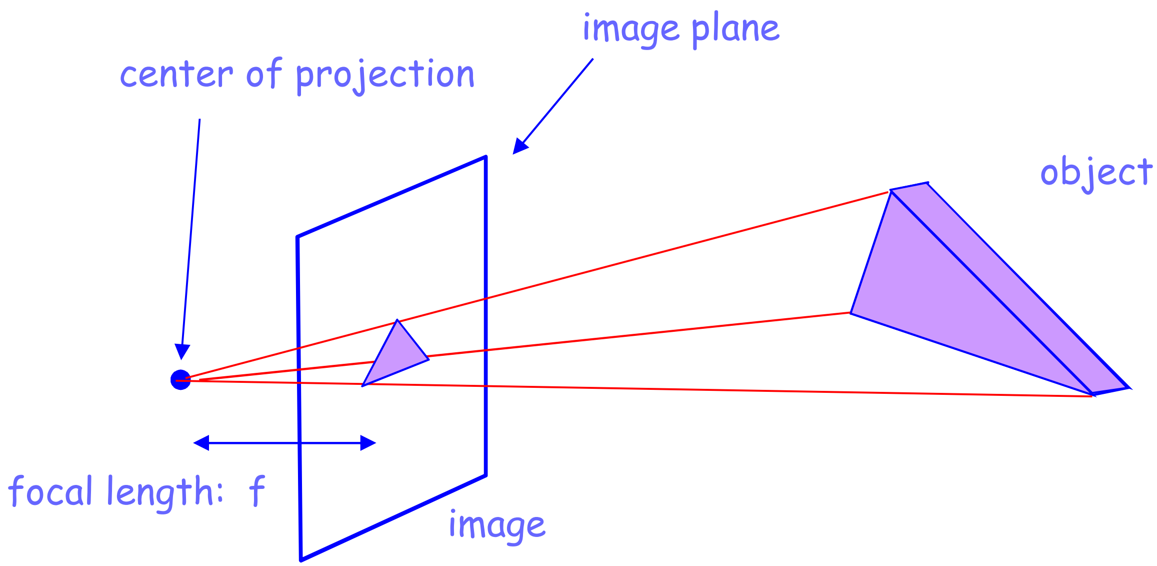
3d → 2d transformation: perspective projection

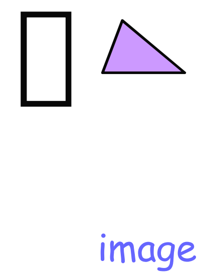


Each point is projected along a ray through the center of projection.

pinhole camera

Perspective Projection

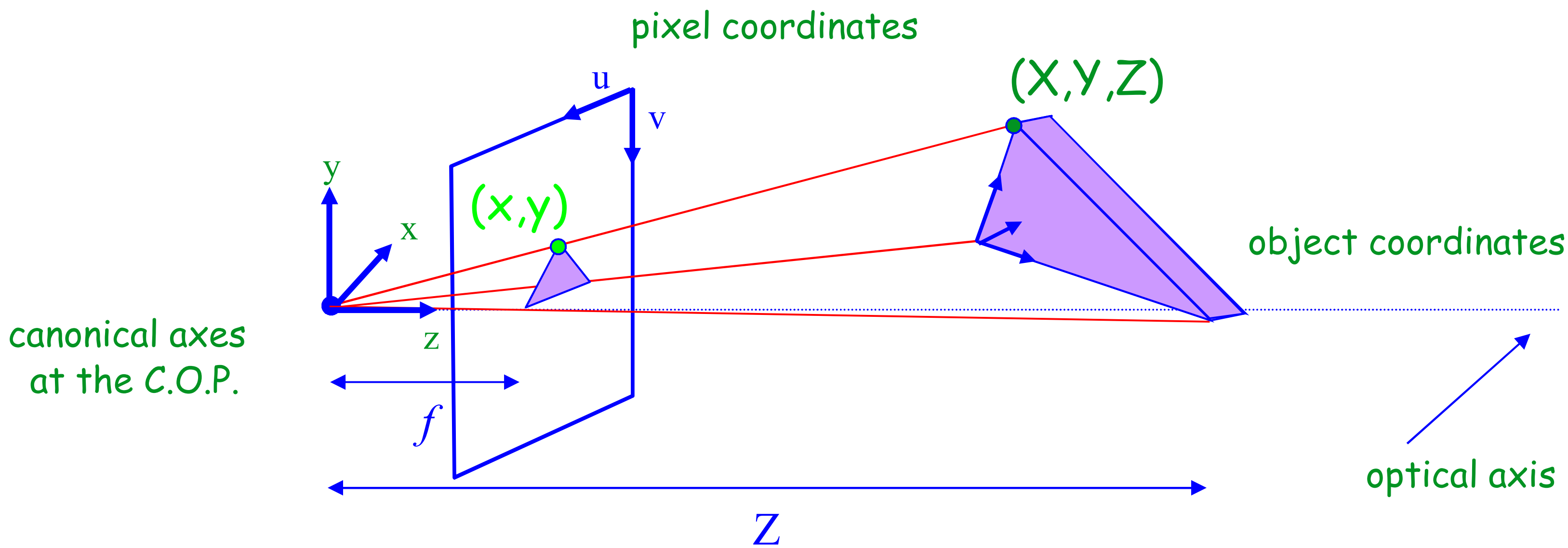


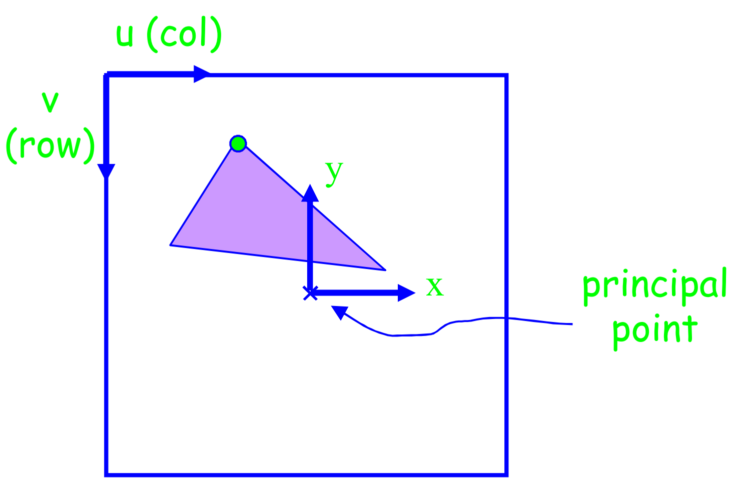


We can save *conscious* mental gyrations by placing the image plane in front of the center.

Add coordinate systems in order to describe feature points...

Coordinate Systems





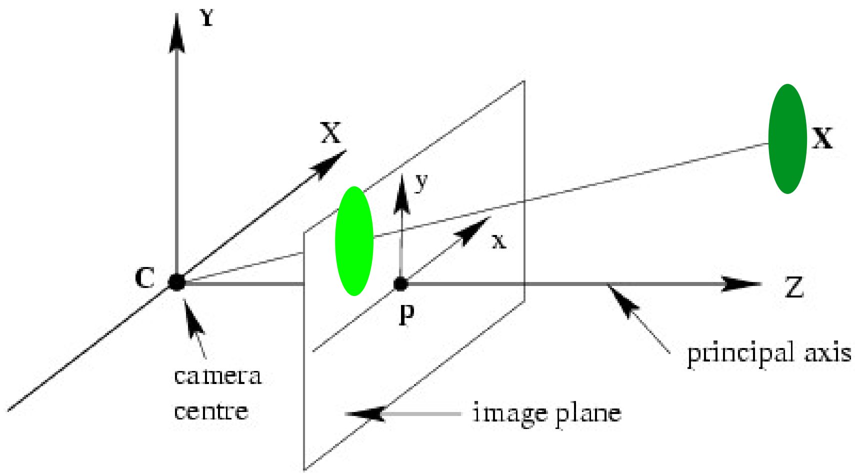
$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

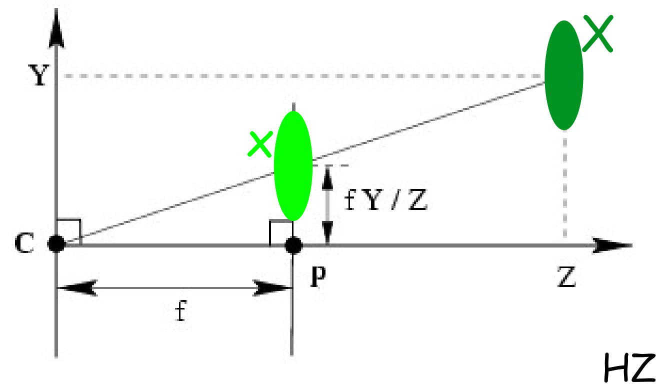
Image structure

Scene structure

how are these related?

Points between 2d and 3d





$$\mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \end{bmatrix}$$

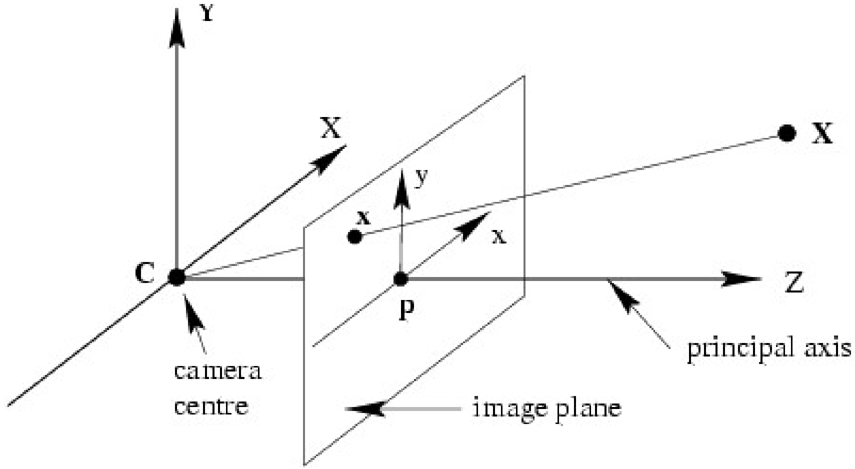
nonlinear!

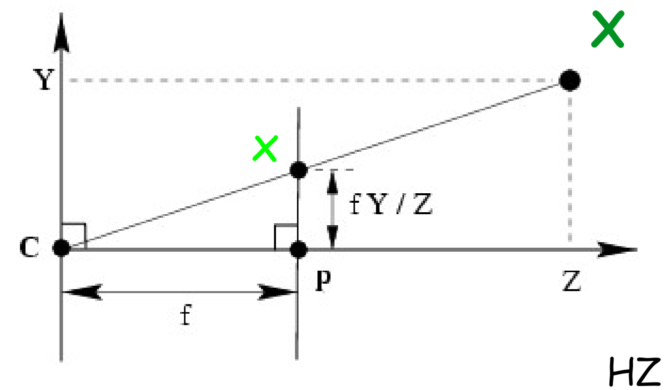
Scene structure

Image structure

But this is only an ideal approximation...

Points between 2d and 3d





$$\mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z} \begin{bmatrix} X \\ Y \end{bmatrix}$$

nonlinear!

Scene structure

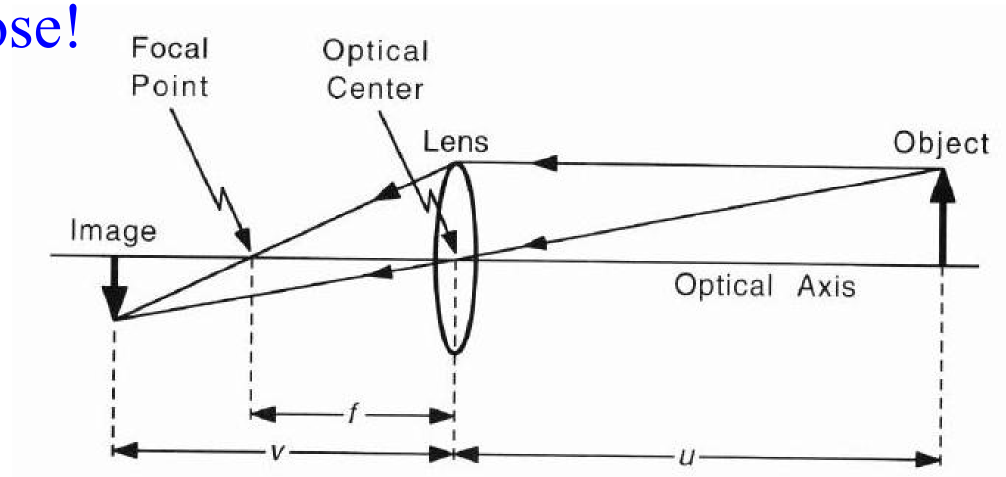
Image structure

But this is only an ideal approximation...

It could be worse… (and often is!)

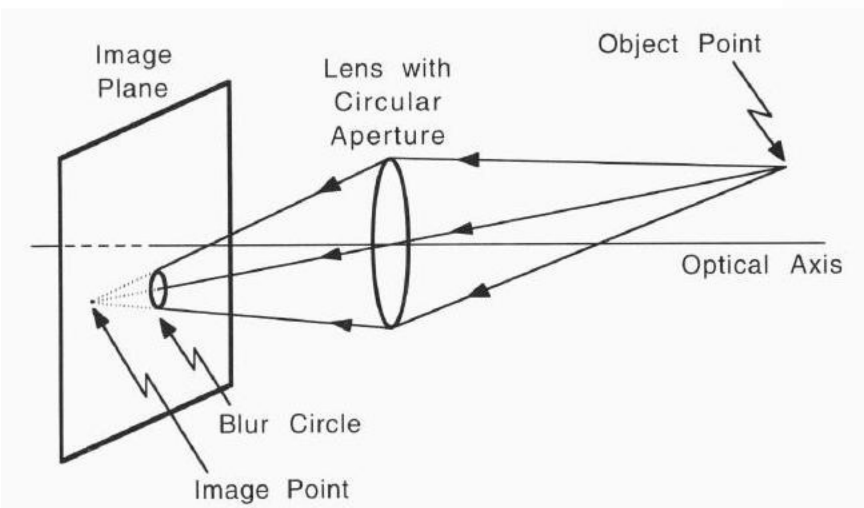
Real cameras don't create exactly a pinhole projection…

But good cameras come close!



Thin-Lens Equation: $\frac{1}{u} + \frac{1}{v} = \frac{1}{f}$

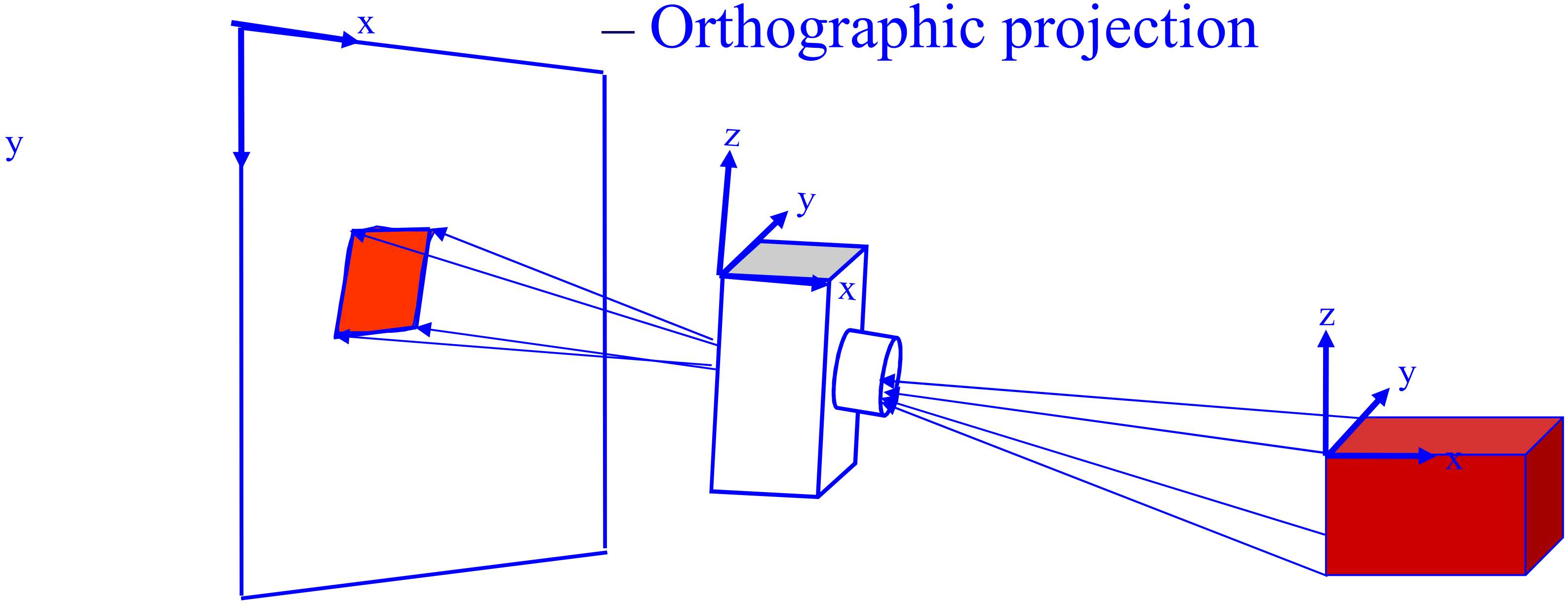
Focus



Lens distortion:
Lines -> curves

Camera models and projections Geometry part 2.

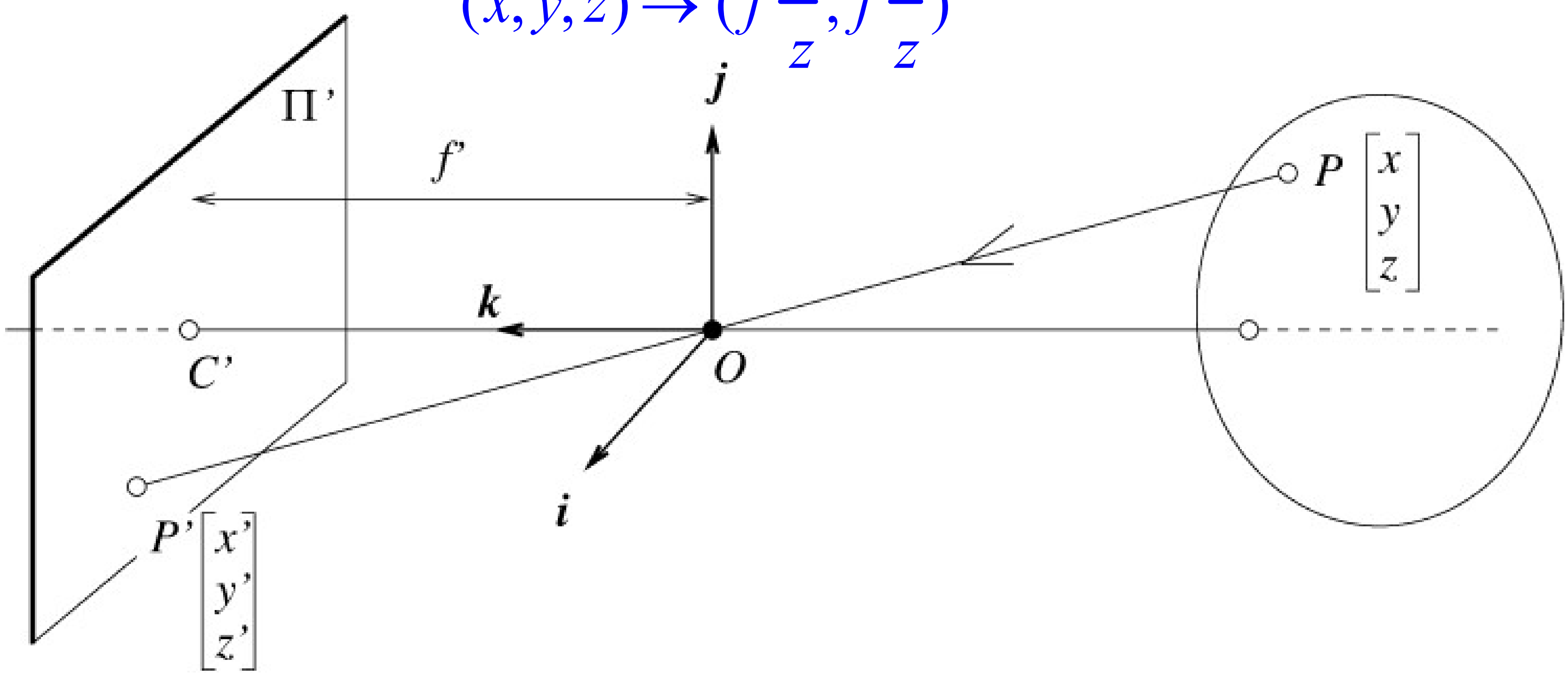
- Using geometry and homogeneous transforms to describe:
  - Perspective projection
  - Weak perspective projection
  - Orthographic projection



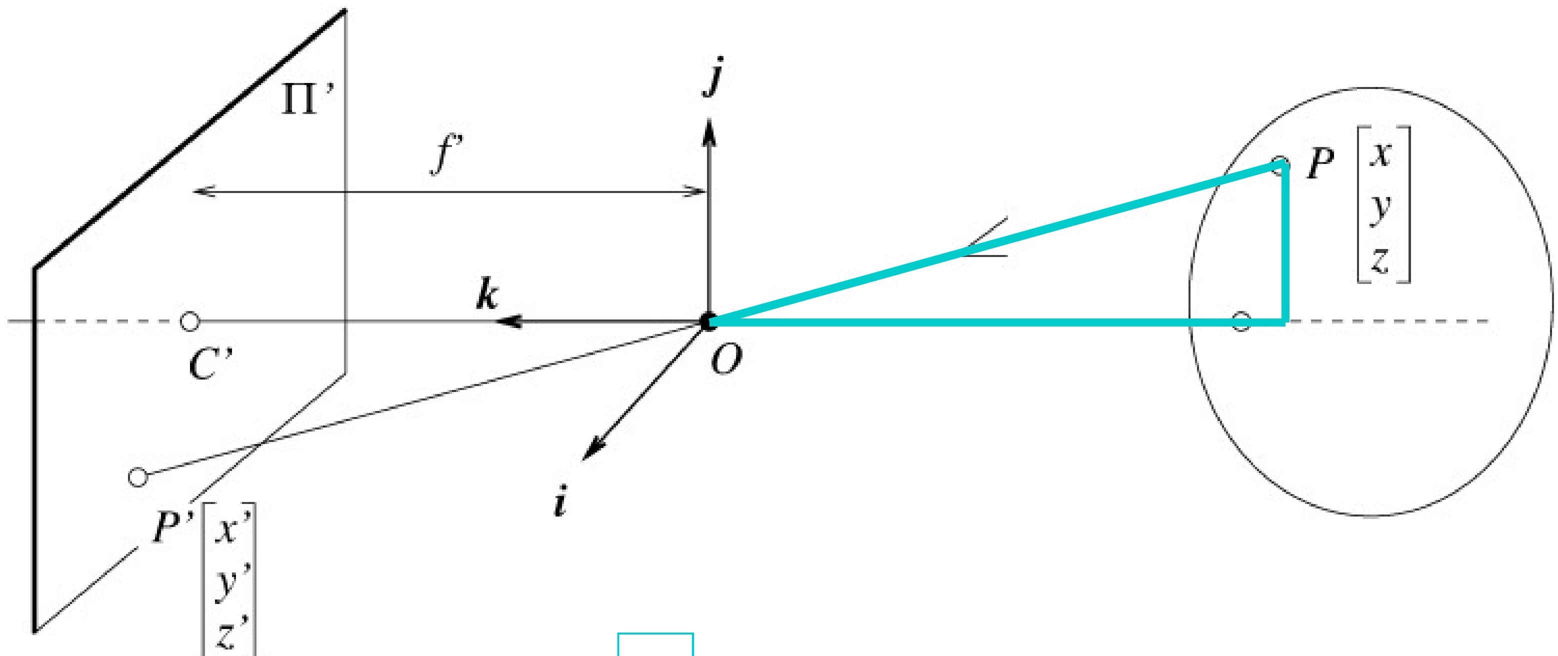
The equation of perspective projection

- Cartesian coordinates:
  - We have, by similar triangles, that (x, y, z) ->
    (f x/z, f y/z, -f)
  - Drop the third coordinate, and get

$$(x, y, z) \rightarrow (f\frac{x}{z}, f\frac{y}{z})$$

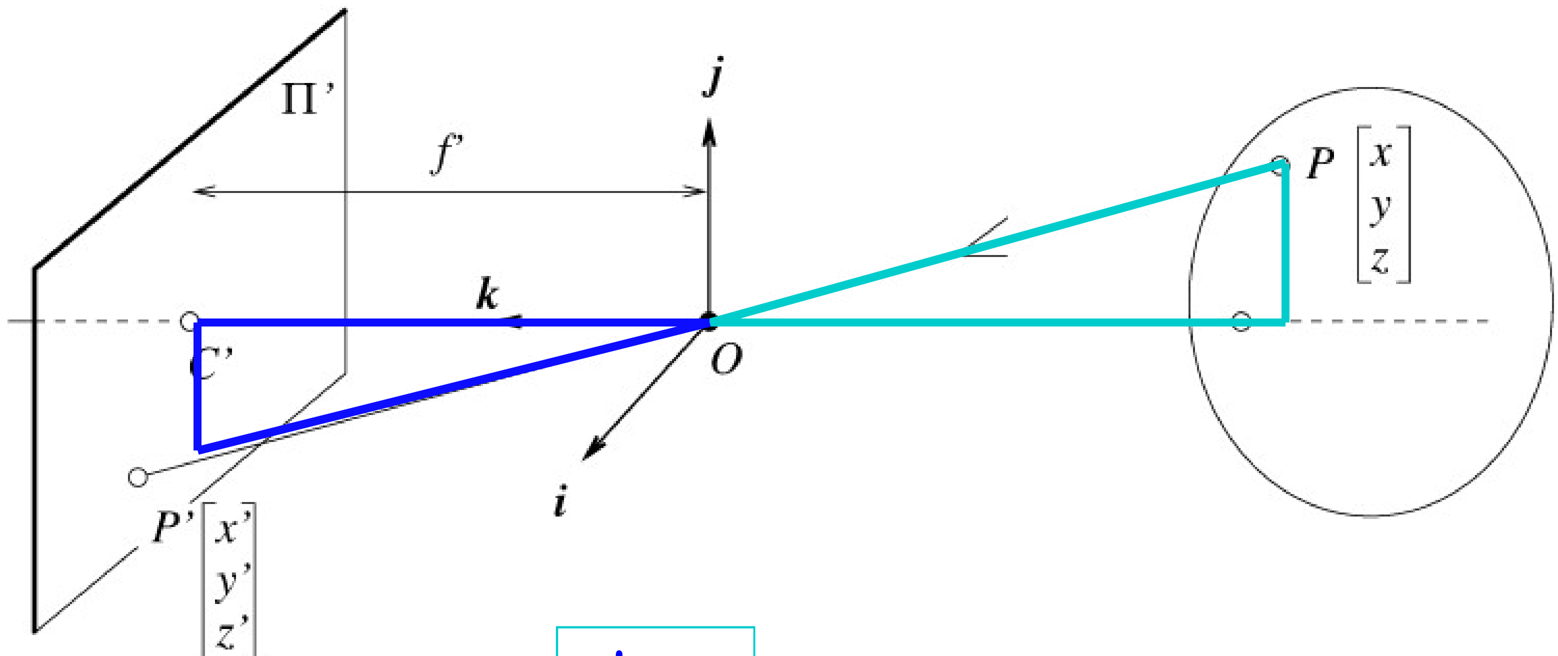


The equation of projection



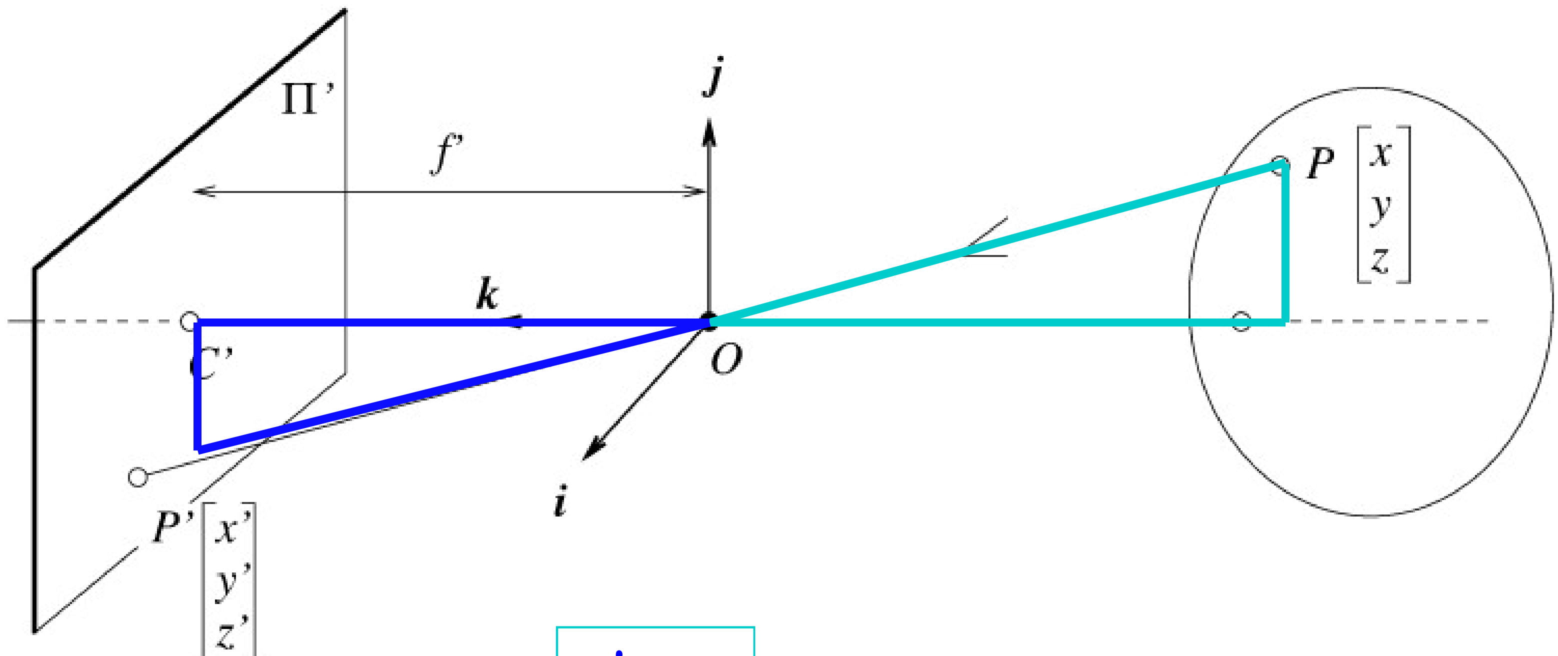
• Similar triangles: $\frac{y}{z}$

The equation of projection



- Similar triangles: $\frac{y'}{f} = \frac{y}{z}$

The equation of projection



- Similar triangles: $\frac{y'}{f} = \frac{y}{z}$

- Projection eq $(x, y, z) \rightarrow (f\frac{x}{z}, f\frac{y}{z})$

Review: Homogeous coordinates
How to translate a 2D point:

- Old way: $$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}$$
- New way:

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Relationship between 3D homogeneous and inhomogeneous

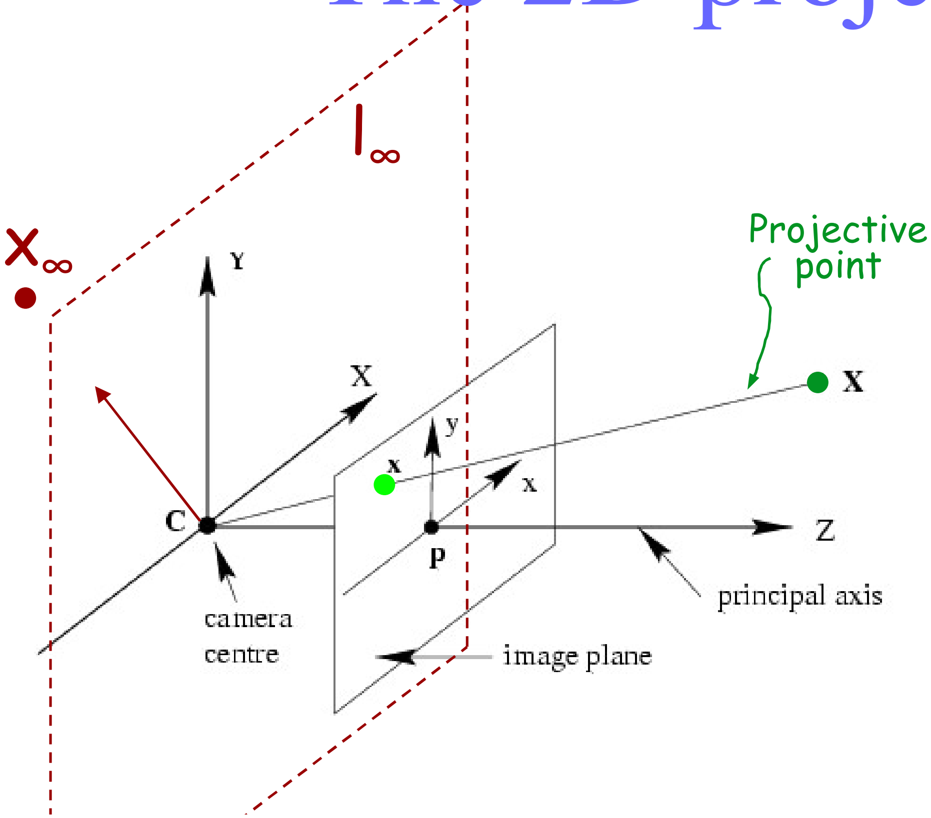
- The Homogeneous coordinate corresponding to the point (x,y,z) is the triple (x_h, y_h, z_h, w) where:

$$x_h = wx$$
$$y_h = wy$$
$$z_h = wz$$

We can (initially) set $w = 1$.

- Suppose a point $P = (x,y,z,1)$ in the homogeneous coordinate system is mapped to a point $P' = (x',y',z',1)$ by a transformations, then the transformation can be expressed in matrix form.

The 2D projective plane



Homogeneous coordinates

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \equiv s \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad s \neq 0$$

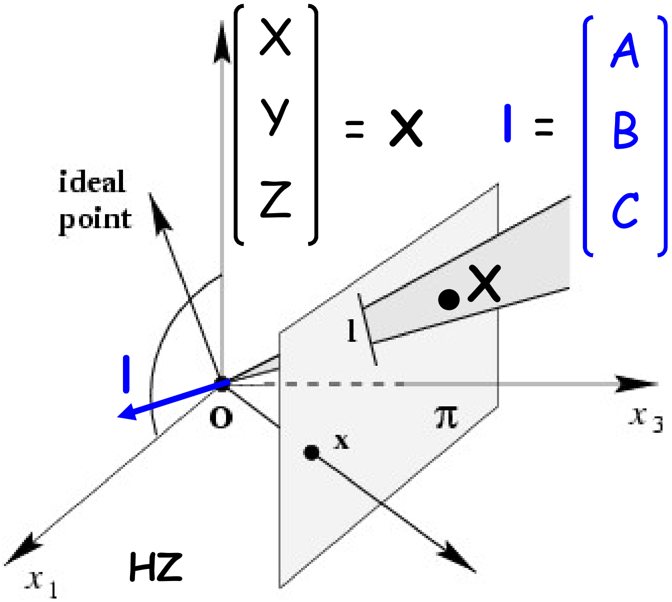
$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} X \\ Y \end{bmatrix}$$

Inhomogeneous equivalent

- Perspective imaging models 2d projective space
- Each 3D ray is a point in P^2 : homogeneous coords.
- Ideal points
- P^2 is R^2 plus a "line at infinity" l_∞

$$X_\infty = \begin{bmatrix} X \\ Y \\ 0 \end{bmatrix}$$

Lines



$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \mathbf{X} \qquad \mathbf{l} = \begin{bmatrix} A \\ B \\ C \end{bmatrix}$$

- Projective line ~ a plane through the origin

$$\mathbf{l}^T\mathbf{X} = \mathbf{X}^T\mathbf{l} = AX + BY + CZ = 0$$

$$\mathbf{X}_\infty = \begin{bmatrix} X \\ Y \\ 0 \end{bmatrix} \qquad \mathbf{l}_\infty = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

"line at infinity"

- Ideal line ~ the plane parallel to the image

Duality: For any 2d projective property, a dual property holds when the role of points and lines are interchanged.

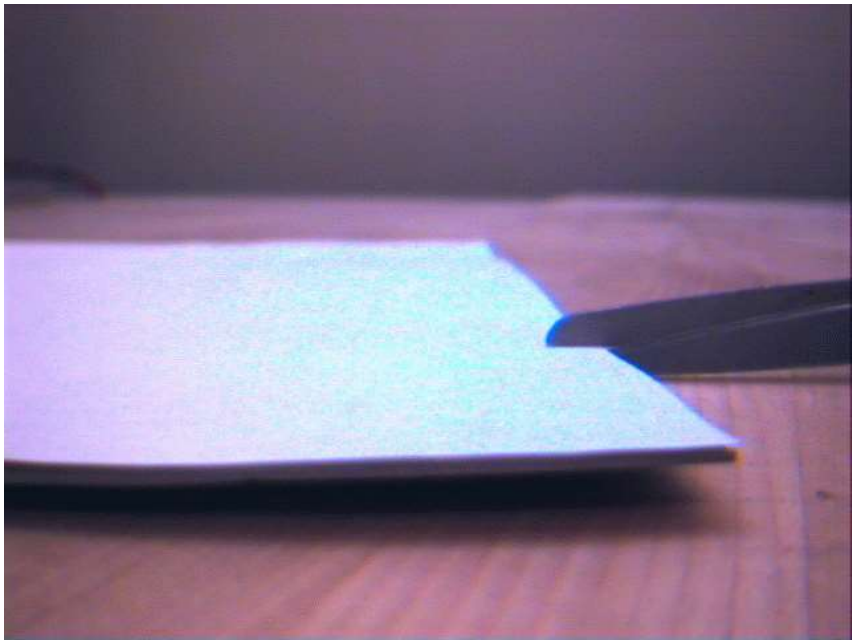
$$\mathbf{l} = \mathbf{X}_1 \times \mathbf{X}_2$$

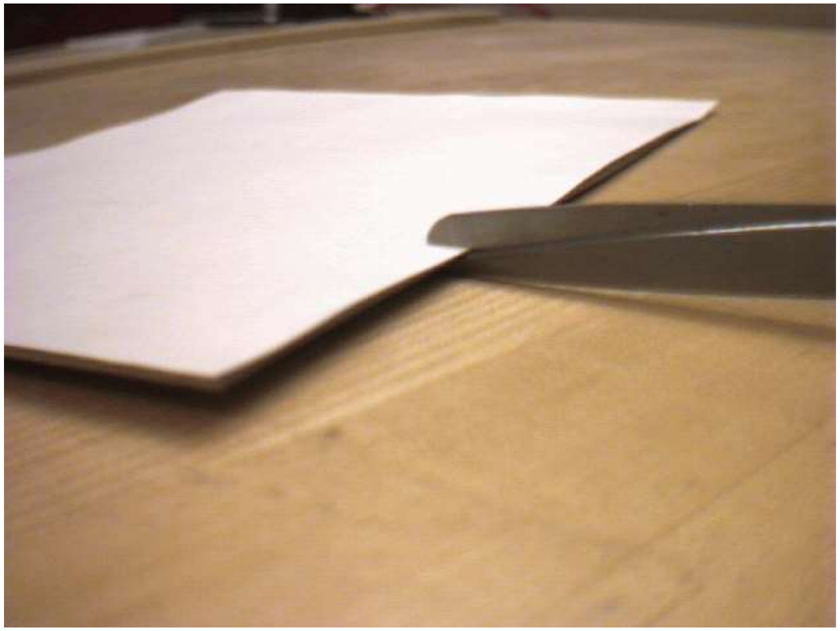
The line joining two points

$$\mathbf{X} = \mathbf{l}_1 \times \mathbf{l}_2$$

The point joining two lines

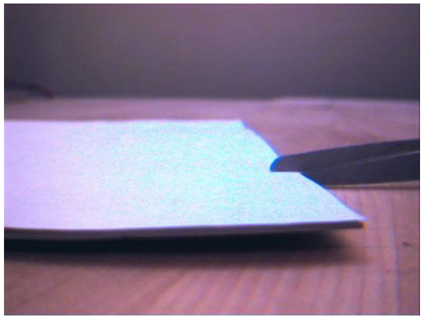
Task ambiguity

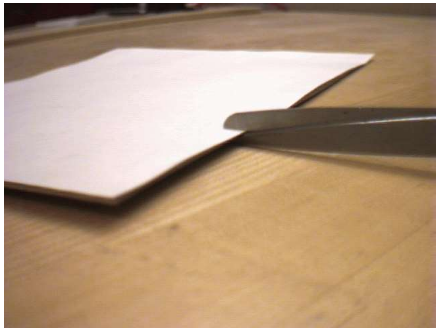




- Will the scissors cut the paper in the middle?

Task ambiguity





- Will the scissors cut the paper in the middle? **NO!**

Solve the cut in the middle task?

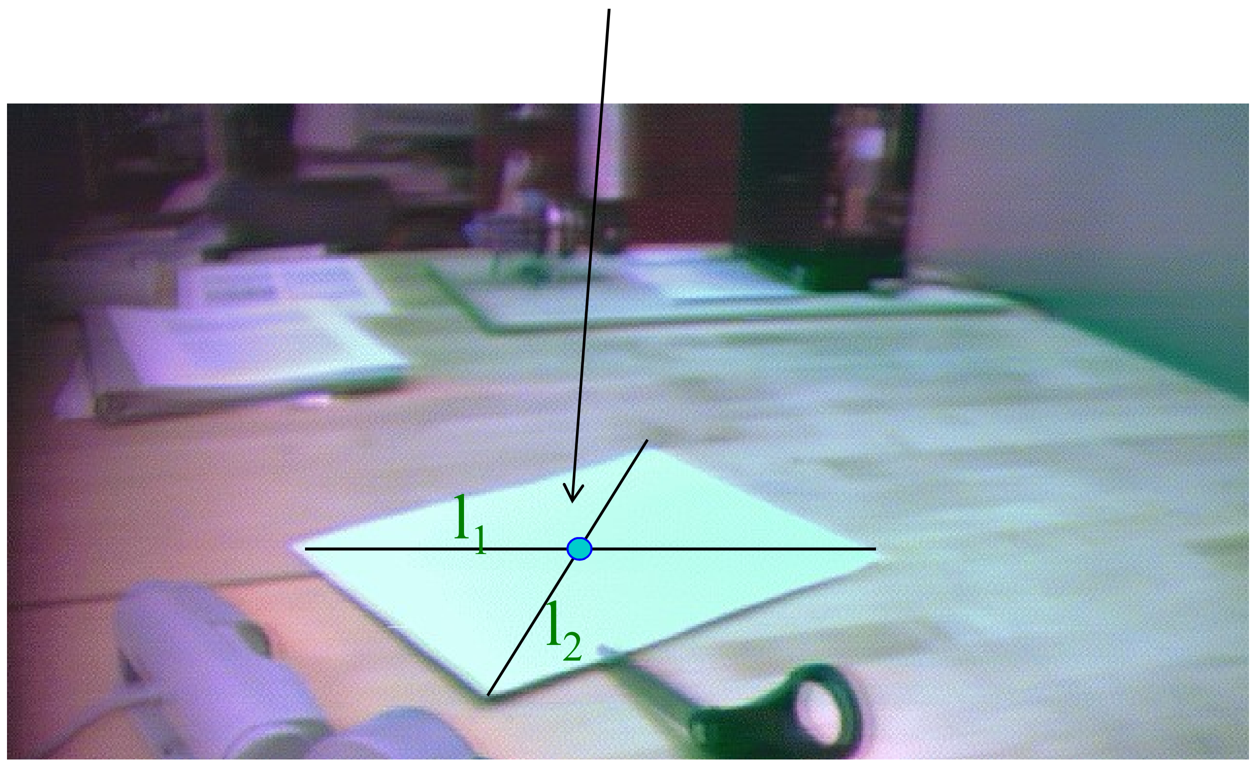
- Compute paper midpoint. How?

Solve the cut in the middle task?

- Compute paper midpoint.

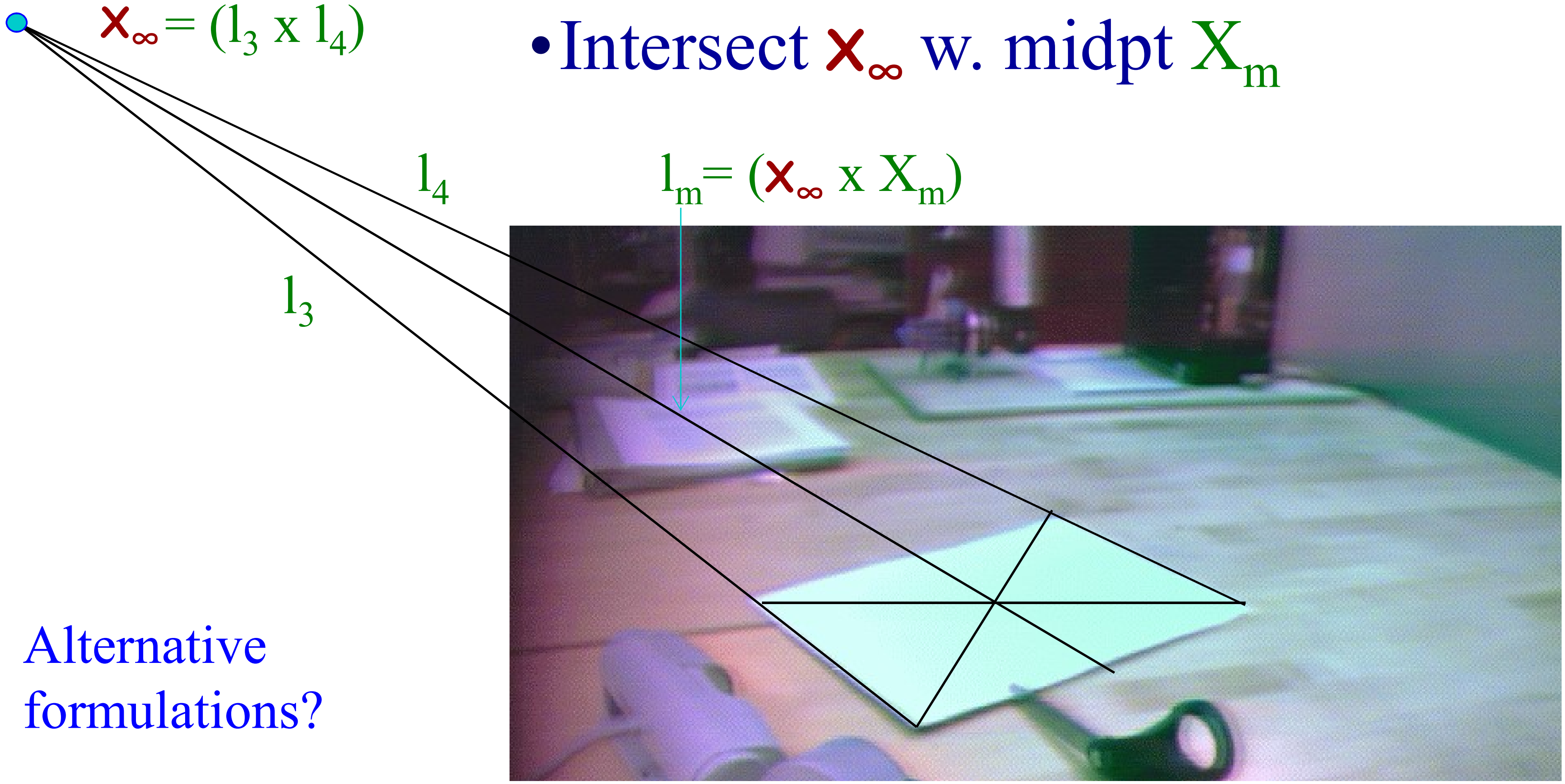
(Are we done yet?)

$x_m = (l_1 \times l_2)$



Solve the cut in the middle task?

- Compute vanishing point $\mathbf{x}_\infty$,
- Intersect $\mathbf{x}_\infty$ w. midpt X_m



Alternative formulations?

The camera matrix

$$(x, y, z) \rightarrow (f\frac{x}{z}, f\frac{y}{z})$$

- Homogenous coordinates for 3D
  - four coordinates for 3D point, 3 for a 2D

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

$$(U, V, W) \rightarrow (\frac{U}{W}, \frac{V}{W}) = (u, v)$$

The camera matrix

$$(x, y, z) \rightarrow (f\frac{x}{z}, f\frac{y}{z})$$

- Homogenous coordinates for 3D
  - Verify homogenous matrix form is the same:

$$\begin{pmatrix} X \\ Y \\ Z/f \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

$$(U, V, W) \rightarrow (\frac{U}{W}, \frac{V}{W}) = (u, v) \quad \rightarrow (f\frac{x}{z}, f\frac{y}{z})$$

The camera matrix

- Homogenous coordinates for 3D
  - equivalence relation (X,Y,Z,T) is the same as (k X, k Y, k Z,k T)

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix} \cong \begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

Canonical form:
Left 3x3
identity matrix

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

The camera matrix

$$(U,V,W) \rightarrow (\frac{U}{W},\frac{V}{W}) = (u,v)$$

- Homogenous coordinates for 3D
  - four coordinates for 3D point, 3 for a 2D

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

- When coordinate systems are not aligned
  - Projective: x image coordinates, X 3D coord, and P an arbitrary 3x4 matrix
  - x = PX
  - Euclidean
  - x = [R|T]X

The camera matrix

$$(U,V,W) \rightarrow (\frac{U}{W},\frac{V}{W}) = (u,v)$$

- Homogenous coordinates for 3D
  - four coordinates for 3D point
  - equivalence relation (X,Y,Z,T) is the same as (k X, k Y, k Z,k T)
- Turn previous expression into HC's
  - HC's for 3D point are (X,Y,Z,T)
  - HC's for point in image are (U,V,W)

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

Camera parameters

- Issue
  - camera may not be at the origin, looking down the z-axis
    - extrinsic parameters
  - one unit in camera coordinates may not be the same as one unit in world coordinates
    - intrinsic parameters - focal length, principal point, aspect ratio, angle between axes, etc.

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} \text{Transformation} \\ \text{representing} \\ \text{intrinsic parameters} \end{pmatrix} \begin{pmatrix} \text{Transformation} \\ \text{representing} \\ \text{projection model} \end{pmatrix} \begin{pmatrix} \text{Transformation} \\ \text{representing} \\ \text{extrinsic parameters} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

Note: f moved from proj to intrinsics!

Intrinsic Parameters

Intrinsic Parameters describe the conversion from metric to pixel coordinates (and the reverse)

$$x_{mm} = -(x_{pix} - o_x)\, s_x$$
$$y_{mm} = -(y_{pix} - o_y)\, s_y$$

or

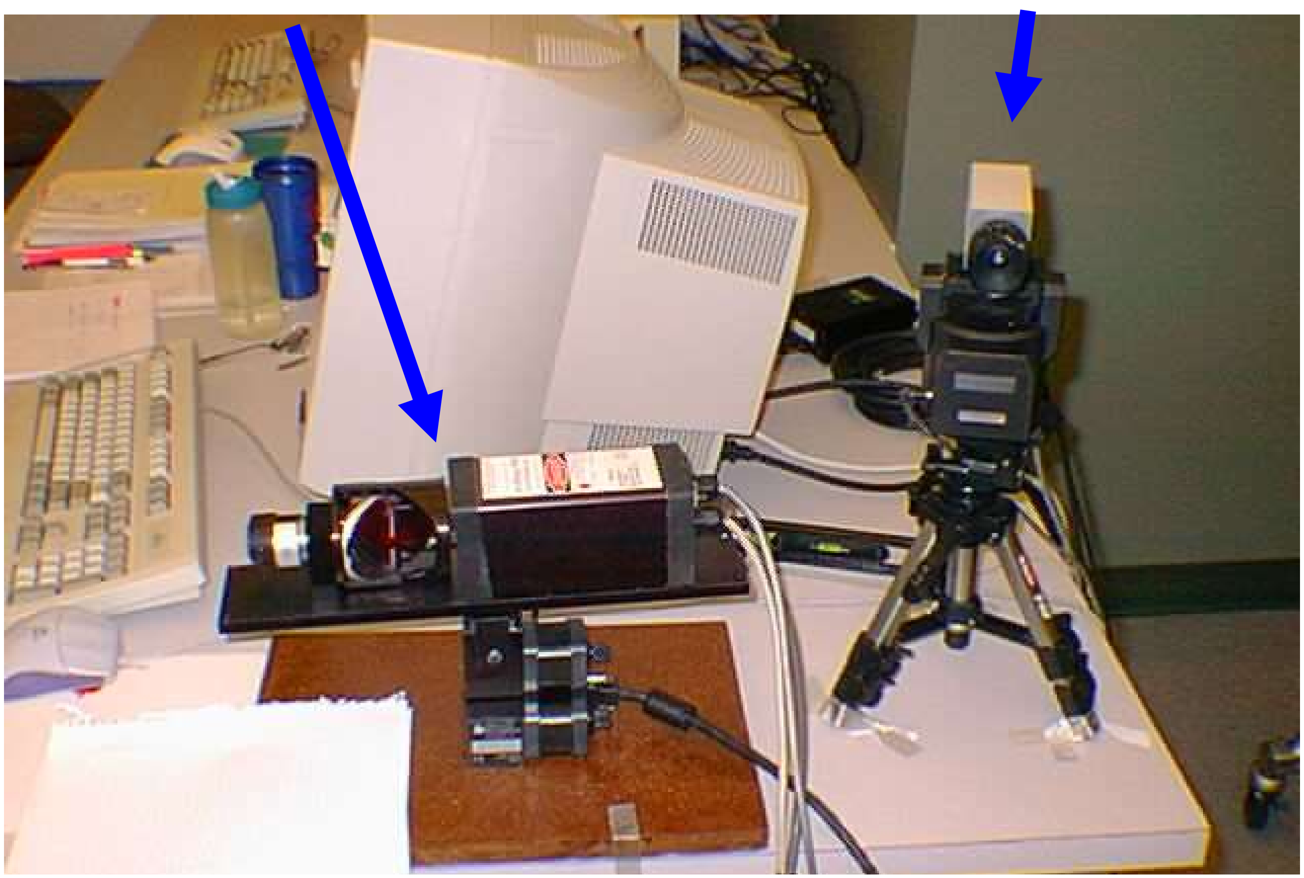
$$\begin{pmatrix} x \\ y \\ w \end{pmatrix}_{pix} = \begin{pmatrix} -f/s_x & 0 & o_x \\ 0 & -f/s_y & o_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ w \end{pmatrix}_{mm} = M_{\text{int}}\, p$$

Note: Focal length is a property of the camera and can be incorporated as above

Example:
A real camera

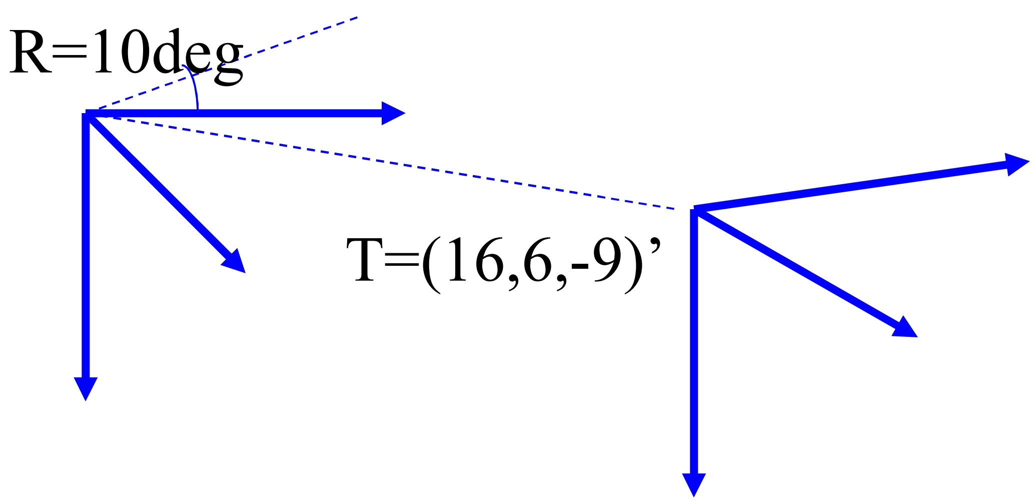
•Laser range finder

•Camera



Relative location Camera-Laser

- Camera
- Laser



In homogeneous coordinates

- Rotation:

$$R = \begin{bmatrix} \cos - 10 & 0 & \sin - 10 \\ 0 & 1 & 0 \\ - \sin - 10 & 0 & \cos - 10 \end{bmatrix}$$

- Translation

$$\mathrm{T} = \begin{pmatrix} 1 & 0 & 0 & 16 \\ 0 & 1 & 0 & 6 \\ 0 & 0 & 1 & -9 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Full projection model

- Camera internal parameters
- Camera projection

$$p_{\text{camera}} = \begin{pmatrix} 1278.6657 & 0 & 256 \\ 0 & 1659.5688 & 240 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0.985 & 0 & -0.174 & 0 \\ 0 & 1 & 0 & 0 \\ 0.174 & 0 & 0.985 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 16 \\ 0 & 1 & 0 & 6 \\ 0 & 0 & 1 & -9 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0.6612 \\ -10.55 \\ 108.0 \\ 1 \end{pmatrix} = \begin{pmatrix} 22262 \\ 16755 \\ 97.47 \end{pmatrix}$$

Extrinsic rot and translation

Full projection model

Coord from clicking in laser scan

$$\begin{pmatrix}22262\\16755\\97.47\end{pmatrix}=\begin{pmatrix}1279&0&256\\0&1660&240\\0&0&1\end{pmatrix}\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\end{pmatrix}\begin{pmatrix}0.985&0&-0.174&16\\0&1&0&6\\0.174&0&0.985&-9\\0&0&0&1\end{pmatrix}\begin{pmatrix}0.6612\\-10.55\\108.0\\1\end{pmatrix}$$

$$\begin{pmatrix}U\\V\\W\end{pmatrix}=\begin{pmatrix}\text{Transformation}\\\text{representing}\\\text{intrinsic parameters}\end{pmatrix}\begin{pmatrix}\text{Transformation}\\\text{representing}\\\text{projection model}\end{pmatrix}\begin{pmatrix}\text{Transformation}\\\text{representing}\\\text{extrinsic parameters}\end{pmatrix}\begin{pmatrix}X\\Y\\Z\\T\end{pmatrix}$$

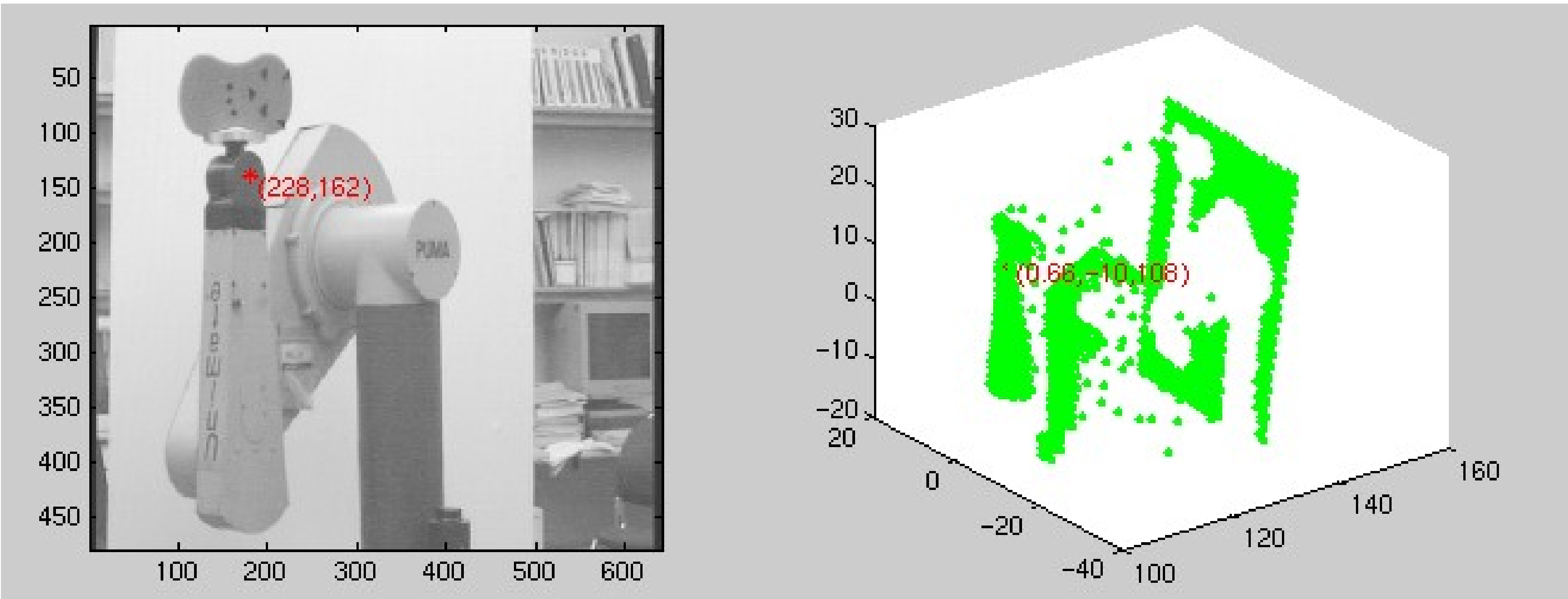
$$(U,V,W)\rightarrow\left(\frac{U}{W},\frac{V}{W}\right)=(u,v)$$

$$\begin{pmatrix}22262\\16755\\97.47\end{pmatrix}\longrightarrow\begin{pmatrix}\frac{22267}{97.47}\\\frac{16755}{97.47}\end{pmatrix}=\begin{pmatrix}228\\162\end{pmatrix}$$

Image pixel coordinates

Result

- Camera image
- Laser measured 3D structure



Camera parameters

- Issue
  - camera may not be at the origin, looking down the z-axis
    - extrinsic parameters
  - one unit in camera coordinates may not be the same as one unit in world coordinates
    - intrinsic parameters - focal length, principal point, aspect ratio, angle between axes, etc.

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \begin{pmatrix} \text{Transformation} \\ \text{representing} \\ \text{intrinsic parameters} \end{pmatrix} \begin{pmatrix} \text{Transformation} \\ \text{representing} \\ \text{projection model} \end{pmatrix} \begin{pmatrix} \text{Transformation} \\ \text{representing} \\ \text{extrinsic parameters} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ T \end{pmatrix}$$

Note: f moved from proj to intrinsics!

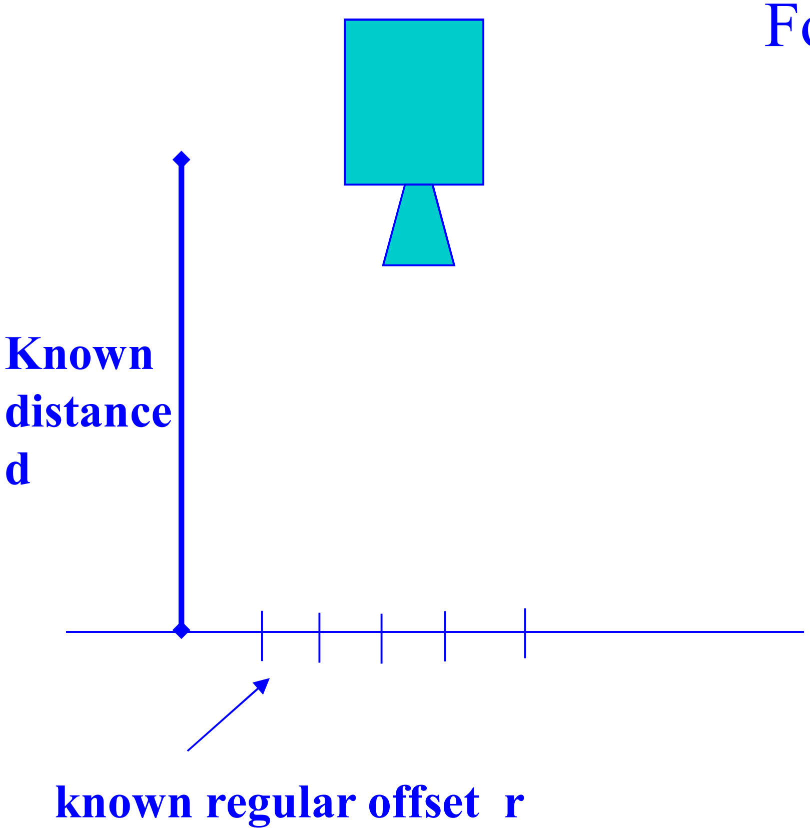
CAMERA INTERNAL CALIBRATION

Compute Sx

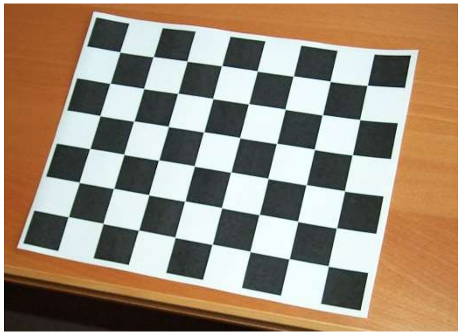
Focal length = 1/ Sx

$$\frac{rk_i}{d} = (x_i - o_x)s_x$$

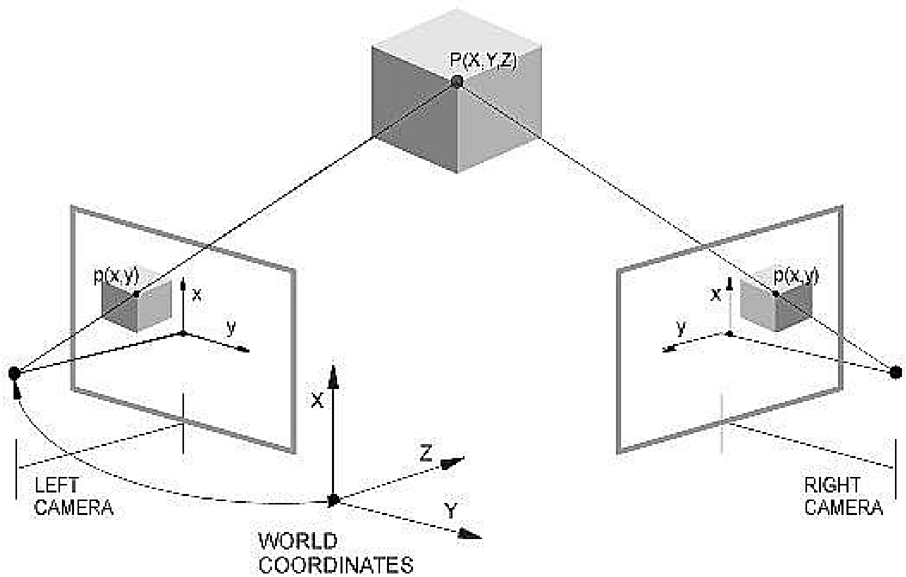
$$\frac{r}{d} = (x_{i+1} - x_i)s_x$$



A simple way to get scale parameters; we can compute the optical center as the numerical center and therefore have the intrinsic parameters



Stereo Vision



- **GOAL:** Passive 2-camera system for triangulating 3D position of points in space to generate a depth map of a world scene.
- Humans use stereo vision to obtain depth

Stereo depth calculation: Simple case, aligned cameras

DISPARITY= (XL - XR)

Similar triangles:

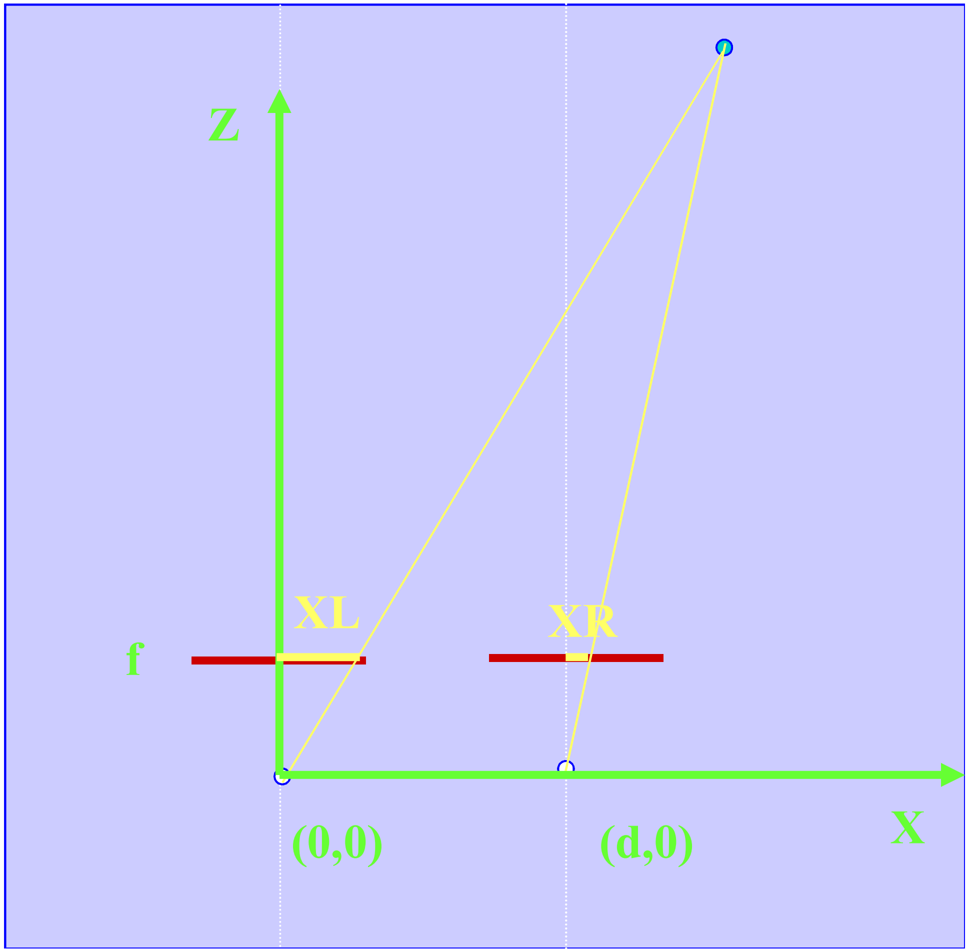
$$Z = (f/XL)\ X$$
$$Z = (f/XR)\ (X-d)$$

Solve for X:

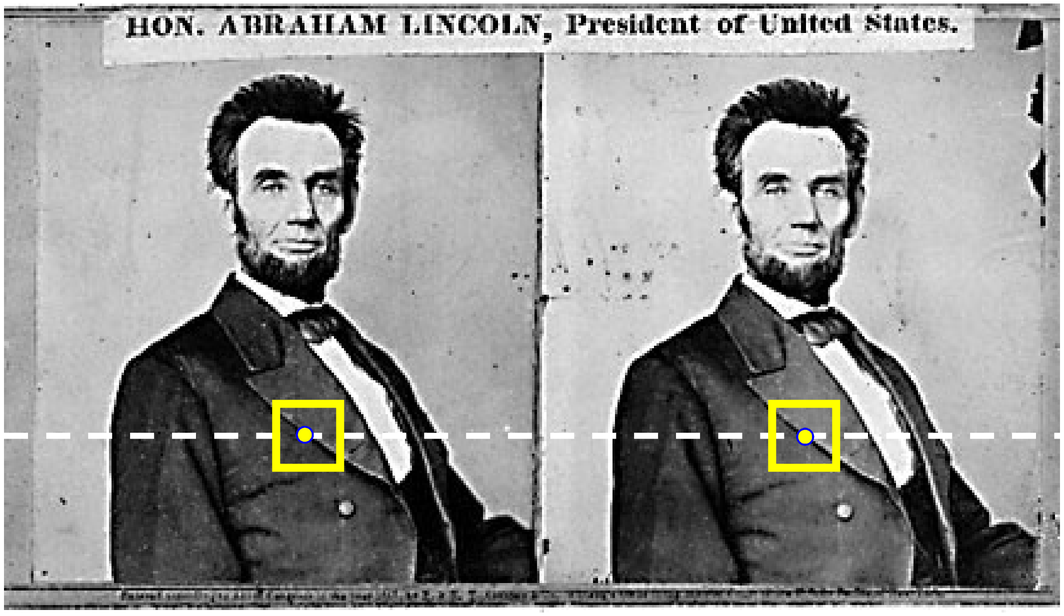
$$(f/XL)\ X = (f/XR)\ (X-d)$$
$$X = (XL\ d)\ /\ (XL - XR)$$

Solve for Z:

$$Z = \frac{d*f}{(XL - XR)}$$



Your basic stereo algorithm



For each epipolar line

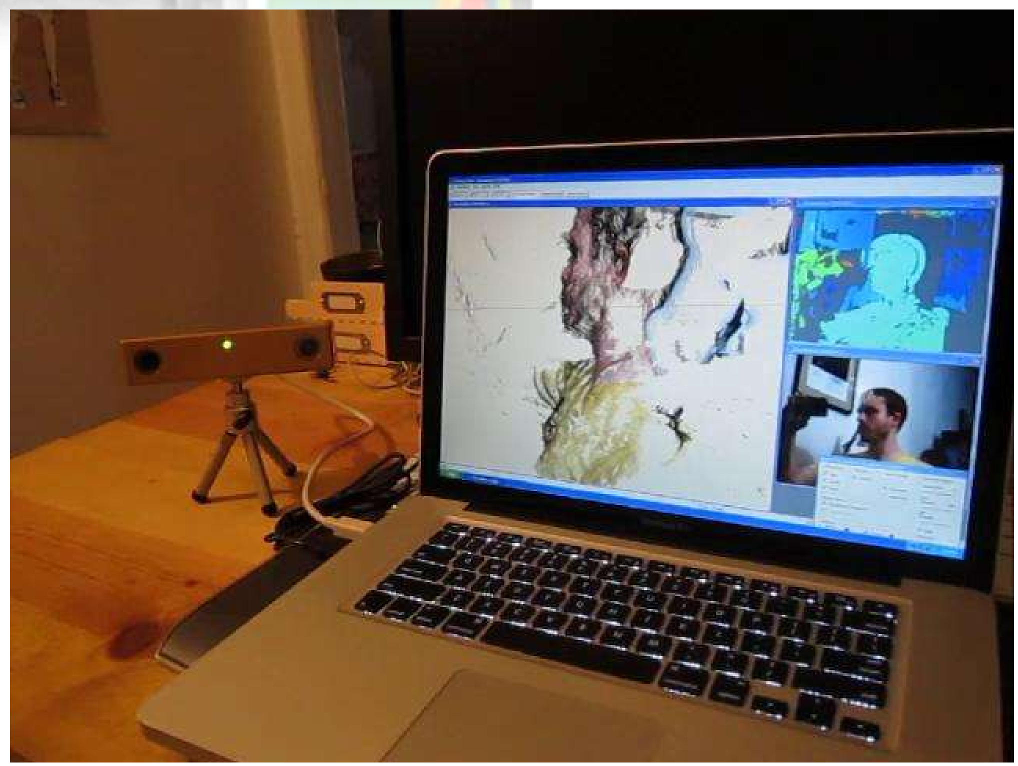
For each pixel in the left image

- compare with every pixel on same epipolar line in right image
- pick pixel with minimum match cost

Improvement: match ***windows***

- This should look familar...

Commercial Stereo Systems



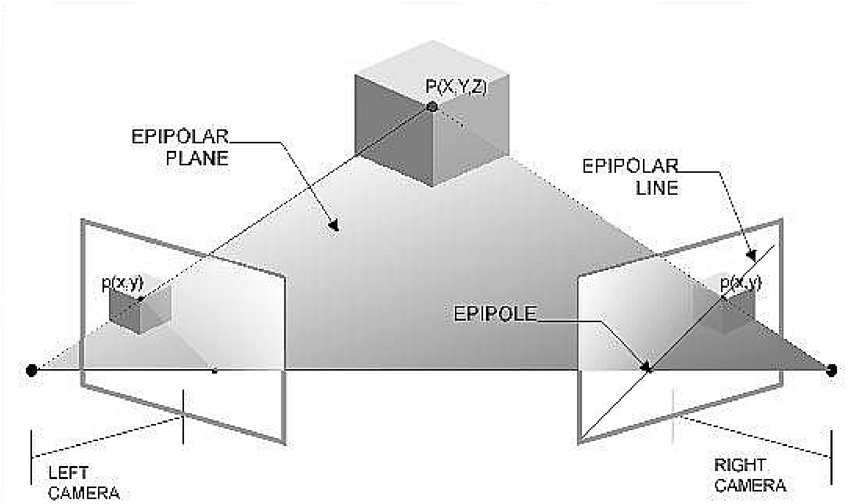
- PointGrey Bumblebee
  - Two cameras
  - Software
- Real-time
- “Infinite” range

http://vimeo.com/12713979

SVM Stereo Head Mounted on an All-terrain Robot

- Stereo Camera
  - Vider Desing
  - www.videredesign.com
- Robot
  - Shrimp, EPFL
- Application of Stereo Vision
  - Traversability calculation based on stereo images for outdoor navigation
  - Motion tracking

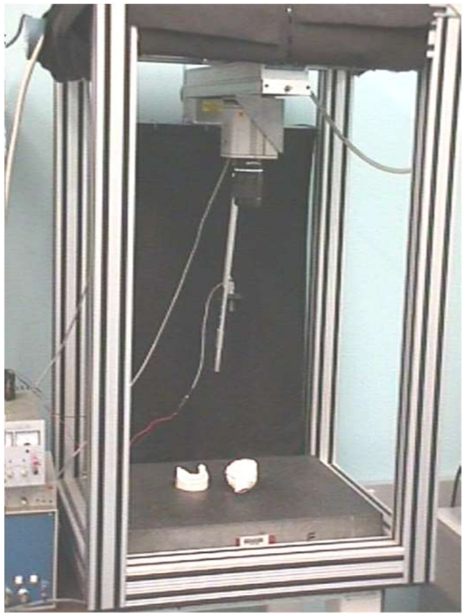
For not aligned cameras: Match along epipolar lines



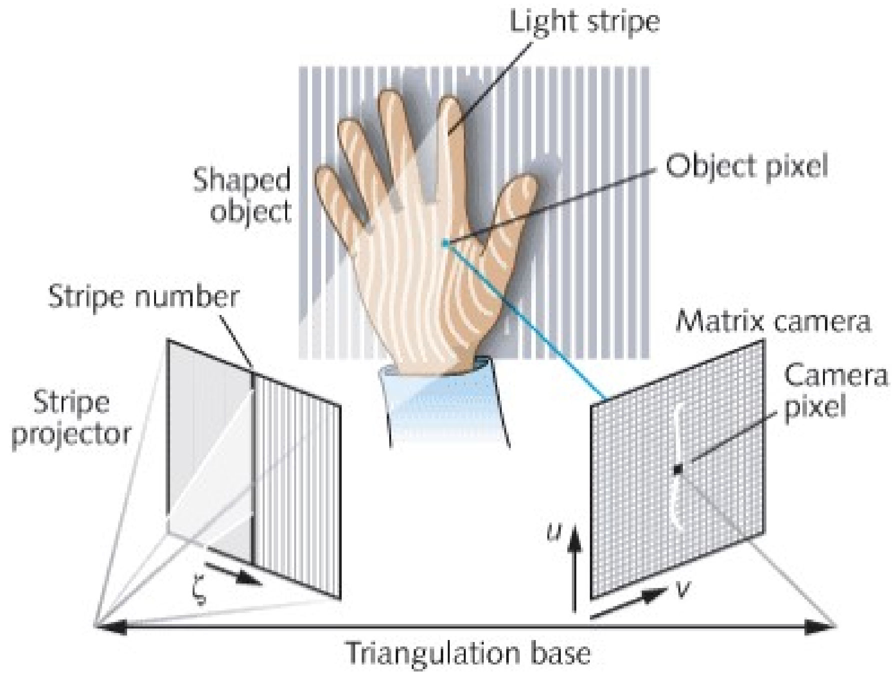
Special case: **parallel cameras** – epipolar lines are parallel and aligned with rows

Structured light method

- Replace one camera with a light strip projector
- Calculate the shape by how the strip is distorted.
- Same depth disparity equation as for 2 cameras



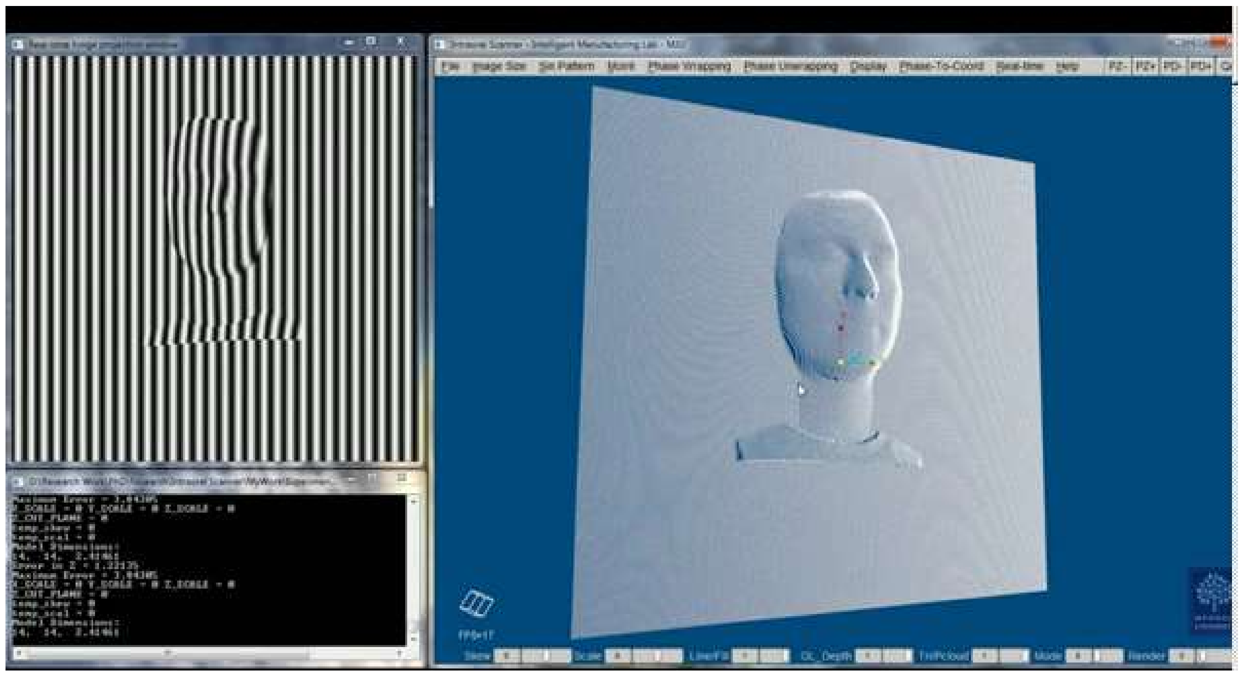
Machine vision setup



http://www.laserfocusworld.com/articles/2011/01/lasers-bring-gesture-recognition-to-the-home.html

Real time Virtual 3D Scanner - Structured Light Technology

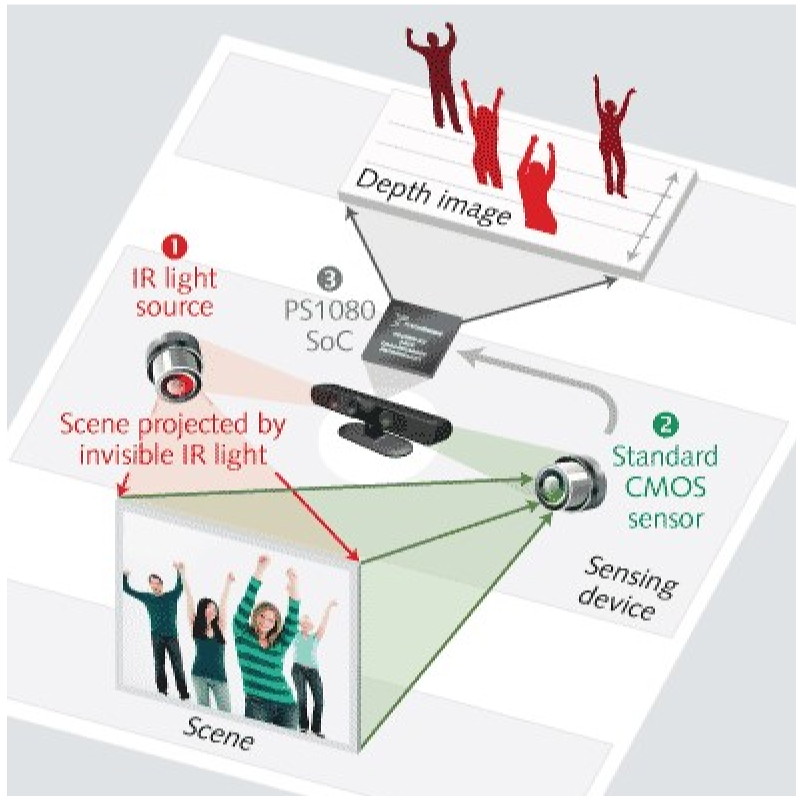
- Demo



http://www.youtube.com/watch?v=a6pgzNUjh_s

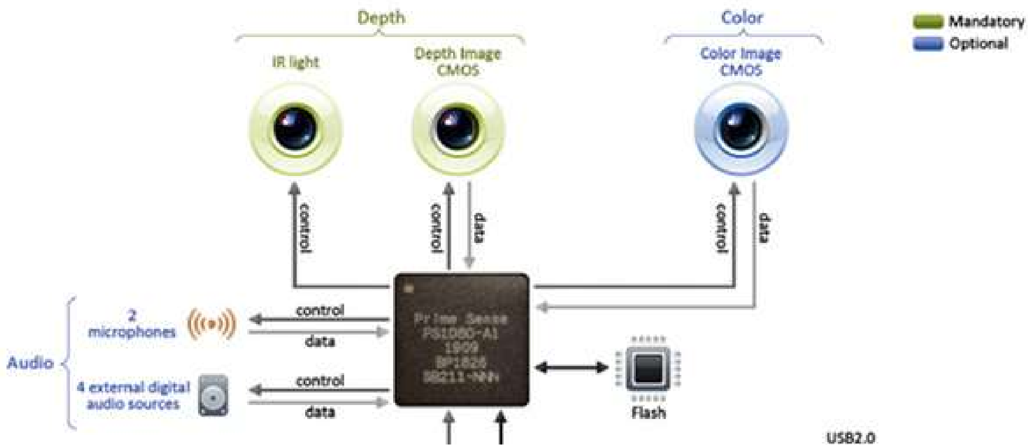
Kinect

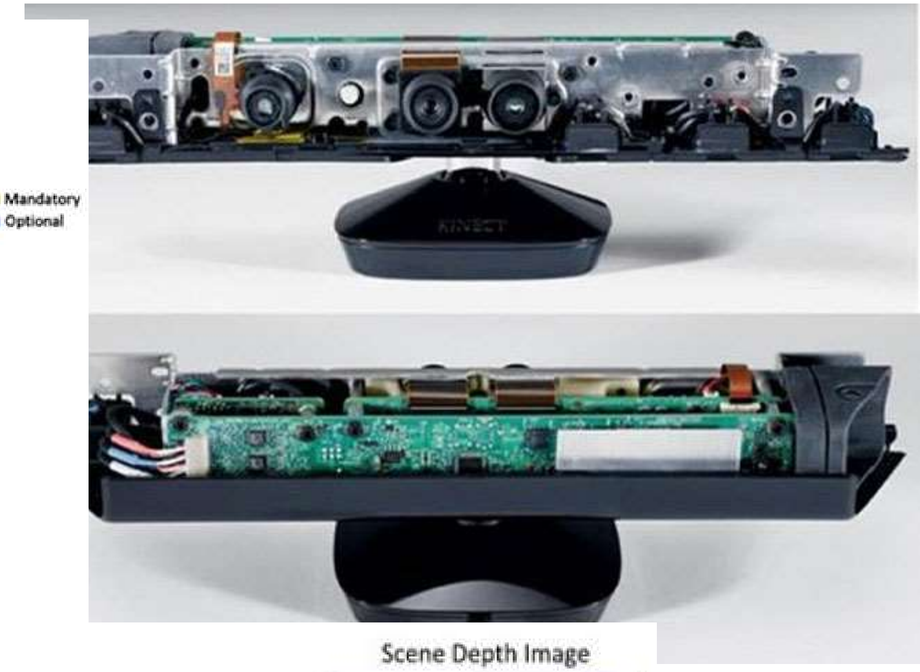
- Another structure light method
- Use dots rather than strips

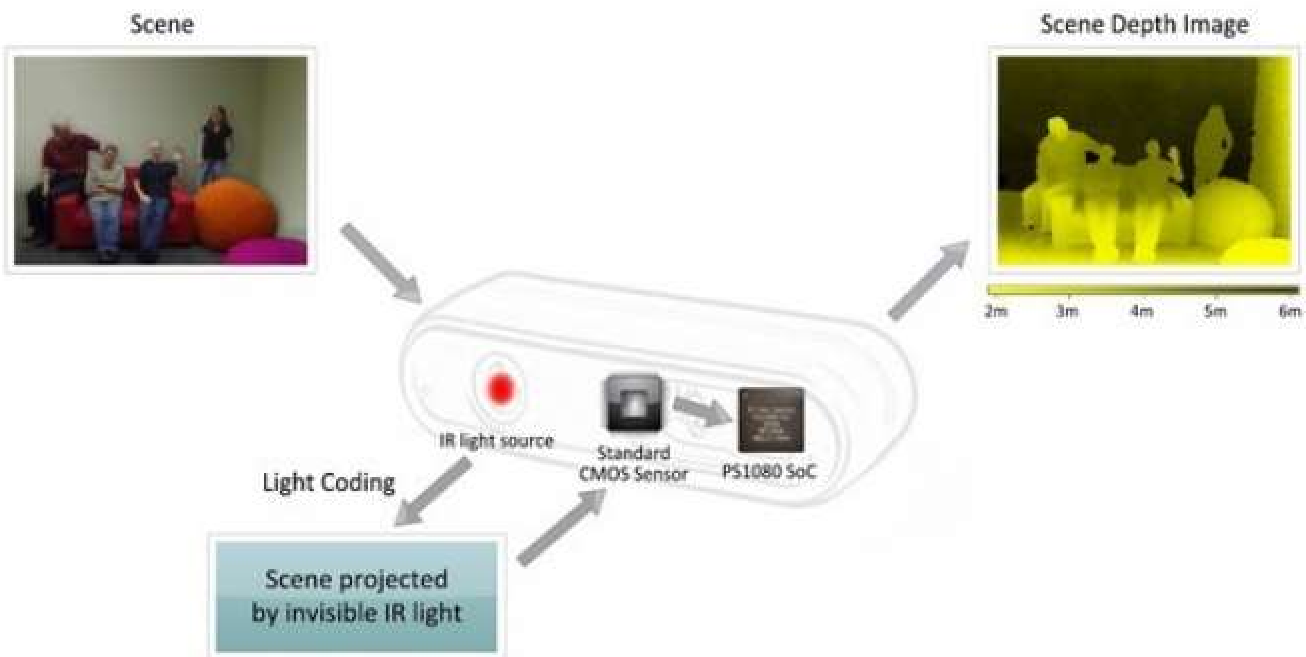


http://www.laserfocusworld.com/articles/2011/01/lasers-bring-gesture-recognition-to-the-home.html

Kinect Hardware







See the IR-dots emitted by KINECT

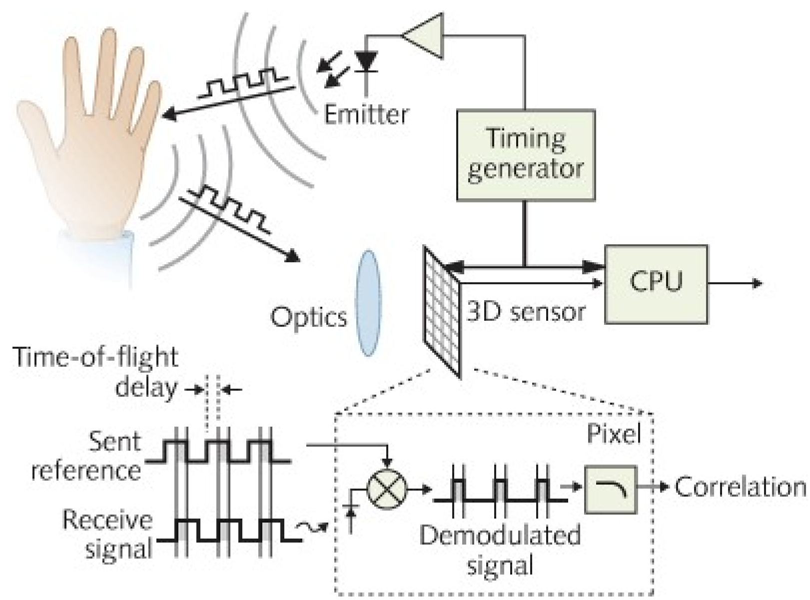
-

http://www.youtube.com/watch?v=-gbzXjdHfJA

http://www.youtube.com/watch?v=dTKlNGSH9Po&feature=related

Time of flight laser method

- Send the IR-laser light to different directions and sense how each beam is delayed.
- Use the delay to calculate the distance of the object point

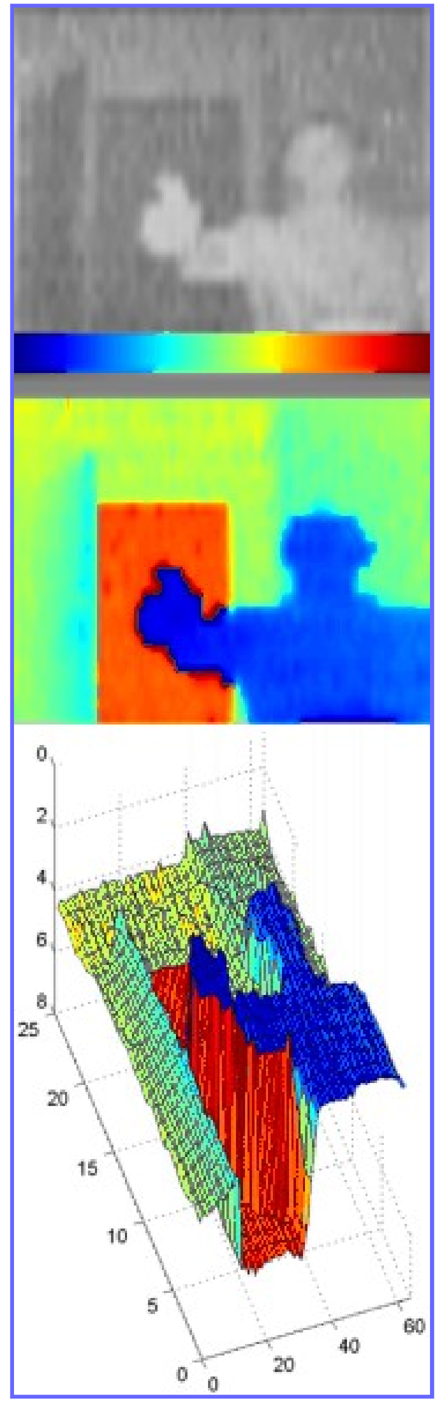


http://www.laserfocusworld.com/articles/2011/01/lasers-bring-gesture-recognition-to-the-home.html

Time of flight laser camera

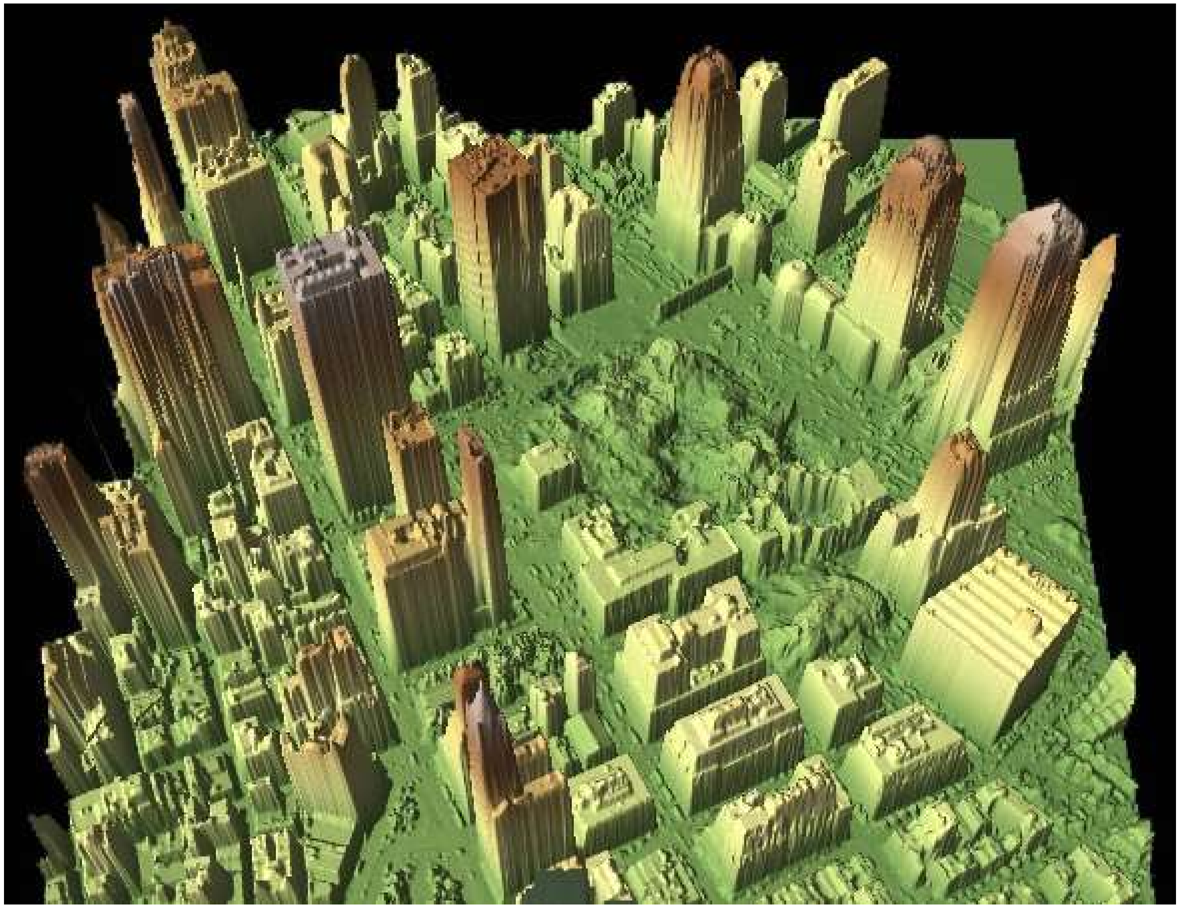
http://www.swissranger.ch/index.php

http://www.advancedscientificconcepts.com



LIDAR light detection and ranging scanner

http://www.youtube.com/watch?v=MuwQTc8KK44

- 

Leica terrestrial lidar (light detection and ranging) scanner

http://hodcivil.edublogs.org/2011/11/06/lidar-%E2%80%93-light-detection-and-ranging/
http://commons.wikimedia.org/wiki/File:Lidar_P1270901.jpg

3D Laser Scanning - Underground Mine Mapping

- Demo

http://www.youtube.com/watch?v=BZbvz8fePeQ

Motion capture for film production (MOCAP)

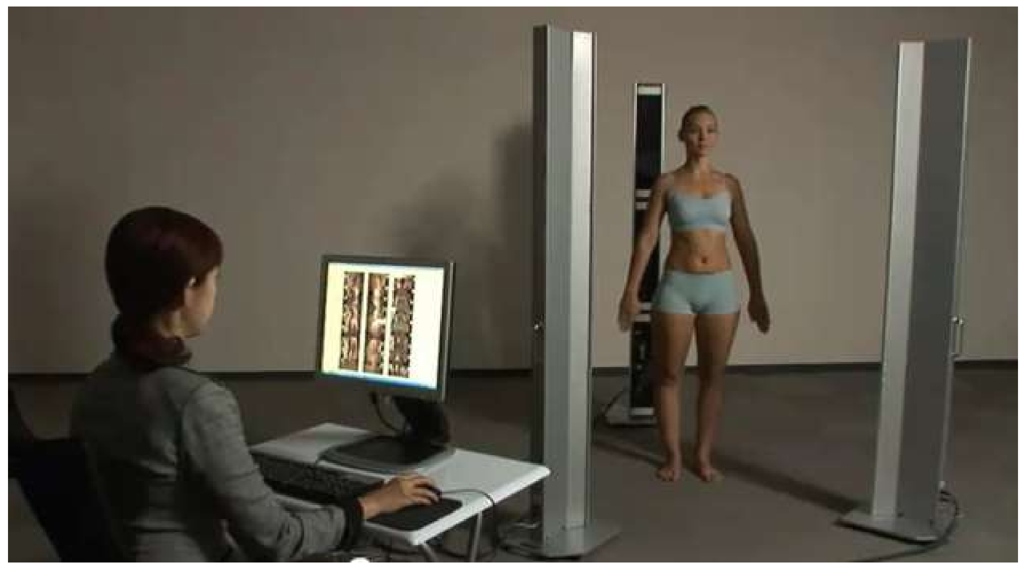
•

IR light
emitter
and
camera

http://www.youtube.com/watch?v=IxJrhnynlN8

- http://upload.wikimedia.org/wikipedia/commons/7/73/MotionCapture.jpg
- http://www.naturalpoint.com/optitrack/products/s250e/indepth.html

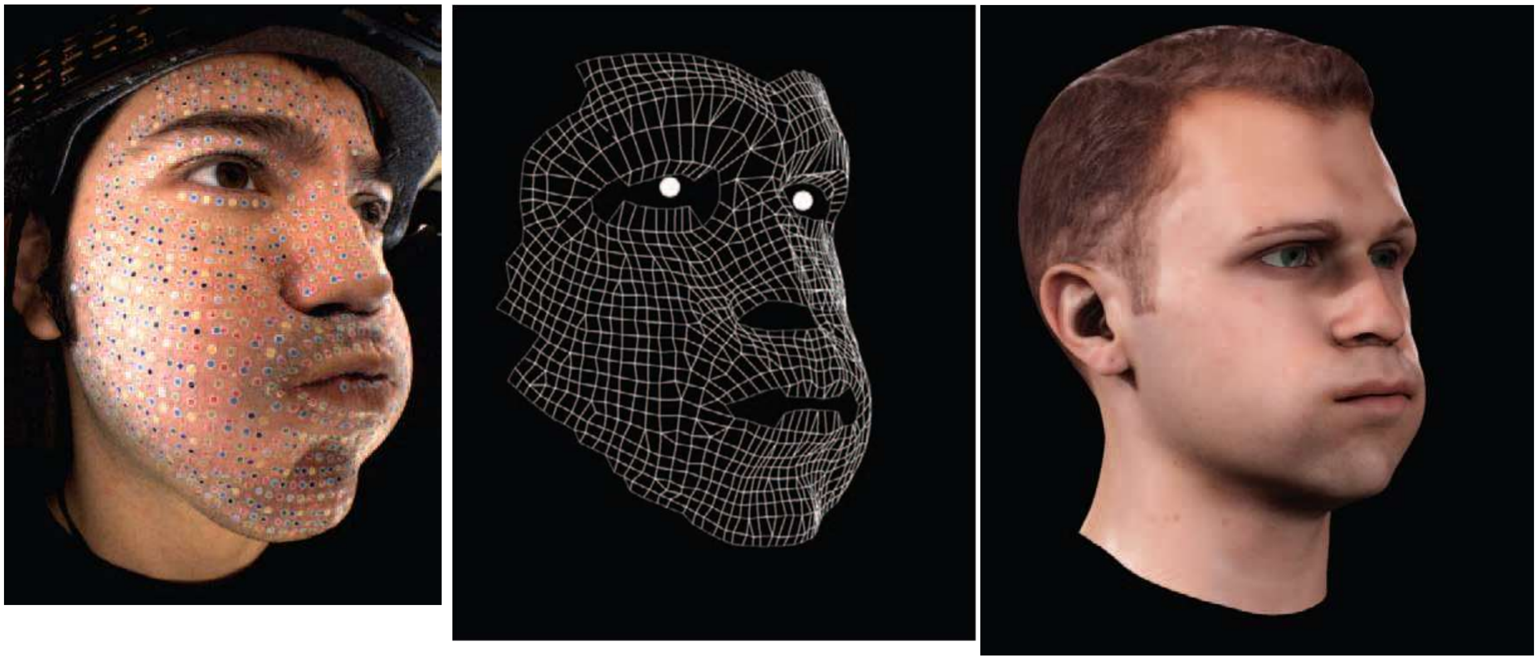
3D body scanner



http://www.youtube.com/watch?v=86hN0x9RycM

http://www.cyberware.com/products/scanners/ps.html
http://www.cyberware.com/products/scanners/wbx.html

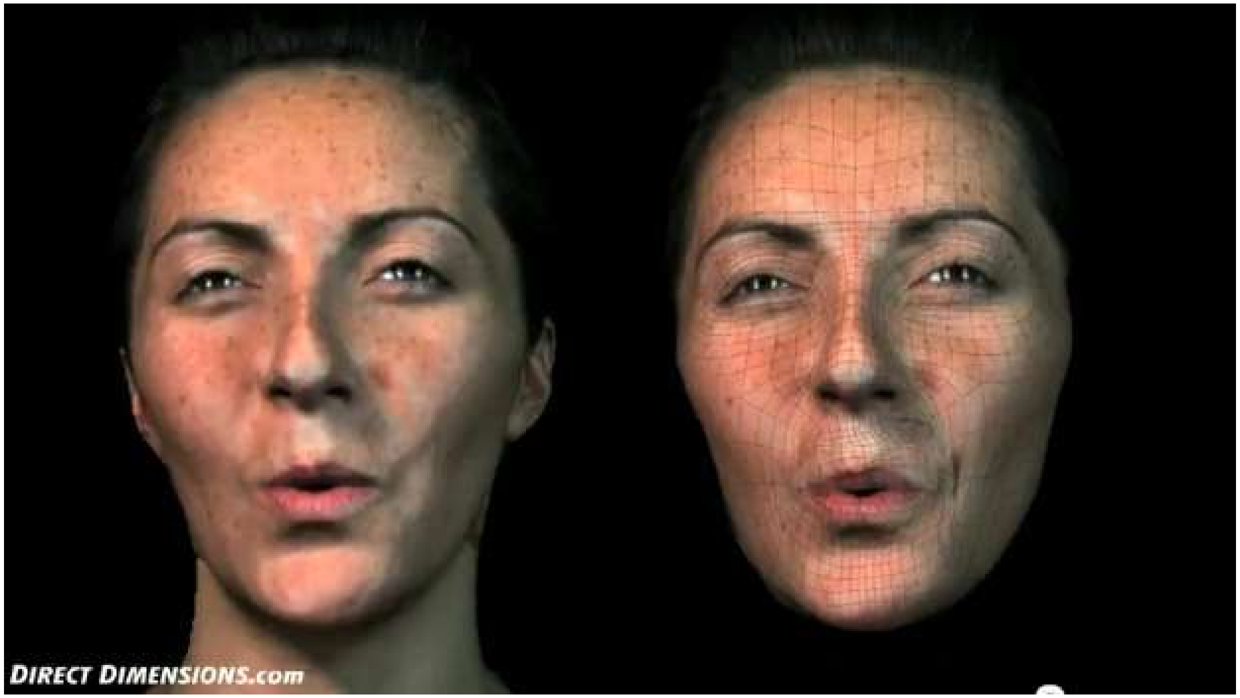
3-D Face capture

- 

http://www.youtube.com/watch?v=-TTR0JrocsI&feature=related

http://www.captivemotion.com/products/

Dimensional Imaging 4D Video Face Capture with Textures

- 

http://www.youtube.com/watch?v=XtTN7tWaXTM&feature=related

Dimensional Imaging 4D Video Face Capture with Textures

Vision and range sensing

- The past:
  - Mobile robots used ring of ultrasound or IR distance sensors for obstacle avoidance or crude navigation
  - Robot arms used VGA cameras to track a few points
- Now:
  - Full camera pose tracking and 3D scene reconstruction possible with inexpensive cameras and processing (e.g. PTAM on RaspberryPI+cam = $50 and 20gram
  - Can track hundreds of interest points for image based visual control.
- The next years
  - Active range sensing RGBD with Kinect growing in popularity indoors
  - Passive camera vision still important. Especially outdoors and on UAV.