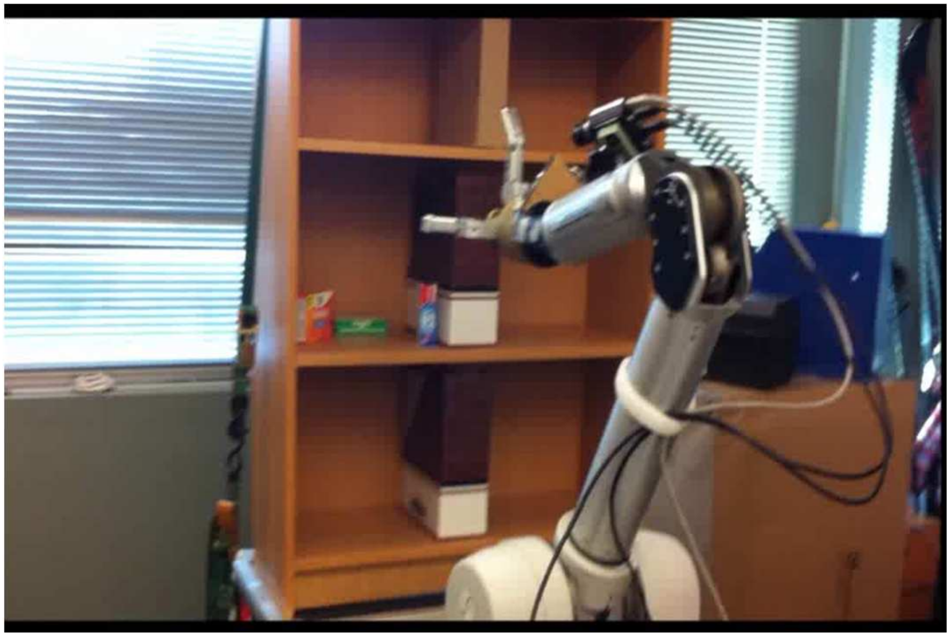
Multiview geometry

Build 3D models from images

Carry out manipulation tasks

http://webdocs.cs.ualberta.ca/~vis/ibmr/



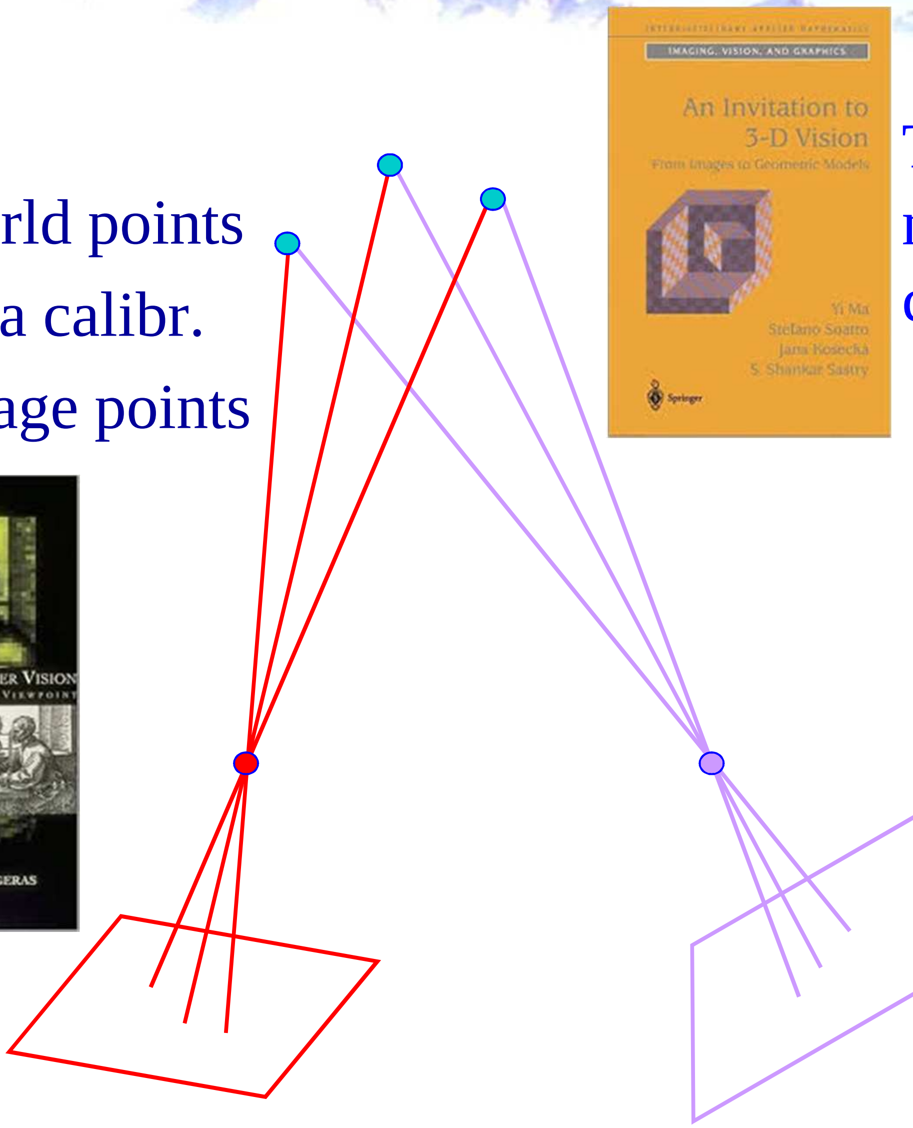
Training for Amazon manipulation challenge

And many other usages…

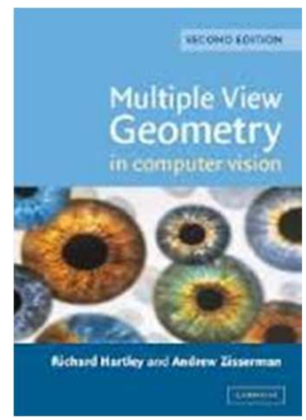
Multi-view Geometry

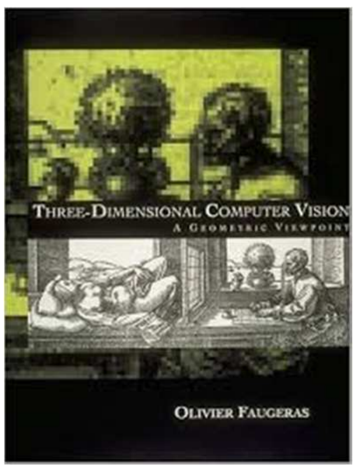
Relates

- 3D world points
- Camera calibr.
- 2D image points



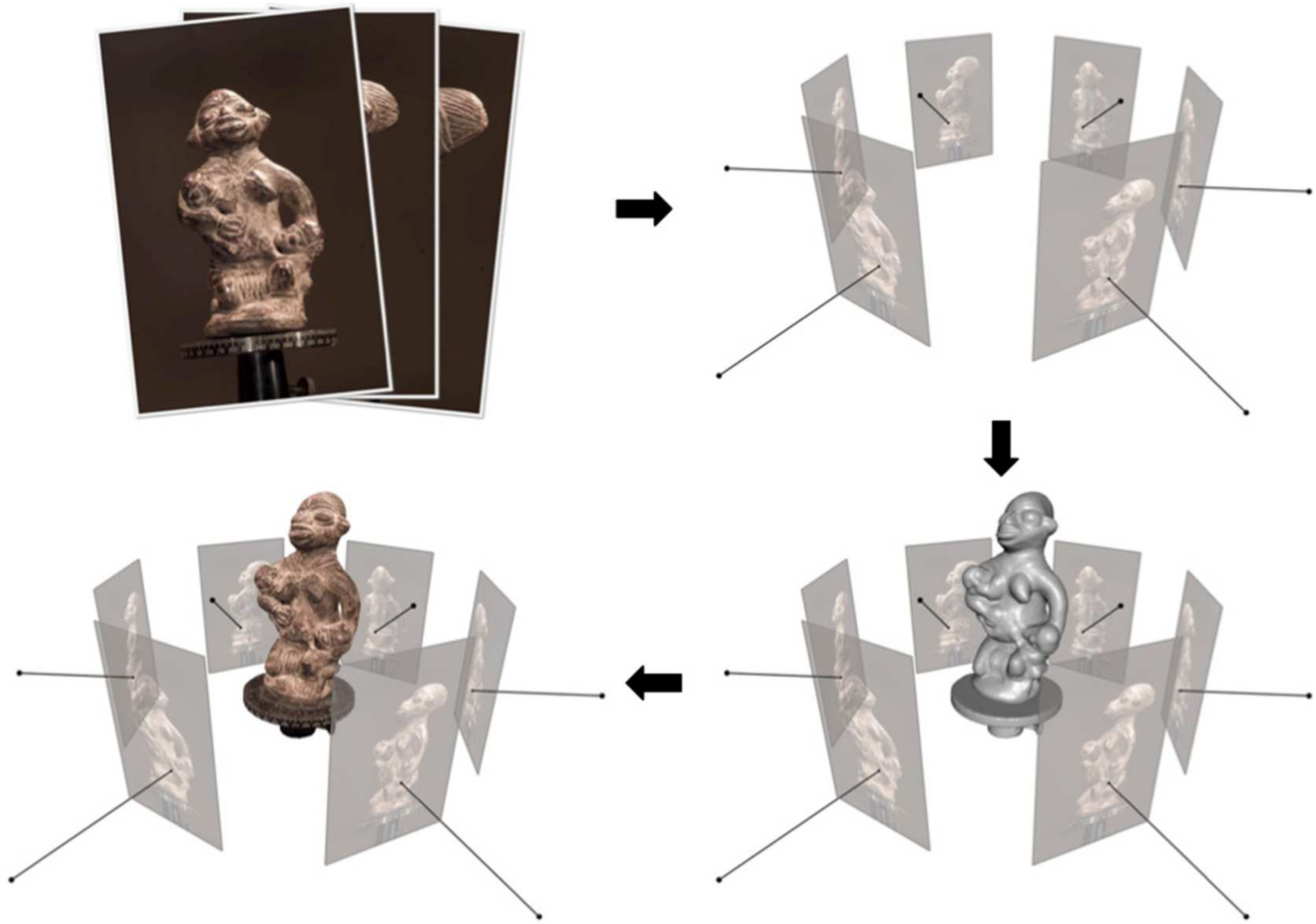
The reconstruction of 3D models of objects from a collection of 2D images



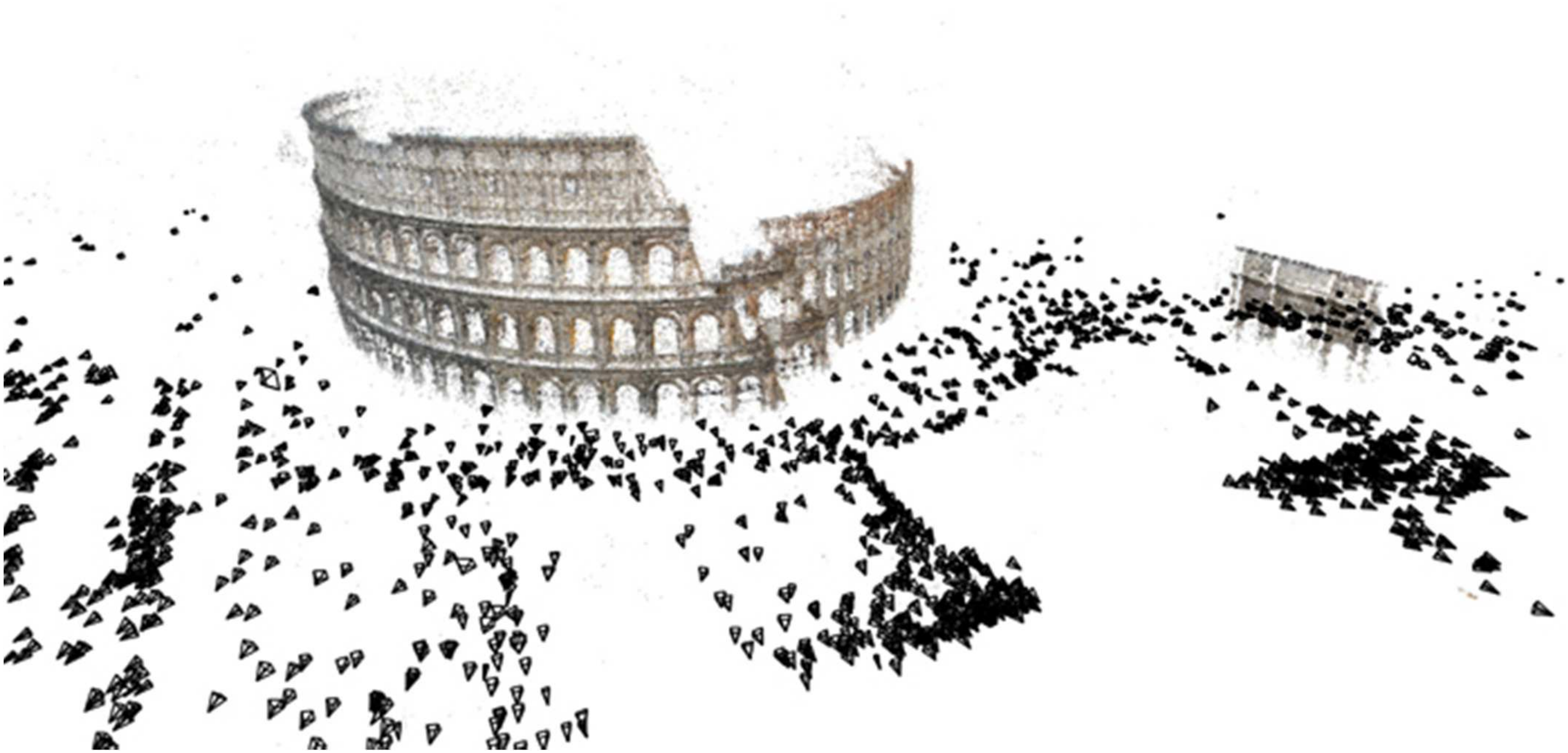


Multi-view Geometry

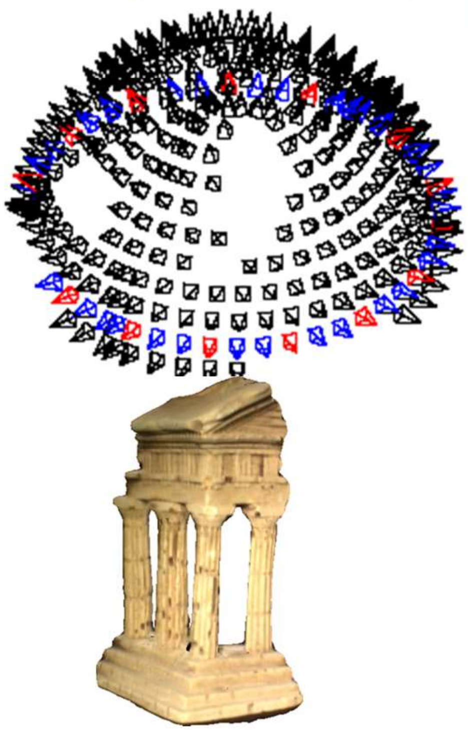
Typical processing pipeline (C. Hernandez MVS tutorial)

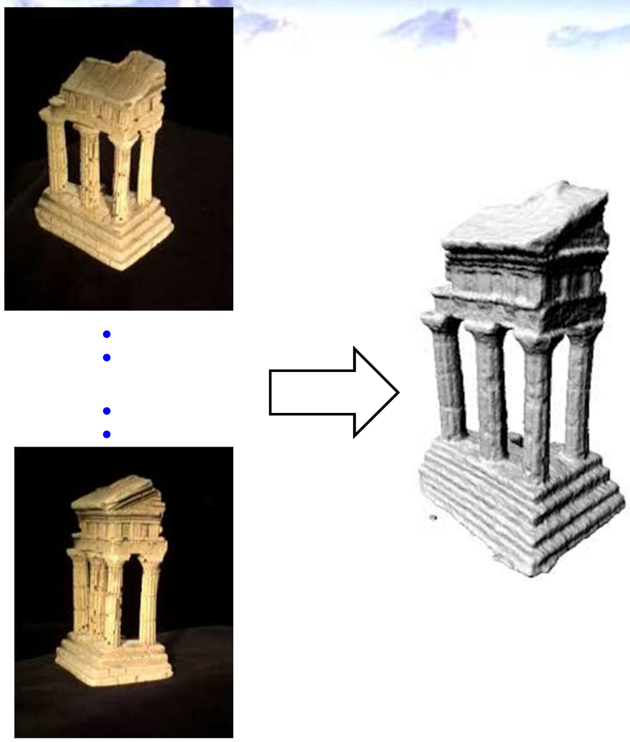


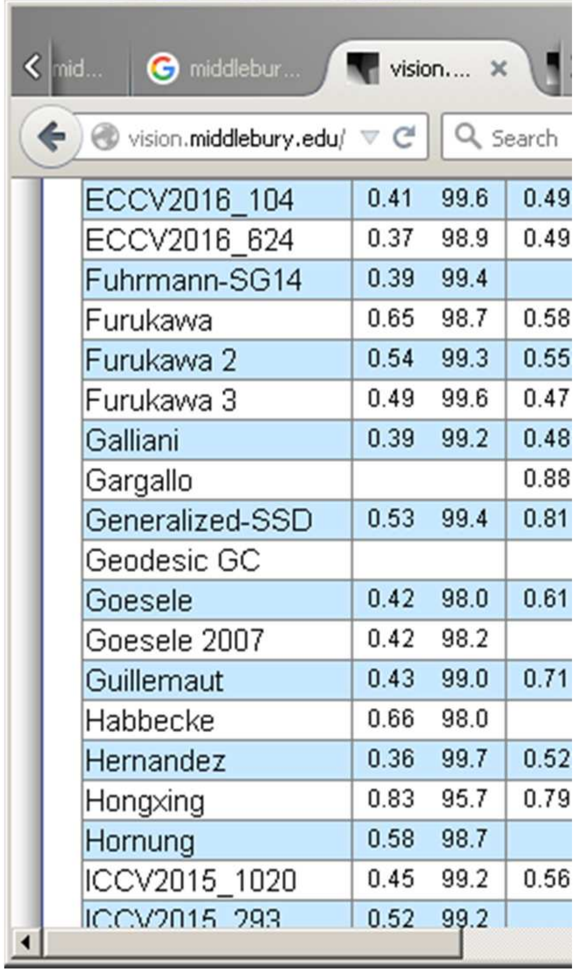
Multi-view Geometry



Middlebury multi-view benchmark







ECCV2016_104	0.41	99.6	0.49
ECCV2016_624	0.37	98.9	0.49
Fuhrmann-SG14	0.39	99.4	
Furukawa	0.65	98.7	0.58
Furukawa 2	0.54	99.3	0.55
Furukawa 3	0.49	99.6	0.47
Galliani	0.39	99.2	0.48
Gargallo			0.88
Generalized-SSD	0.53	99.4	0.81
Geodesic GC			
Goesele	0.42	98.0	0.61
Goesele 2007	0.42	98.2	
Guillemaut	0.43	99.0	0.71
Habbecke	0.66	98.0	
Hernandez	0.36	99.7	0.52
Hongxing	0.83	95.7	0.79
Hornung	0.58	98.7	
ICCV2015_1020	0.45	99.2	0.56
ICCV2015_293	0.52	99.2	

A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms, Seitz, Curless, Diebel, Szeliski
CVPR 2006, vol. 1, pages 519-526. Cited by 1479

Calibration accuracy on these datasets appears to be on the order of a pixel (a pixel spans about 1/4mm on the object).

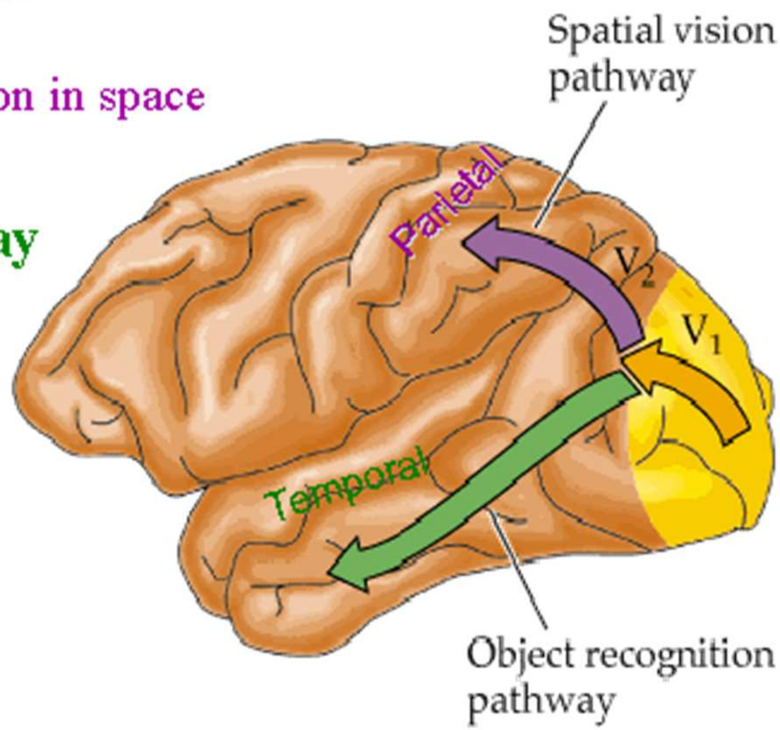
Dorsal and Ventral Pathways Where/What or Action/Perception?

Dorsal (magno) Pathway

- to parietal lobe
- spatial vision – localization in space
- “WHERE”

Ventral (parvo) Pathway

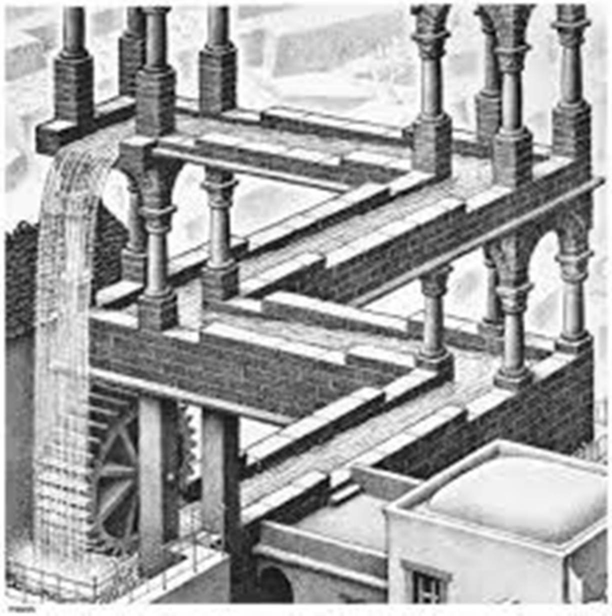
- to temporal lobe
- object recognition
- “WHAT”



- Humans don’t internalize detailed 3D maps!
- Use external world as map. (Hayhoe, Pelz, Rensink, Goodale etc)

Uses of partial information from 2D-3D camera geometry

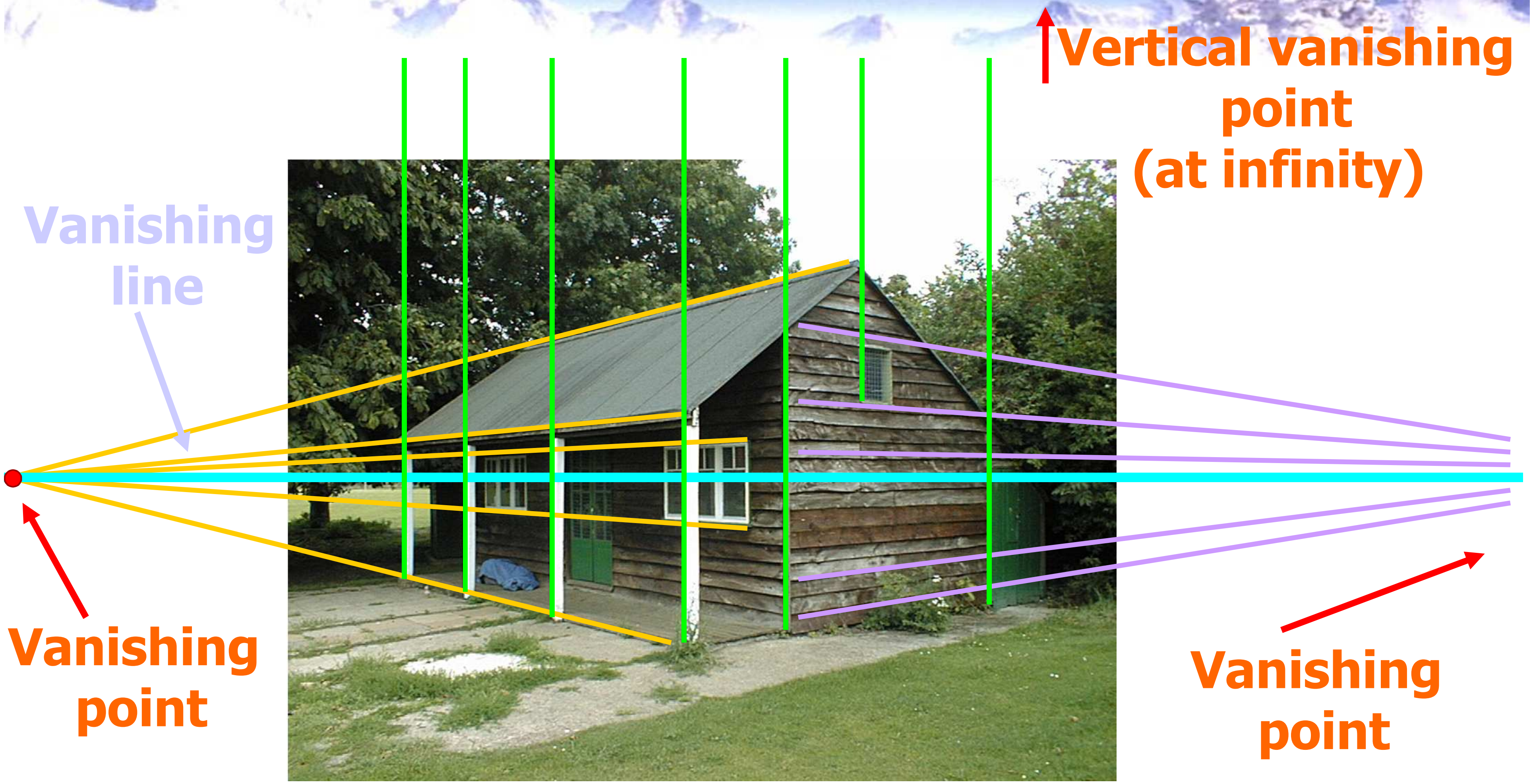
- Measurements in single images
- Visual constraints: verify alignments, detect impossible configurations (Escher paintings)
- Visual servoing
- Video tracking
- Rendering
- … many more



Why? Compact, accurate

Few relative alignments vs. complete global geometry

Geometric Cues



Single View Metrology

Estimating Height

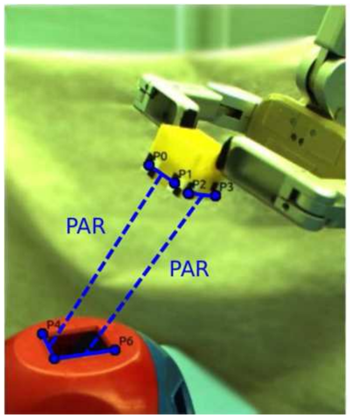
- The distance $\| t_r - b_r \|$ is known
- Used to estimate the height of the man in the scene

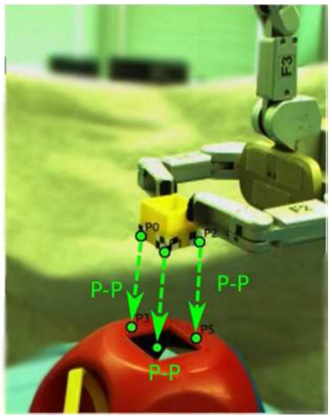
Single View Metrology (559 citations)

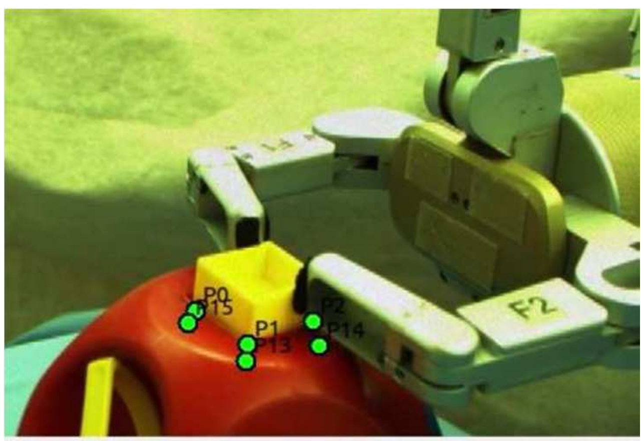
A. Criminisi, I Reid, A Zisserman
International Journal of Computer Vision 40 (2), 123-148, 2000

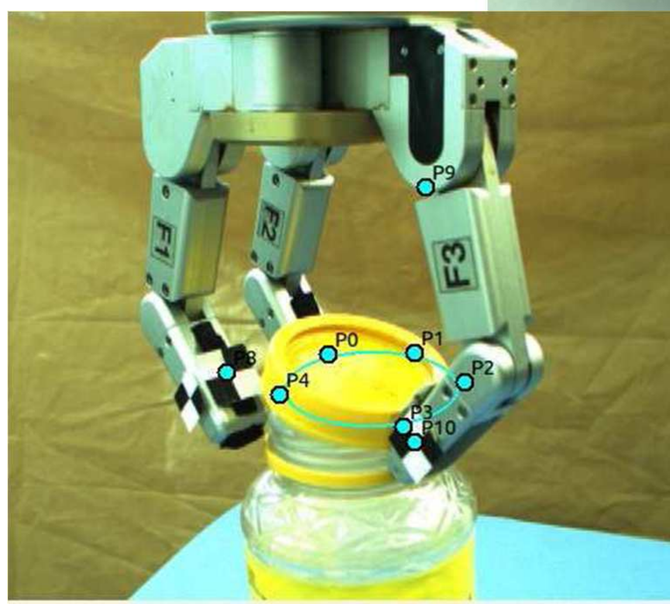
Geometry for hand-eye coordination

Image-Based Visual Servoing (IBVS)

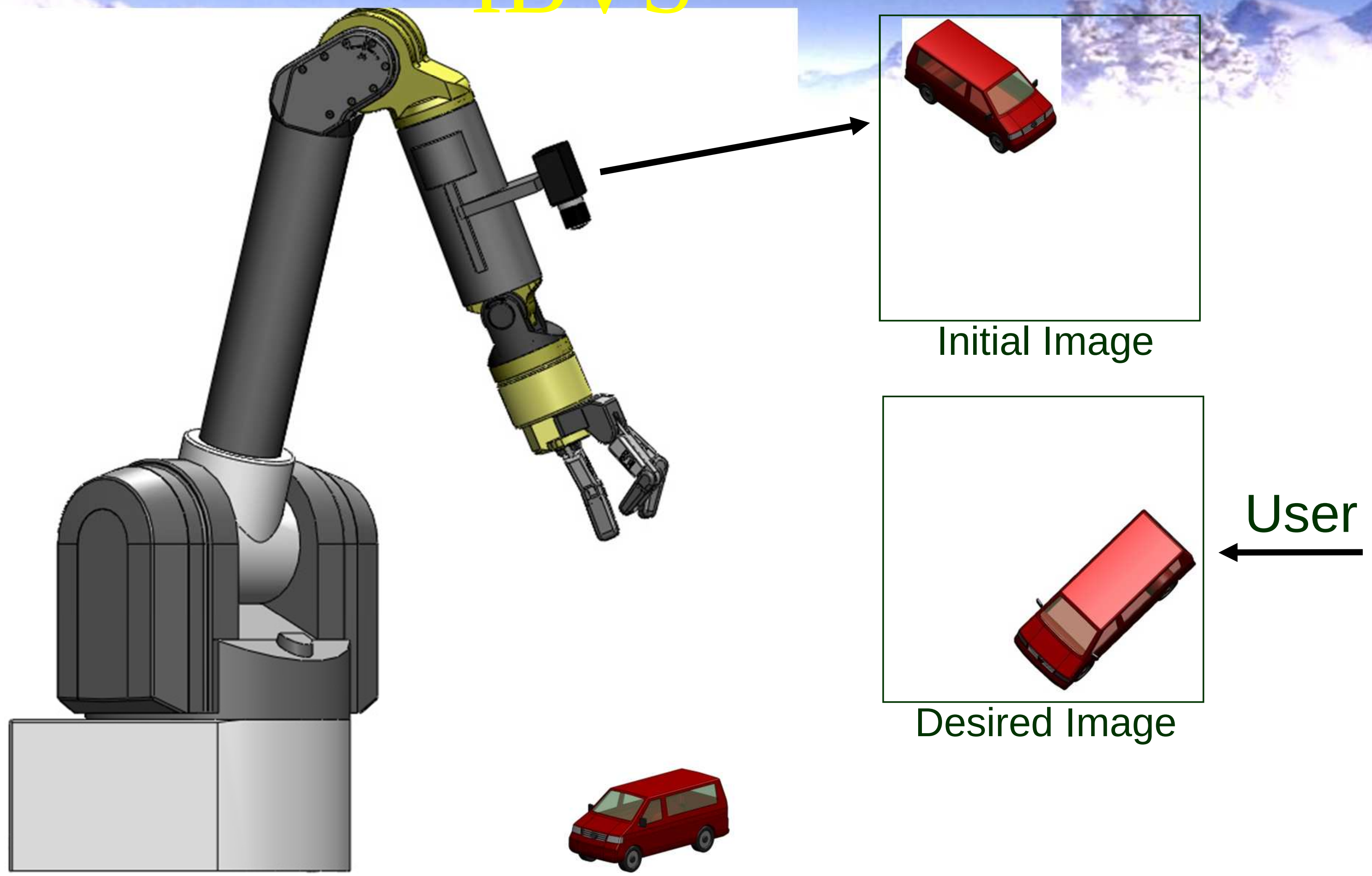




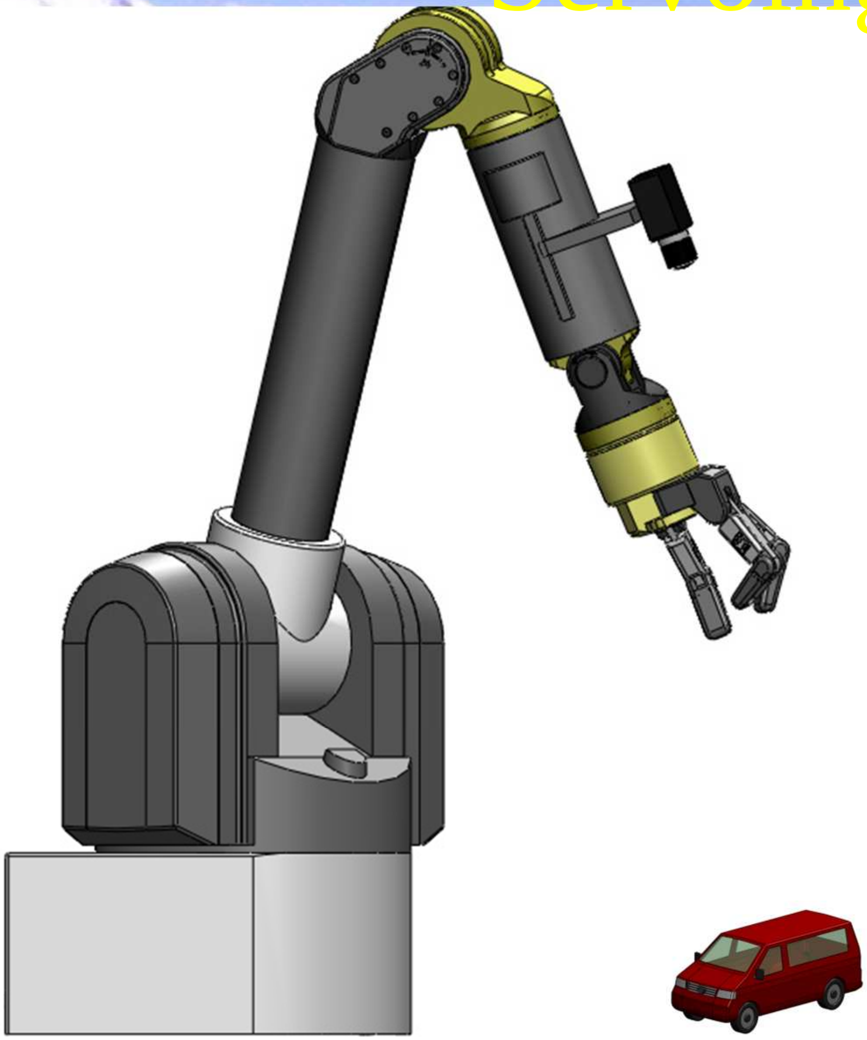


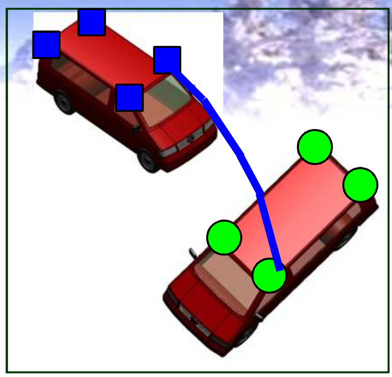


Intro to Image-based Visual Servoing IBVS



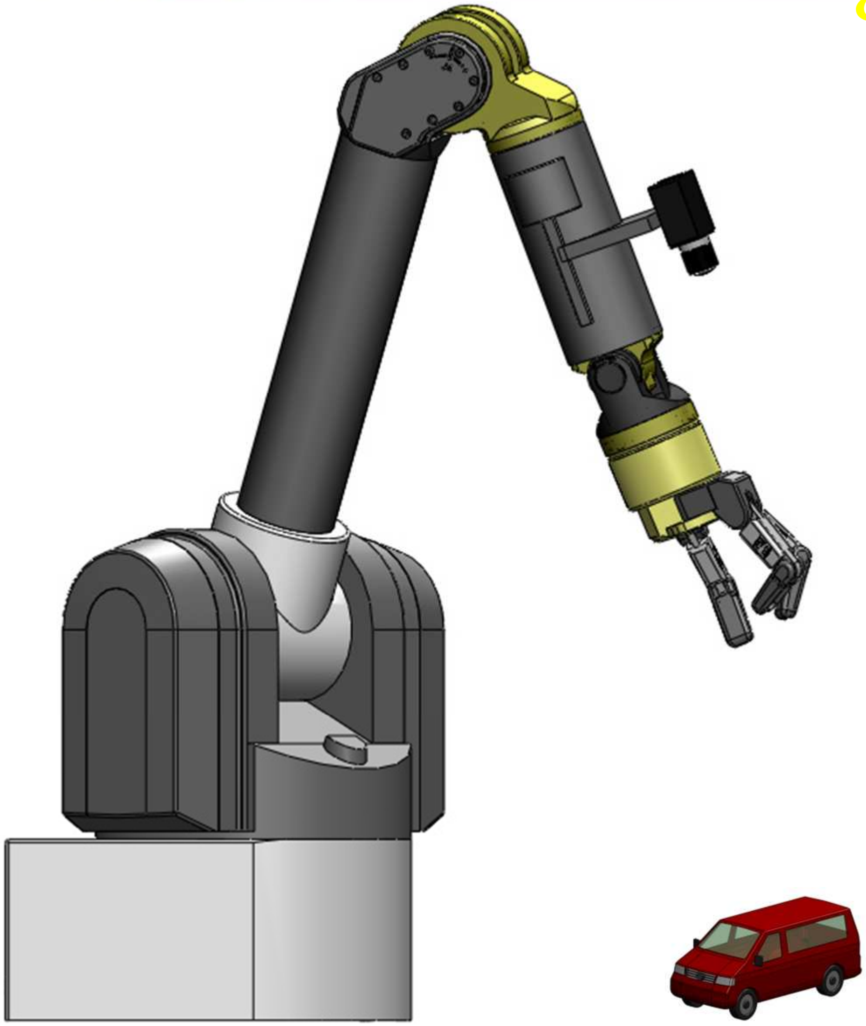
Vision-Based Control (Visual Servoing)

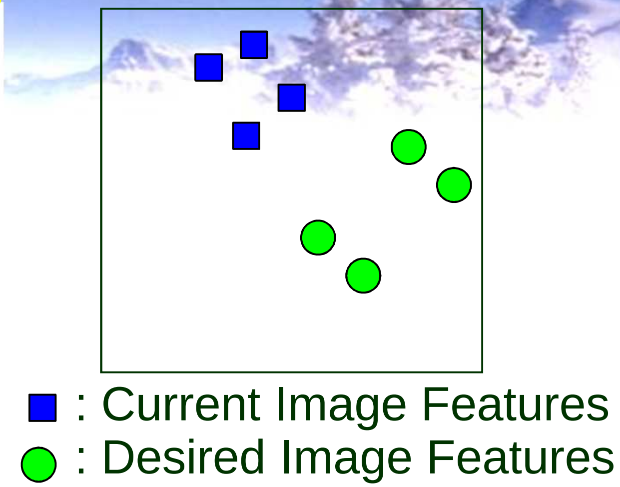




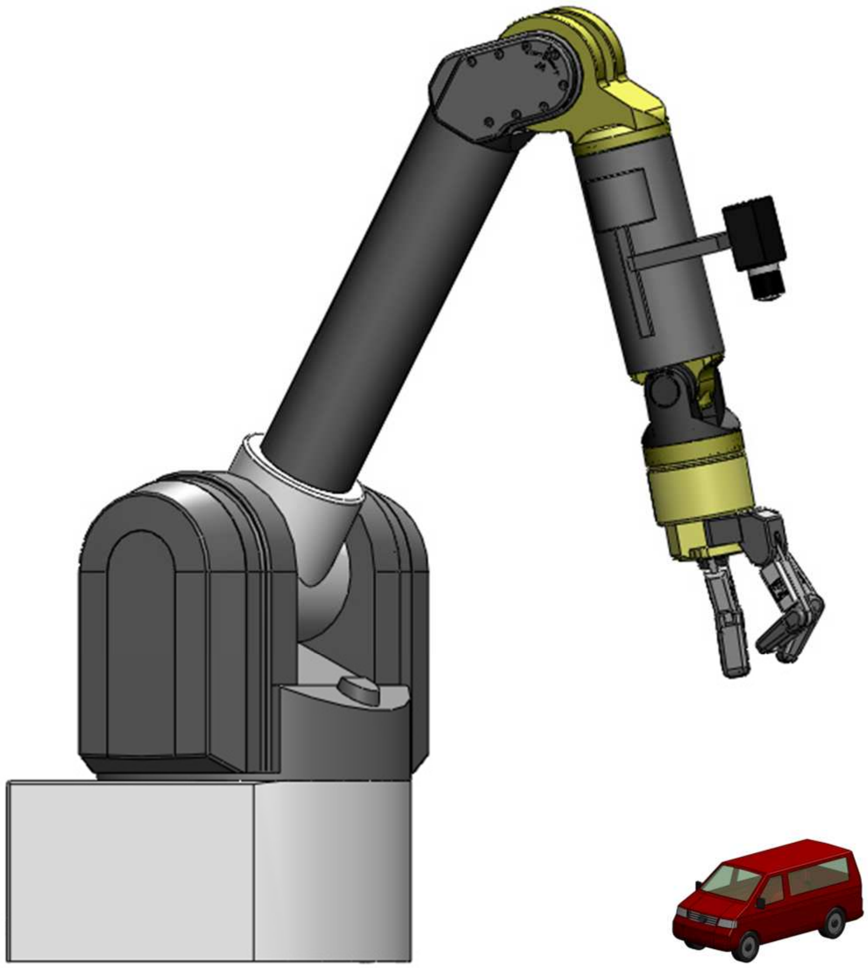
: Current Image Features
: Desired Image Features

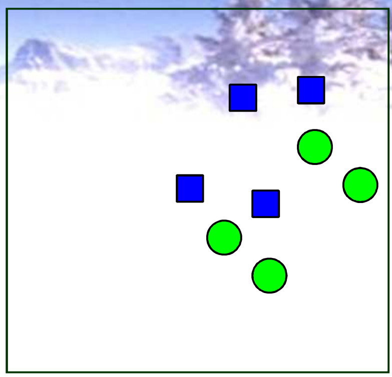
Vision-Based Control (Visual Servoing)





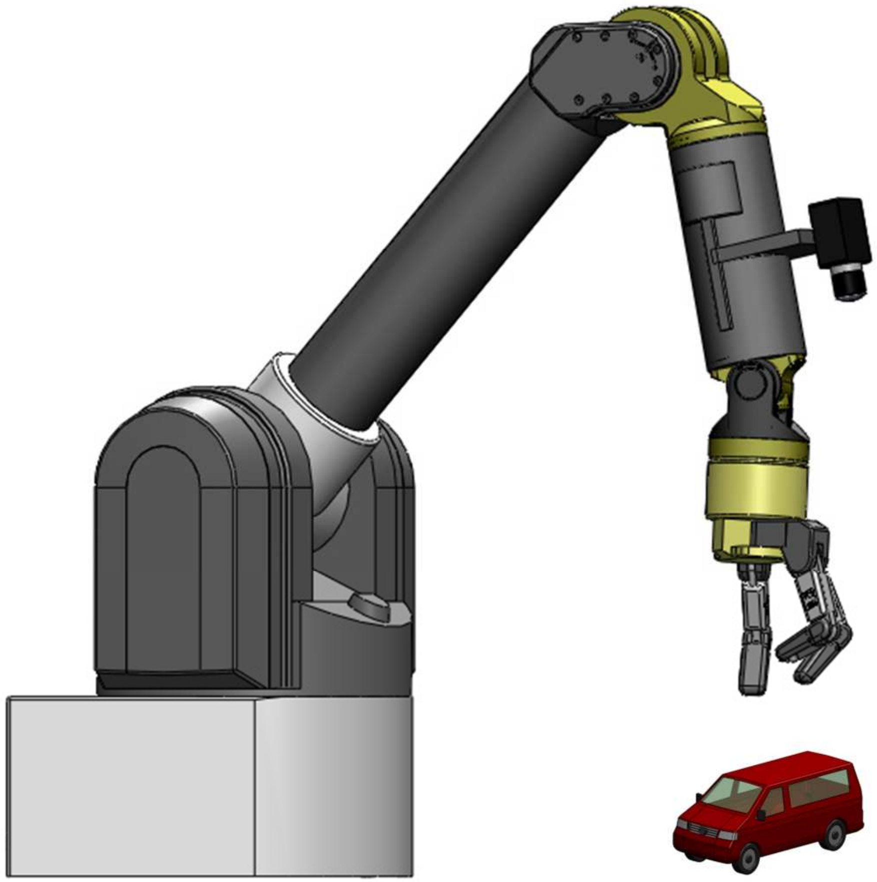
Vision-Based Control (Visual Servoing)

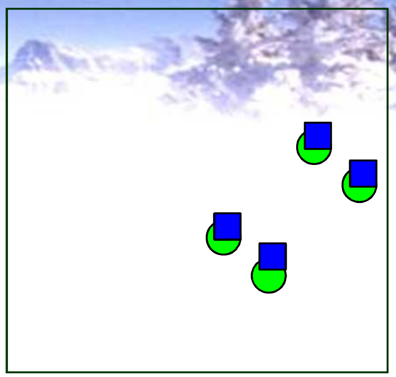




■ : Current Image Features
● : Desired Image Features

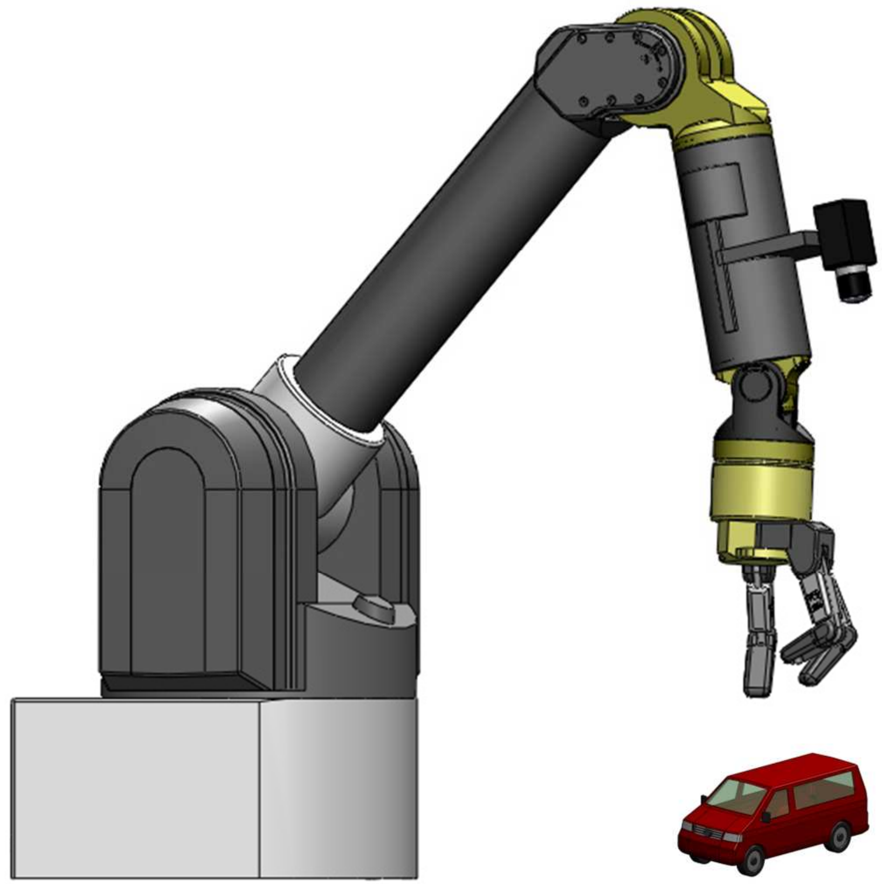
Vision-Based Control (Visual Servoing)

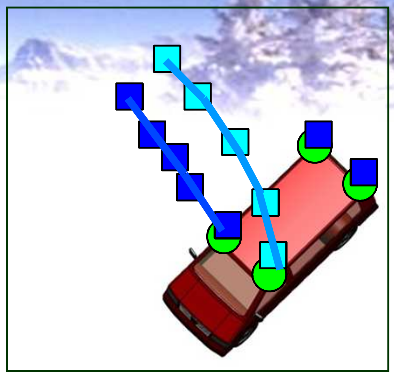




■ : Current Image Features
● : Desired Image Features

Vision-Based Control (Visual Servoing)

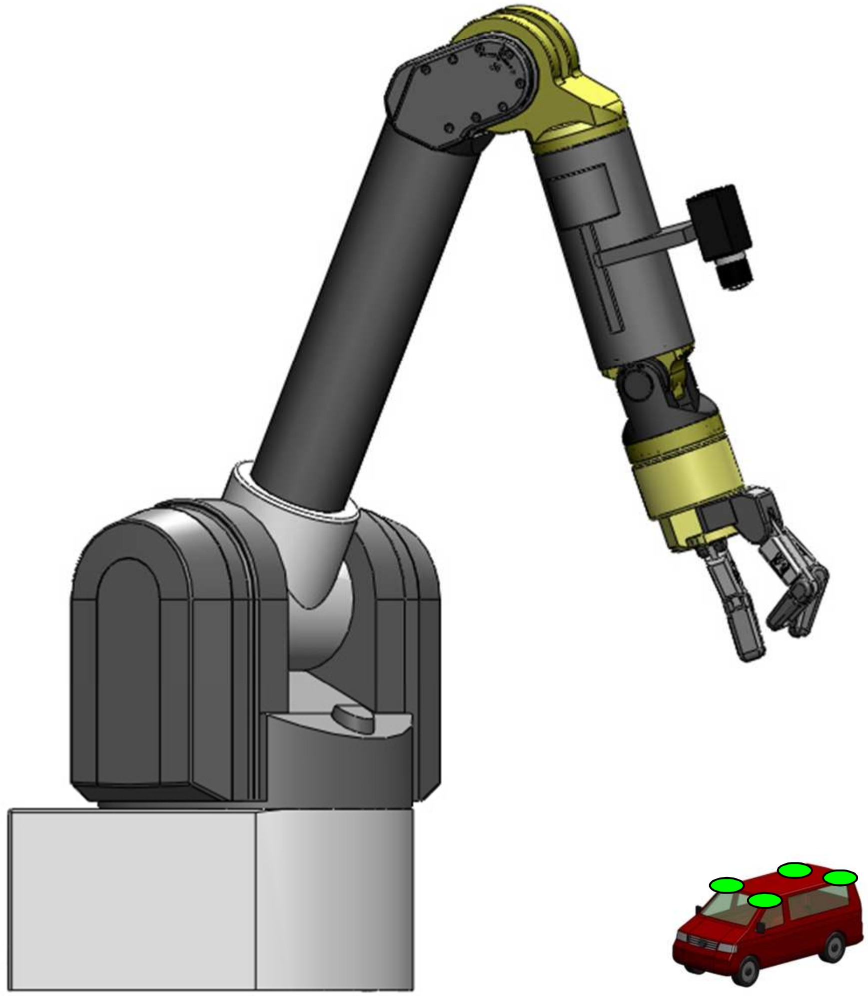


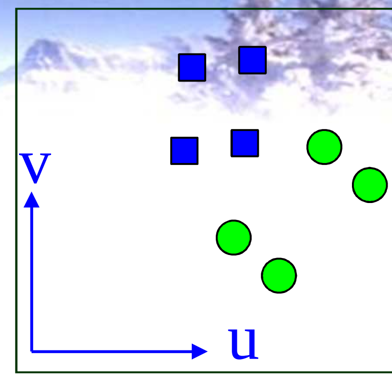


: Current Image Features

: Desired Image Features

u,v Image-Space Error





■ : Current Image Features

● : Desired Image Features

$$\mathbf{E} = [\,\bullet \;-\; \blacksquare\,]$$

One point

$$\mathbf{E} = [\mathbf{y}_0 - \mathbf{y}^*]$$

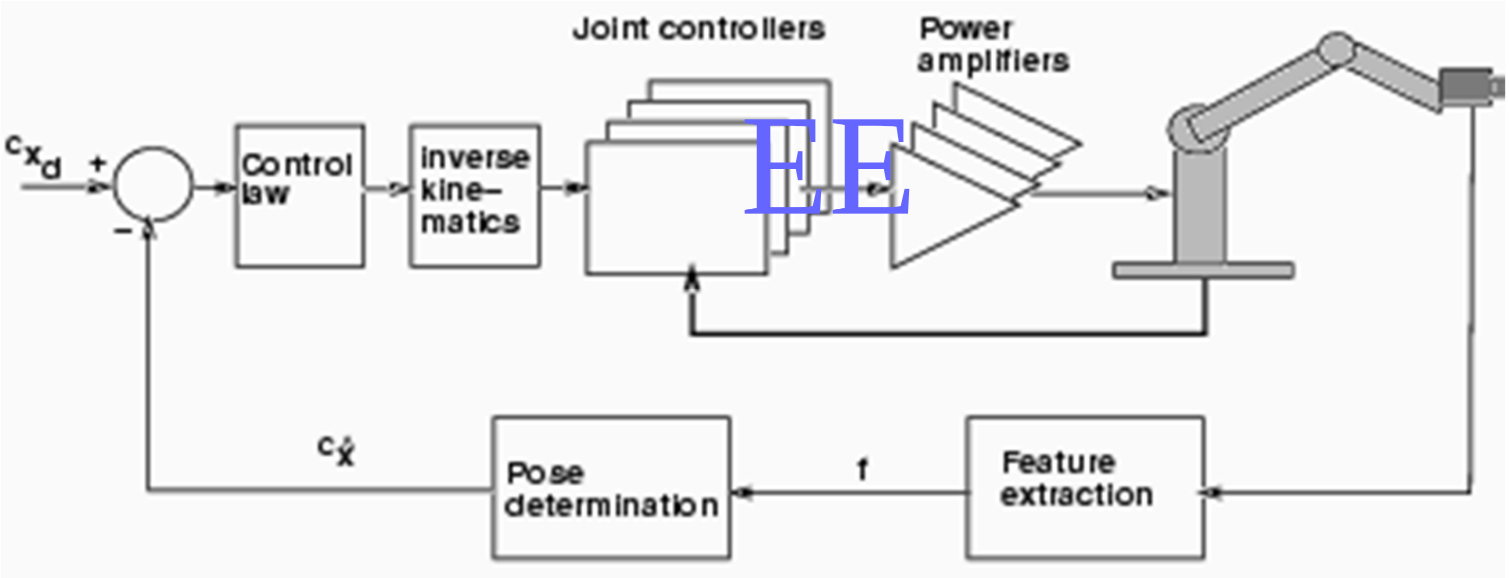
Pixel coord

$$\mathbf{E} = \begin{bmatrix} y_u \\ y_v \end{bmatrix} - \begin{bmatrix} y_u \\ y_v \end{bmatrix}^*$$

Many points

$$\mathbf{E} = \begin{bmatrix} y_1 \\ \vdots \\ y_8 \end{bmatrix}^* - \begin{bmatrix} y_1 \\ \vdots \\ y_8 \end{bmatrix}_0$$

Conventional Robotics: Motion command in Eucl. base coord



- We focus on the geometric transforms “visual-motor kinematics”

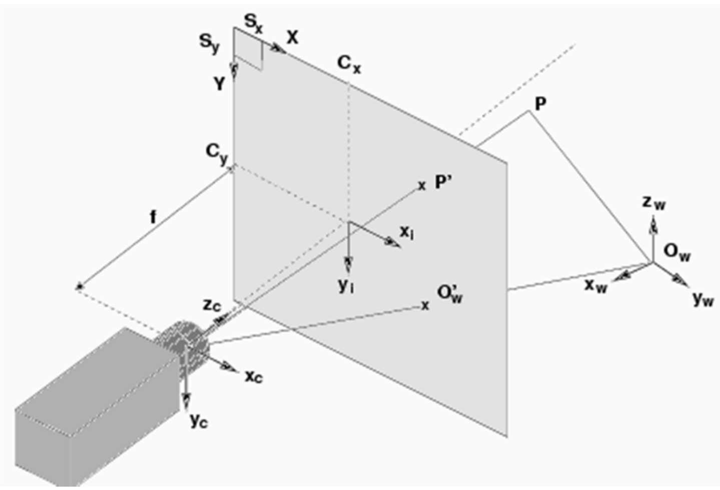
Problem: Lots of coordinate frames to calibrate

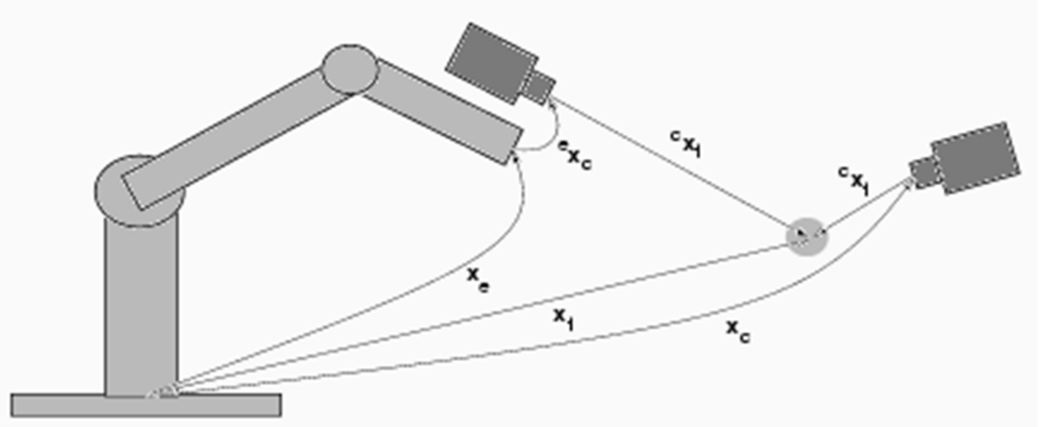
Camera

- Center of projection
- Different models

Robot and scene

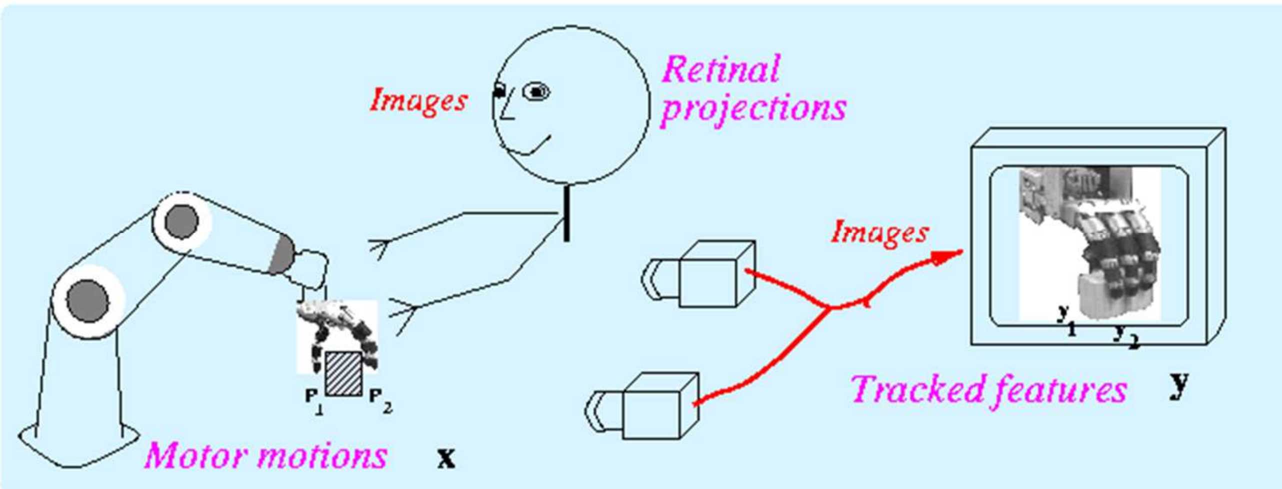
- Base frame
- End-effector frame
- Object





Use only camera coord. Uncalibrated Visual Servoing

Jagersand'94,96,00 Hosoda, Asada'94, 97



Only assume

y = f(x)

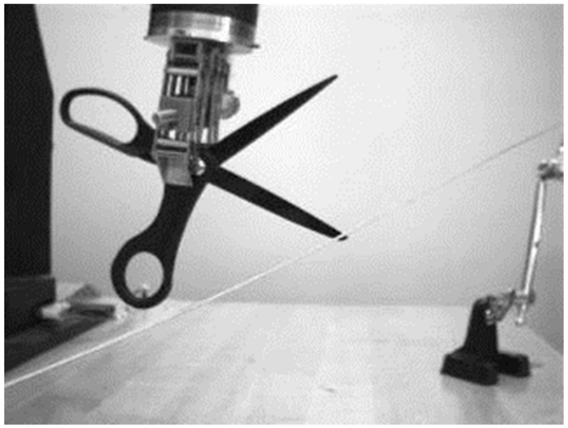
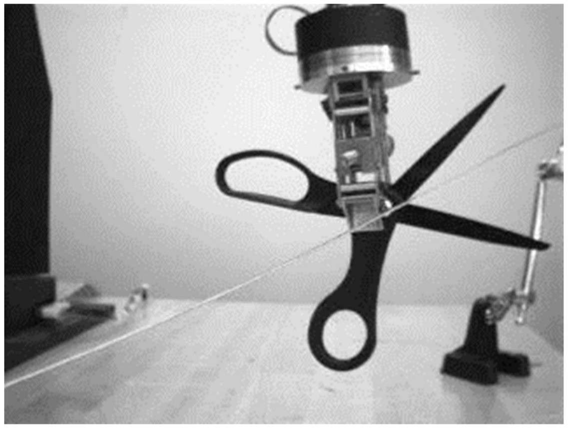
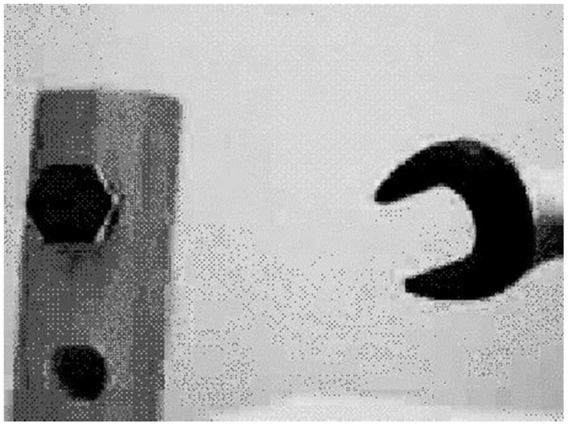
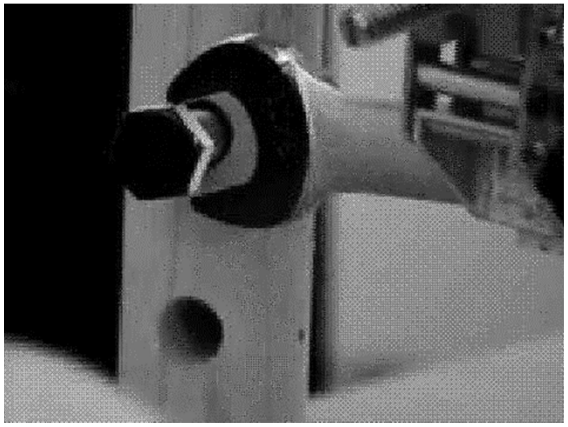
smooth

repeat

1. Solve for motion: $[\mathbf{y}^* - \mathbf{y}_k] = \mathbf{J}\Delta\mathbf{x}$
2. Move robot joints: $\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta\mathbf{x}$
3. Read actual visual move Δy
4. Update Jacobian: $\hat{J}_{k+1} = \hat{J}_k + \frac{(\Delta\mathbf{y} - \hat{J}_k\Delta\mathbf{x})\Delta\mathbf{x}^T}{\Delta\mathbf{x}^T\Delta\mathbf{x}}$

Downloadable templated library: http://ugweb.cs.ualberta.ca/~vis/ros-uvs/

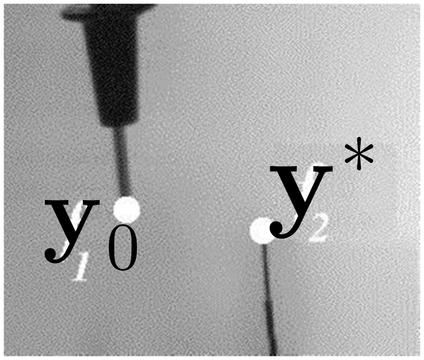
How to specify a visual task?

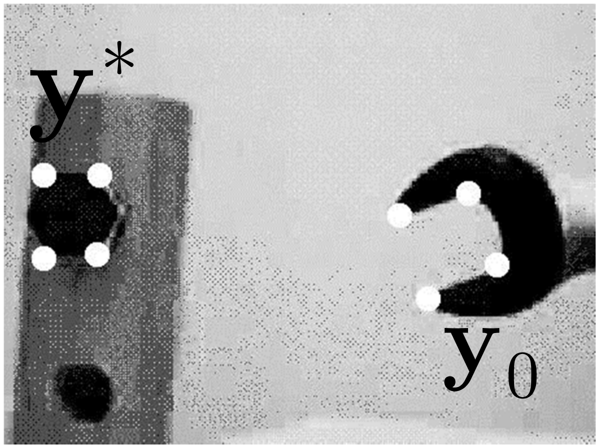
Visual specifications

- Point to Point task "error":

$$\mathbf{E} = [\mathbf{y}^* - \mathbf{y}_0]$$



$$\mathbf{E} = \begin{bmatrix} y_1 \\ \vdots \\ y_{16} \end{bmatrix}^* - \begin{bmatrix} y_1 \\ \vdots \\ y_{16} \end{bmatrix}_0$$



Why 16 elements?

Visual specifications 2

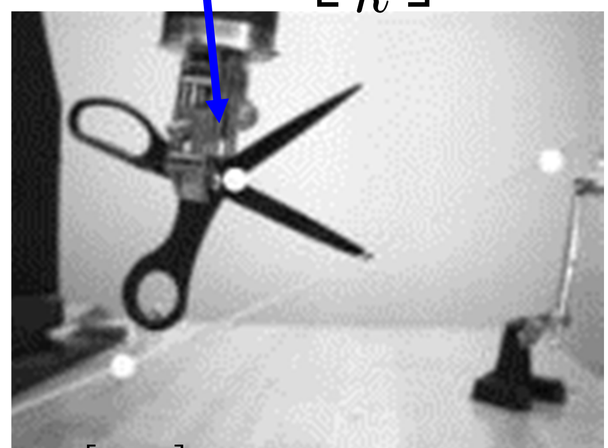
- Point to Line

Note: y's in homogeneous coord.

$$\mathbf{y}_l = \begin{bmatrix} u_l \\ v_l \\ k \end{bmatrix}$$

Line: $$\mathbf{E}_{pl}(\mathbf{y}, \mathbf{l}) = \begin{bmatrix} \mathbf{y}_l \cdot \mathbf{l}_l \\ \mathbf{y}_r \cdot \mathbf{l}_r \end{bmatrix}$$

$$\mathbf{l}_l = [\, y_2 \times y_3 \,]_l$$



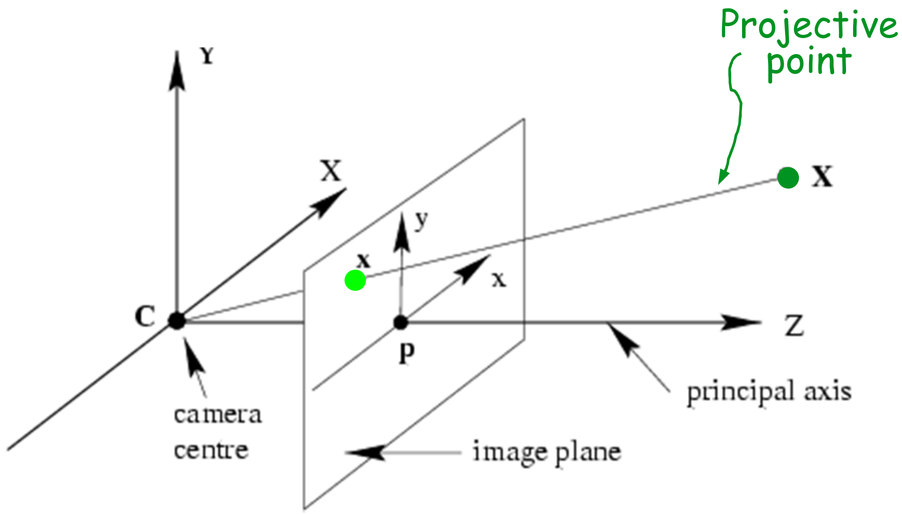
$[y_3]$

$[y_2]$

How to design visual specifications in a principled way?

Review of projective geometry

The 2D projective plane



Homogeneous coordinates equivalence relation

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

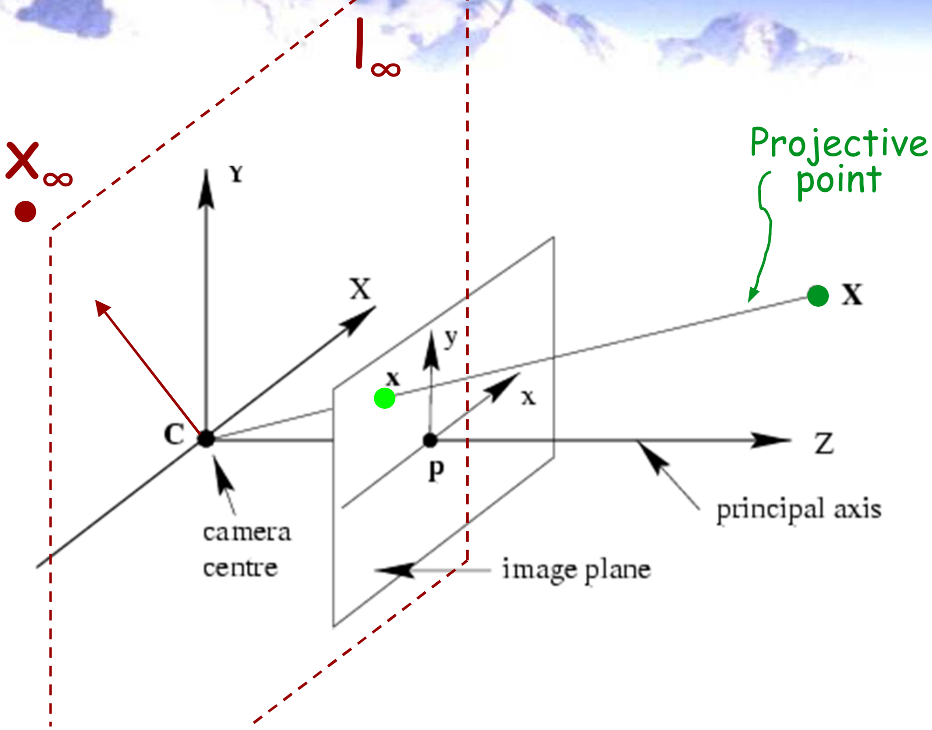
$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} X \\ Y \end{bmatrix}$$

Inhomogeneous equivalent

- 2D projective space models perspective imaging
- Each 3D ray is a point in P^2 : homogeneous coords.

Review of projective geometry

The 2D projective plane



Homogeneous coordinates

$$\begin{bmatrix} X \\ y \\ k \end{bmatrix} \equiv s \begin{bmatrix} X \\ y \\ k \end{bmatrix} \qquad s \neq 0$$

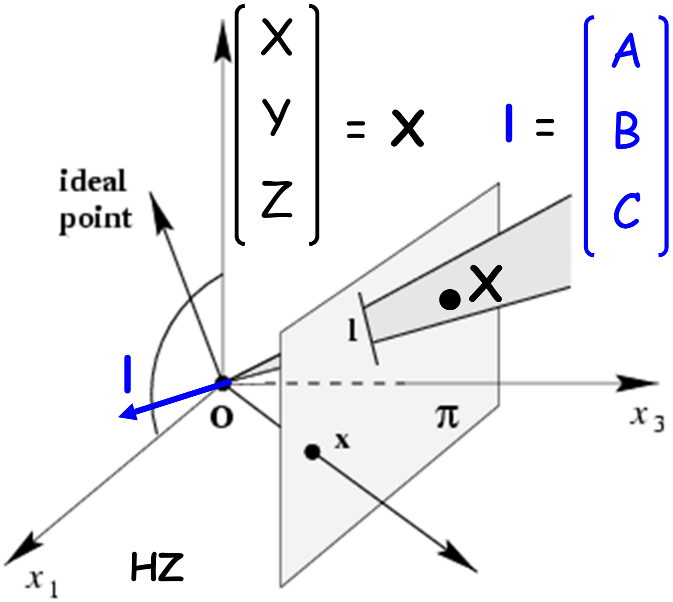
$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{k} \begin{bmatrix} X \\ y \end{bmatrix}$$

Inhomogeneous equivalent

- 2D projective space models perspective imaging
- 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.

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

The line joining two points — The point joining two lines

Projective transformations

- Homographies, collineations, projectivities
- 3x3 nonsingular H

  maps $\boldsymbol{P}^2$ to $\boldsymbol{P}^2$
  8 degrees of freedom
  determined by 4 corresponding points

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

$$x' = Hx$$

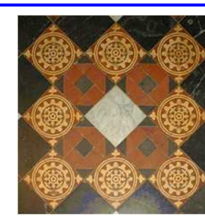
- Transforming Lines?

subspaces preserved $\mathbf{x}^T\mathbf{l} = 0 \qquad \mathbf{x}'^T\mathbf{l}' = 0$

substitution $x^T H^T \mathbf{l}' = 0$

dual transformation $\mathbf{l}' = H^{-T}\mathbf{l}$

Homographies a generalization of affine and Euclidean transforms

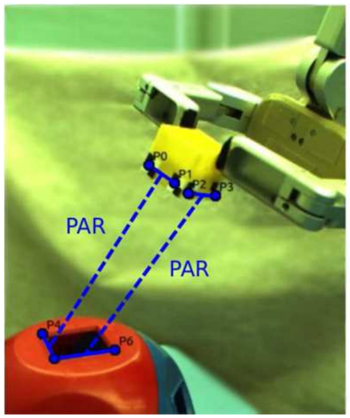
Group	Transformation	Invariants	Distortion
Projective 8 DOF	$H_P = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{v}^T & v \end{bmatrix}$	• Cross ratio • Intersection • Tangency	
Affine 6 DOF	$H_A = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Parallelism • Relative dist in 1d • Line at infinity $\mathbf{l}_\infty$	
Metric 4 DOF	$H_S = \begin{bmatrix} sR & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Relative distances • Angles • Dual conic C^*_∞	
Euclidean 3 DOF	$H_E = \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Lengths • Areas	

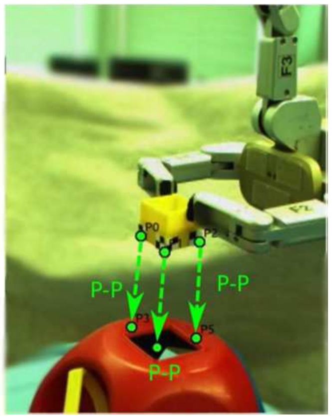
2 dof $\mathbf{l}_\infty$

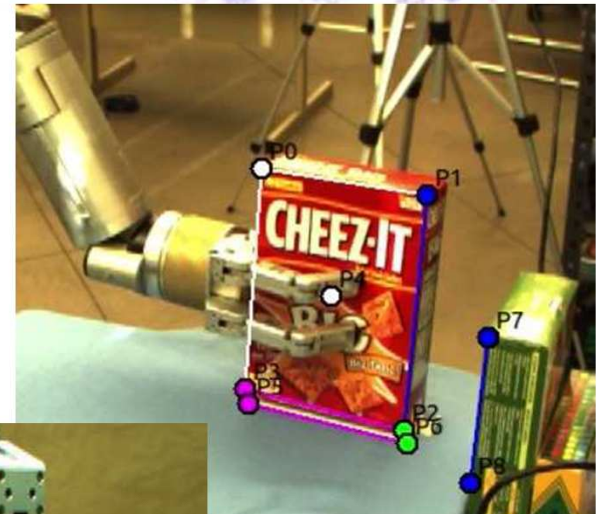
2 dof C^*_∞

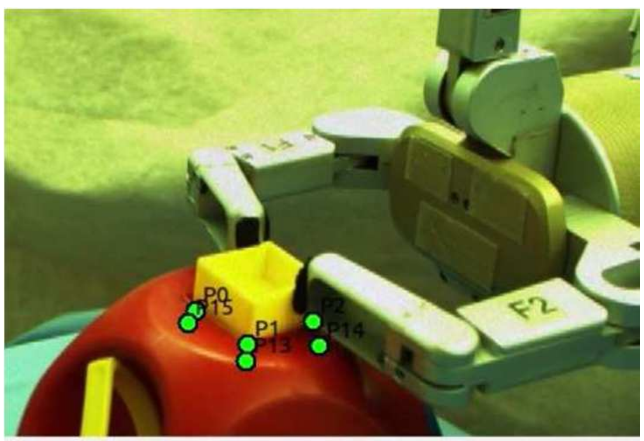
- Goto slide 59

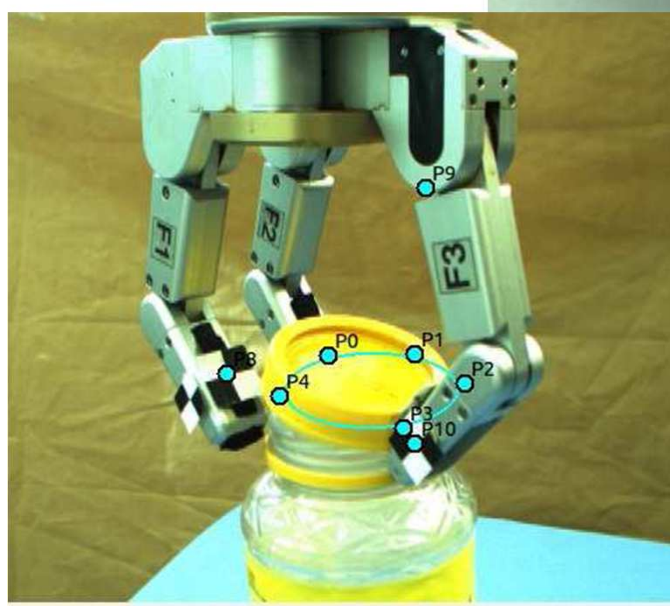
How to define a visual task?



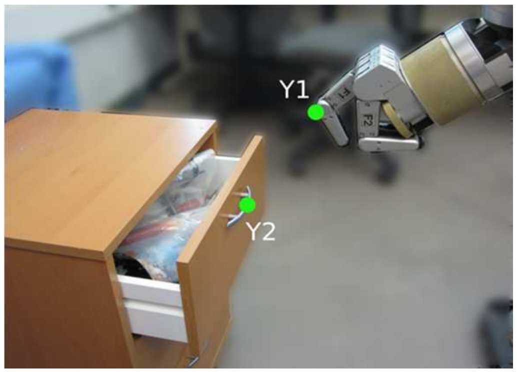


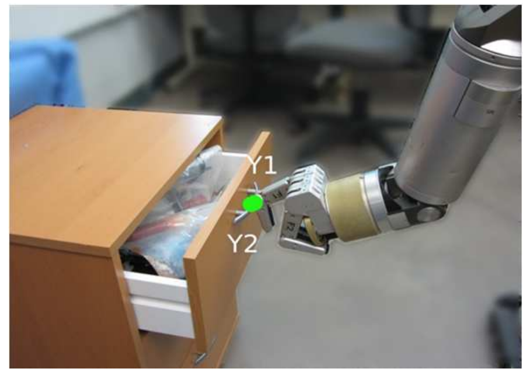






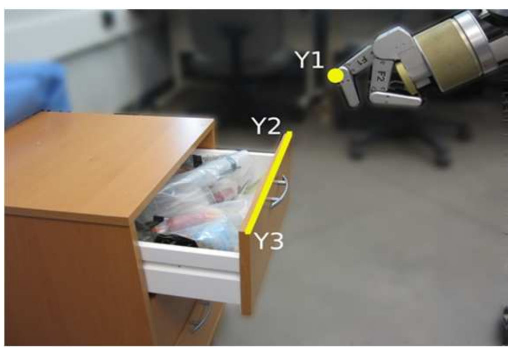
Visually defined alignment: basic

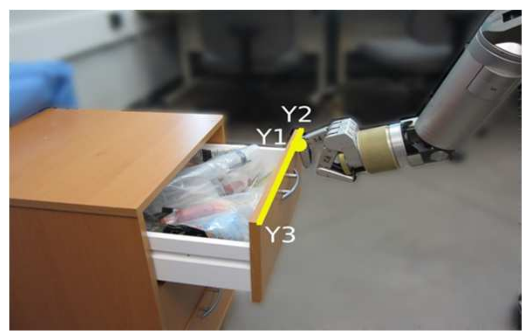




point-to-point: $e_{pp}(y) = y_2 - y_1$

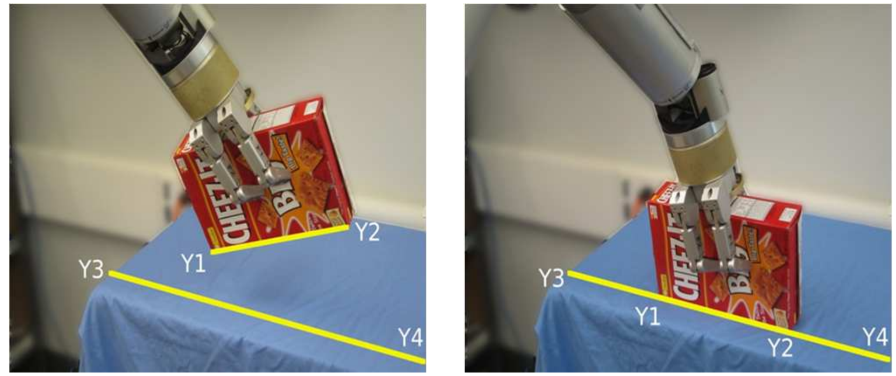
or (homogenous coord) $e_{pp}(y) = y_2 \times y_1$



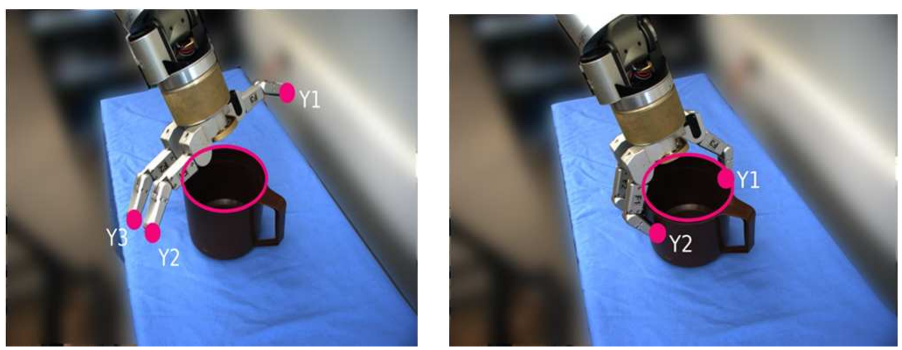


point-to-line: $e_{pl}(y) = y_1 \cdot (y_2 \times y_3)$

Some more visual alignments



parallel lines: $e_{par}(l) = (l_1 \times l_2) \times (l_3 \times l_4)$ line-to-line: $e_{ll}(y) = y_1 \cdot (y_3 \times y_4) + y_2 \cdot (y_3 \times y_4)$

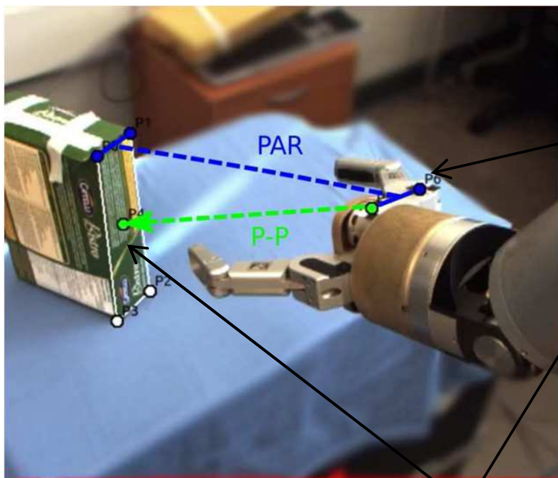


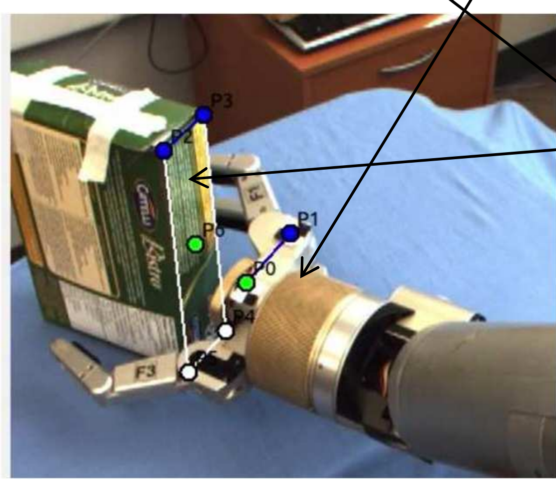
point-to-ellipse: $e_{pe}(y) = y^T{}_1 \, C_{ellipse} \, y_1$

Now: Language for visual alignments What else do we need?

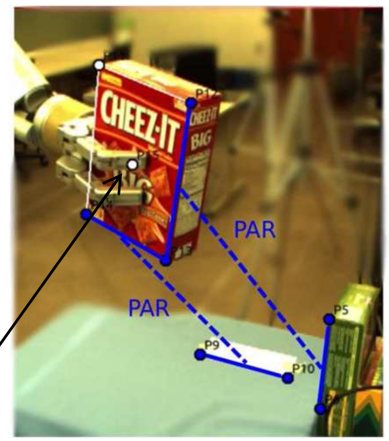
Need: 1. Some way of entering alignments in images

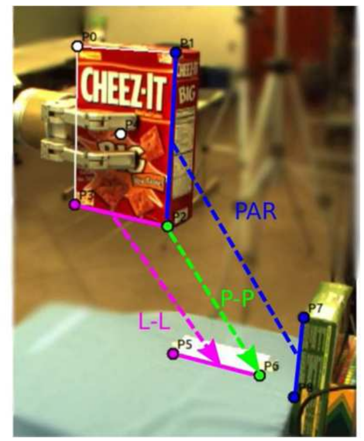
2. video tracking to perform servoing!



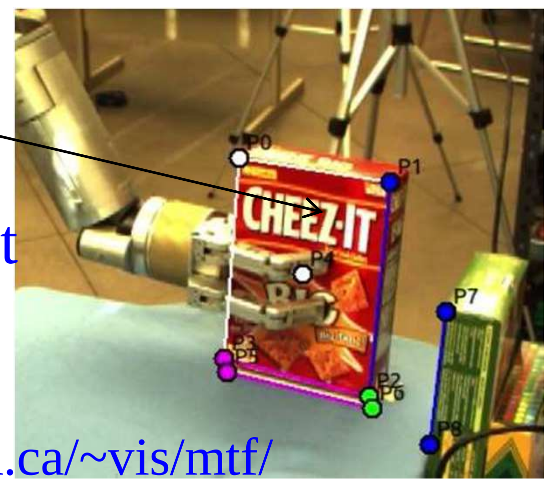


Feature trackers





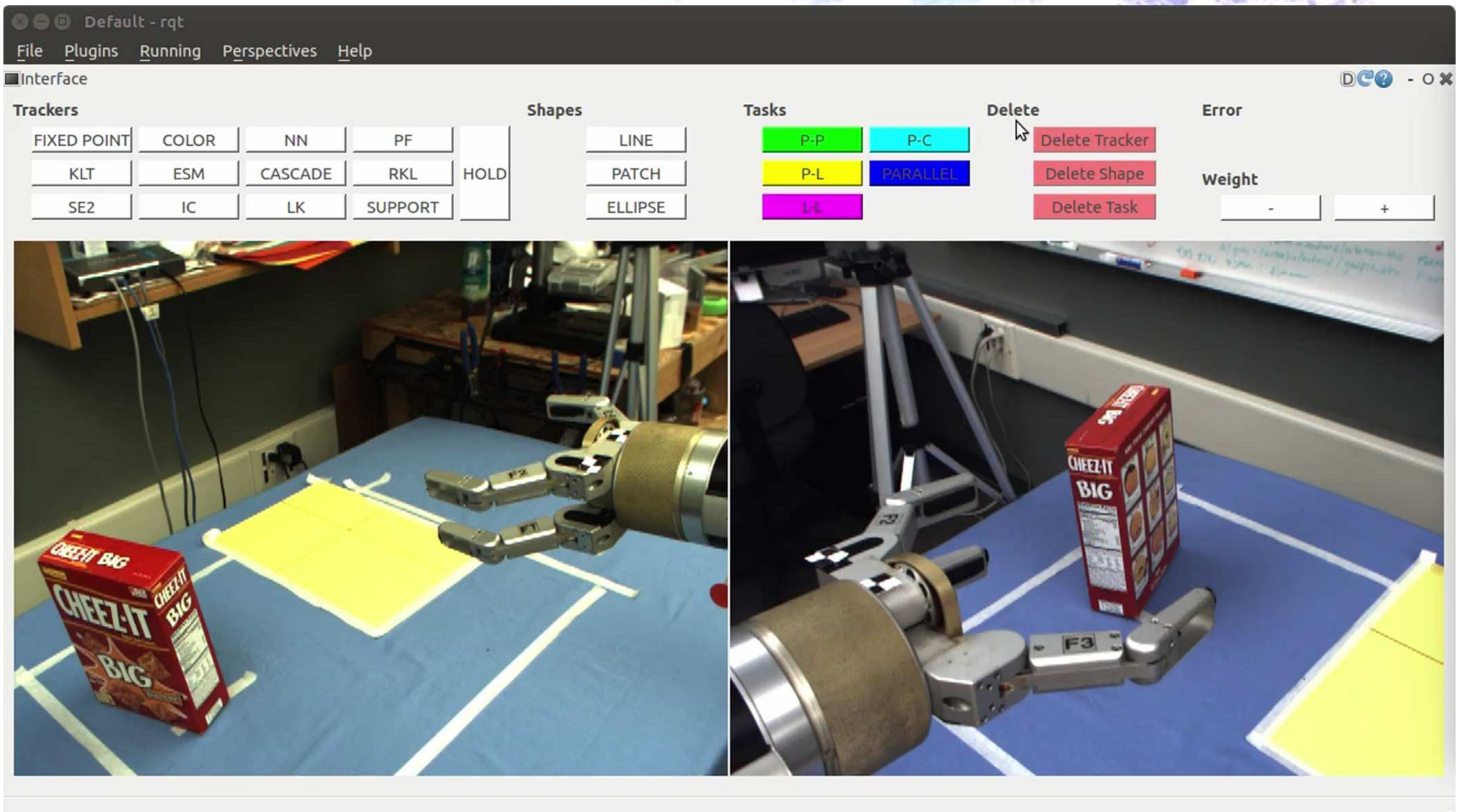
Registration trackers: Thu 9:50 Abhineet



Download: http://webdocs.cs.ualberta.ca/~vis/mtf/

ViTa: Visual Task spec. for Manip.

Mona Gridseth, Martin Jagersand ICRA'16



Composing basic visual alignments
1. Parallel

Conceptual task

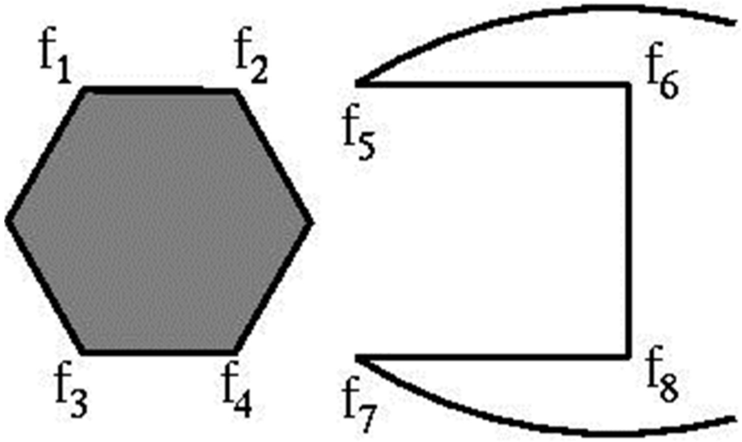
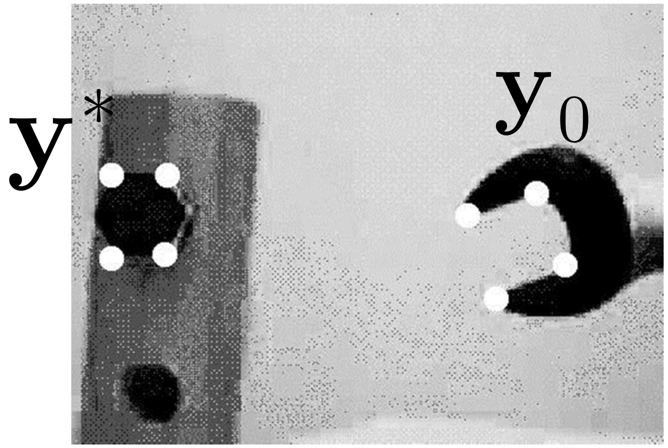


Image of task



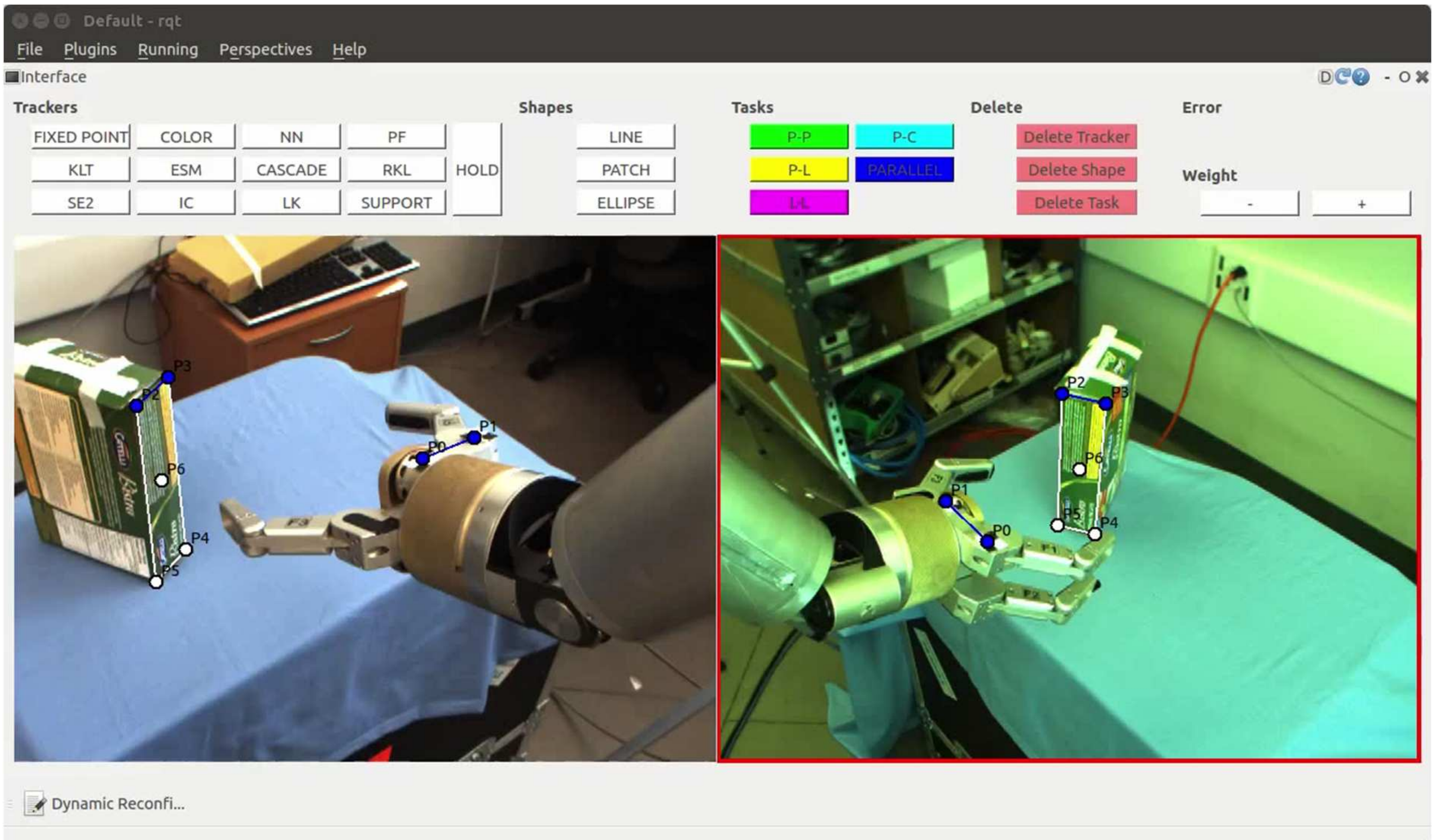
$$\mathbf{E} = \begin{bmatrix} y_1 \\ \vdots \\ y_{16} \end{bmatrix}^* - \begin{bmatrix} y_1 \\ \vdots \\ y_{16} \end{bmatrix}_0$$

$$E_{wrench}(y) = \begin{pmatrix} y_2 - y_5 \\ y_4 - y_7 \\ y_6 \bullet (y_1 \times y_2) \\ y_8 \bullet (y_3 \times y_4) \end{pmatrix}$$

Which encoding E(y) is better? Can both reach E(y) = 0 ?

Do we need 1 or 2 cameras? Why?

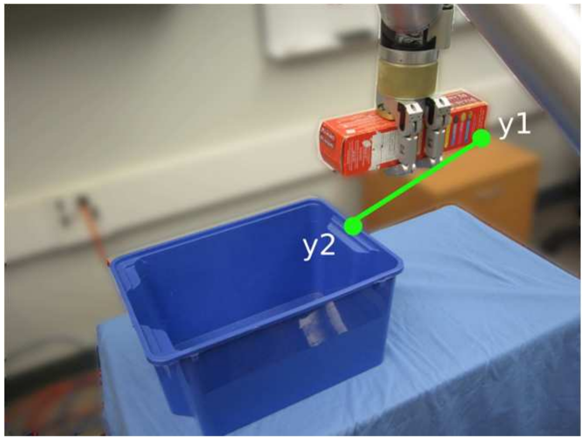
Composing basic visual alignments
1. Parallel constrains

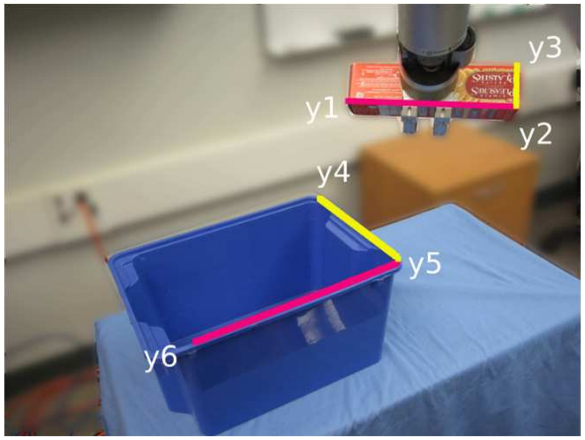


Composing basic visual alignments
2. Serial seqencing

Many tasks divide naturally into

1. Transportation / reaching
   - Coarse primitive for large movements
   - Low DOF control (e.g. of object centroid)
   - Robust to disturbances
2. Fine Manipulation
   - For high precision control of both position and orientation
   - 6DOF control based on several object features



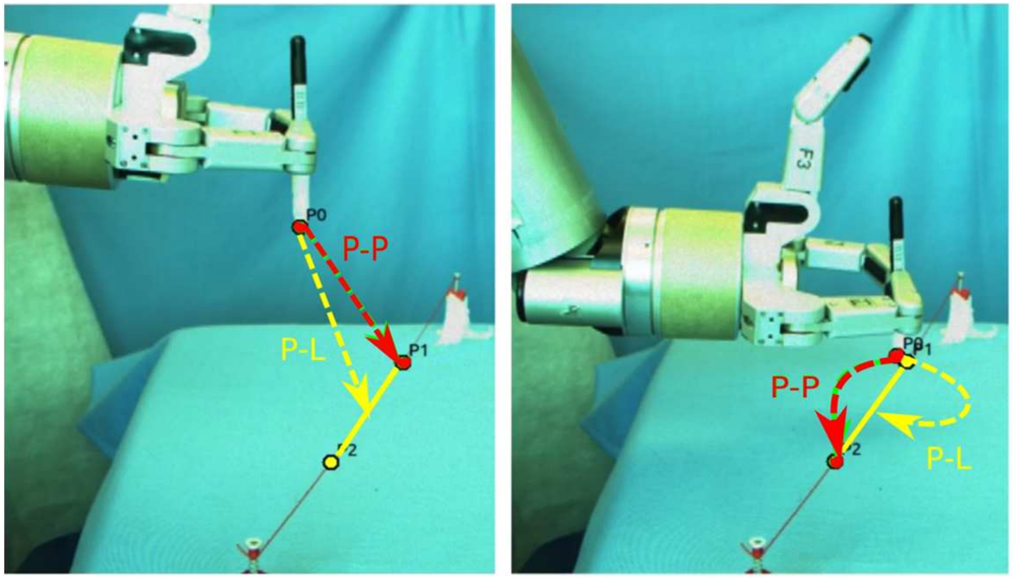


Serial Composition
Solving whole real tasks

- Linked task primitive

$$A = (\mathbf{E}^{\text{init}}, M, \mathbf{E}^{\text{final}})$$

1. Acceptable initial (visual) conditions
2. Visual or Motor constraints to be maintained
3. Final desired condition



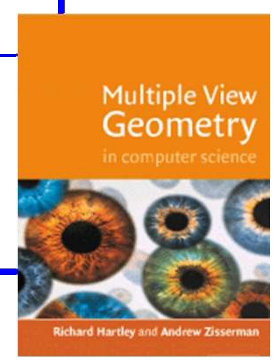
- Task = $A_1 A_2 \ldots A_k$

Previous alignments just examples. Can use any invariants for the level of calibration available

Group	Transformation	Invariants	Distortion
Projective 8 DOF	$H_P = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{v}^T & v \end{bmatrix}$	• Cross ratio • Intersection • Tangency	
Affine 6 DOF	$H_A = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Parallelism • Relative dist in 1d • Line at infinity $\mathbf{l}_\infty$	
Metric 4 DOF	$H_S = \begin{bmatrix} sR & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Relative distances • Angles • Dual conic C^*_∞	
Euclidean 3 DOF	$H_E = \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$	• Lengths • Areas	

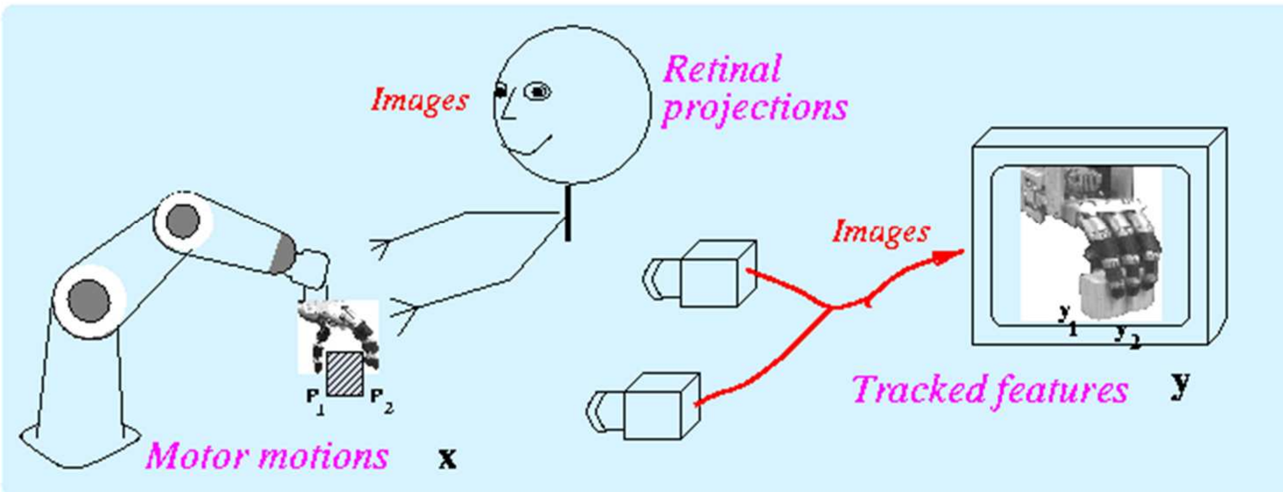
2 dof $\mathbf{l}_\infty$

2 dof C^*_∞



Use only camera coord. Uncalibrated Visual Servoing

Jagersand'94,96,00 Hosoda, Asada'94, 97



repeat

1. Solve for motion: $[\mathbf{y}^* - \mathbf{y}_k] = \mathbf{J}\Delta\mathbf{x}$
2. Move robot joints: $\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta\mathbf{x}$
3. Read actual visual move Δy

$\frac{(\Delta\mathbf{y} - \hat{J}_k \Delta\mathbf{x})\Delta\mathbf{x}^T}{\Delta\mathbf{x}^T\Delta\mathbf{x}}$

Can we always guarantee when a task is achieved/achievable?

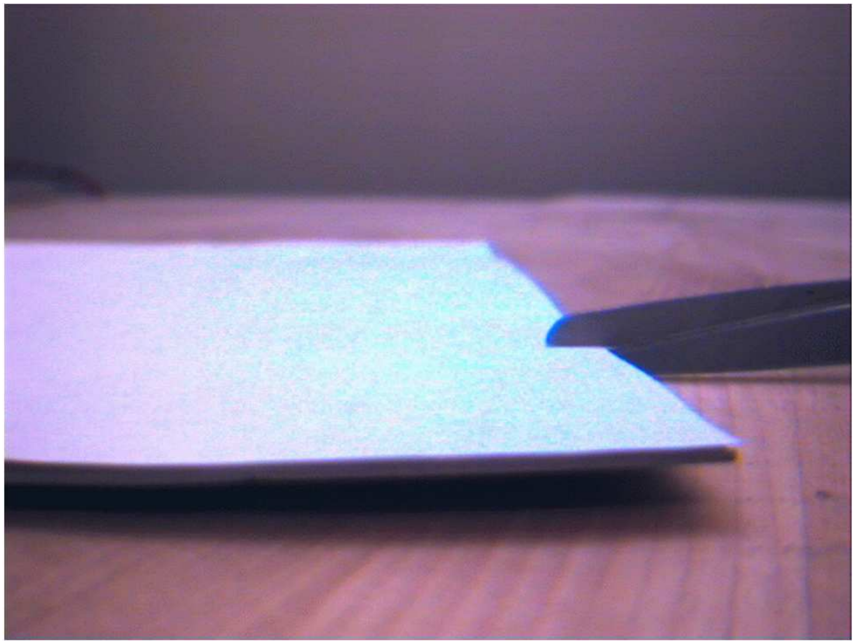
Downloadable templated library: http://ugweb.cs.ualberta.ca/~vis/ros-uvs/

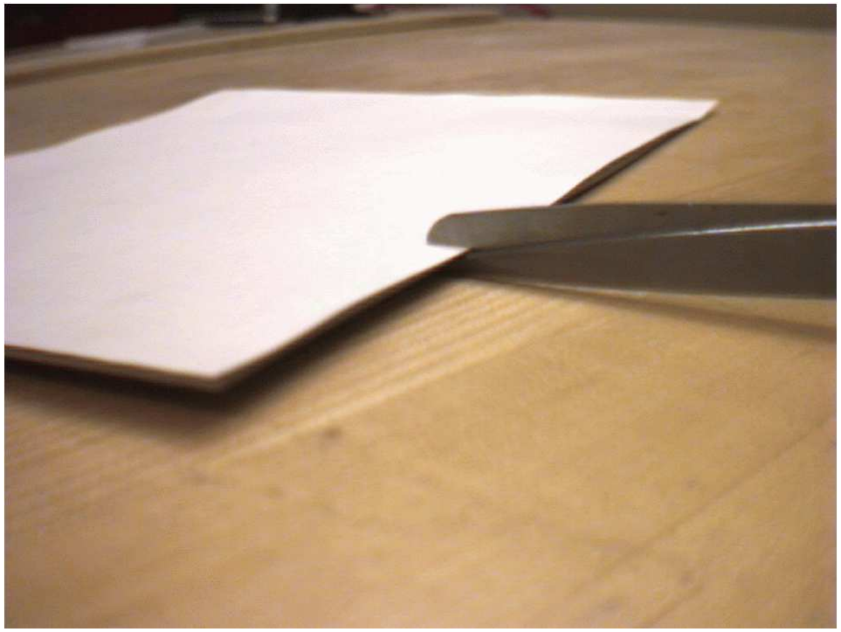
Visually guided motion control

Issues:

1. What tasks can be performed?
   – Camera models, geometry, visual encodings
2. How to do vision guided movement?
   – H-E transform estimation, feedback, feedforward motion control
3. How to plan, decompose and perform whole tasks?

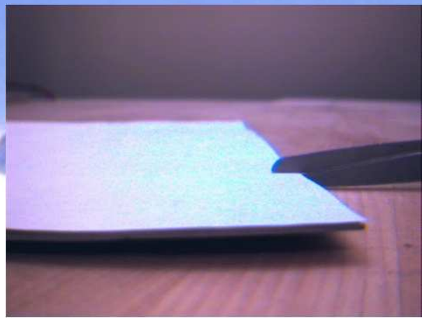
Task ambiguity





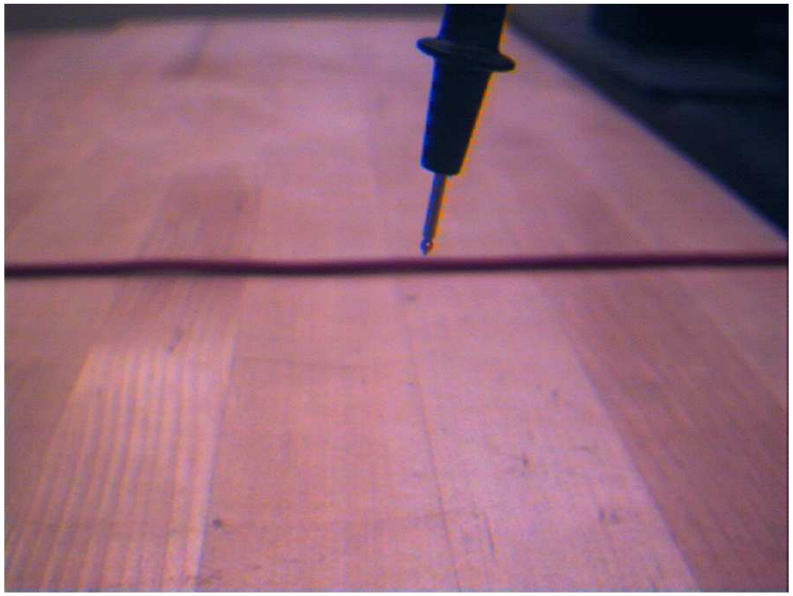
- Will the scissors cut the paper in the middle?

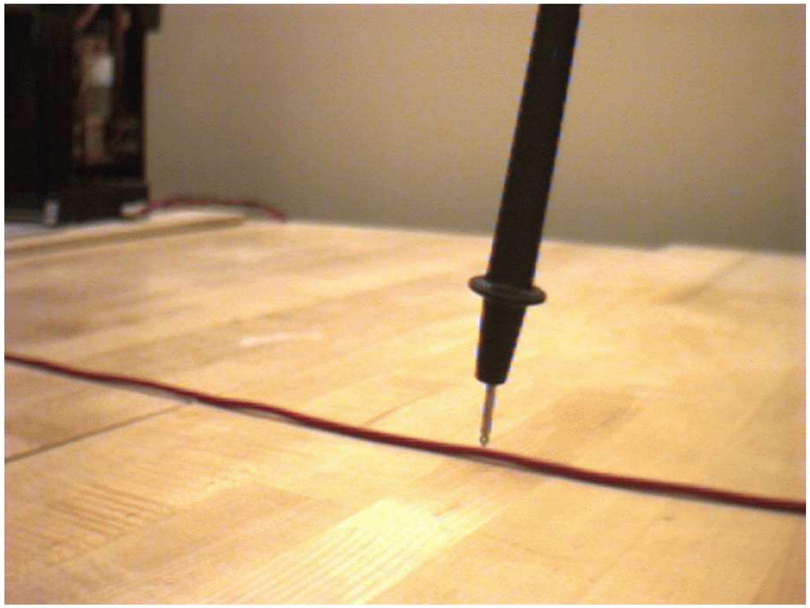
Task ambiguity



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

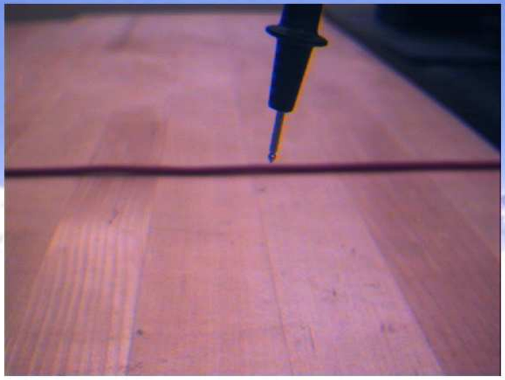
Task Ambiguity

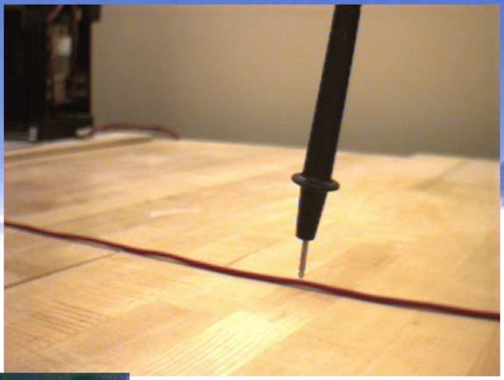


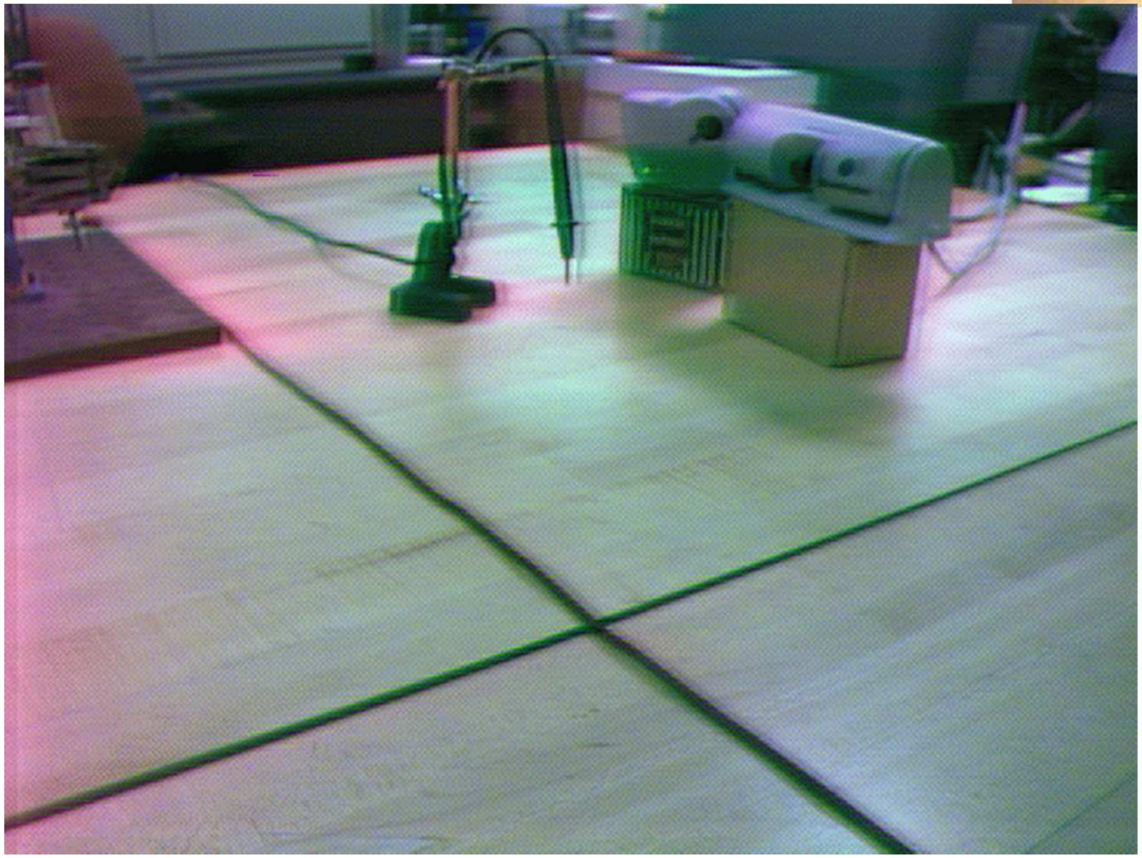


- Is the probe contacting the wire?

Task Ambiguity







- Is the probe contacting the wire? **NO!**

Solve the cut in the middle task?

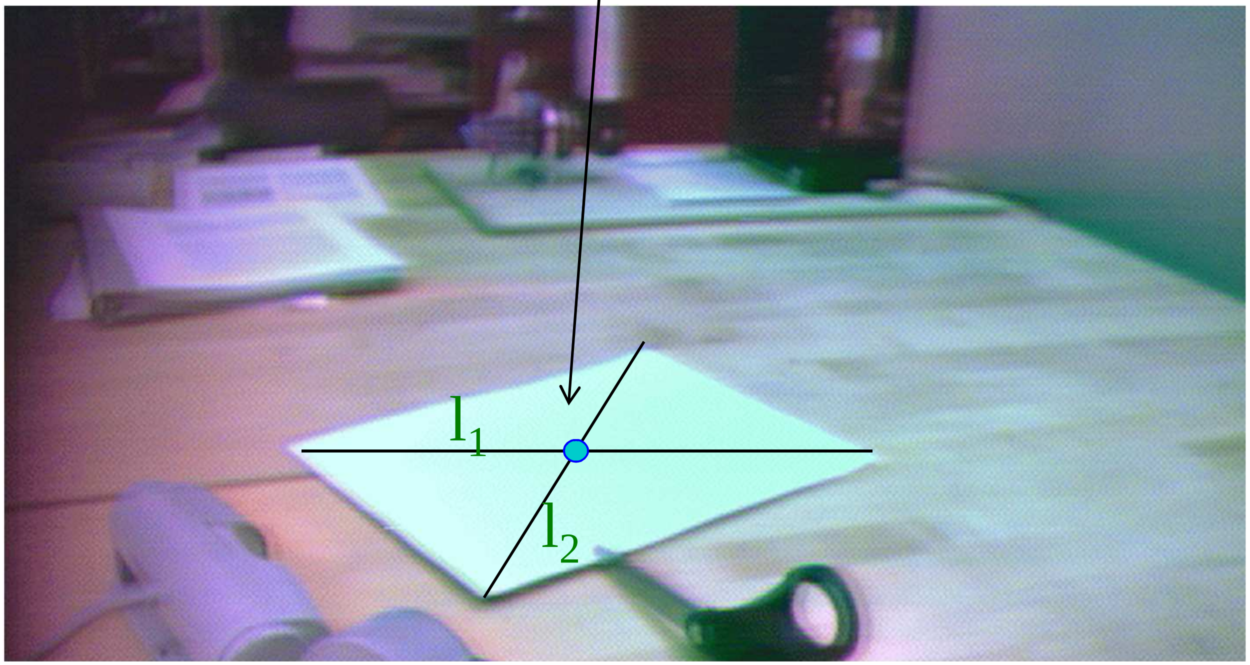
- Compute paper midpoint. How?

Solve the cut in the middle task?

- Compute paper midpoint.

(Are we done yet?)

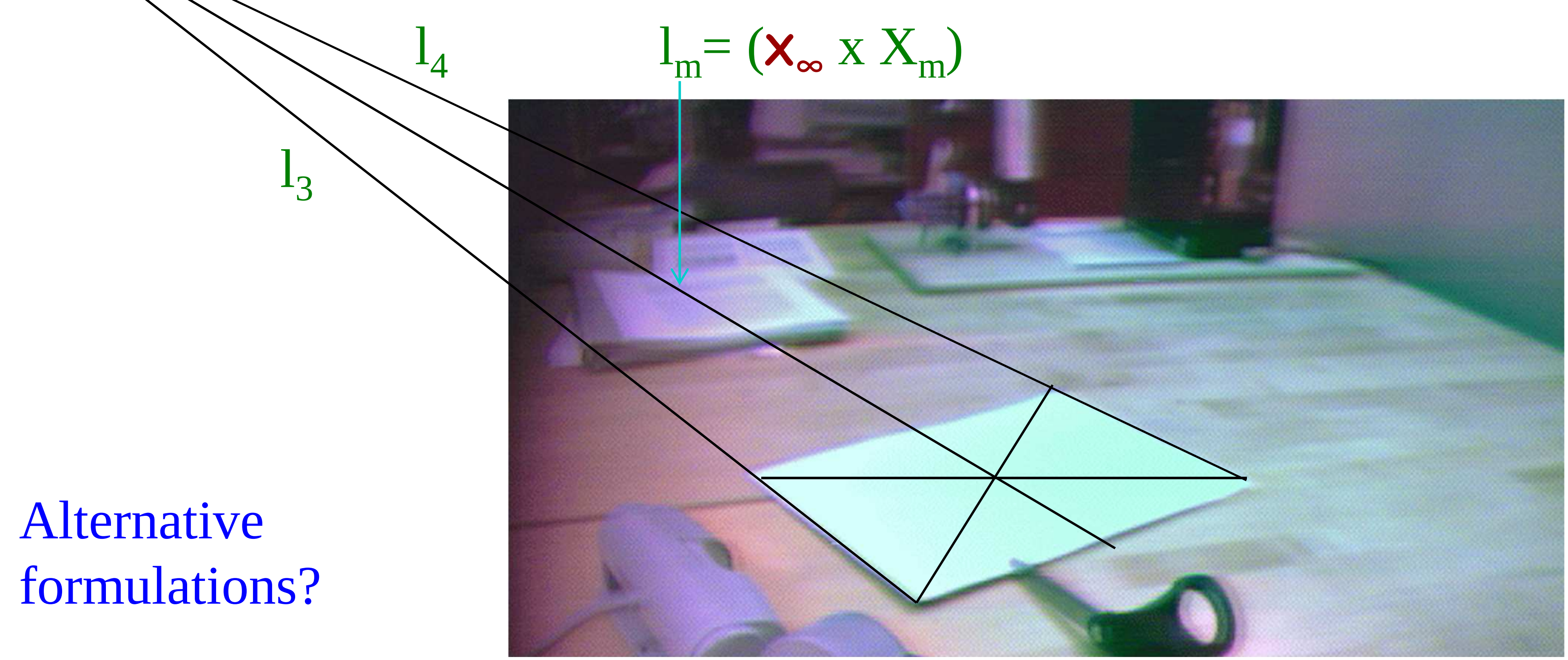
$x_m = (l_1 \times l_2)$



Solve the cut in the middle task?

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

$\mathbf{x}_\infty = (l_3 \text{ x } l_4)$

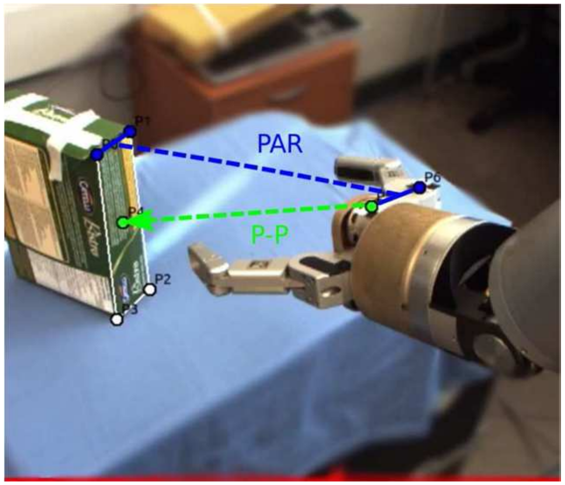


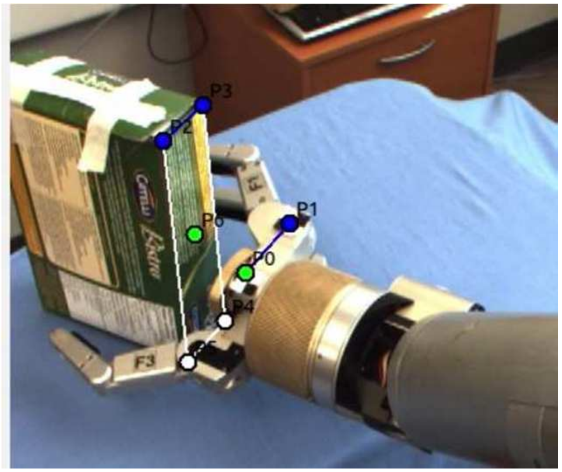
Alternative formulations?

Visual Servoing Summary

Steps in Image-Based Visual Servoing

1. Specify image alignments
2. Video tracking of regions and features
3. Minimize visual error by moving robot



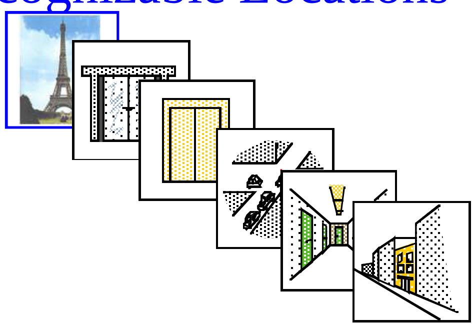


- Difficulties:
  - Tracking: finding trackable regions. Tracker robustness and accuracy
  - Creating correct and complete specifications can be tedious.

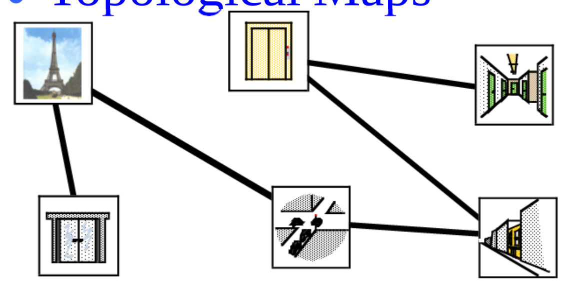
What information do we use to move?

Mobile robot navigation: From appearance to metric SLAM

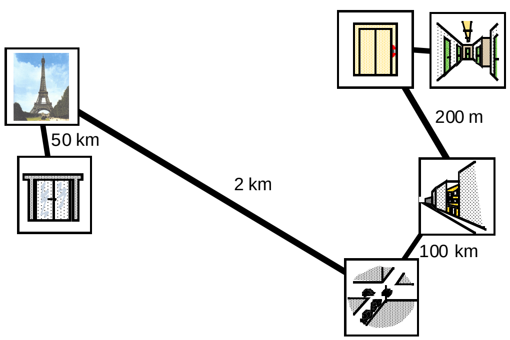
- Recognizable Locations



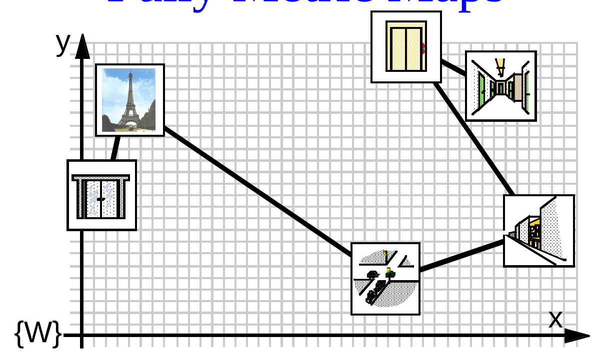
- Topological Maps



- Metric Topological Maps



- Fully Metric Maps



Courtesy K. Arras

How about arm/hand manipulation?

Almost all work use global metric world coordinates

Underappreciated topic?

A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms, Seitz, Curless, Diebel, Szeliski
CVPR 2006, vol. 1, pages 519-526. — 1479 citations

Single View Metrology — 559 citations
A. Criminisi, I Reid, A Zisserman
International Journal of Computer Vision 40 (2), 123-148, 2000

What tasks can be performed with an uncalibrated stereo vision system? JP Hespanha, Z Dodds, GD Hager, AS Morse
International Journal of Computer Vision 35 (1), 65-85, 1999 — 34 citations

Untapped opportunity?

Beyond projective camera vision and screen GUI: Pointing

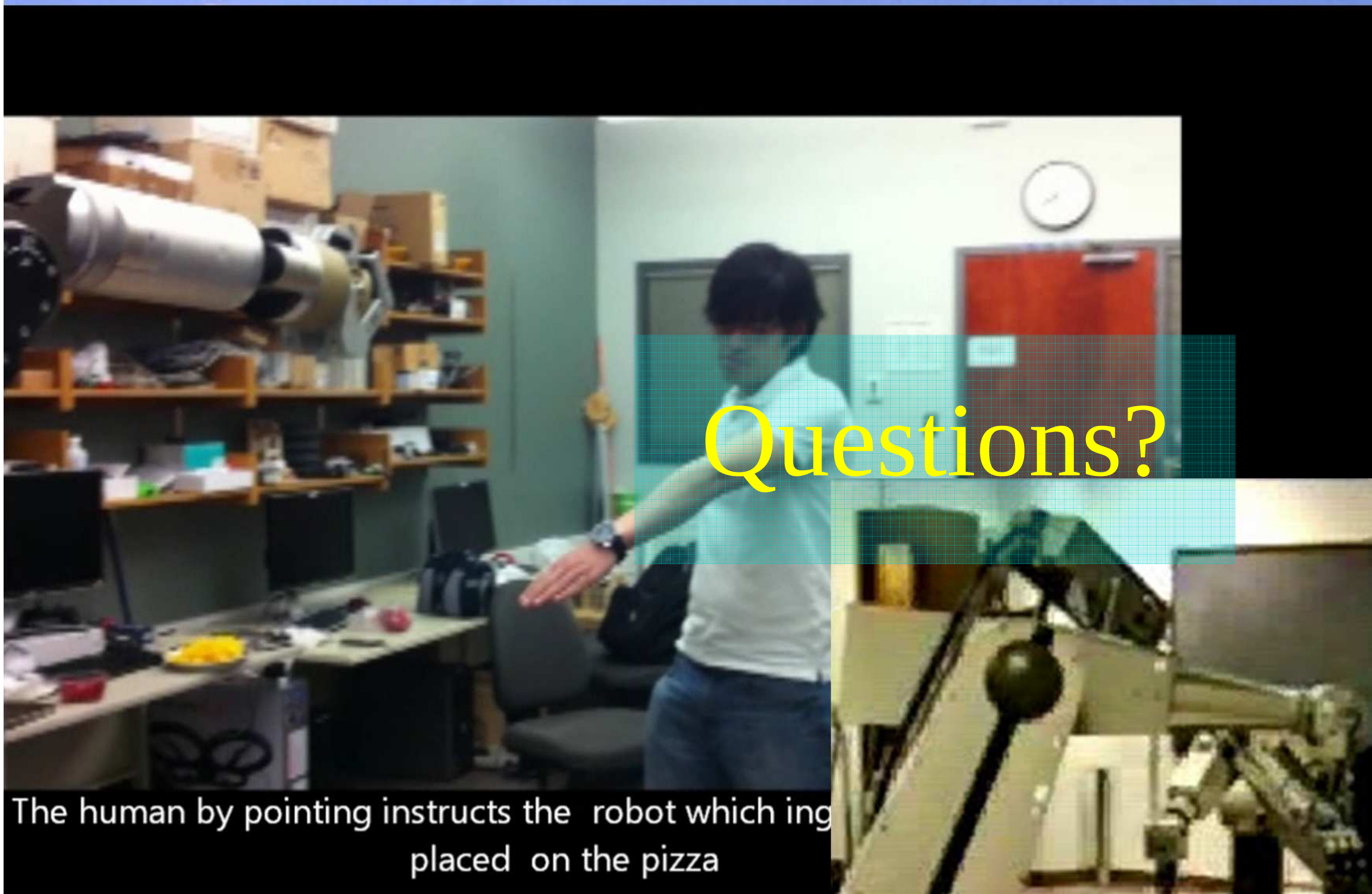
The human by pointing instructs the robot which ingredients are to be placed on the pizza

Beyond projective camera vision and screen GUI

- Without physical GUI: Gestures and pointing
- Combine modalities: Camera vision, RGBD, force …
- Use sensory feedback and calibration level most appropriate for task
  - Precise visual alignments can be combined with approximate Euclidean, (e.g. hold glass up)
  - Force feedback can maintain contact with surface, and visual feedback define the motion on the surface.

Making Pizza with my robot

3rd Prize ICRA'15 Video competition



Questions?

Change a Lightbulb

ICRA'97 Video