Computer Vision
Week 13

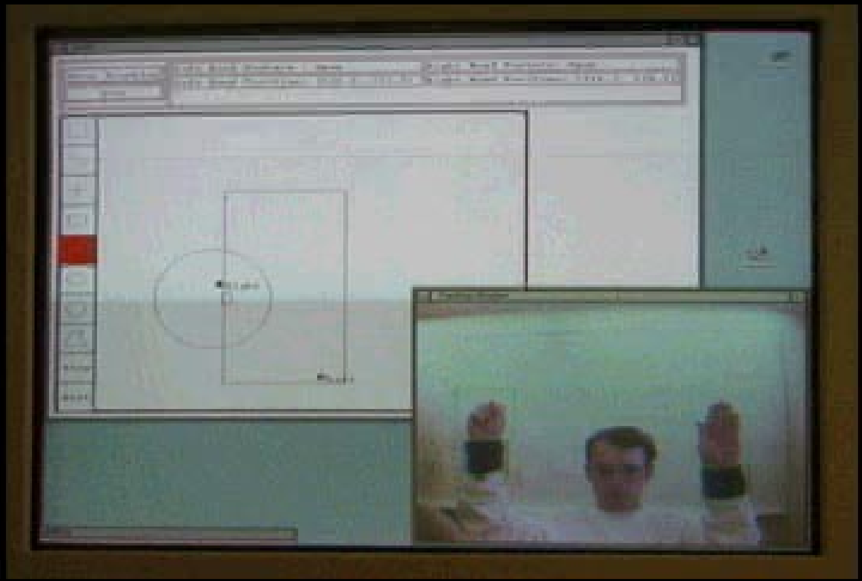
Visual Tracking

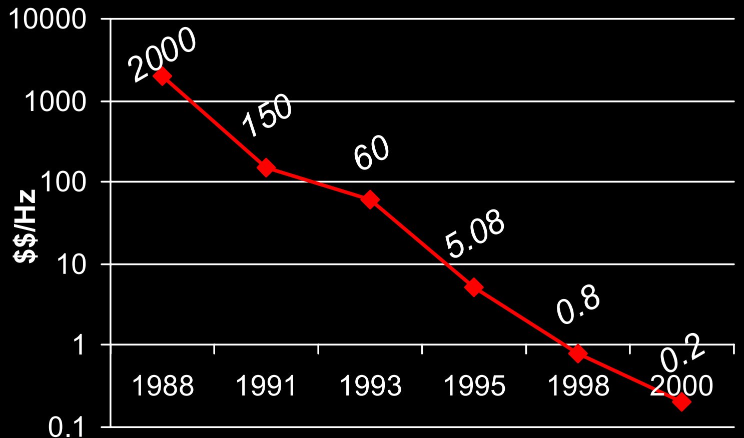
Efficient Region Tracking with Parametric Models of Geometry and Illumination (with P. Belhumeur). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(10), pp. 1125-1139, 1998.

The XVision System: A General-Purpose Substrate for Portable Real-Time Vision Applications (with K. Toyama). *Computer Vision and Image Understanding*, 69(1), pp. 23-37, 1998.

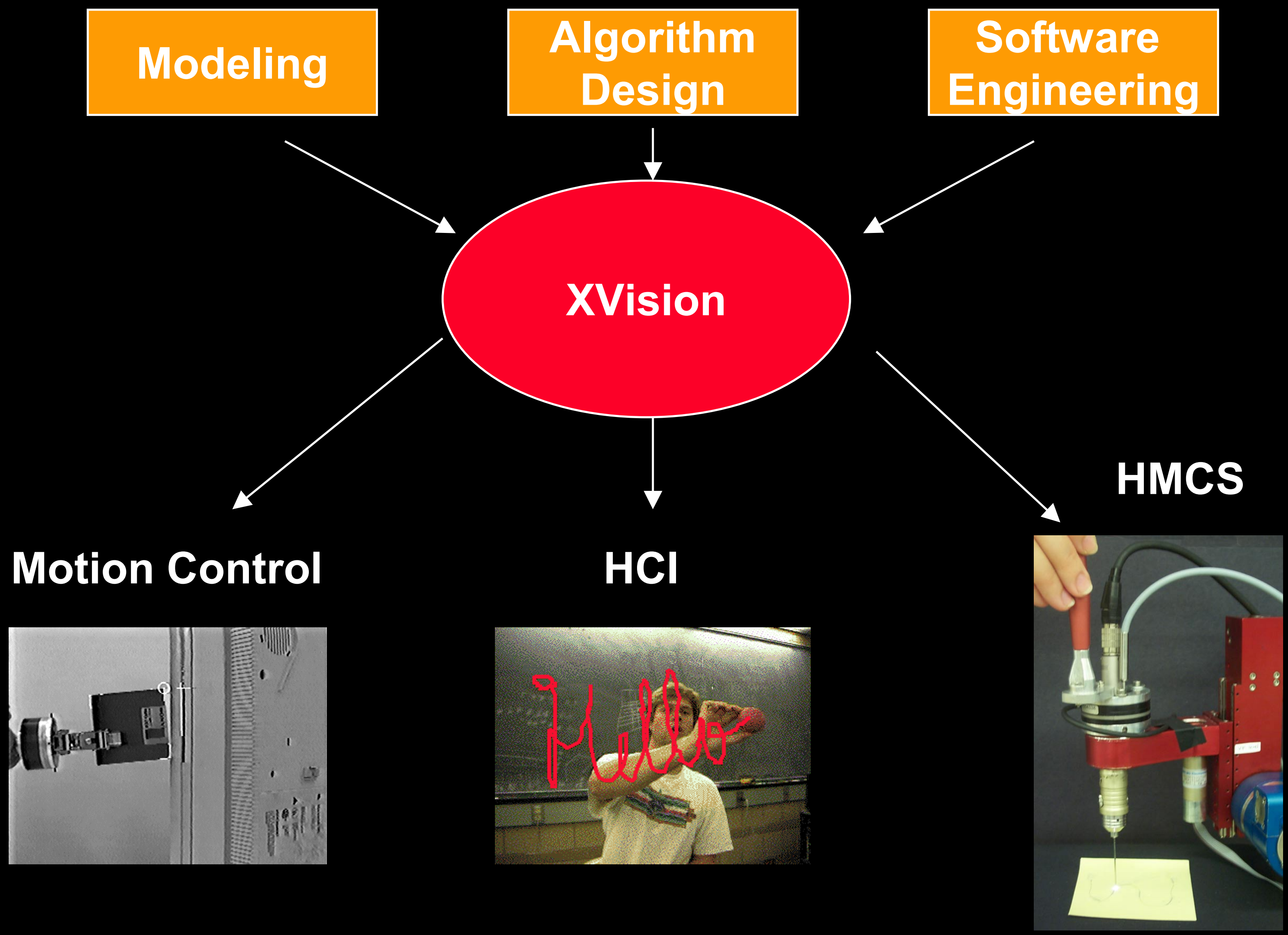
http://www.cs.jhu.edu/CIRL

Motivation



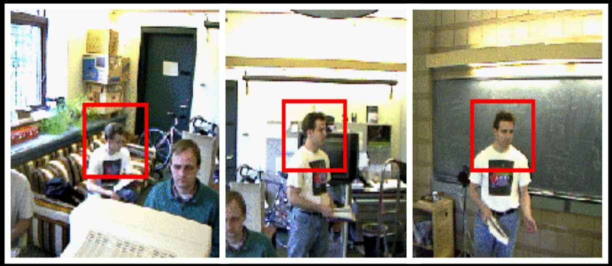

10000
1000
100
10
1
0.1
$$/Hz
2000
150
60
5.08
0.8
0.2
1988
1991
1993
1995
1998
2000

Overview

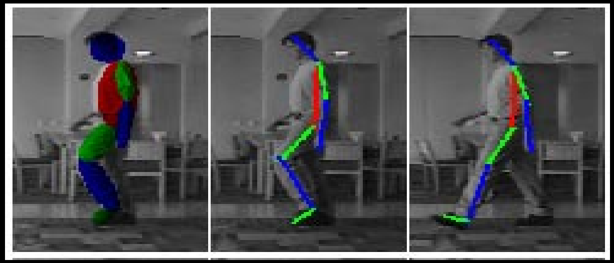

Modeling
Algorithm Design
Software Engineering
XVision
Motion Control
HCI
HMCS

VISUAL TRACKING

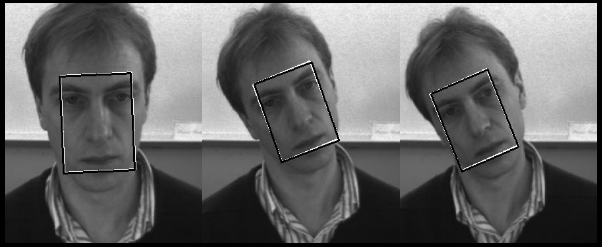
What Is Visual Tracking?



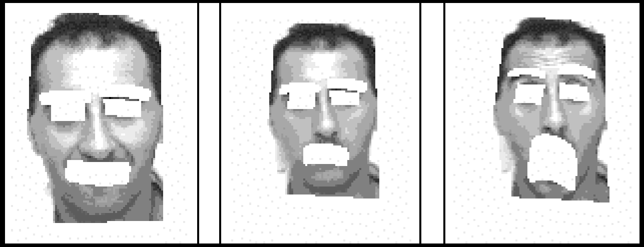
Hager & Rasmussen 98



Bregler and Malik 98



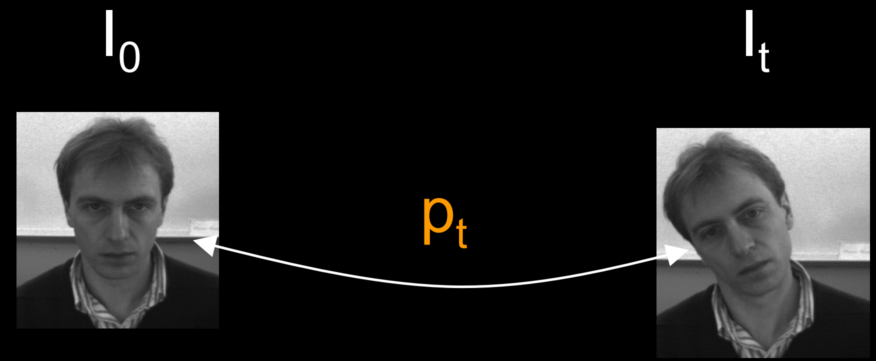
Hager & Belhumeur 98



Black and Yacoob 95

Bascle and Blake 98

Principles of Visual Tracking

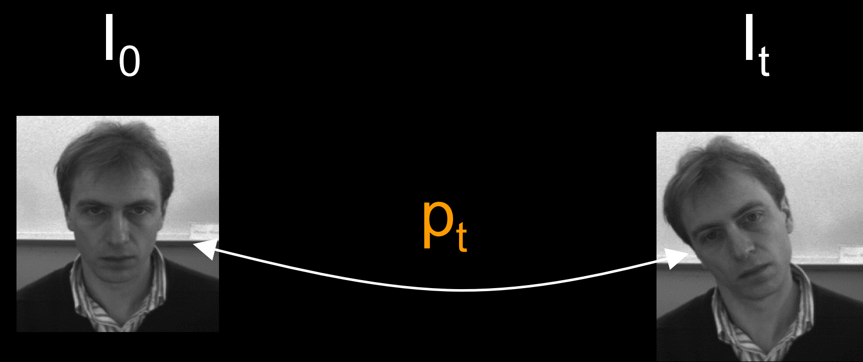


Variability model: $I_t = g(I_0, p_t)$

Incremental Estimation: From I_0, I_{t+1} and p_t compute Δp_{t+1}

$$|| I_0 - g(I_{t+1}, p_{t+1}) ||^2 \implies \min$$

Principles of Visual Tracking

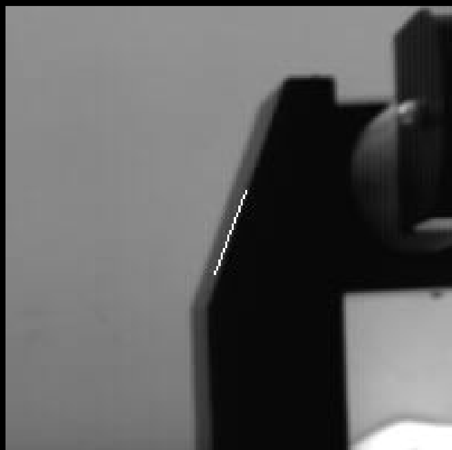


Variability model: $I_t = g(I_0, p_t)$

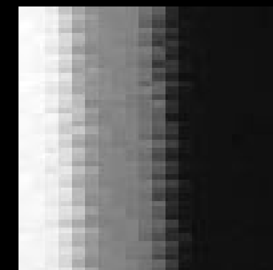
Incremental Estimation: From I_0, I_{t+1} and p_t compute Δp_{t+1}

Visual Tracking = Visual Stabilization

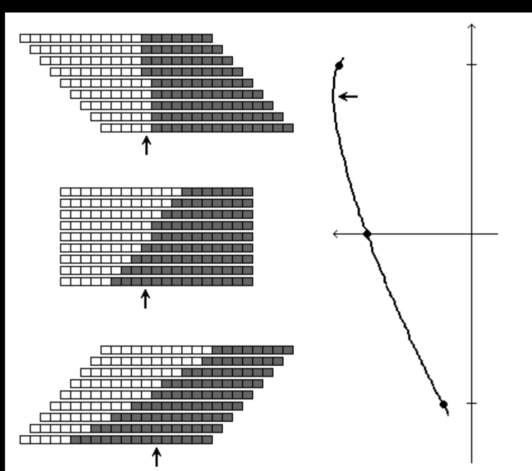
A Simple Example: Edges



Rotational warp



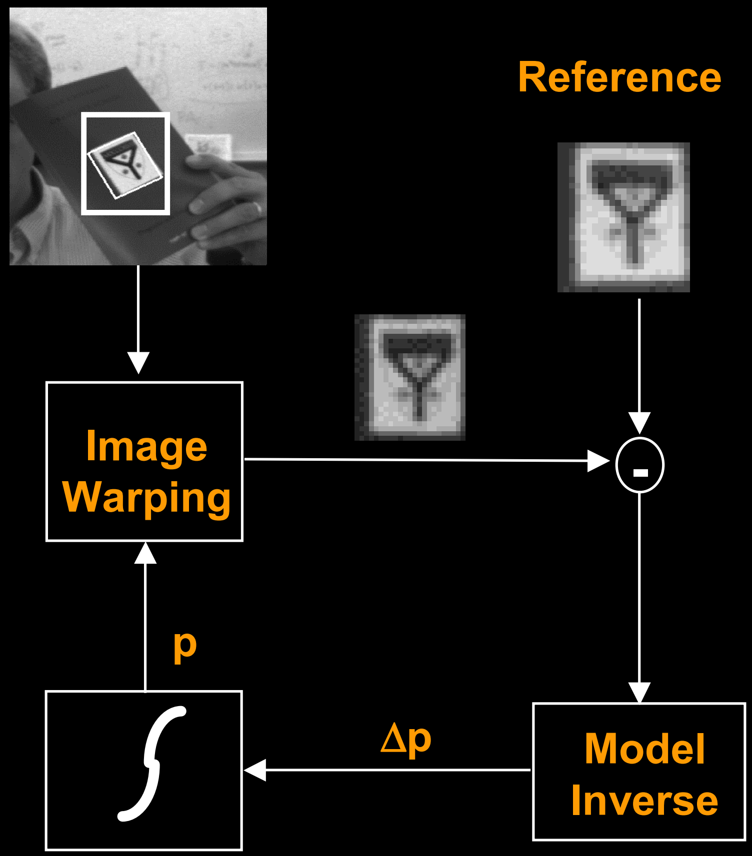
Apply an edge detector across rows



Sum and interpolate to get position and orientation

Tracking Cycle

- ⇨ Prediction
  - Prior states predict new appearance
- ⇨ Image warping
  - Generate a "normalized view"
- ⇨ Estimation
  - Compute change in parameters from changes in the image
- ⇨ State integration
  - Apply correction to state

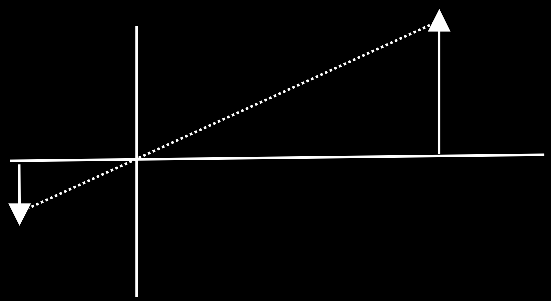


Some Background

- ⇨ Perspective (pinhole) camera

  X' = x/z

  Y' = y/z



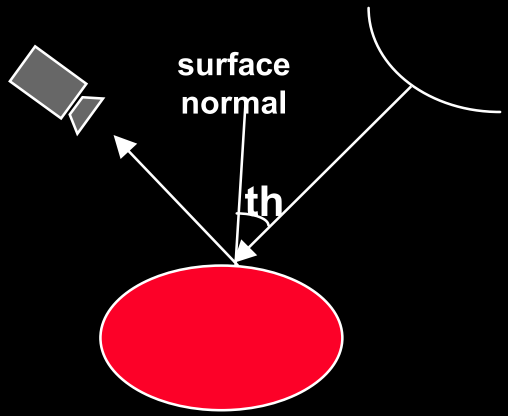
- ⇨ Para-perspective

  X' = s x

  Y' = s y

- ⇨ Lambert's law

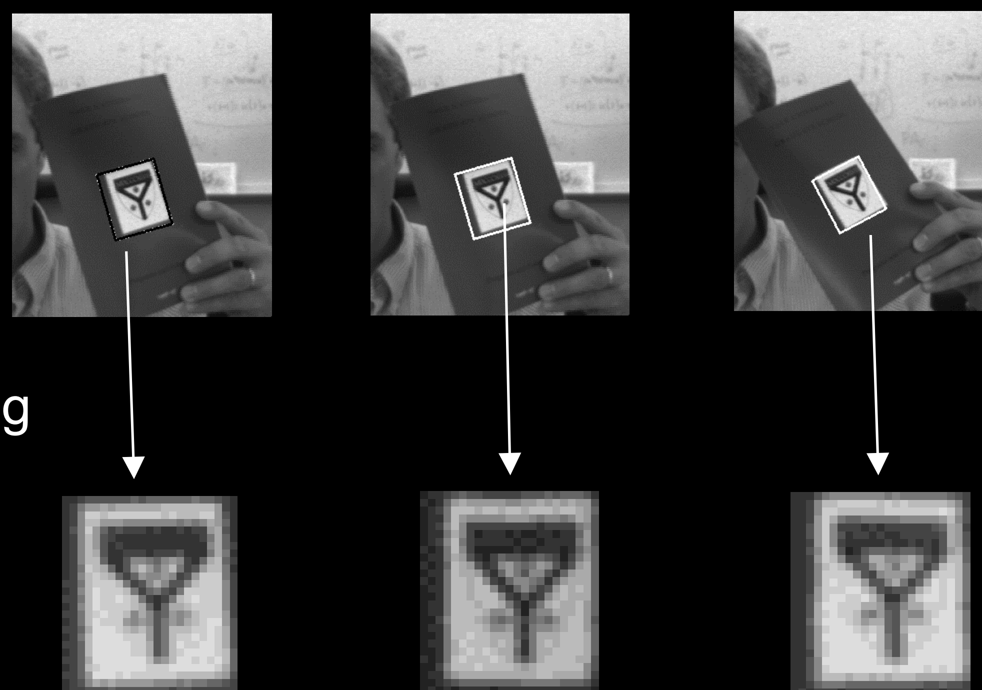
  B = a cos(th)



Regions: A More Interesting Case

Planar Object => Affine motion model: $u'_i = A u_i + d$

Warping



$$I_t = g(p_t, I_0)$$

Stabilization Formulation

⇨ Model

$$I_0 = g(p_t, I_t) \qquad \text{(image I, variation model g, parameters p)}$$

$$\Delta I = \mathbf{M}(p_t, I_t)\,\Delta p \qquad \text{(local linearization } \mathbf{M}\text{)}$$

⇨ Define an error

$$e_{t+1} = g(p_t, I_t) - I_0$$

M is N x m and
is time varying!

⇨ Close the loop

$$p_{t+1} = p_t - (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \; e_{t+1} \quad \text{where } \mathbf{M} = \mathbf{M}(p_t, I_t)$$

A Factoring Result

(Hager & Belhumeur 1998)

Suppose $I = g(I_t, p)$ at pixel location u is defined as

$$I(u) = I_t(f(u,p))$$

and

$$\frac{\partial f}{\partial u}^{-1} \frac{\partial f}{\partial p} = \mathbf{L}(u)\mathbf{S}(p)$$

Then

$$\mathbf{M}(p, I_t) = \mathbf{M}_0\, S(p) \quad \text{where } \mathbf{M}_0 = \mathbf{M}(0, I_0)$$

Stabilization Revisited

⇨ In general, solve

$$[\mathbf{S}^T\ \mathbf{G}\ \mathbf{S}]\ \Delta p = \mathbf{M}_0^T\ e_{t+1} \quad \text{where } \mathbf{G} = \mathbf{M}_0^T\mathbf{M}_0 \quad \text{constant!}$$

$$p_{t+1} = p_t + \Delta p$$

⇨ If S is invertible, then

$$p_{t+1} = p_t - \mathbf{S}^{-T}\ \mathbf{G}\ \ e_{t+1} \quad \text{where } \mathbf{G} = (\mathbf{M}_0^T\mathbf{M}_0)^{-1}\mathbf{M}_0^T$$

G is m x N,
e is N x 1
S is m x m

→ O(mN) operations

Stabilization Revisited

⇨ In general, solve

$[\mathbf{S}^T\ \mathbf{G}\ \mathbf{S}]\ \Delta p = \mathbf{M}_0^T\ e_{t+1}$ where $\mathbf{G} = \mathbf{M}_0^T\mathbf{M}_0$ constant!

$p_{t+1} = p_t + \Delta p$

⇨ If S is invertible, then

$p_{t+1} = p_t - \mathbf{S}^{-T}\ \mathbf{G}\ \ e_{t+1}$ where $\mathbf{G} = (\mathbf{M}_0^T\mathbf{M}_0)^{-1}\mathbf{M}_0^T$

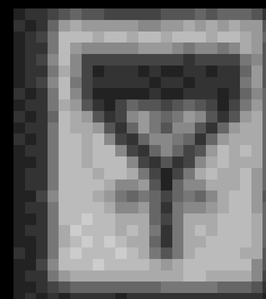
G is constant!

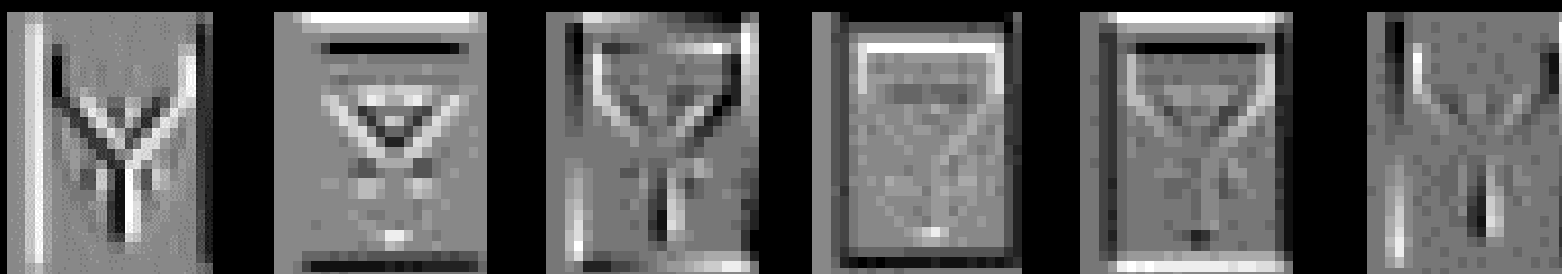
S is small and time varying

→ Local asymptotic stability!

On The Structure of M

Planar Object -> Affine motion model: $u'_i = \mathbf{A} u_i + d$



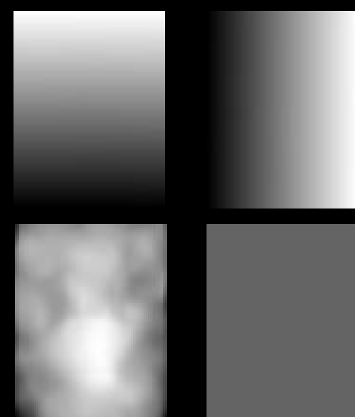


X **Y** **Rotation** **Scale** **Aspect** **Shear**

$$M(p) = \partial g / \partial p$$

3D Case : Global Geometry

Non-Planar Object: $u_i = \mathbf{A} u_i + b z_i + d$



Observations:

- Image coordinates lie in a 4D space
- 3D subspace can be fixed
- Motion in two images gives affine structure

3D Case: Local Geometry

Non-Planar Object: $u_i = \mathbf{A}\, u_i + b\, z_i + d$

x y rot z scale aspect rot x rot y

Tracking 3D Objects

What is the set of all images of a 3D object?

Motion | **Illumination** | **Occlusion**

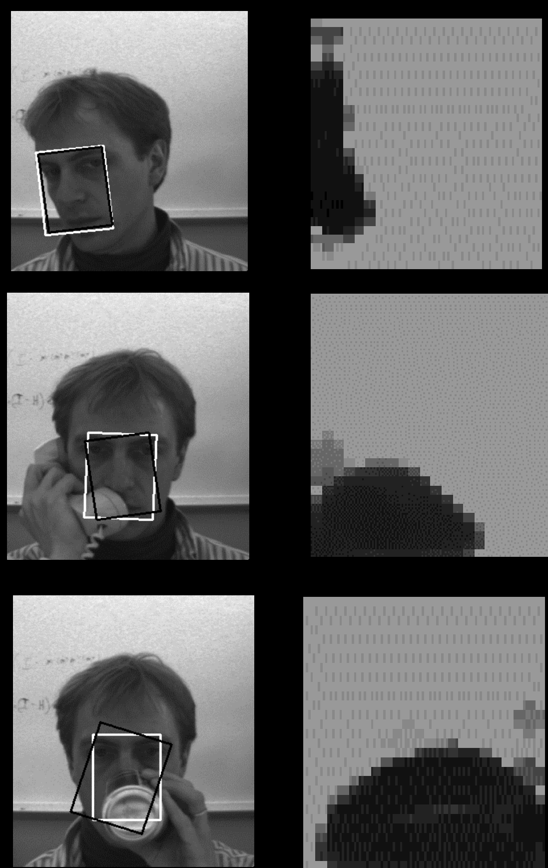
3D Case: Illumination Modeling

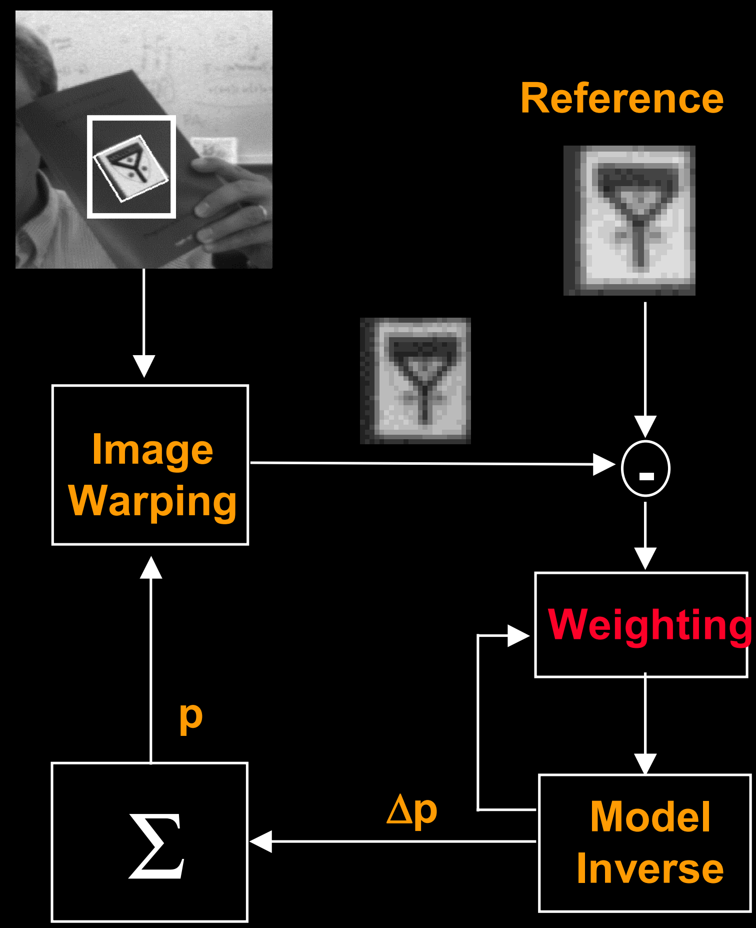
Non-Planar Object: $I_t = \mathbf{B}\,\alpha + I_0$

Observations:

- Lambertian object, single source, no cast shadows => 3D image space
- With shadows => a cone
- Empirical evidence suggests 5 to 6 basis images suffices

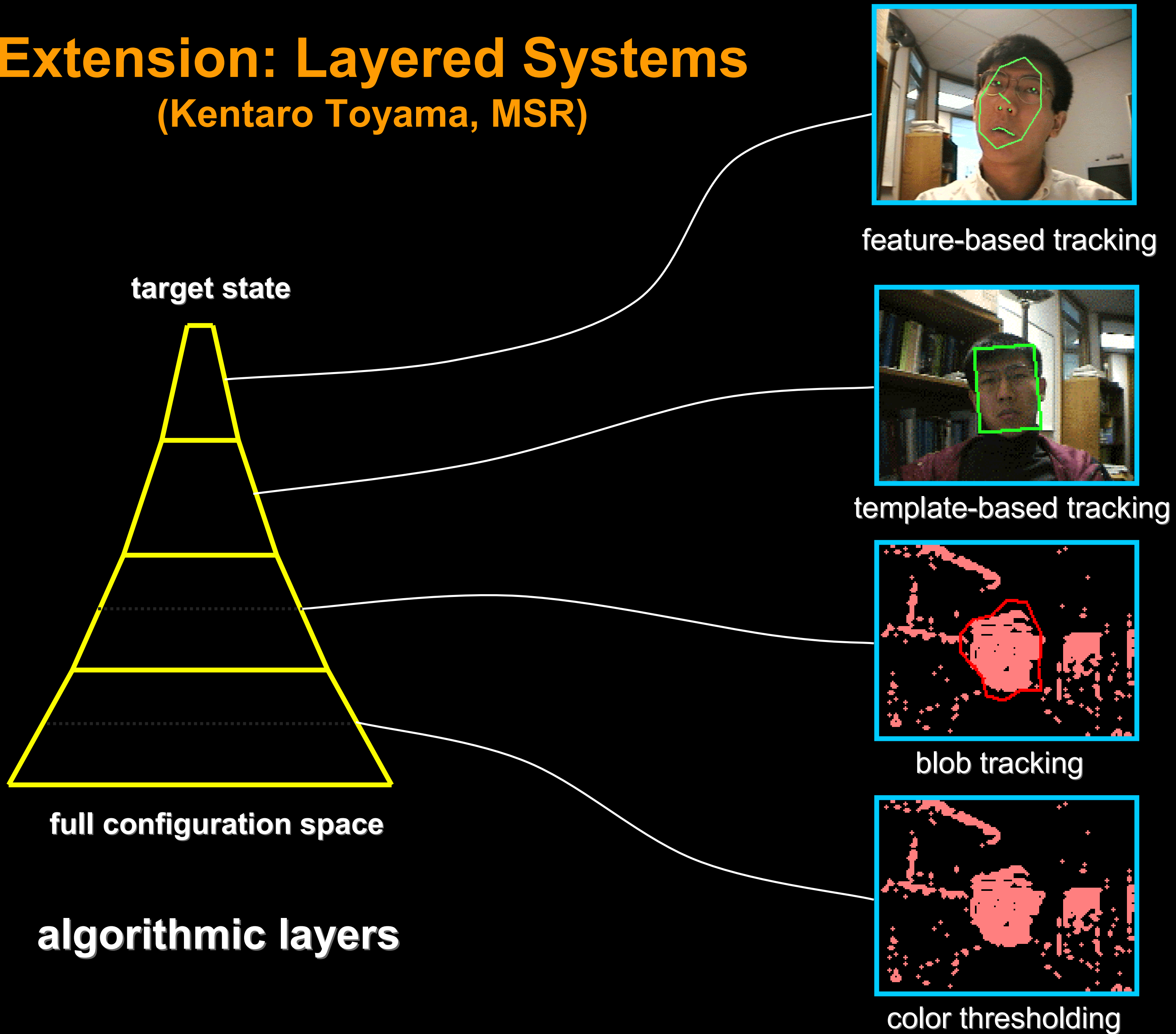
Handling Occlusion





A Complete Implementation

Extension: Layered Systems
(Kentaro Toyama, MSR)



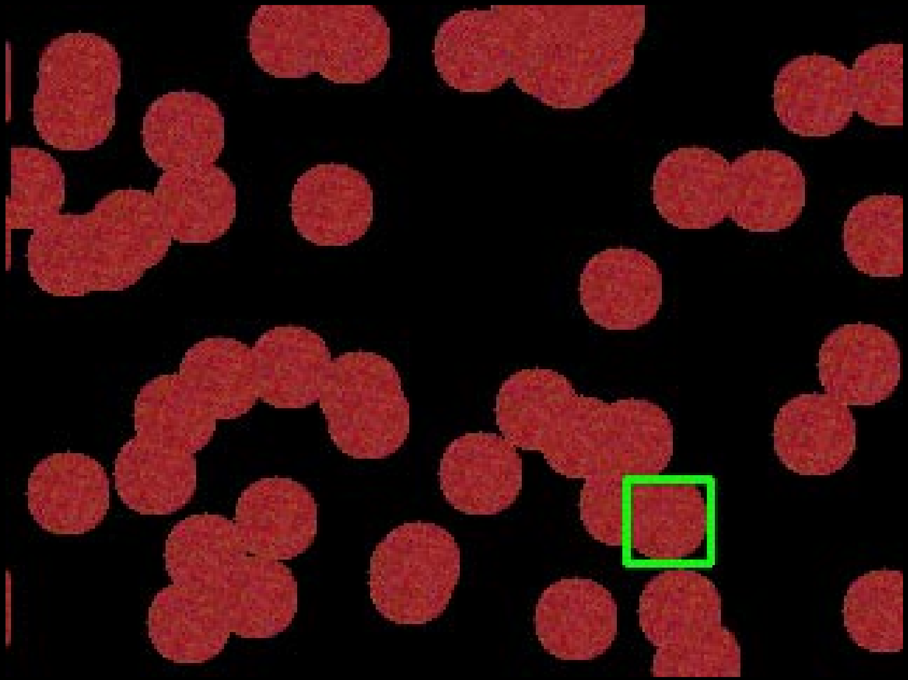
Layered System: Example

Green: tracking   Red: searching

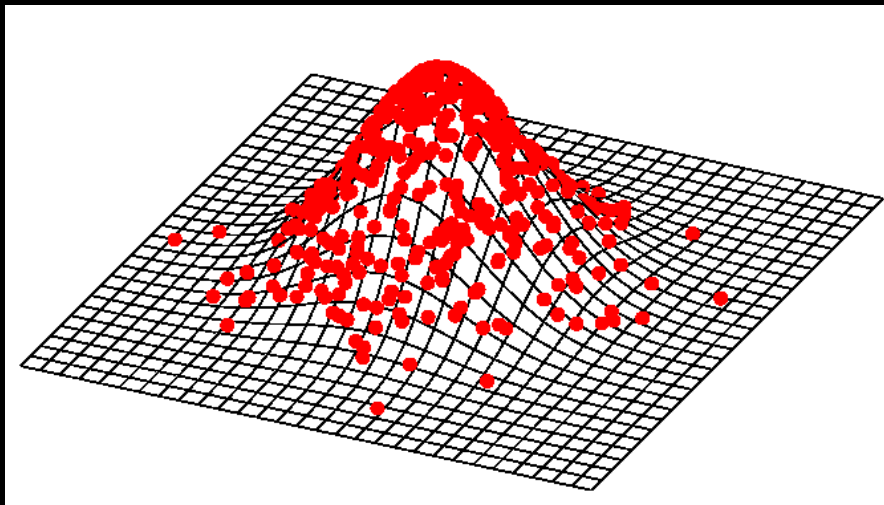
Extension: Dealing with Distraction

(Christopher Rasmussen, NIST)

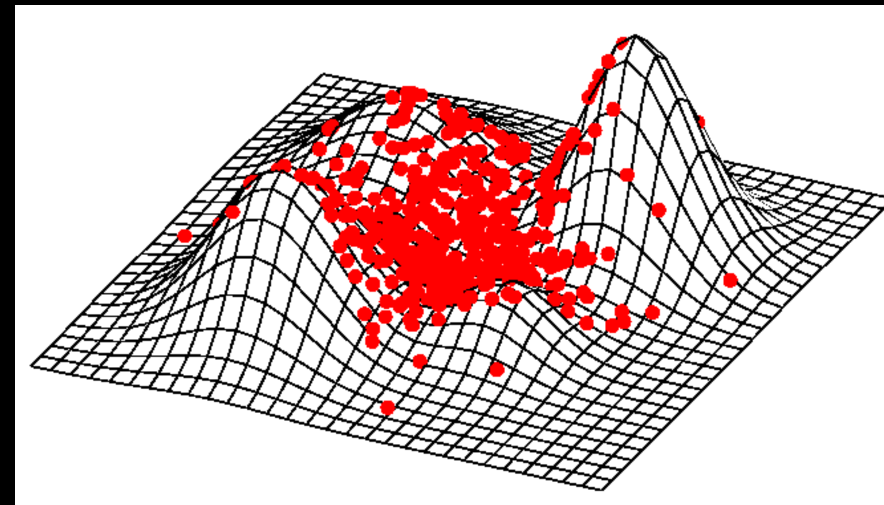
Tracking an orbit (50 distractors)
1 measurement: 5/20 successes
10 measurements: 17/20



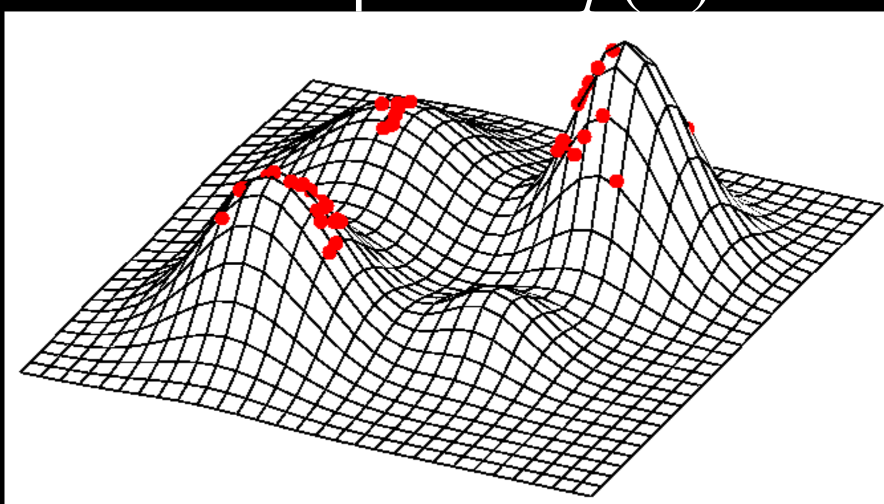
Measurement Generation



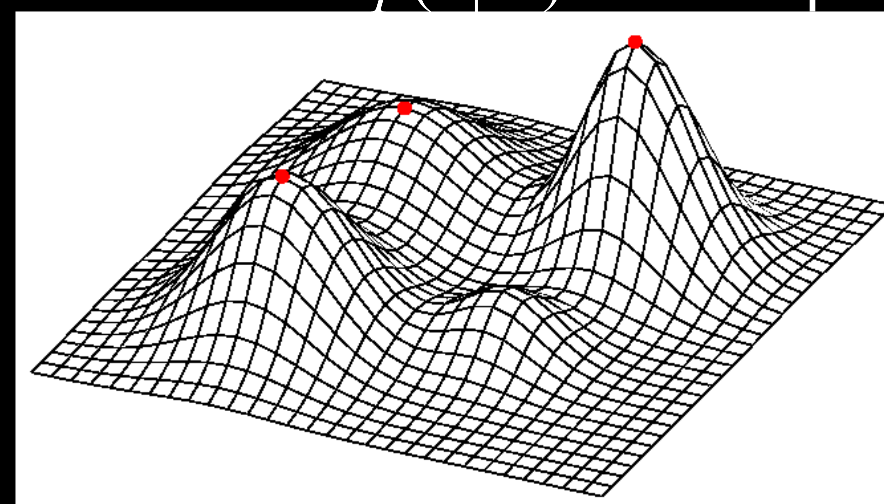
Sample from $p(\mathbf{X})$



Evaluate $p(\mathbf{I} \mid \mathbf{X})$ at samples



Keep high-scoring samples



Ascend gradient & pick exempla

Measuring: Textured Regions

Predicted state

Initial samples

Top fraction

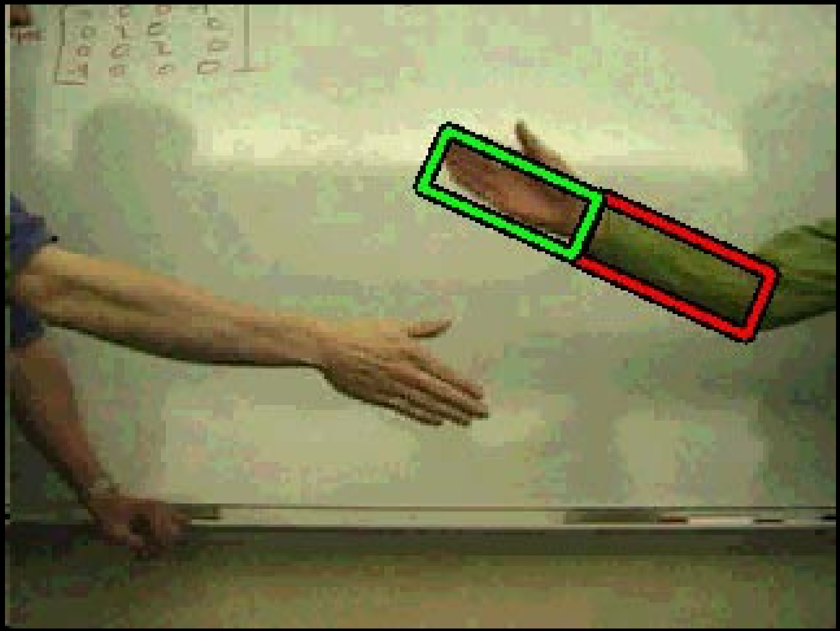
Hill-climbed

Example: Combined homogeneous region & contour trackers

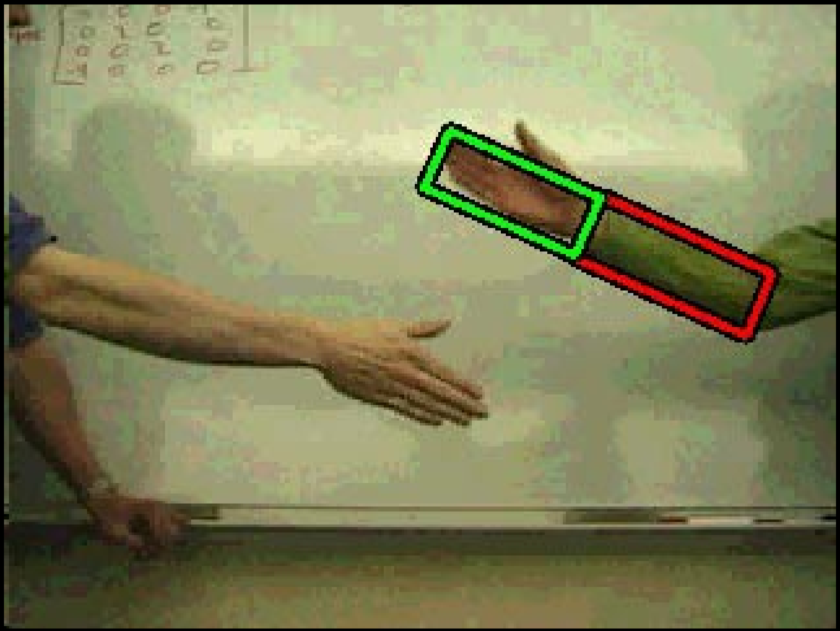
Homogeneous region

Homogeneous region and snake

Example: Hinge between homogeneous regions



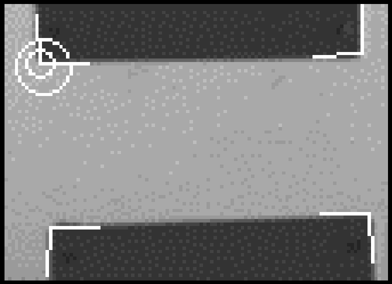
JLF



CJLF

VISUAL TRACKING: Programming

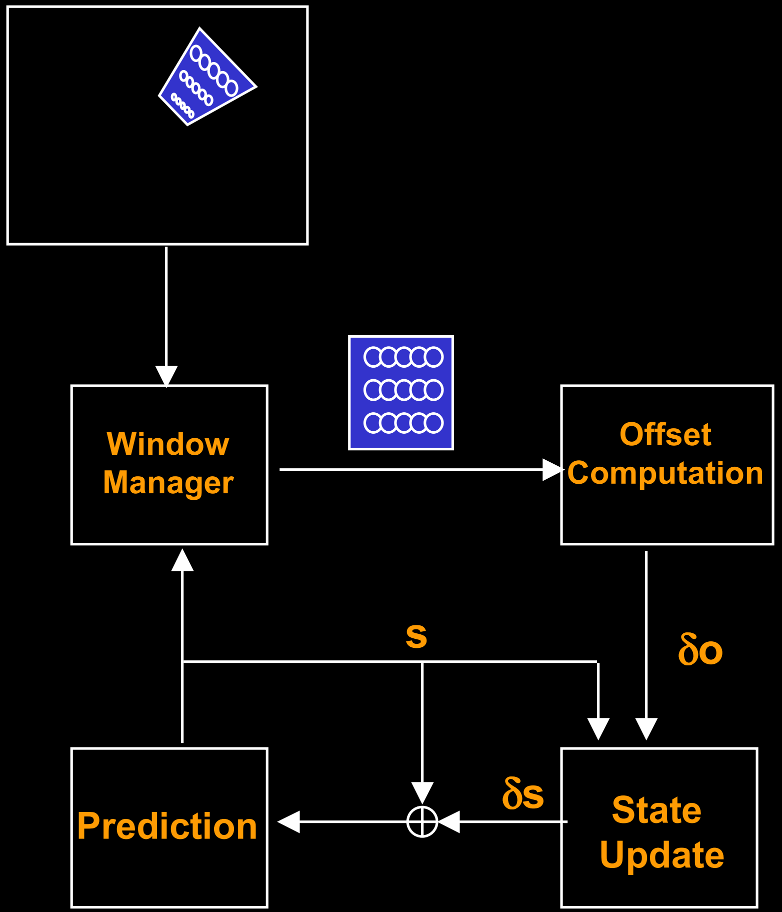
XVision: Desktop Feature Tracking



- Graphics-like system
  - Primitive features
  - Geometric constraints
- Fast local image processing
  - Perturbation-based algorithms
- Easily reconfigurable
  - Set of C++ classes
  - State-based conceptual model of information propagation
- Goal
  - Flexible, fast, easy-to-use substrate

Abstraction: Feature Tracking Cycle

- ⇨ Prediction
  prior states predict new appearance
- ⇨ Image rectification
  generate a “normalized view”
- ⇨ Offset computation
  compute error from nominal
- ⇨ State update
  apply correction to fundamental state



Abstraction: Feature Composition

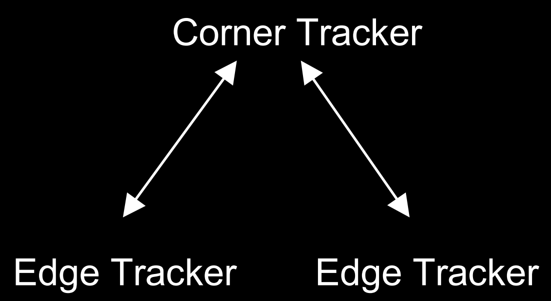
- ⇨ Features related through a projection-embedding pair

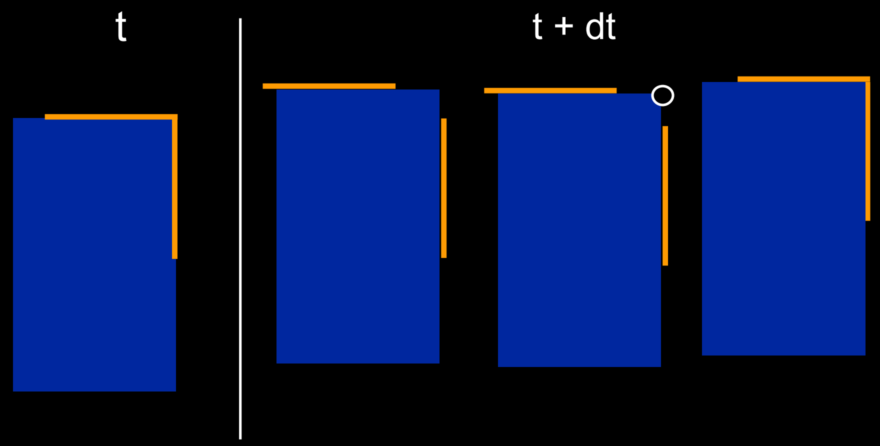
  an $f: R^n \rightarrow R^m$, and $g: R^m \rightarrow R^n$, with $m \leq n$ s.t. $f \circ g$ = identity

- ⇨ Example: corner composed of two edges

  each edge provides one positional parameter and one orientation.

  two edges define a corner with position and 2 orientations.

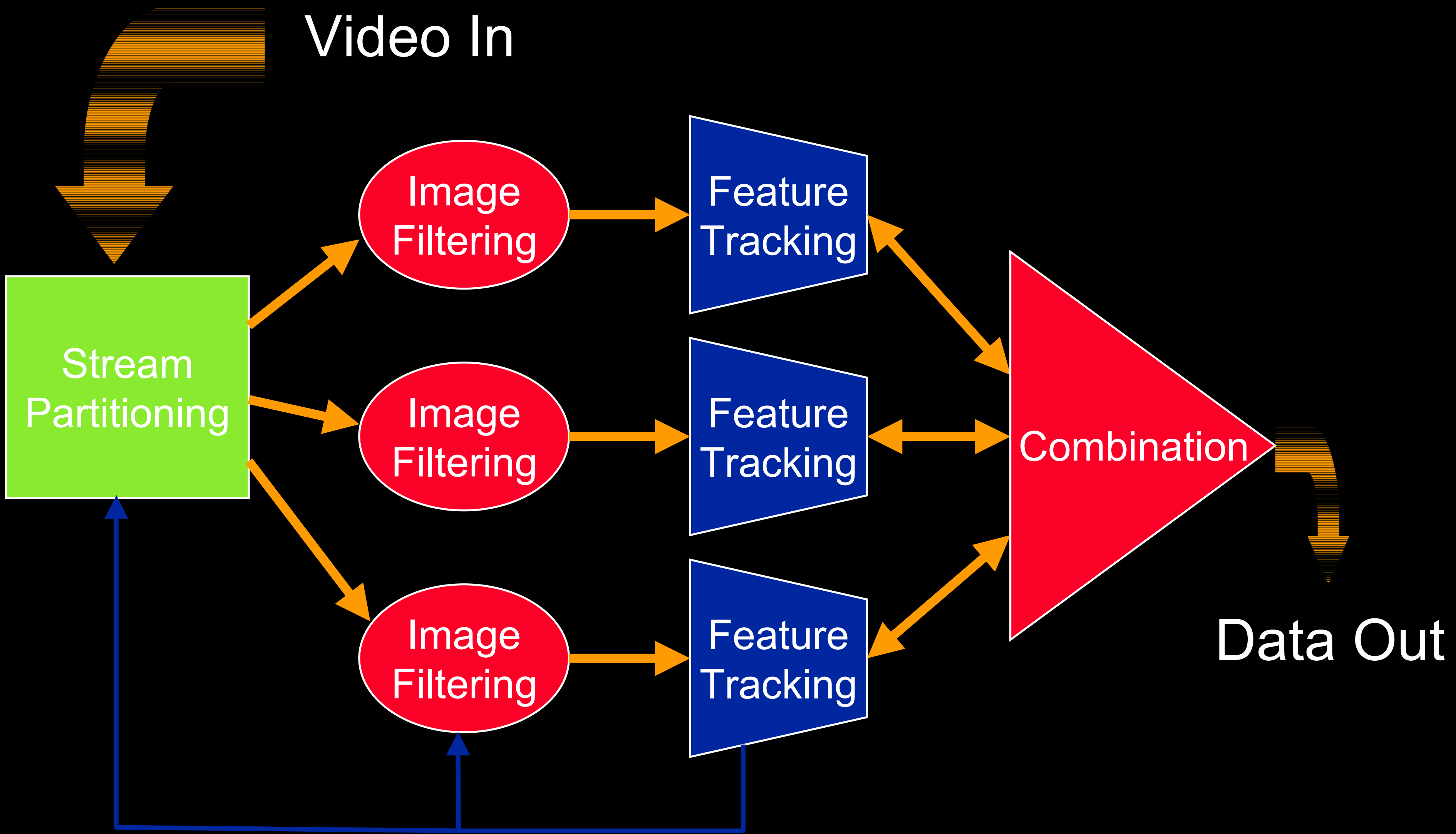




XVision2

- ⇨ New camera interfaces
  - – Firewire
  - – Stereo
- ⇨ Extensive support for color
  - – RGB, YUV, …
  - – Abstract class support for non-intensive applications
- ⇨ New programming models
  - – Separation of feature detection and tracking
  - – Dataflow model innate to system

New XVision Programming Model



APPLICATIONS

Mobile Navigation
(Darius Burschka)

Sensor-Based Control

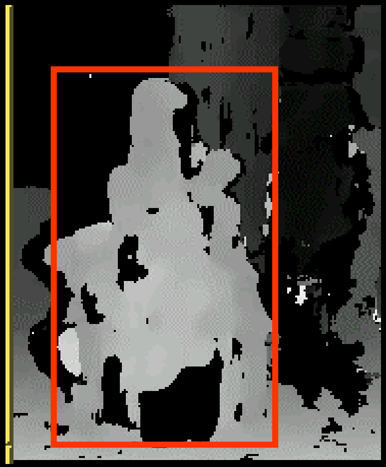
control signals for the robot are generated directly from the visual input

$$O_i = \mathfrak{I}^{-1} \cdot I_i$$

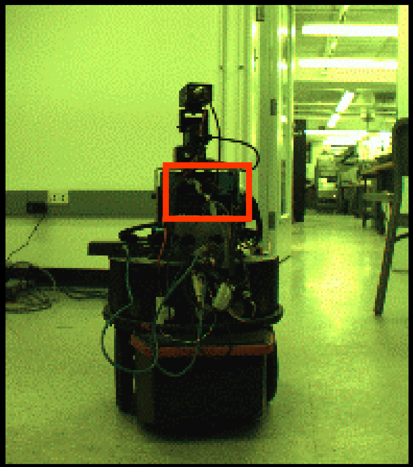
Map-Based Navigation

pre-processed sensor data is stored in a geometrical representation of the envi-ronment (map). Path plan-ning+strategy algorithms are used to define the actions of the robot

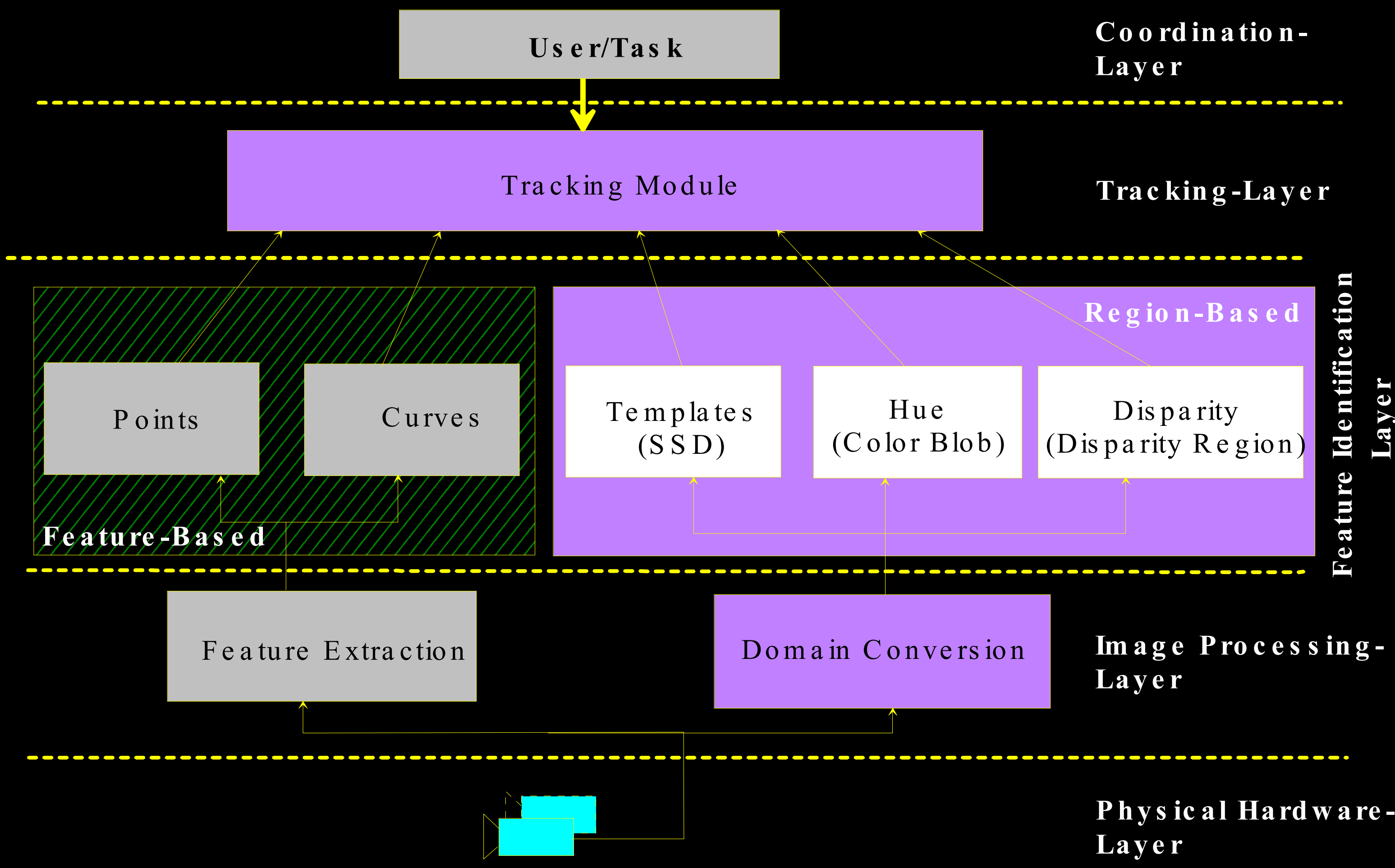
Dynamic Composition of Tracking Cues



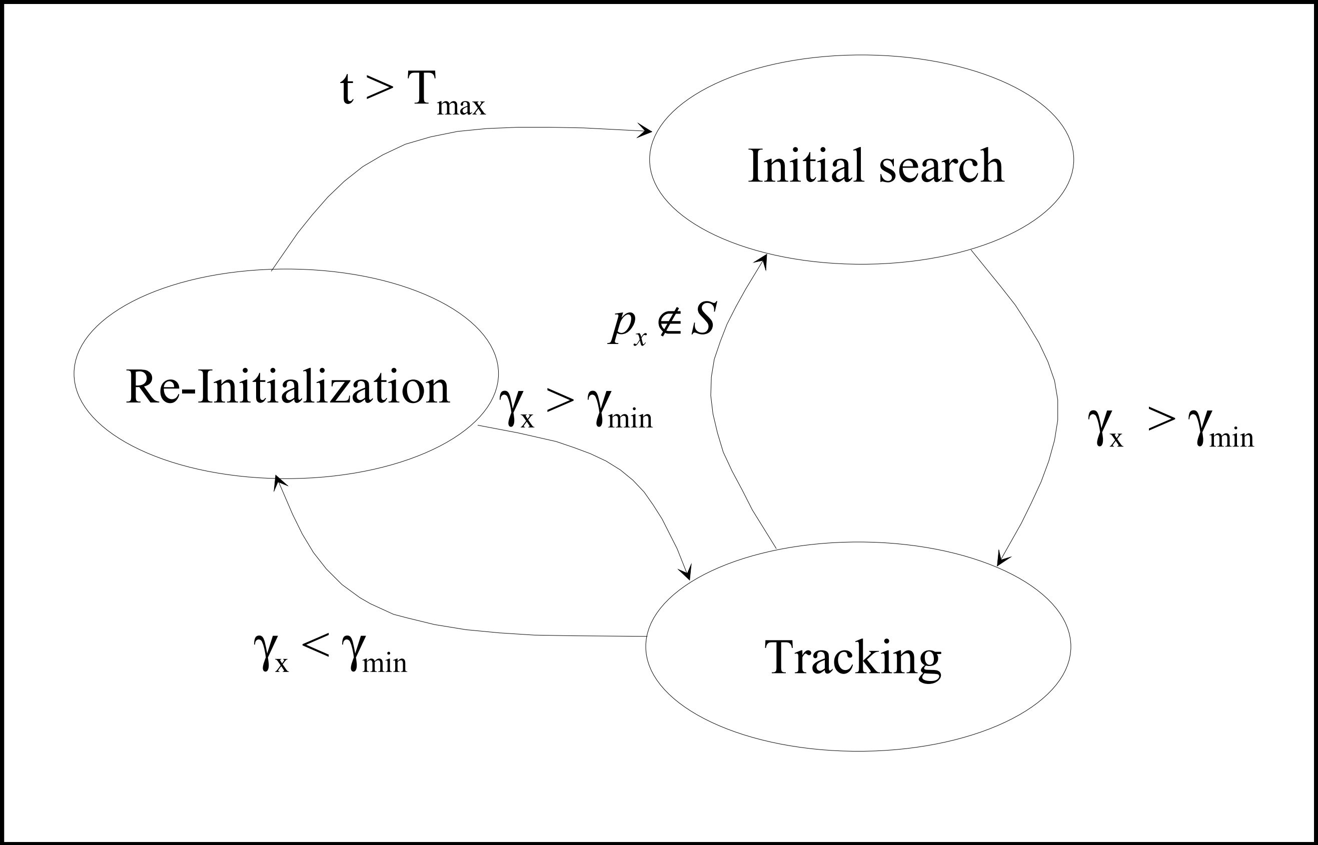
Basic problem: no single cue suffices



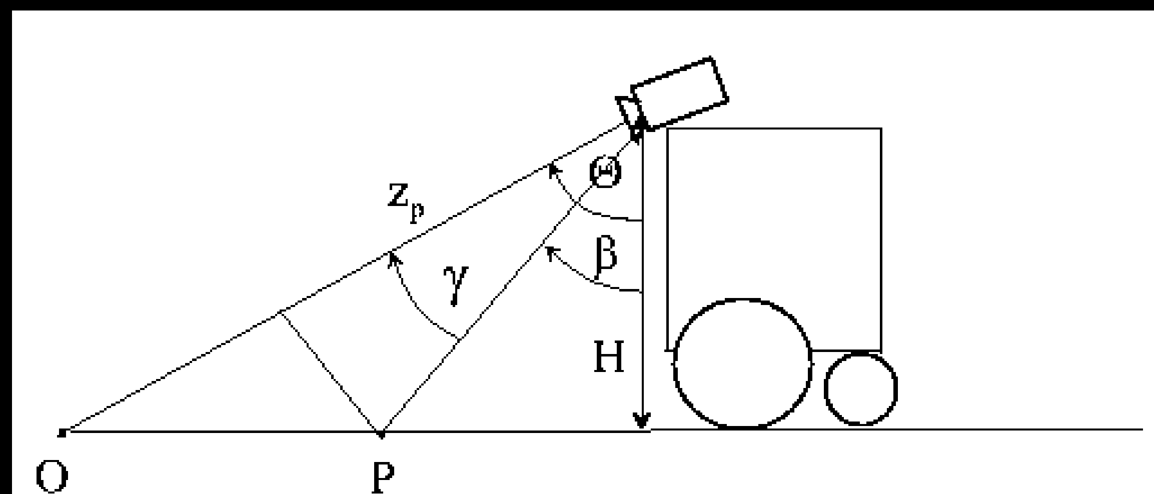
Tracking-System Architecture



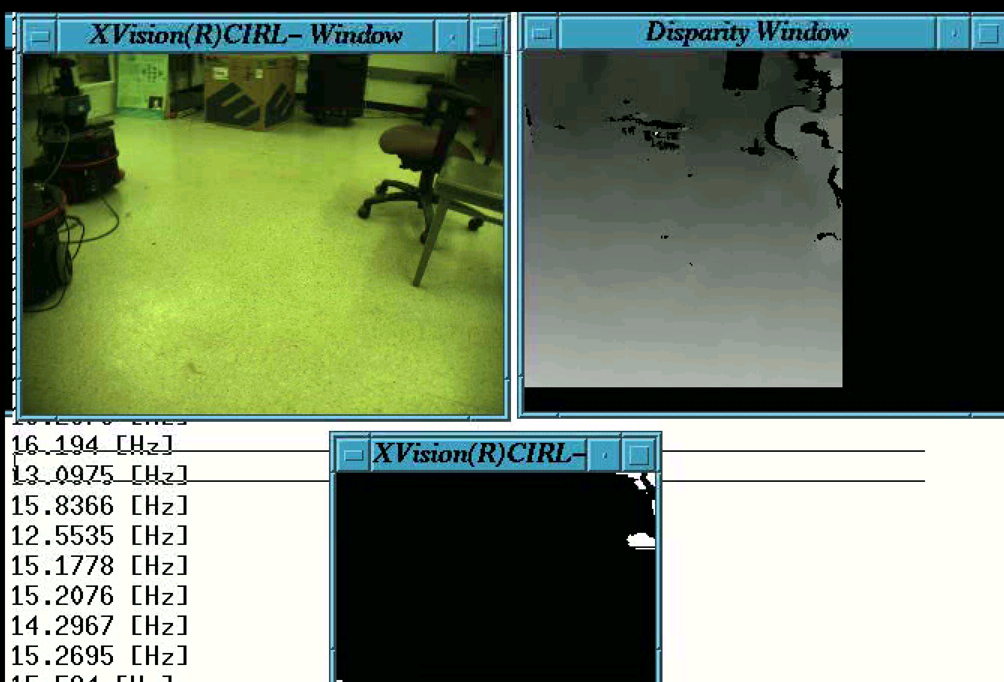
State Transitions in the Tracking Process



Problem in the Disparity Domain



Results Obstacle Detection



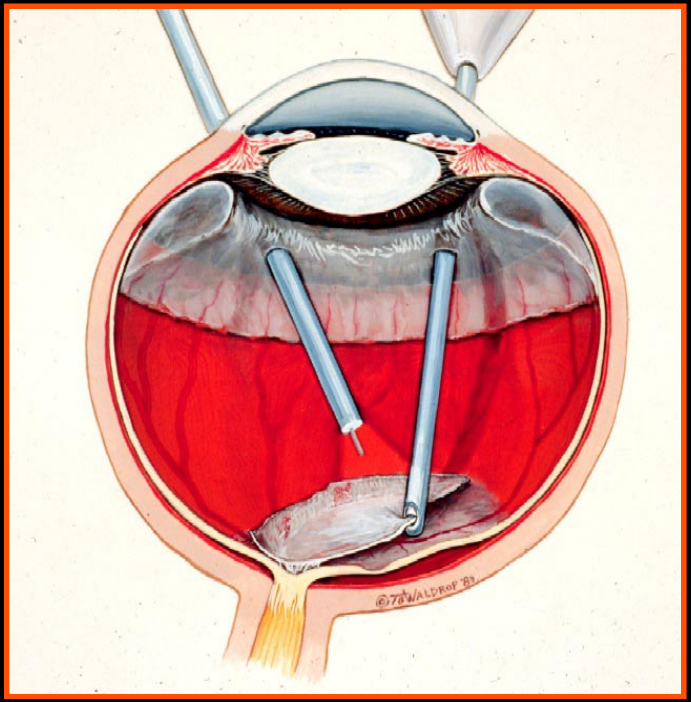
Results Dynamic Composition

scene	disparity γ_i	color γ_i
before door	0.33	0.32
in door	0.22	0.33
behind door	0.42	0.30

Medical Systems

Two basic questions:

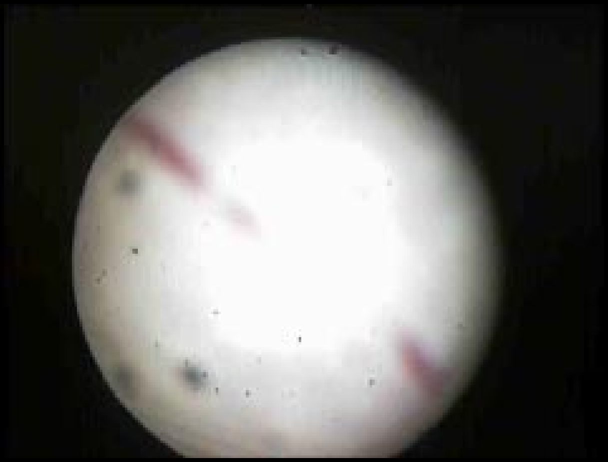
1. Sensory augmentation: How can we use vision techniques to provided integrated information display to surgeon.
2. Physical augmentation: How can we physically improve surgeon dexterity?



Medical Systems

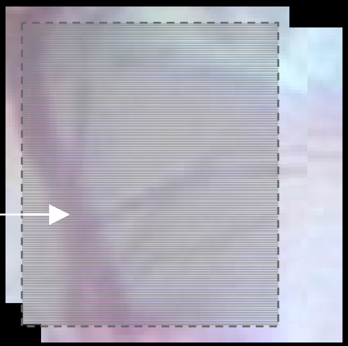
Two basic questions:

1. Sensory augmentation: How can we use vision techniques to provided integrated information display to surgeon.
2. Physical augmentation: How can we physically improve surgeon dexterity?



Some Observations

Common reference area



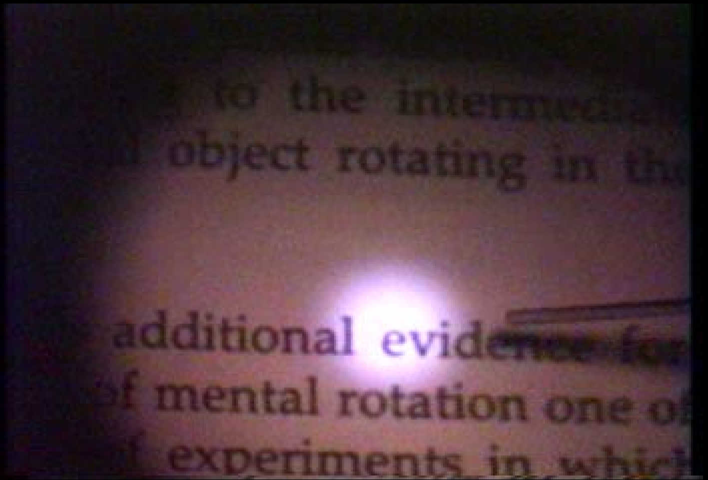
Tracking:

Quick tracking from common reference area provides initial fit

$$\delta_\mu = -(M^T M)^{-1} M^T \left[I(\mu, t+\tau) - I(0, t_0) \right]$$

Medical Applications
(Myron Brown, APL)

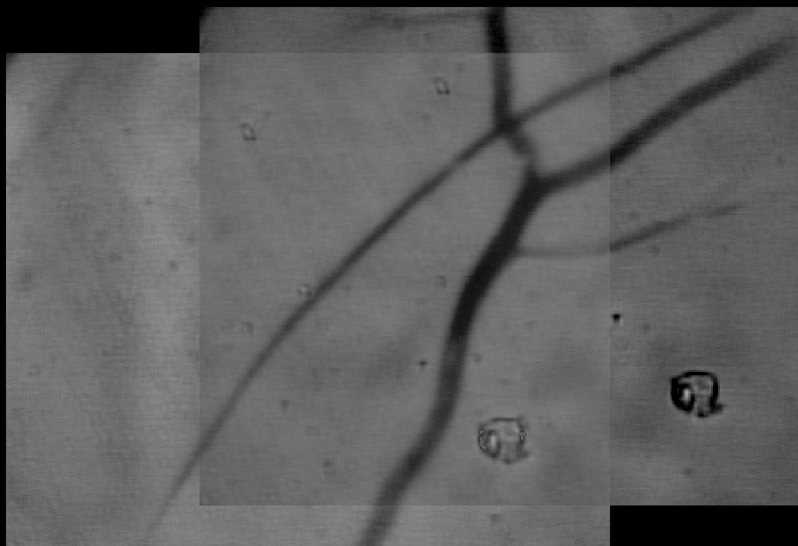
Endoscopic Mosaic

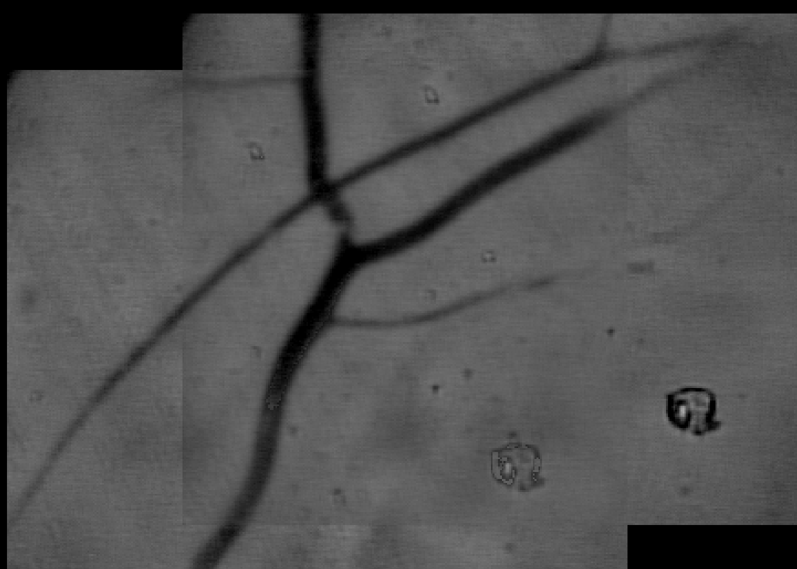


Mosaic of Vitreoretinal Sequence

- Basis of registration is tracking
  - additional "feed-forward" term including robot motion
- Combination through warping
  - locally stabilized images "averaged" together to produce a mosaic
- Problem: stark changes in illumination

Mosaic of Vitreoretinal Sequence

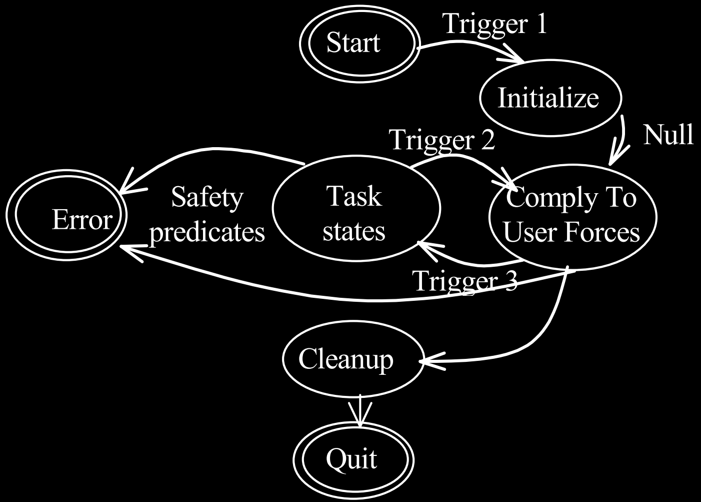


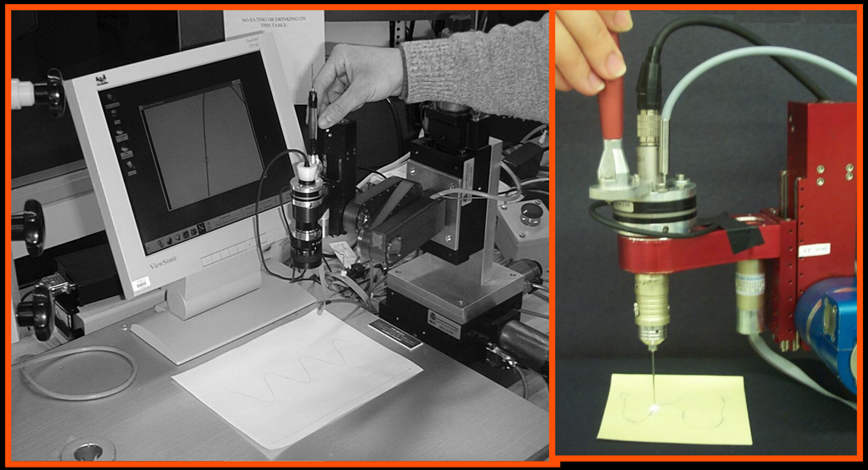


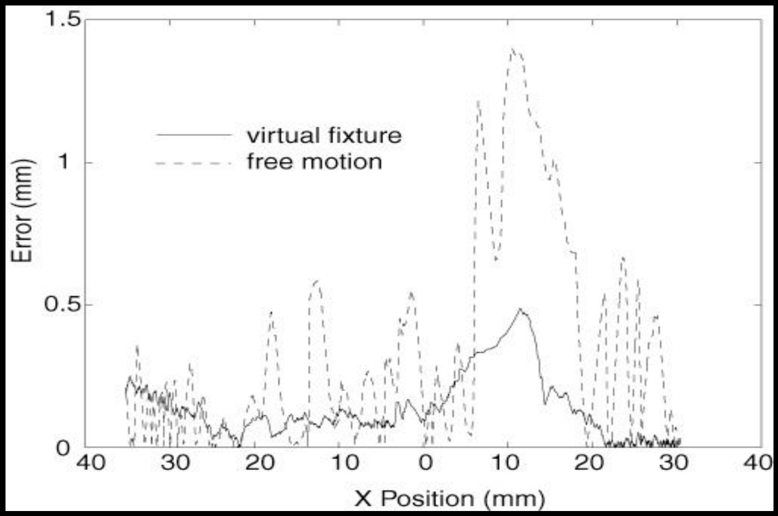
- ⇨ Basis of registration is tracking
  - – additional “feed-forward” term including robot motion
- ⇨ Combination through warping
  - – locally stabilized images “averaged” together to produce a mosaic
- ⇨ Solution: non-parametric illumination model

Human-Machine Cooperative Systems

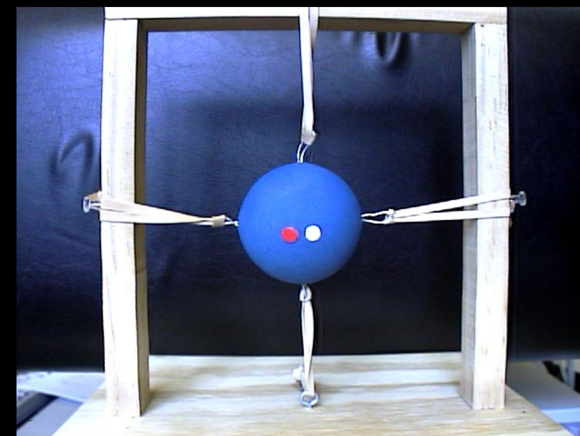
Goal: To augment surgeons' ability to perform *complex* procedures through sensor-based feedback

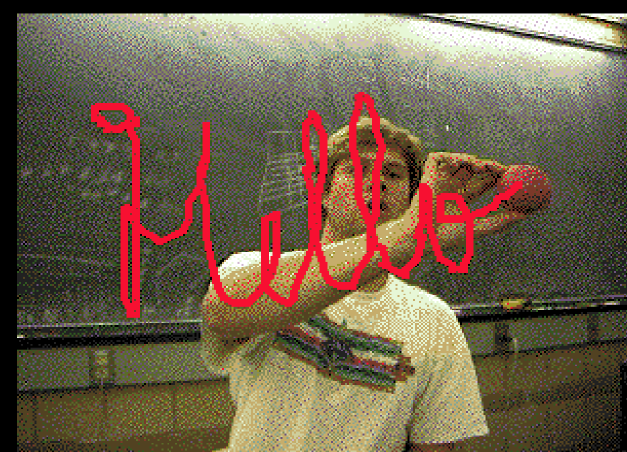






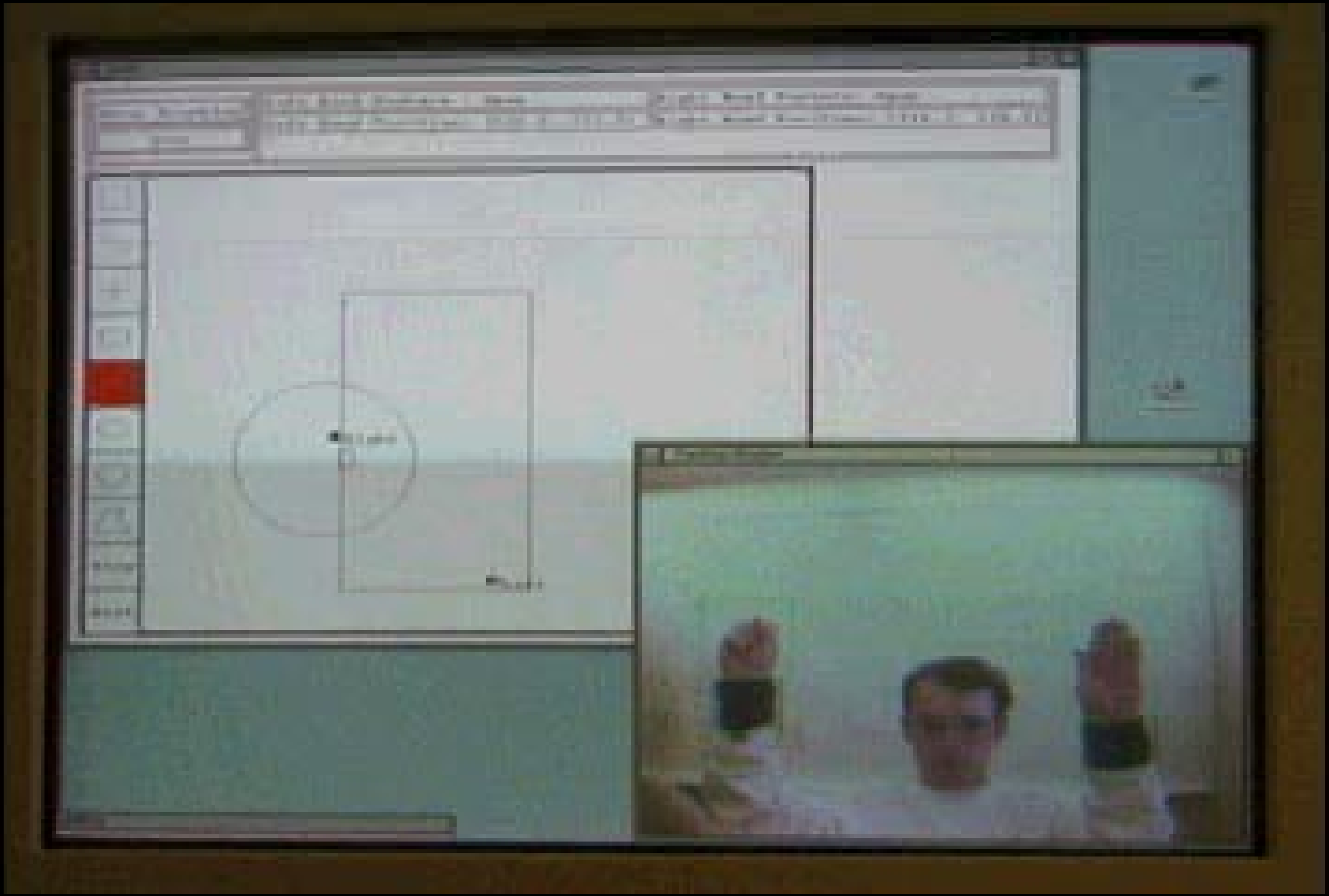
Human-Computer Interaction





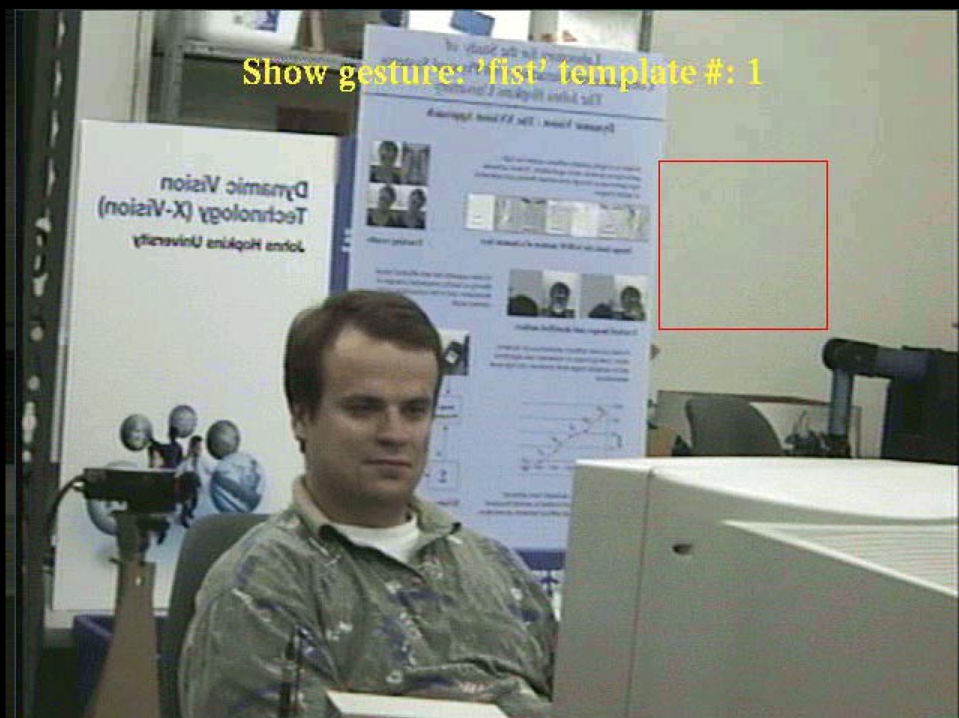
Human-Computer Interaction

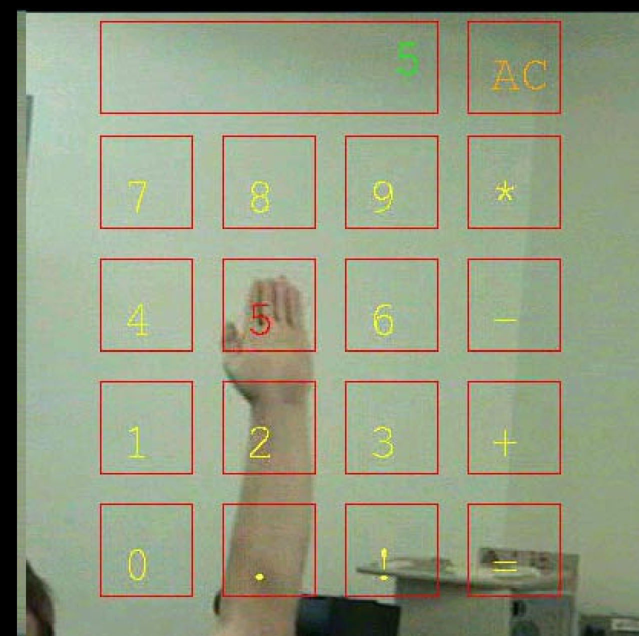
"The Past"



Visual Interaction Cues
(Jason Corso)

Idea: Develop a set of shared visual representations that support *local* processing and structured context





Applications of Computer Vision: Image Databases

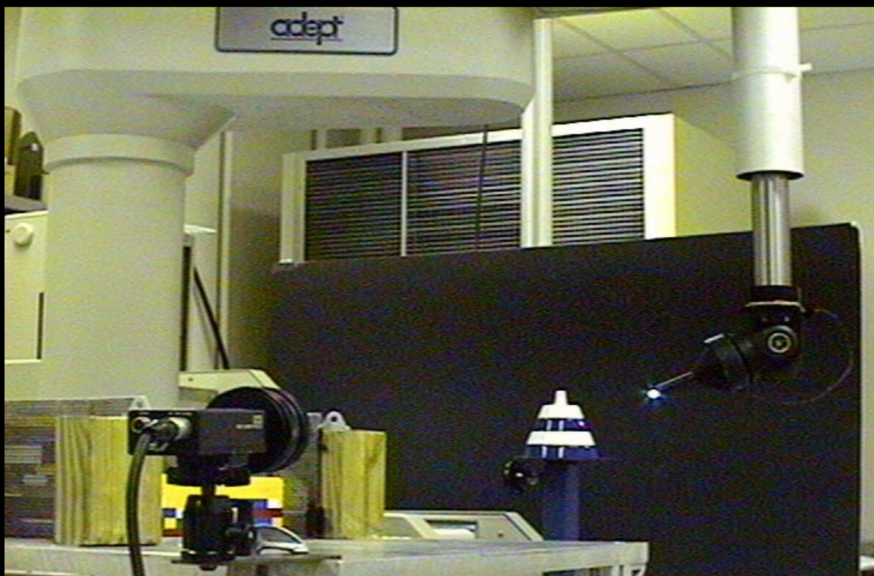
(Courtesy D. Forsyth & J. Ponce)

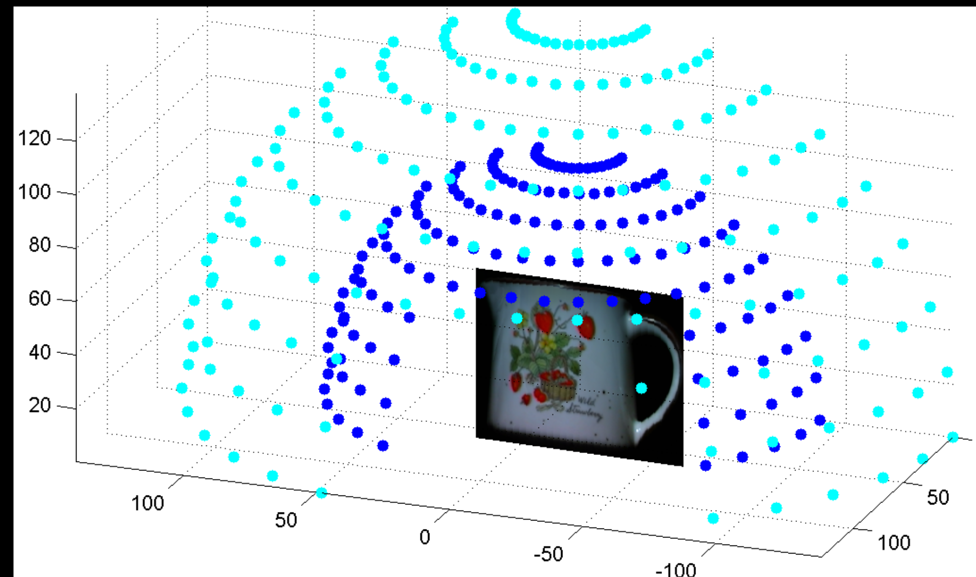
From a search for horse pix in 100 horse images and 1086 non-horse images

Applications of Computer Vision: Data Acquisition

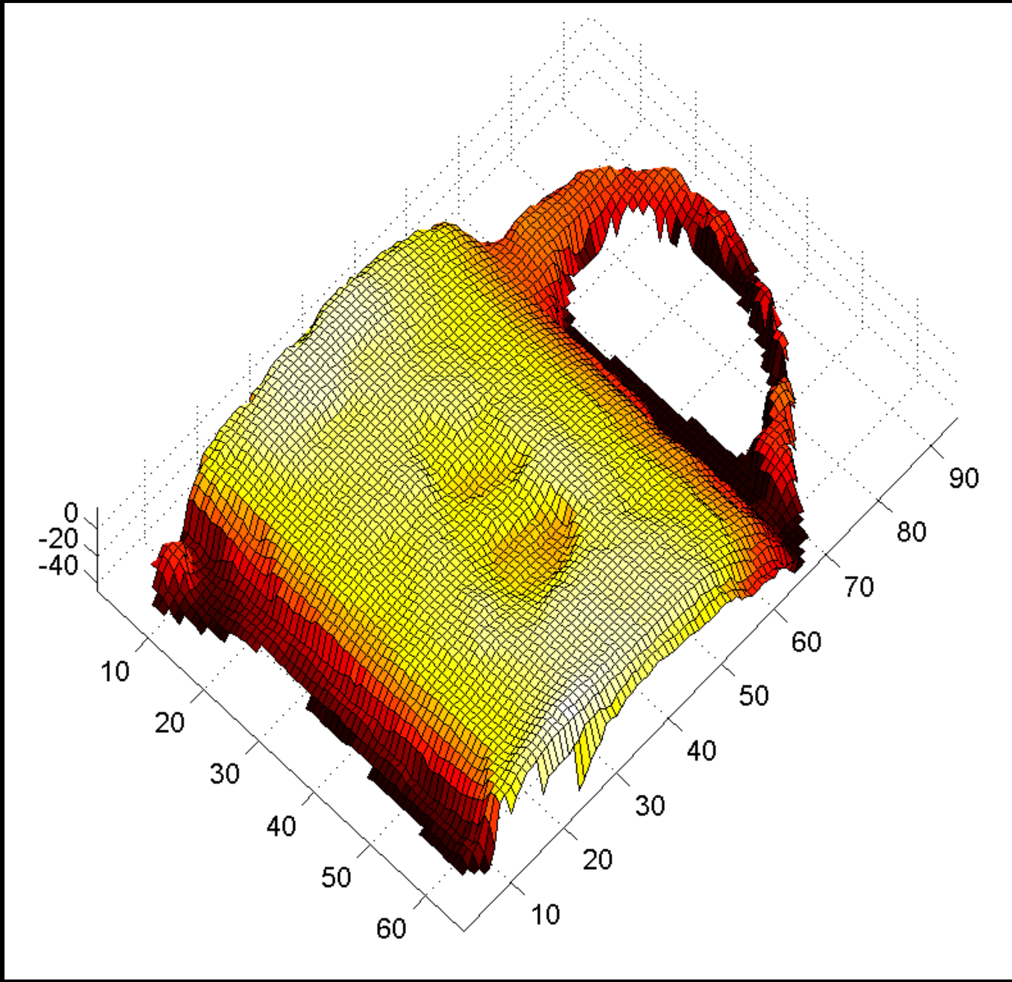
Applications of Computer Vision: Motion Control

Applications of Computer Vision: Rendering





A Reconstructed Depth Map

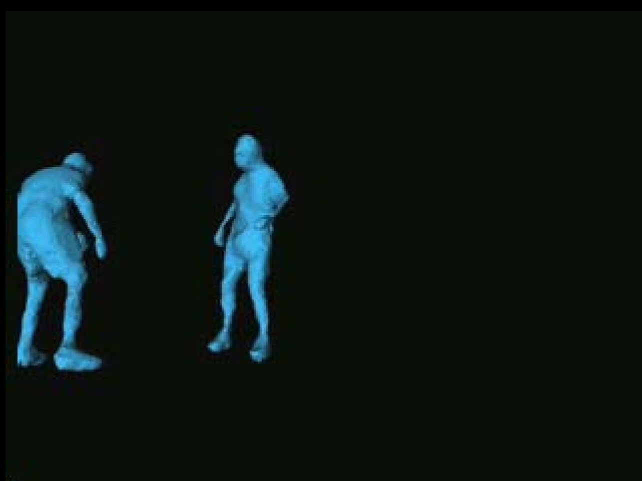


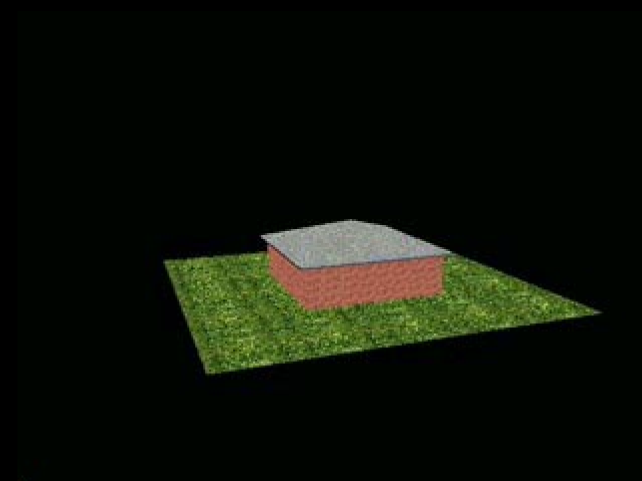
143 Images on each surface

Rendered Images

Synthetic Sequences

Application of Computer Vision: CMU virtualized reality





The Challenge

Develop a system that can:

1. Deal with approx. 30 objects in a “generic” fashion
2. Interact with a human both spatially and iconically
3. Interact with the physical world in a controlled, reliable, and SAFE fashion.