Hierarchies in representations Latent structure discovery and Dimensionality reduction

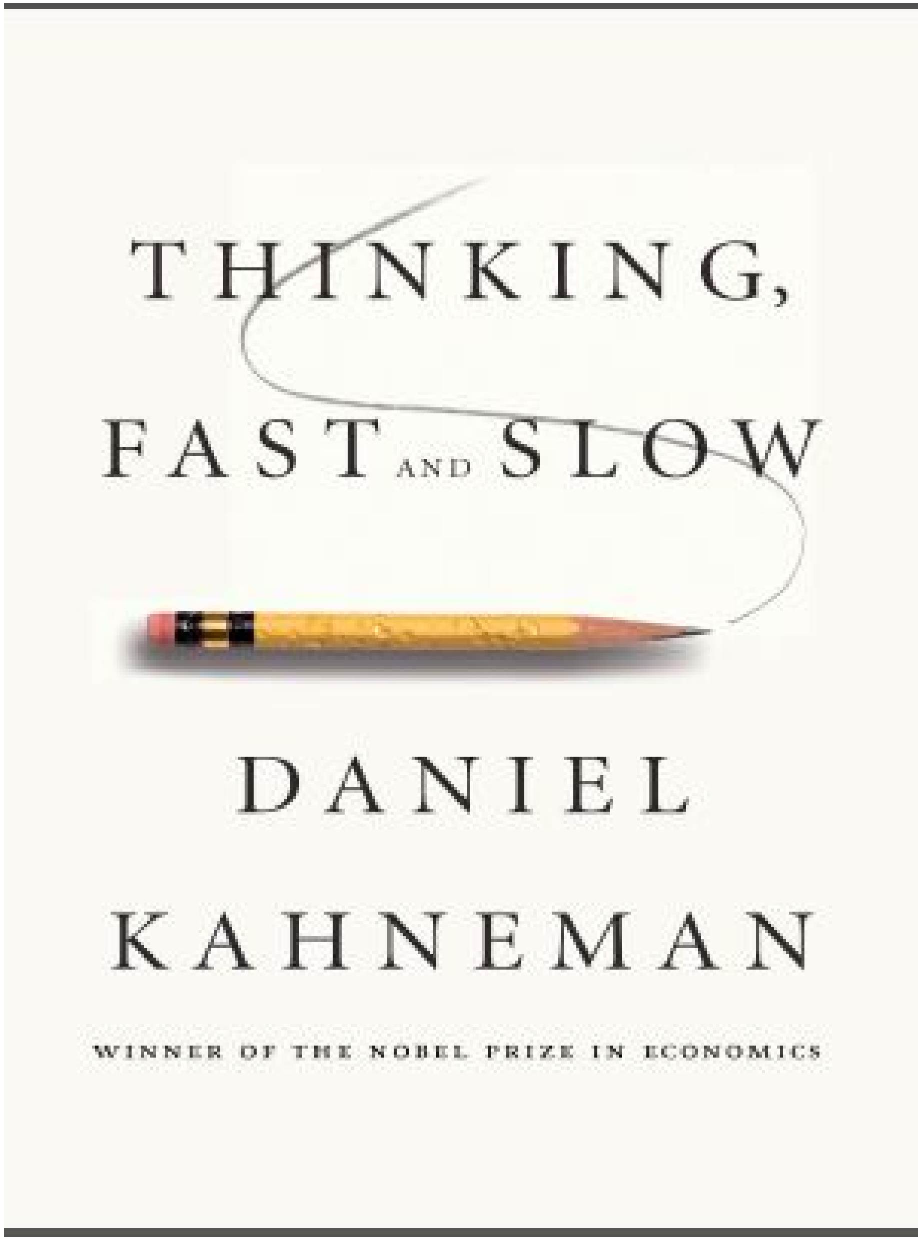
Amitabha Mukerjee

IIT Kanpur, India

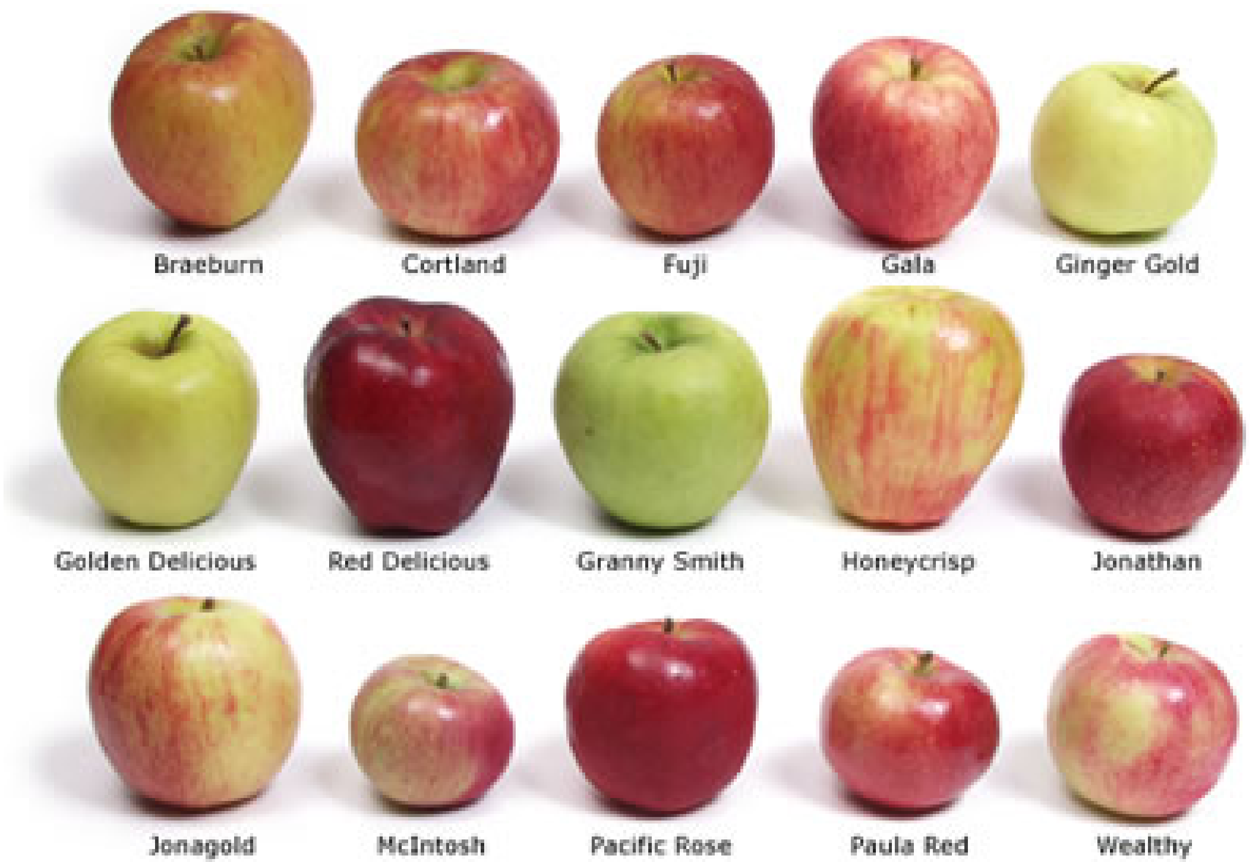
Tacit
Knowing
(Thinking fast)

→

Latent
Relations



Latent Representation

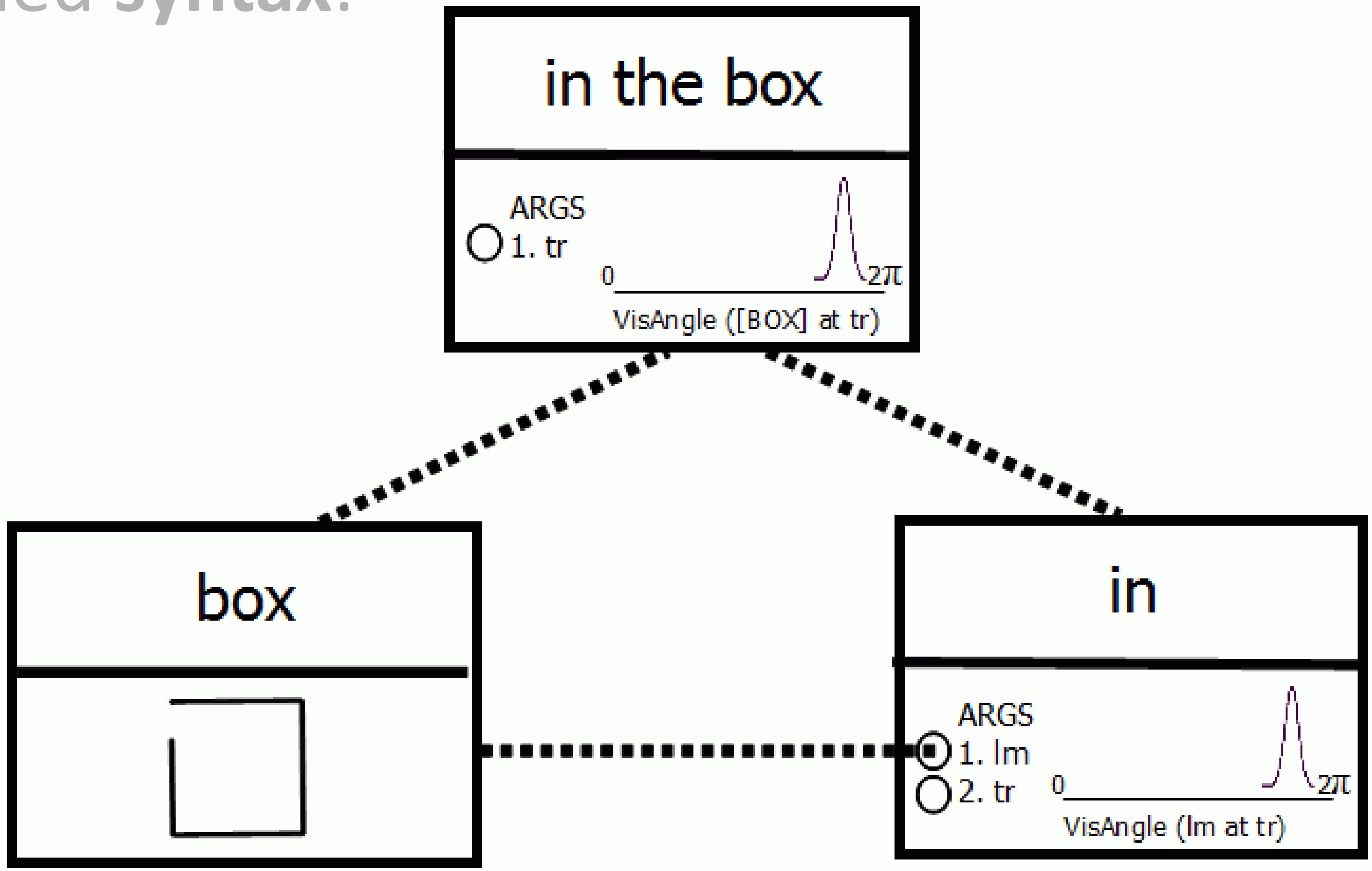


images: 100 x 100 pixels

Ack: A. Efros, original images from hormel corp.

Manifolds in language

- grounded **syntax**:



[Nayak and Mukerjee AAAI-12]

Visuo-Motor expertise

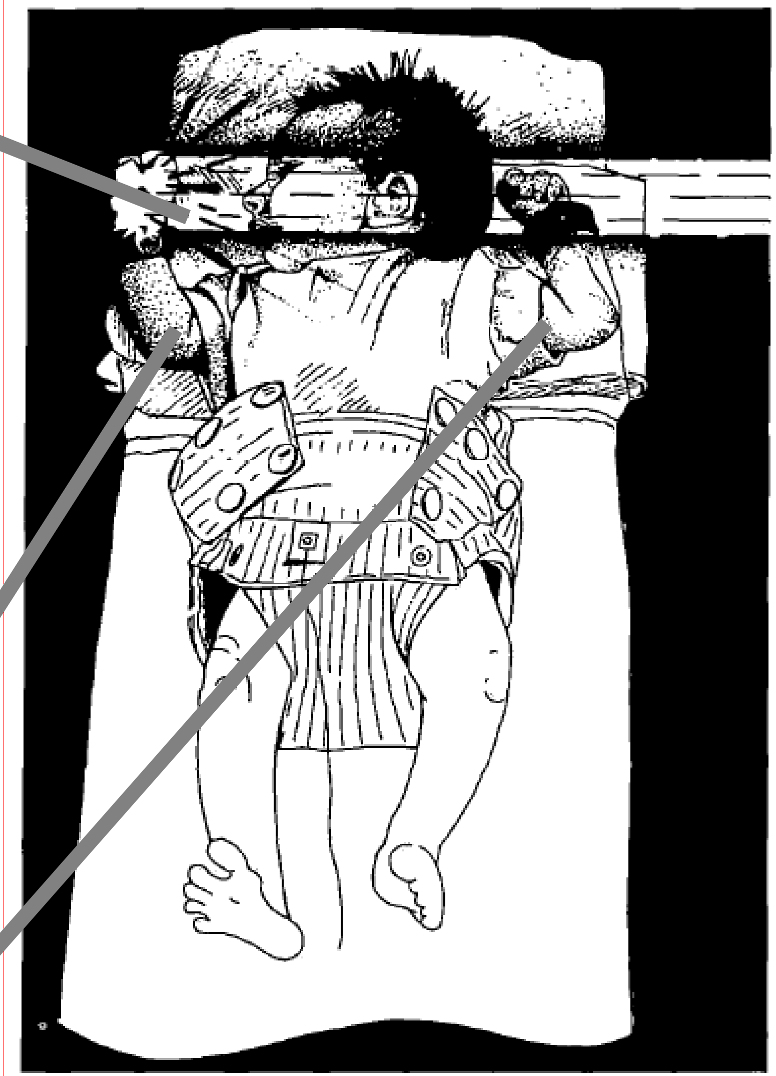
in darkened room,
works hard to position arm
in a narrow beam of light

Newborns
(10-24 days)

Small weights
tied to wrists

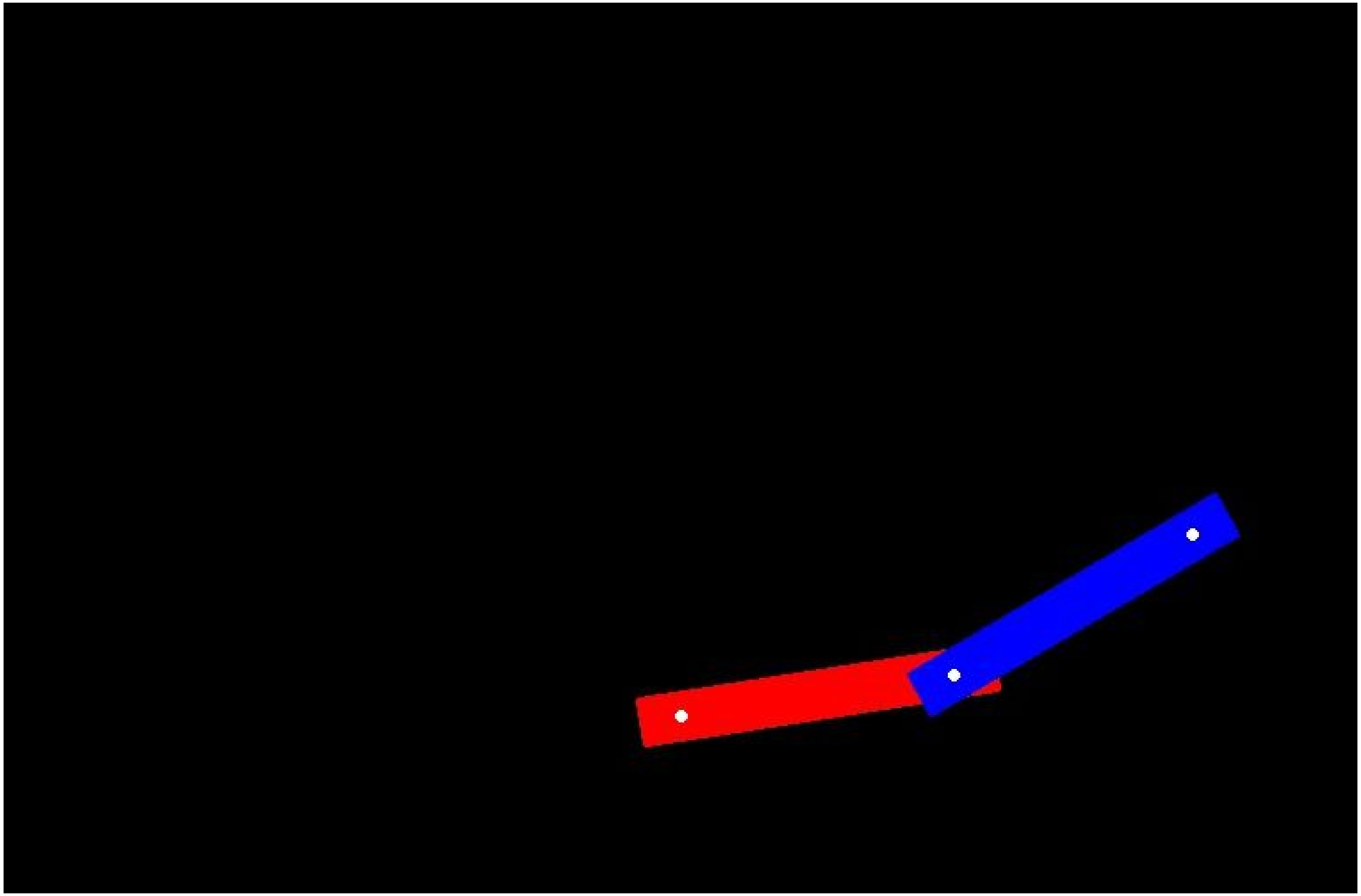
Will resist weights to move
the arm they can see

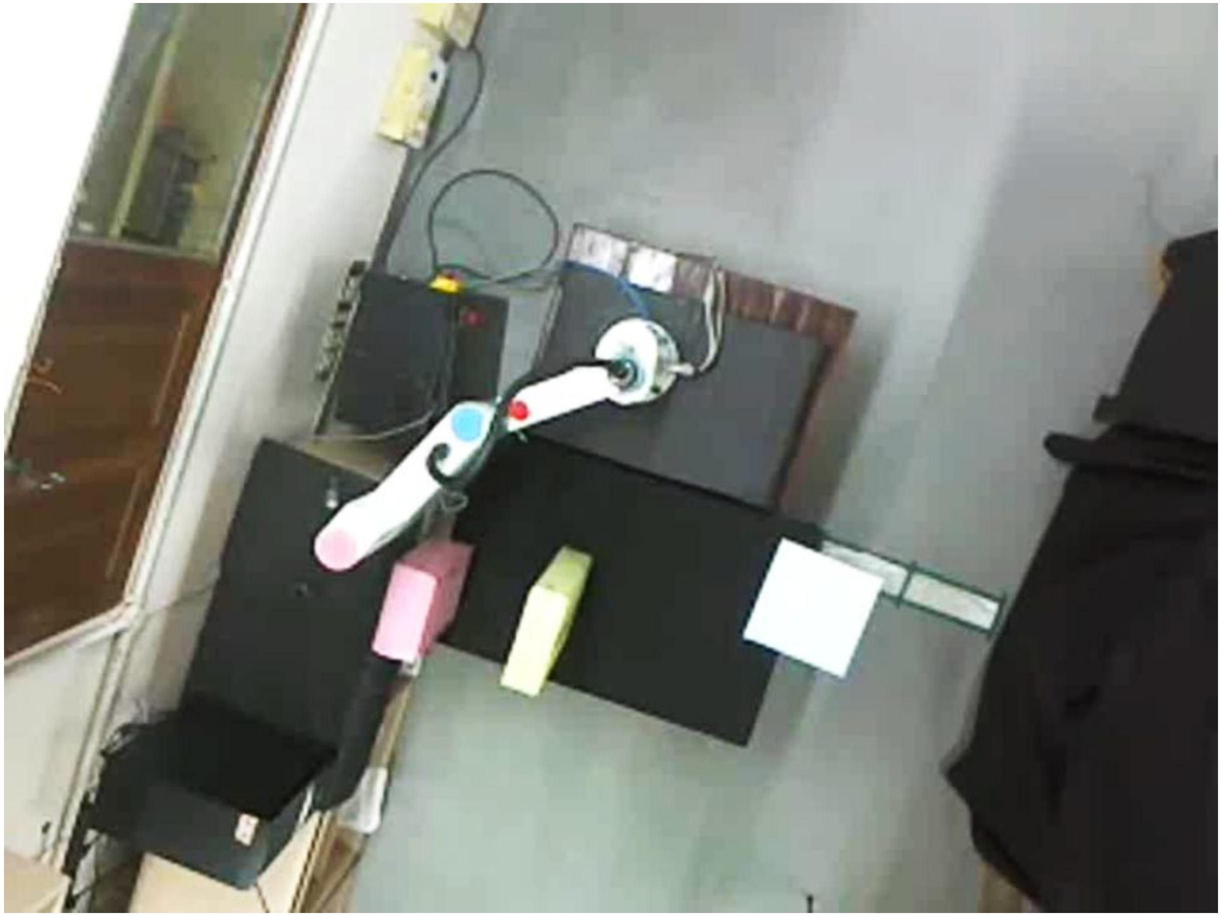
Will let it droop if
they can't see it

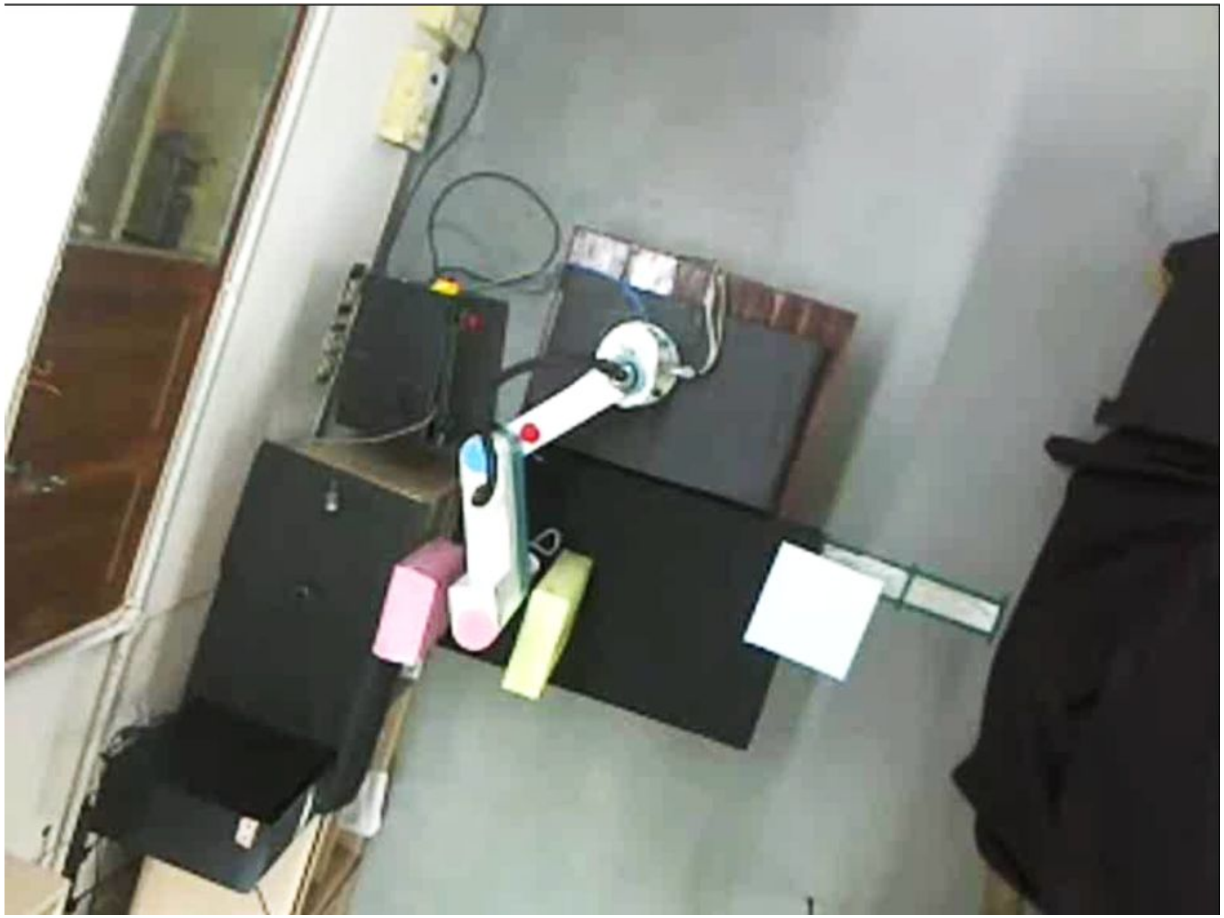


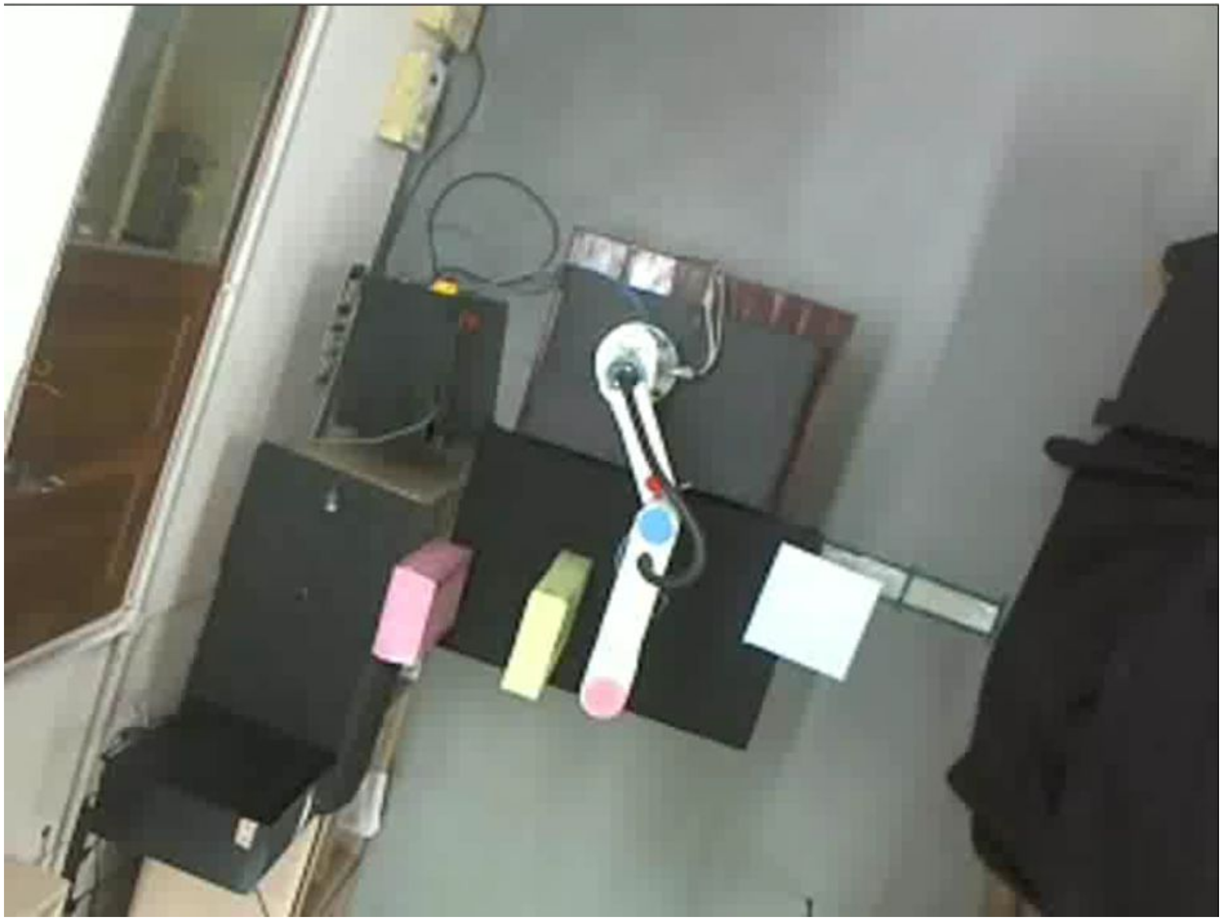
[A. van der Meer, 1997: Keeping the arm in the limelight]

Simulation

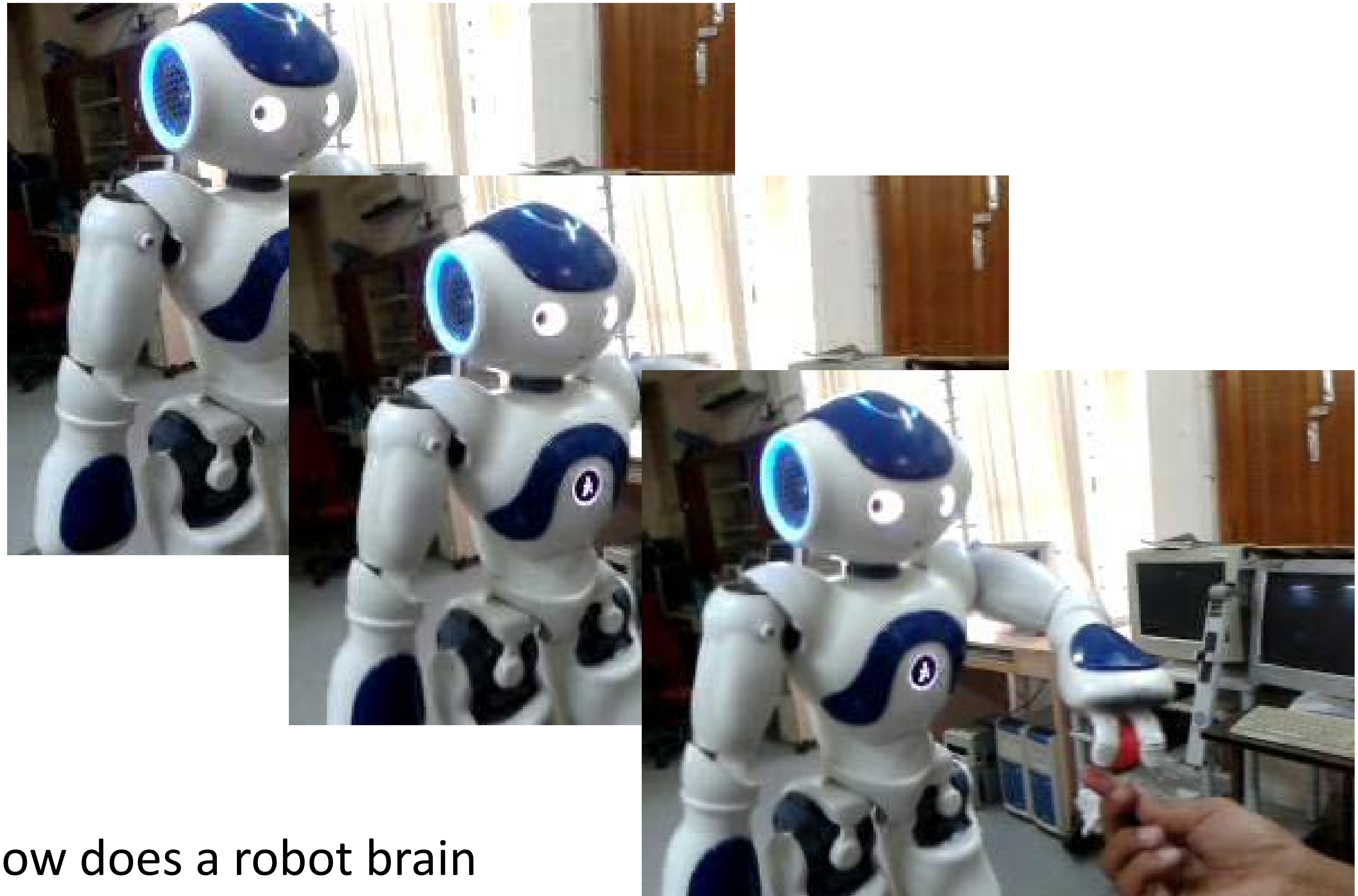








Robot self-discovery



How does a robot brain
discover the world?

Camera Motion and Shape via Factorization

Homogeneous Coordinates

- 2-D point (x,y) represented as (x_1,x_2,x_3) in homogeneous system

$$\mathbf{x = x_1/x_3 \qquad y = x_2/x_3}$$

- (x_1,x_2,x_3) same as (kx_1,kx_2,kx_3) same as $(x_1/x_3,x_2/x_3,1)$
- Similarly for 3-D point – (x_1,x_2,x_3,x_4)

Camera Models

- Orthographic
- Weak perspective/Scaled orthographic
- Para-perspective
- Perspective
- Affine
- Projective

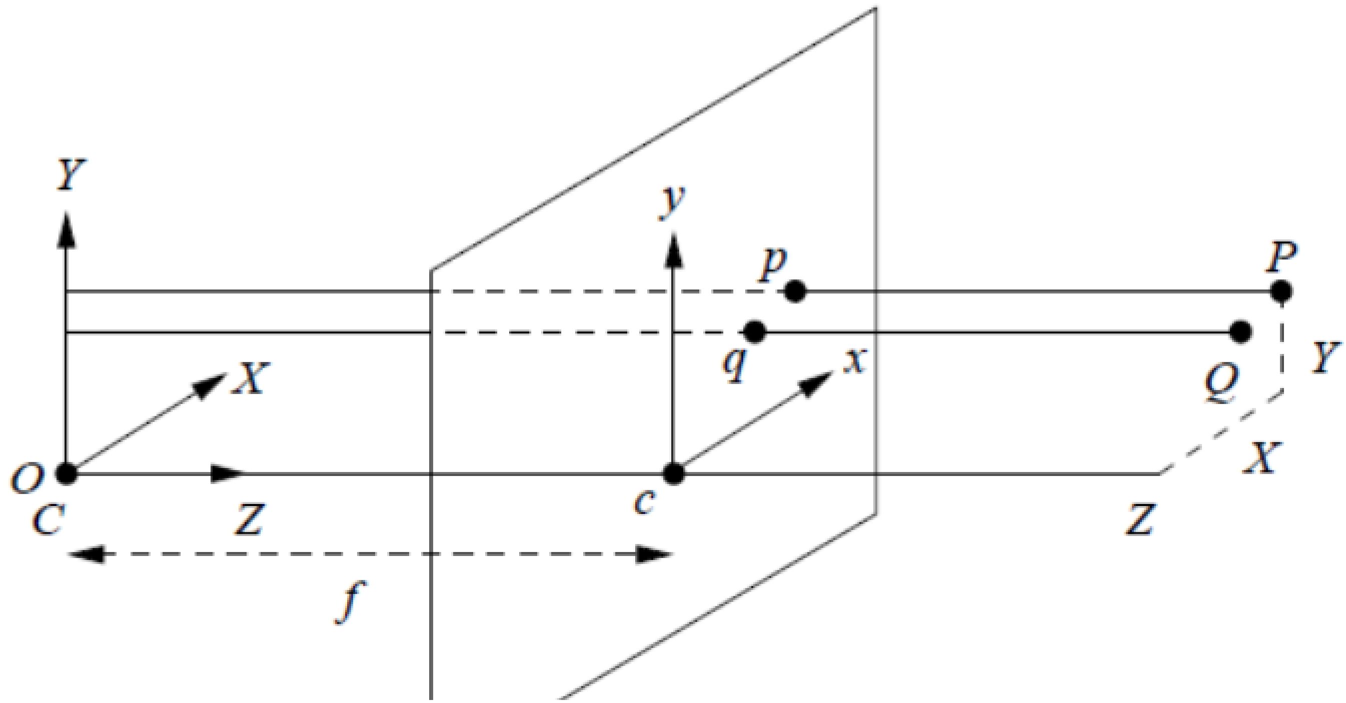
Camera Models

- $x = (u,v,1)^T$ or $(x_1,x_2,x_3)^T$=> homogeneous image coordinates
- $X = (x,y,z,1)^T$ or (X_1,X_2,X_3,X_4)=> homogeneous 3D coordinates

$$\mathbf{x}_{(3x1)} = \mathbf{T}_{(3x4)}\mathbf{X}_{(4x1)}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ T_{31} & T_{32} & T_{33} & T_{34} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

Orthographic Camera Model



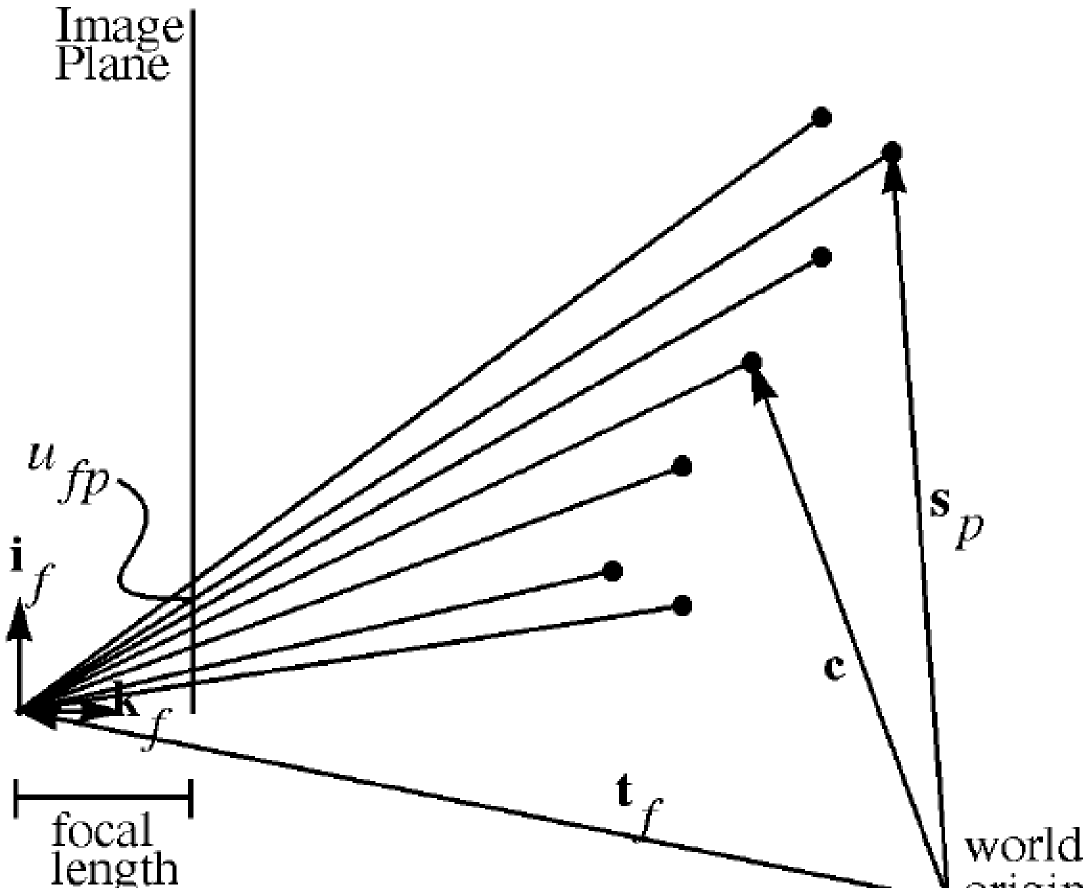
u=x   v=y

Orthographic Camera Model

$$\mathbf{T}_{orth} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

Perspective Camera Model



$$u = fx/Z$$
$$v = fy/Z$$

Perspective Camera Model

$$\mathbf{T}_p = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

Weak Perspective or Scaled Orthographic

$$\mathbf{T}_{wp} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & Z_{ave}/f \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & Z_{ave}/f \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

Affine Camera Model

$$\mathbf{T} = \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ 0 & 0 & 0 & T_{34} \end{bmatrix}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ 0 & 0 & 0 & T_{34} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

Projective Camera

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

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ T_{31} & T_{32} & T_{33} & T_{34} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}$$

Structure from Motion by Factorization

Tomasi, Carlo, and Takeo Kanade. 1992 "Shape and motion from image streams under orthography: a factorization method.

Factorization Method

- Given: Image stream, where P points have been tracked over F frames
- Image coordinates of p-th point in f-th frame = u_{fp}, v_{fp}
- **W** = 2F x P Measurement Matrix
- Column => observation for point
  Row => observation for frame

$$W = \begin{bmatrix} u_{11} & \dots & u_{1P} \\ \dots & \dots & \dots \\ u_{F1} & \dots & u_{FP} \\ v_{11} & \dots & v_{1P} \\ \dots & \dots & \dots \\ v_{F1} & \dots & v_{FP} \end{bmatrix}$$

Factorization Method : Orthography

- $\mathbf{W} = \mathbf{M}_{(2F\times3)}\mathbf{S}_{(3\times P)} + \mathbf{t}_{(2F\times1)}\,[1\ldots1]_{(1\times P)}$
  translation vector **t**: element f = mean of row f in **W**

- registered measurement matrix W*
  $$\mathbf{W^*} = \mathbf{W} - \mathbf{t}\,[1\ldots1] = \mathbf{M}\,\mathbf{S} \quad \text{(rank 3)}$$

$\mathbf{M}_{(2F\times3)}$: row f = camera horiz/ vert axes in frame f
$\mathbf{S}_{(3\times P)}$: column p = (x,y,z) of tracked point p

OBJECTIVE : obtain **M** and **S** from W* via SVD

Hotel Stream Data

frame 1

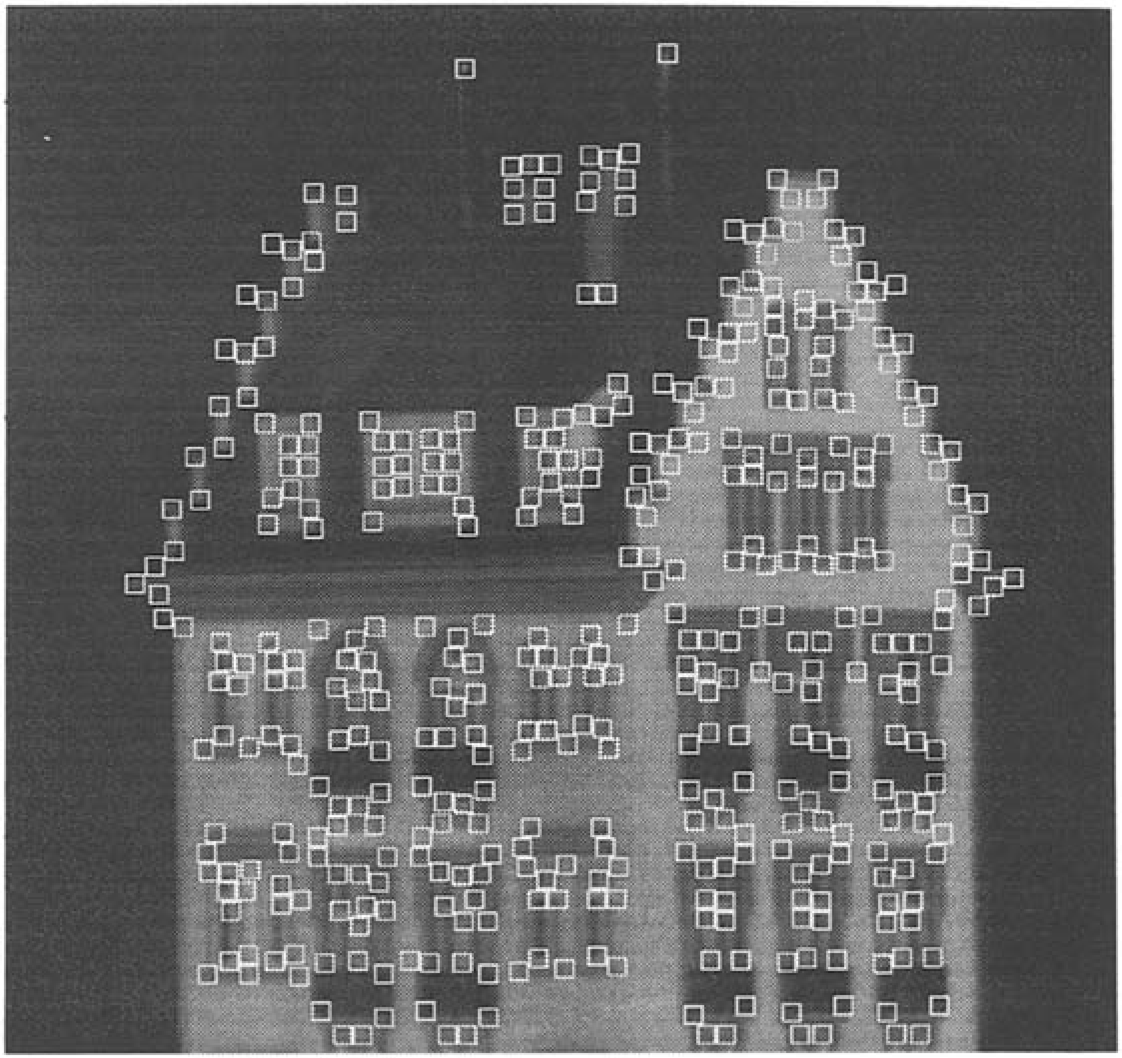
Hotel Stream Data

frame 50

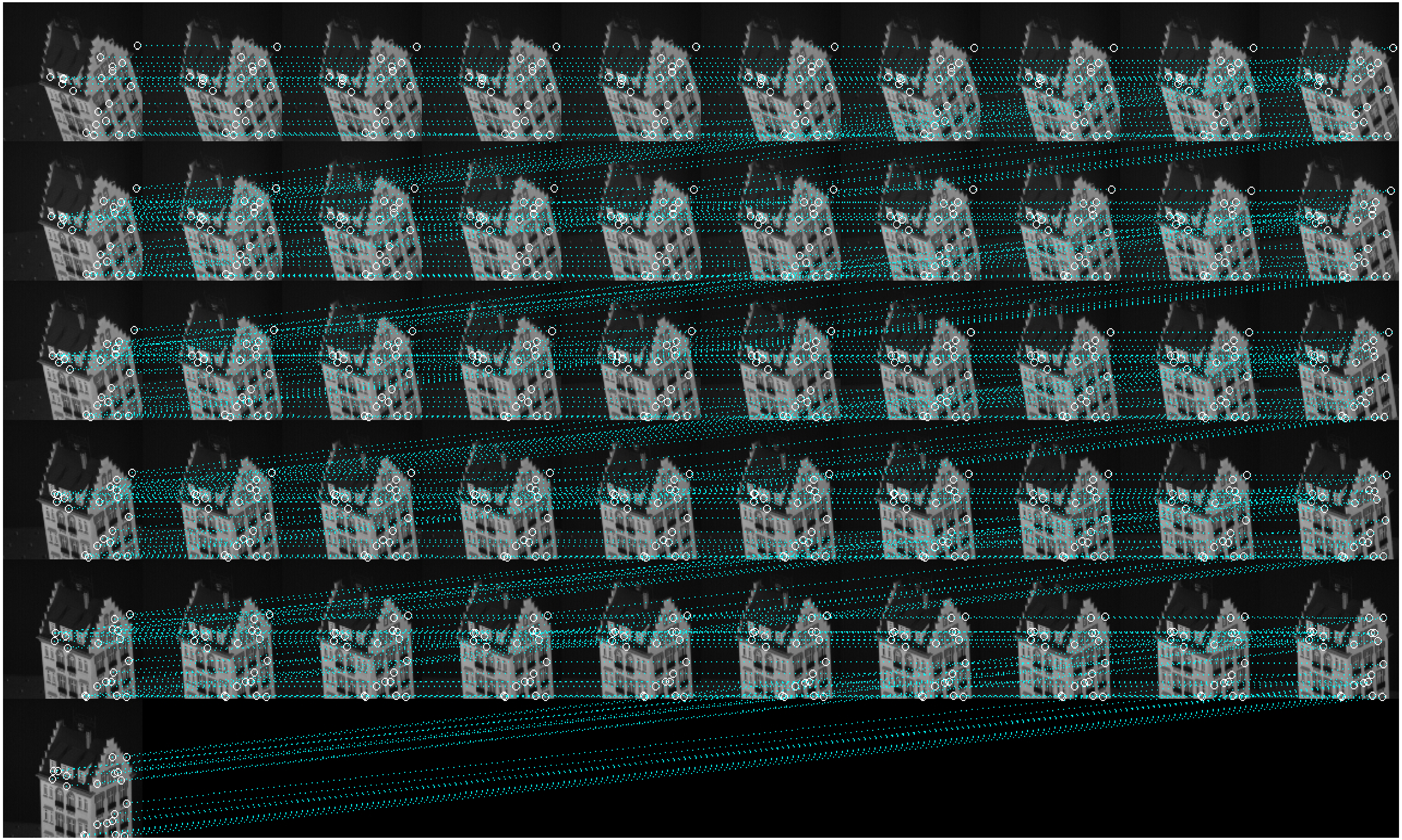
Hotel Stream Data

frame 100

Tracked Features



Tracking



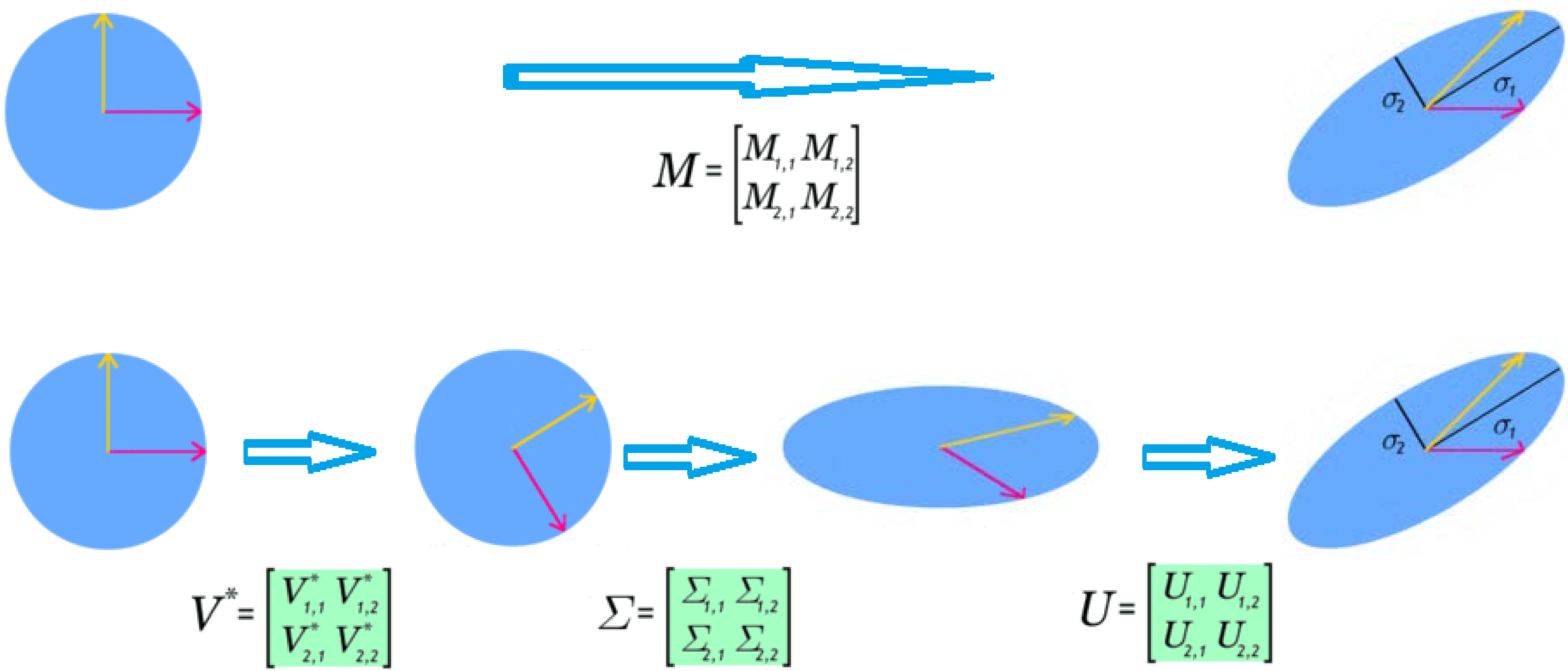
Factorizatation Method : Orthography

- Creation of measurement matrix **W**
- Obtaining registered measurement matrix **W***
- Performing **SVD** to obtain

  $$\mathbf{W^* = UEV^T = MS}$$

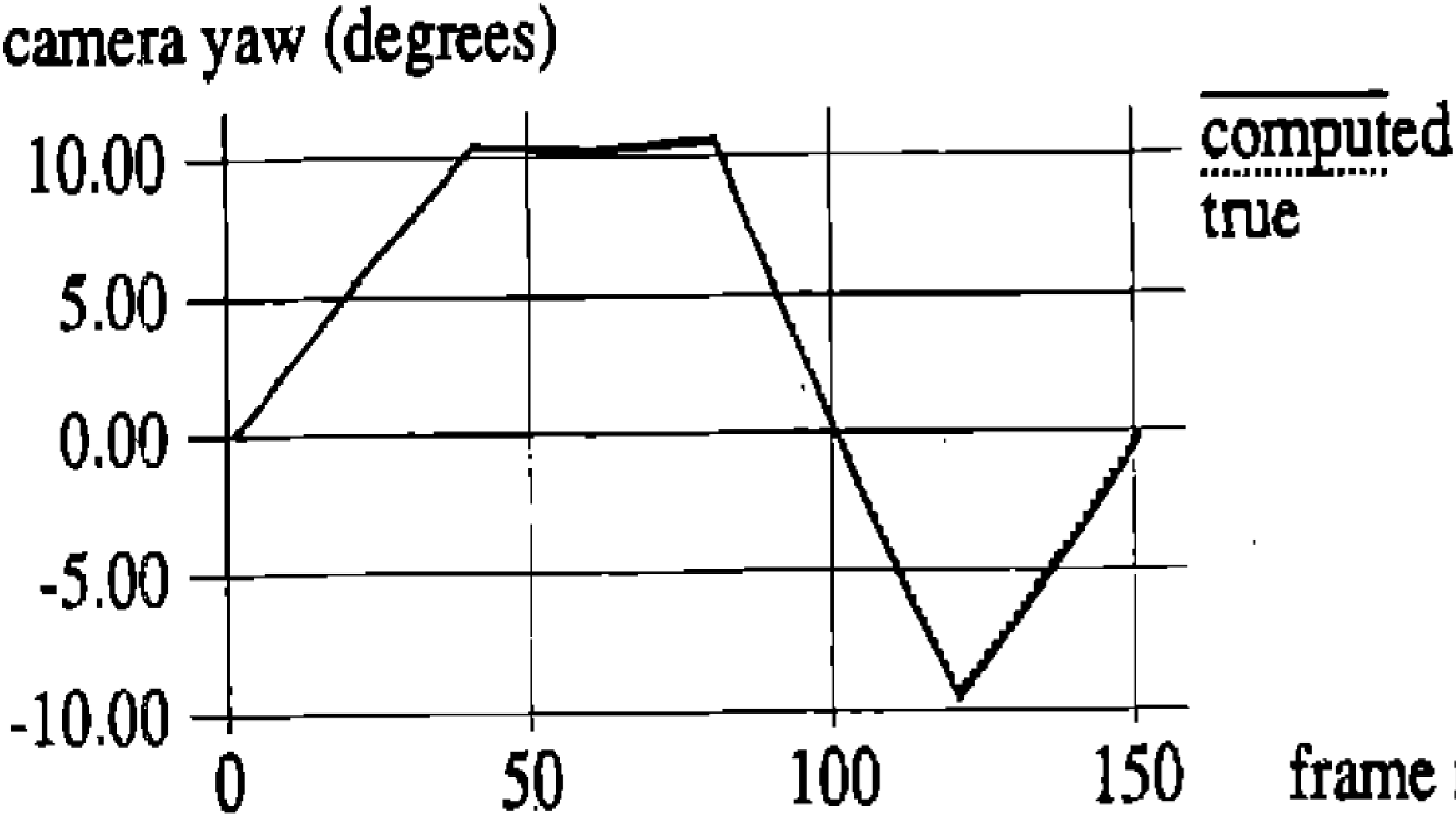
- $M = UE^{0.5}$ $S = E^{0.5}V^T$ …not unique solution
- MS or $(M_2A)(A^{-1}S_2)$ …both are solutions
- Constraints to obtain unique A
- Directions can be aligned to camera directions in first frame

SVD



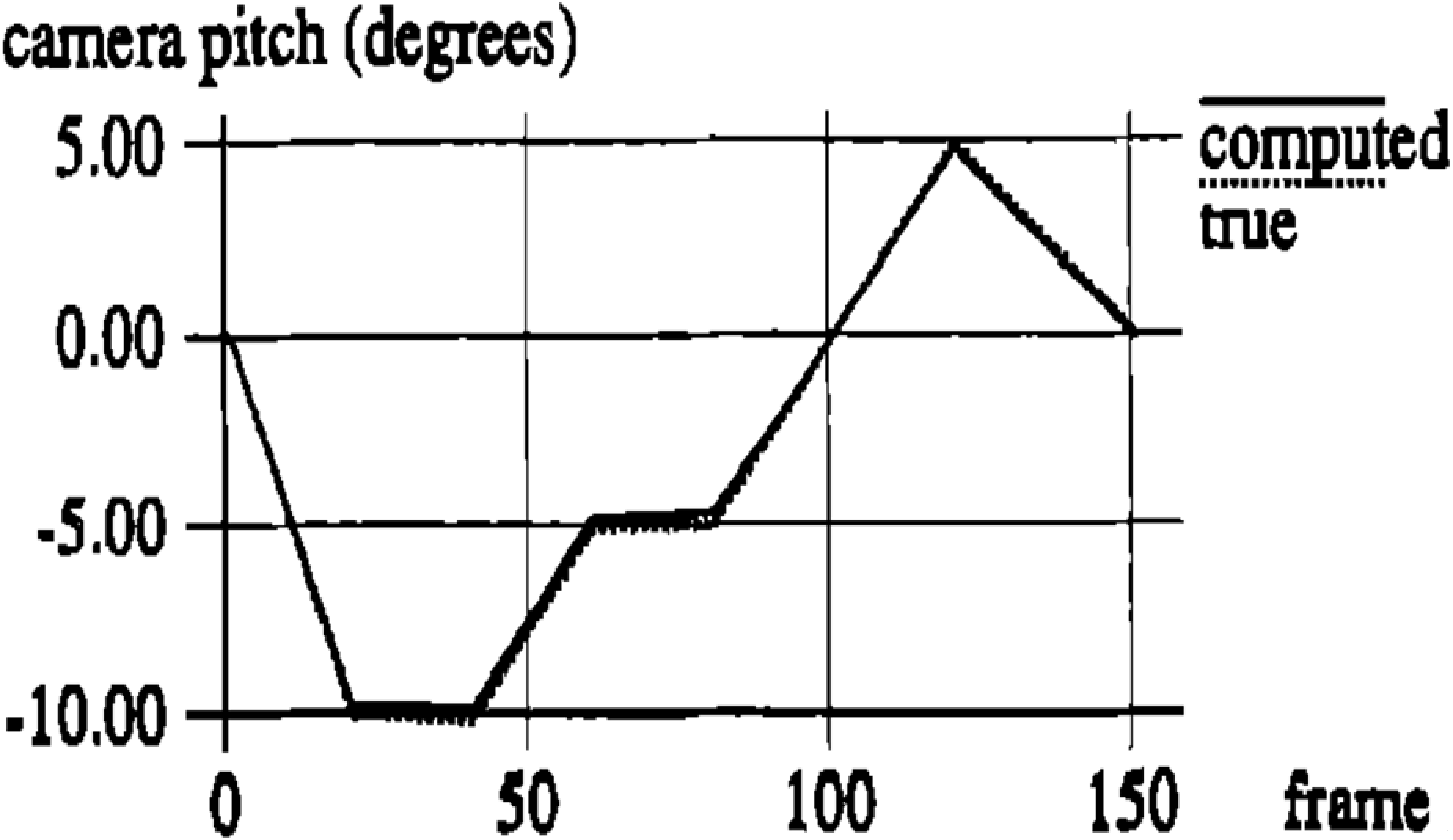
σ_2
σ_1
$M = \begin{bmatrix} M_{1,1} & M_{1,2} \\ M_{2,1} & M_{2,2} \end{bmatrix}$
σ_2
σ_1
$V^* = \begin{bmatrix} V^*_{1,1} & V^*_{1,2} \\ V^*_{2,1} & V^*_{2,2} \end{bmatrix}$
$\Sigma = \begin{bmatrix} \Sigma_{1,1} & \Sigma_{1,2} \\ \Sigma_{2,1} & \Sigma_{2,2} \end{bmatrix}$
$U = \begin{bmatrix} U_{1,1} & U_{1,2} \\ U_{2,1} & U_{2,2} \end{bmatrix}$

Camera Motion Estimation [M]



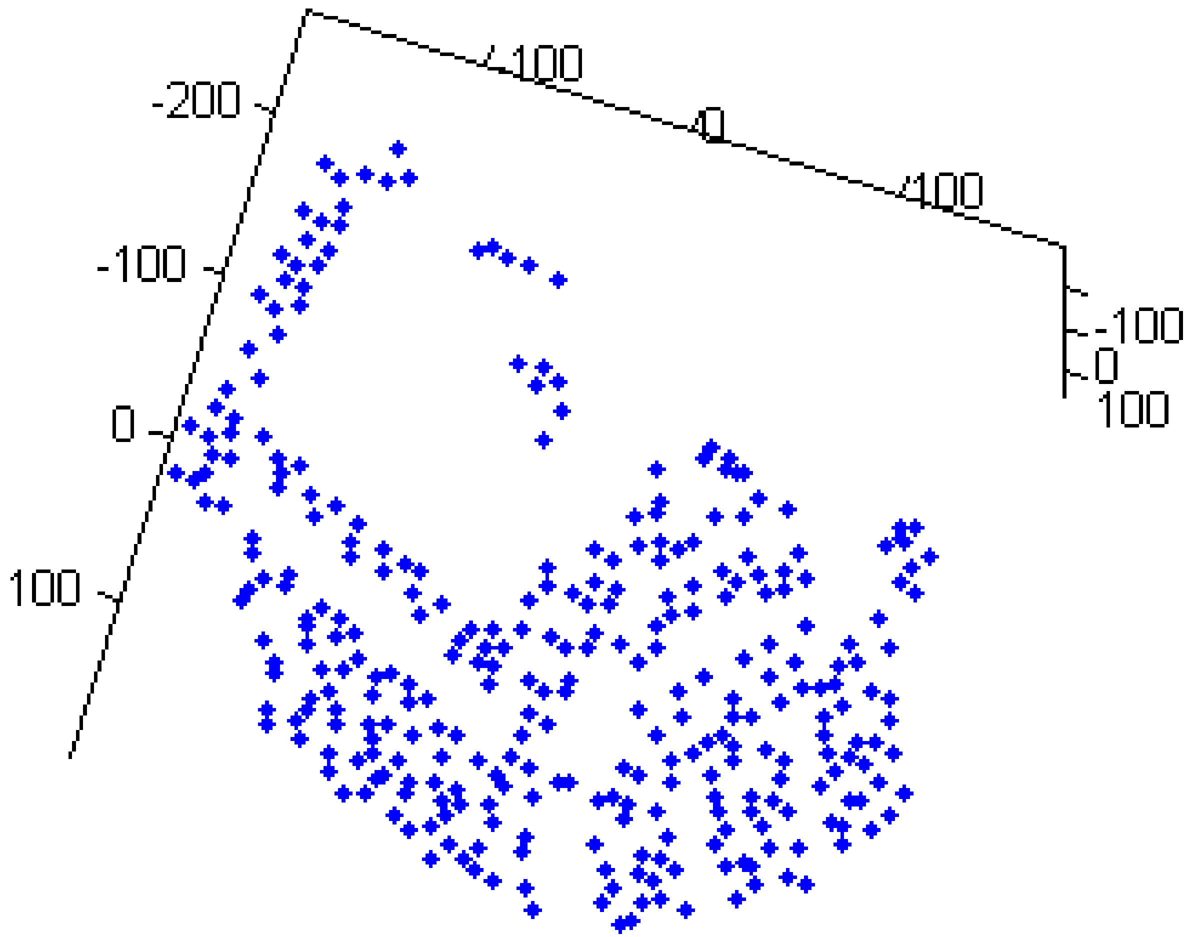
error < 0.3 degrees

Camera Motion Estimation [M]



error < 0.35 degrees

3D Shape Estimation [S]



Multi-body Factorization Method

- Multiple objects in a video sequence moving independently
- Earlier approach => from object frame (as if object stationary and camera moving)
- Common representation required for all objects in multi-object scenario
- Why not look from **camera frame**!!!
- (Moving camera+static object) equivalent to (static camera +moving object)
- unique for all objects for a frame

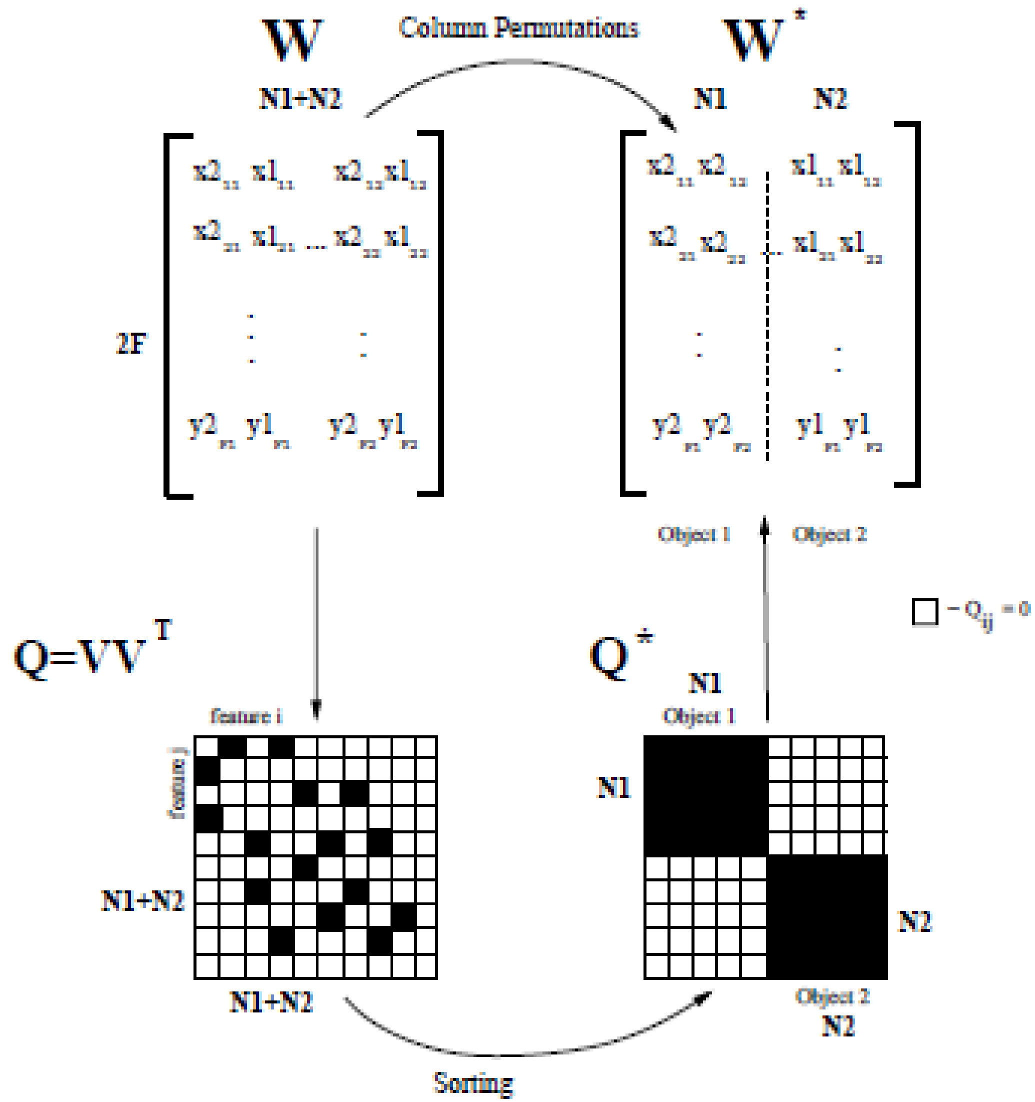
Multi-body Factorization Method

- Points from all objects collectively taken
- Expected Rank <= 4*number of objects
- Shape Interaction Matrix

$$\mathbf{Q} = \mathbf{V}\mathbf{V}^T$$

- Q independent of camera orientation, coordinate transformation
- Permuting columns of V does not change values of Q matrix(only row/column number changed in same way as V permutation)
- Only the interaction values of points from same object get non-zero value.
- Block-diagonalization to obtain the points corresponding to same object

Multi-body Factorization Method



Aspects of Factorization

- All images treated uniformly
- Alternate approaches are initialization based and may do poorly if improper.
- Convergence guaranteed by numerical approach
- In multi-object factorization, can determine number of objects and point clusters

Unsupervised Action Learning and Anomaly Detection

3.2. *Topic Modelling*

- Given: Document and Vocabulary

  * Document is histogram over vocabulary
- Goal: Identify topics in a given set of Documents

Idea: Topics are latent variables

Alternate view :

- Clustering in topic space
- Dimensionality reduction

3.3. *Models in practice*

- *LSA*

Non-parametric clustering into topics using SVD.

- *pLSA :*

Learns probability distribution over fixed number of topics; Graphical model based approach.

- *LDA :*

Extension of pLSA with dirichlet prior for topic distribution. Fully generative model.

4. *Vision to NLP : Notations*

Text Analysis	Video Analysis
Vocabulary of words	Vocabulary of visual words
Text documents	Video clips
Topics	Actions/Events

5.1. *Video Clips*

- 45 minute video footage of traffic available
- 25 frames per second
- 4 kinds of anomaly
- Divided into clips of fixed size of say 4 – 6 seconds

5.2. *Visual Words*

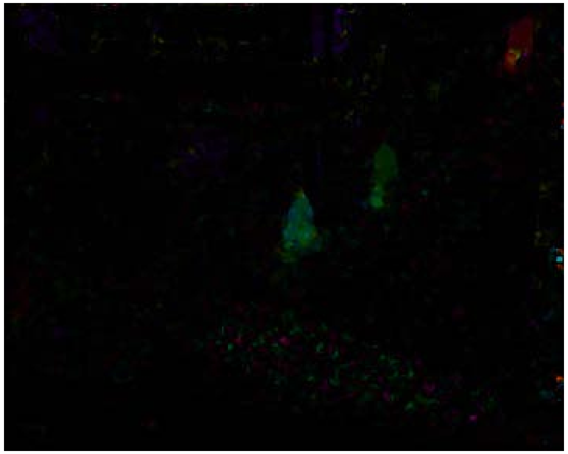
- Each frame is 288 x 360
- Frame is divided into 15 x 18 parts, each part containing 400 pixels
- Features
  - Optical flow
  - Object size
- Background subtraction is performed on each frame to obtain the objects in foreground. Features are computed only for these objects
- Foreground objects consist of vehicles, pedestrians and cyclists

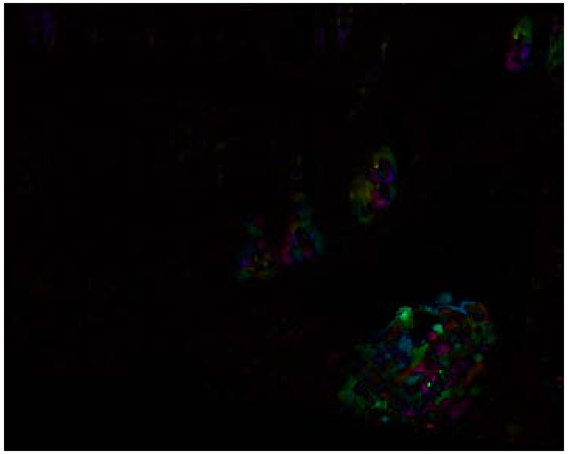
5.2. *Visual Words (contd..)*

- Foreground pixels then divided into “big” and “small” blobs (connected components)
- Optical flow computed on foreground
- Flow vector quantised into 5 values :
  - Static
  - Dynamic- up, down, left and right
- 15x28x5x2 different “words” obtained

5.3. *Foreground Extraction*

5.4. *Optical Flow: Heat Map*





6. *Modelling pLSA*

- Training Dataset: no or very less "anomaly"
- Test Dataset: usual + anomalous events

Procedure

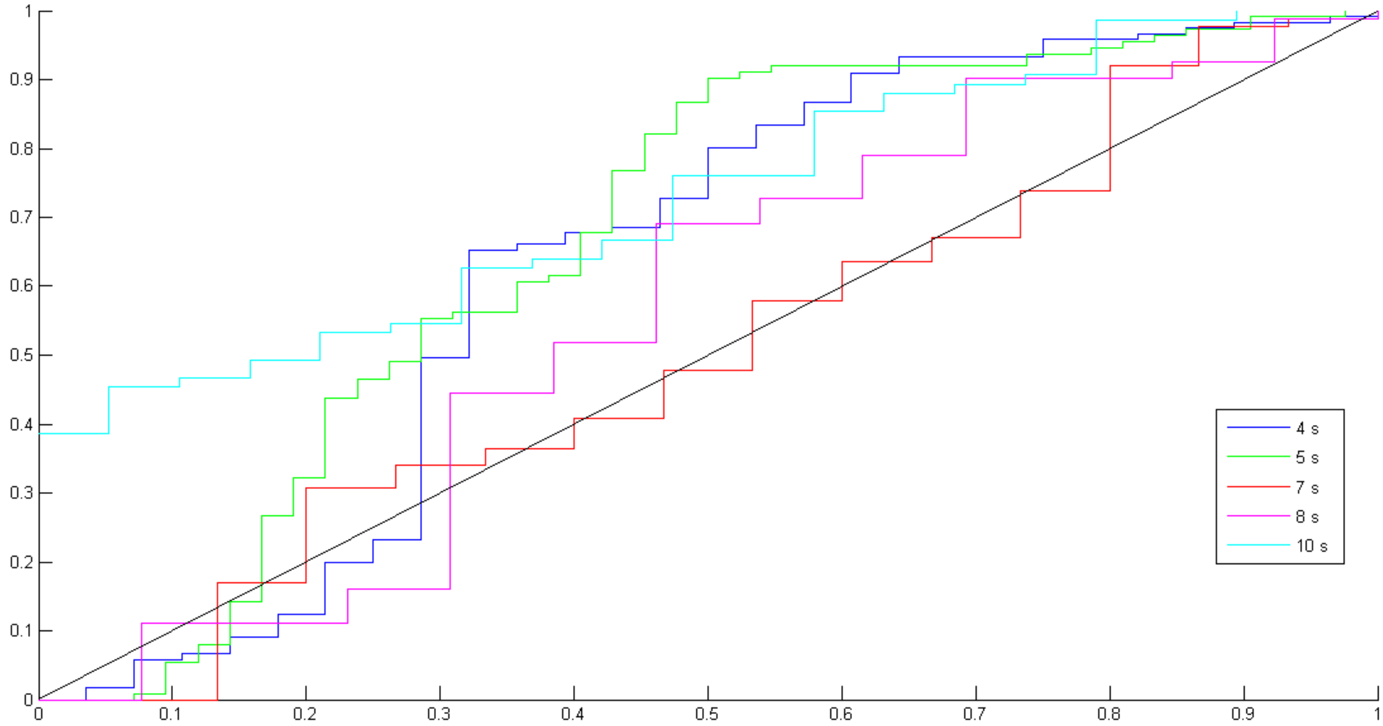
- Learn $\mathrm{P(w|z)}$ and $\mathrm{P(z|d)}$ from training data
- Keeping $\mathrm{P(w|z)}$, estimate $\mathrm{P(z|d)}$ on test data
- Threshold on likelihood estimate of individual test video clips

7. *Results Demo*

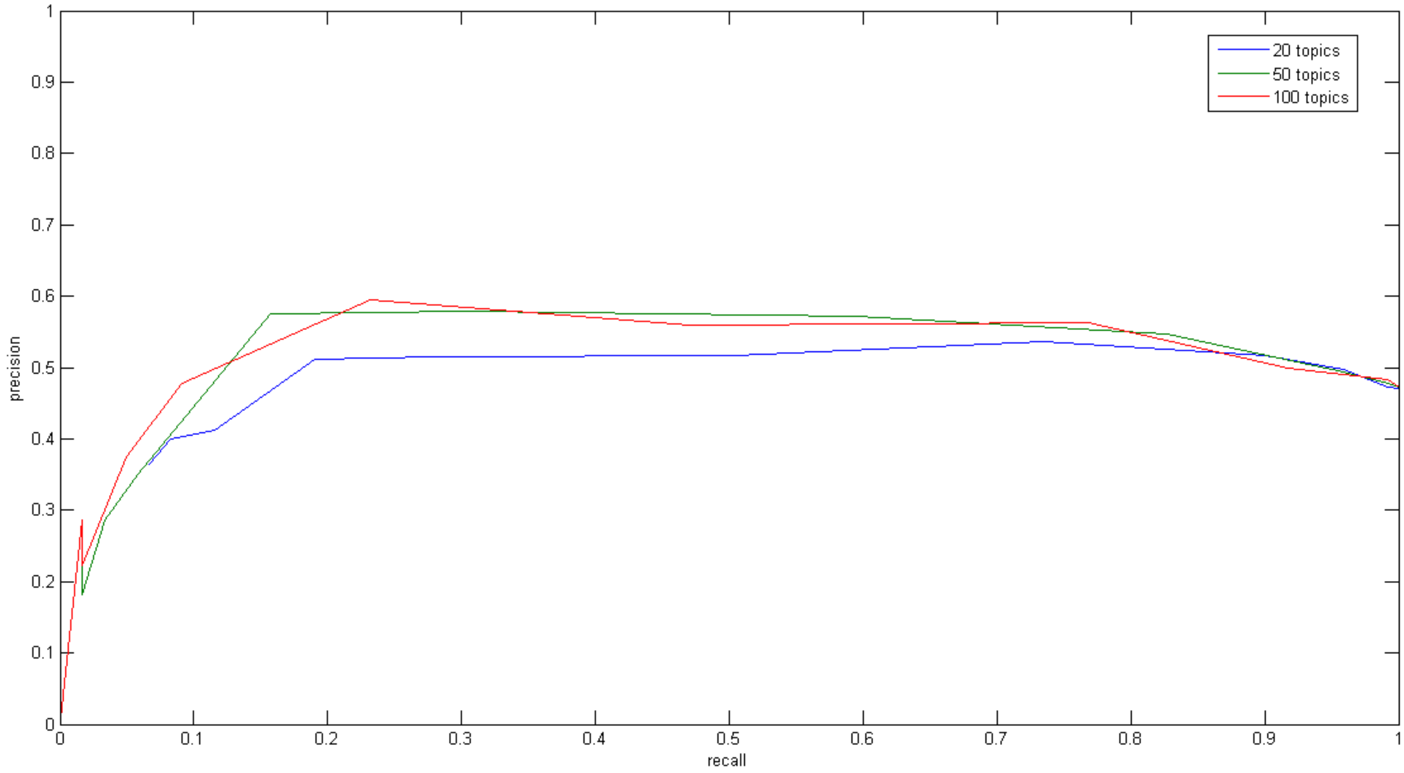
- 3 clips

- 3 different types of anomalies

8. Results (ROC plot)

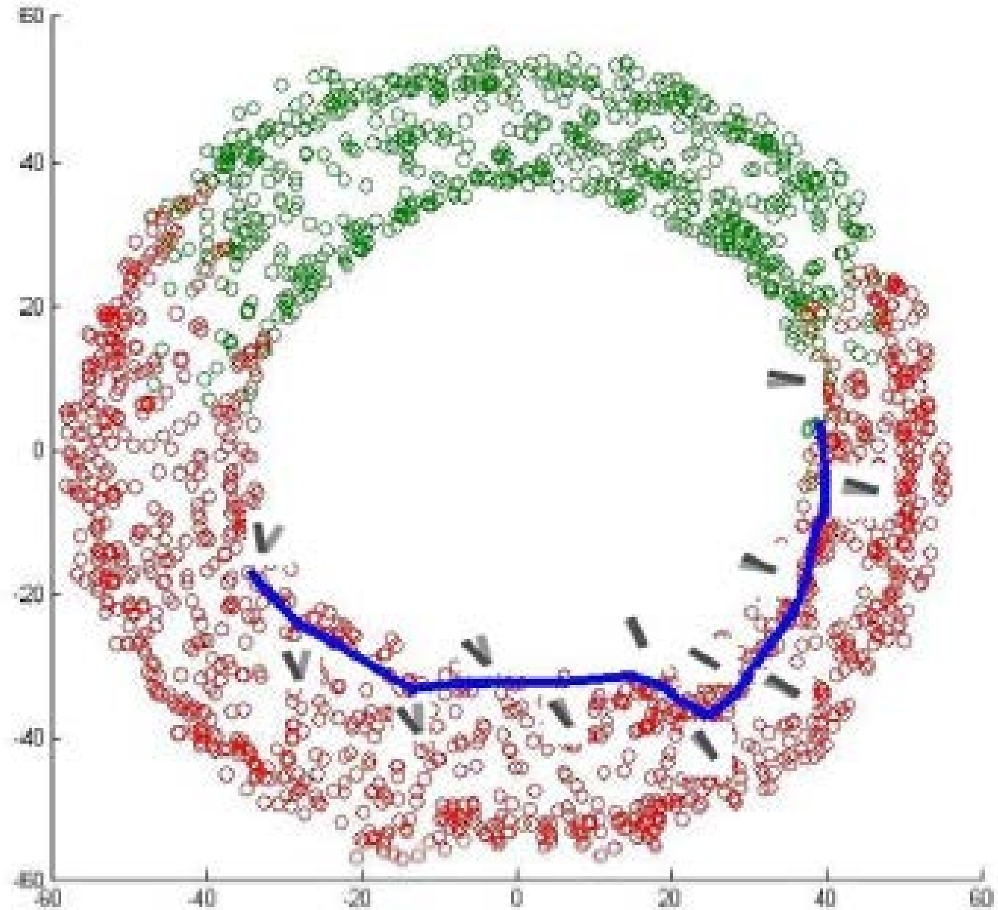


8. Results (PR curve)



Motion Planning

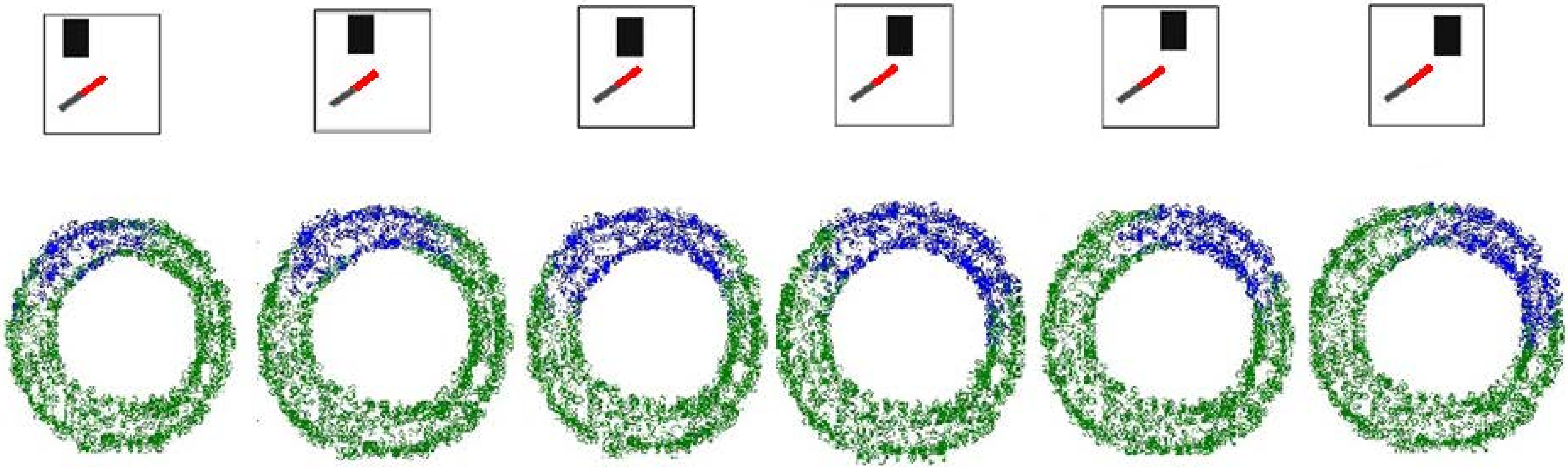
Motion planning



Given start / goal image, map to manifold using local interpolation

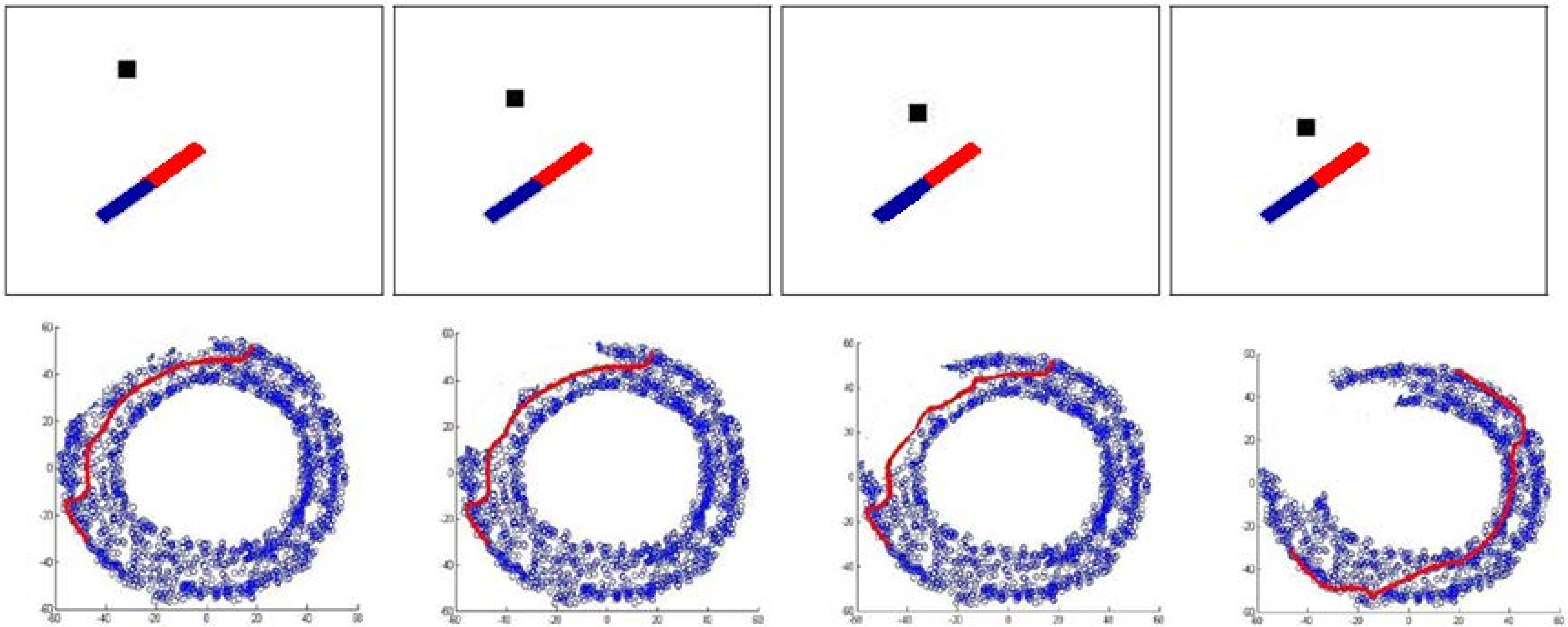
Use k-nn connectivity in manifold as “roadmap” for motion planning

Obstacle modeling by node deletion

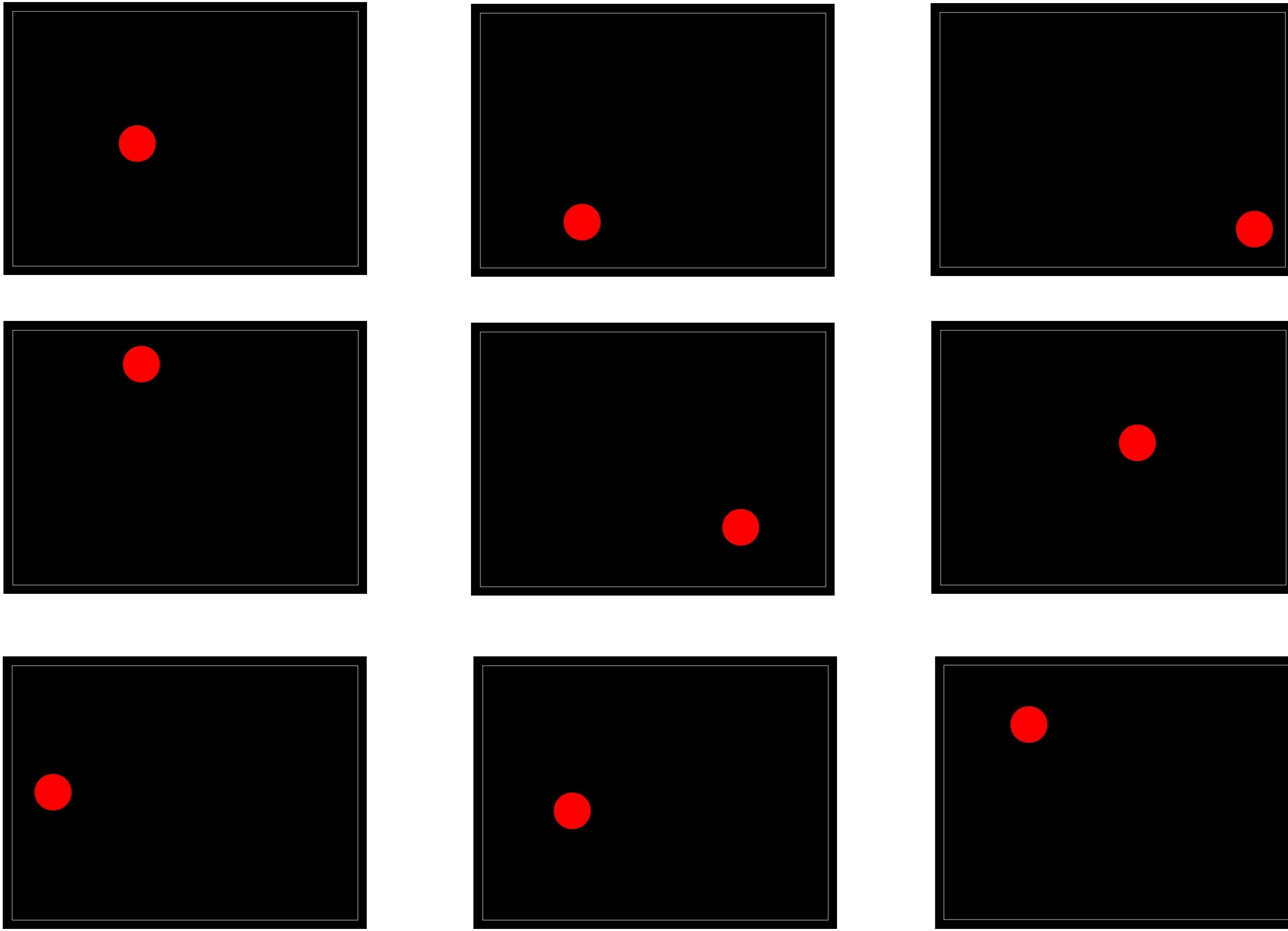


If obstacle intersects robot in image space →
delete corresponding nodes from “visual roadmap”

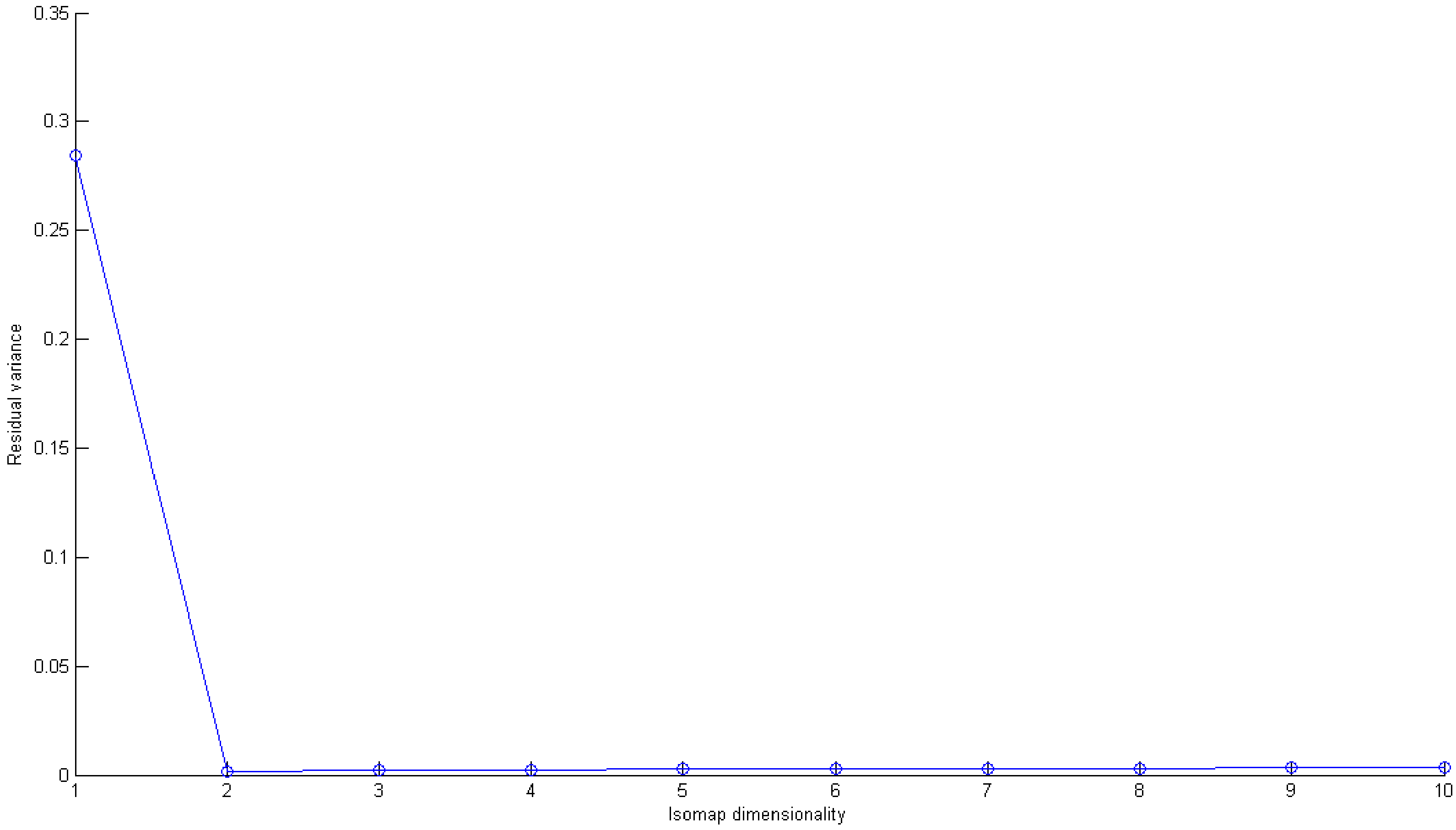
Path planning as obstacle moves



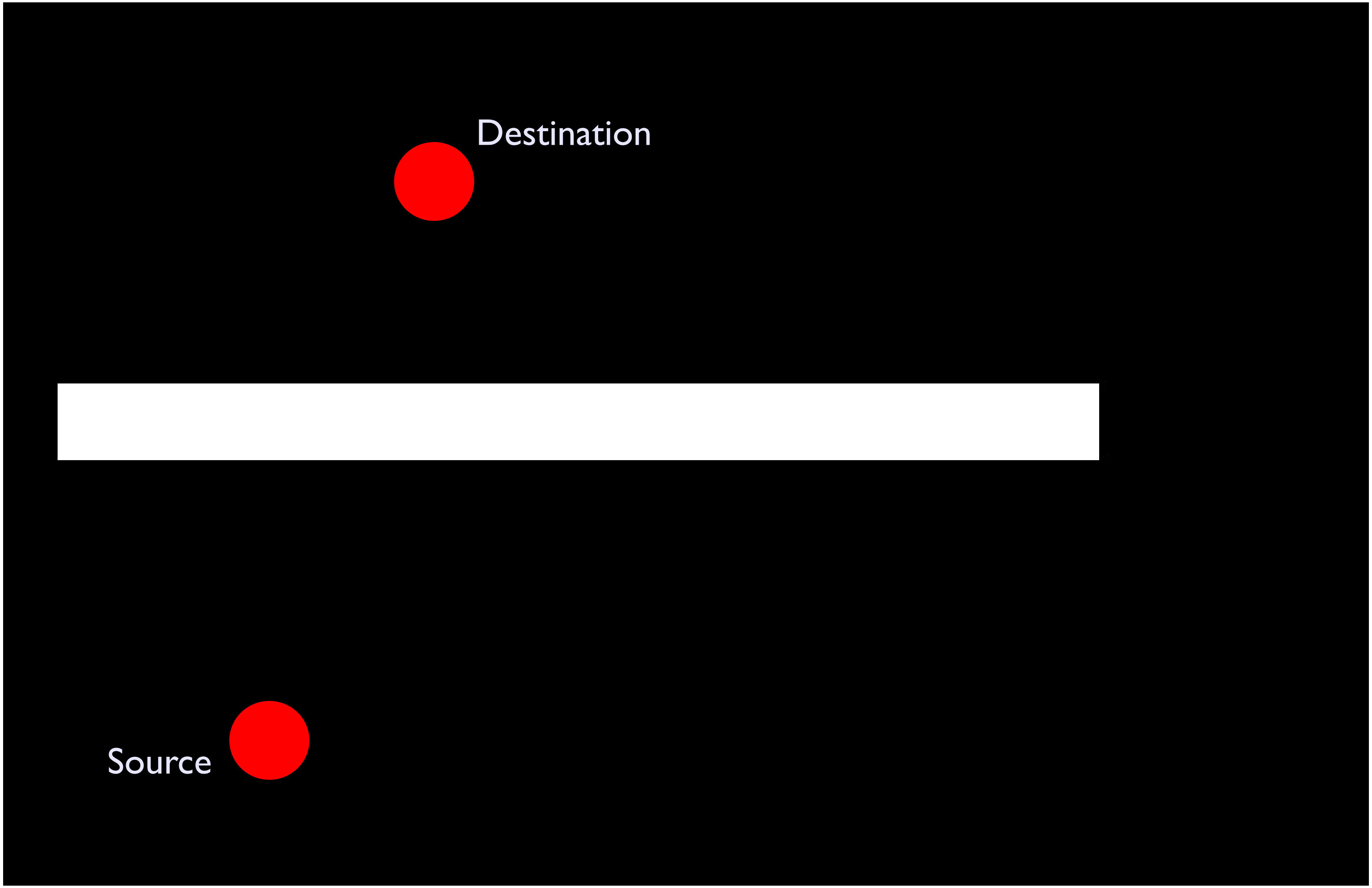
Mobile robots



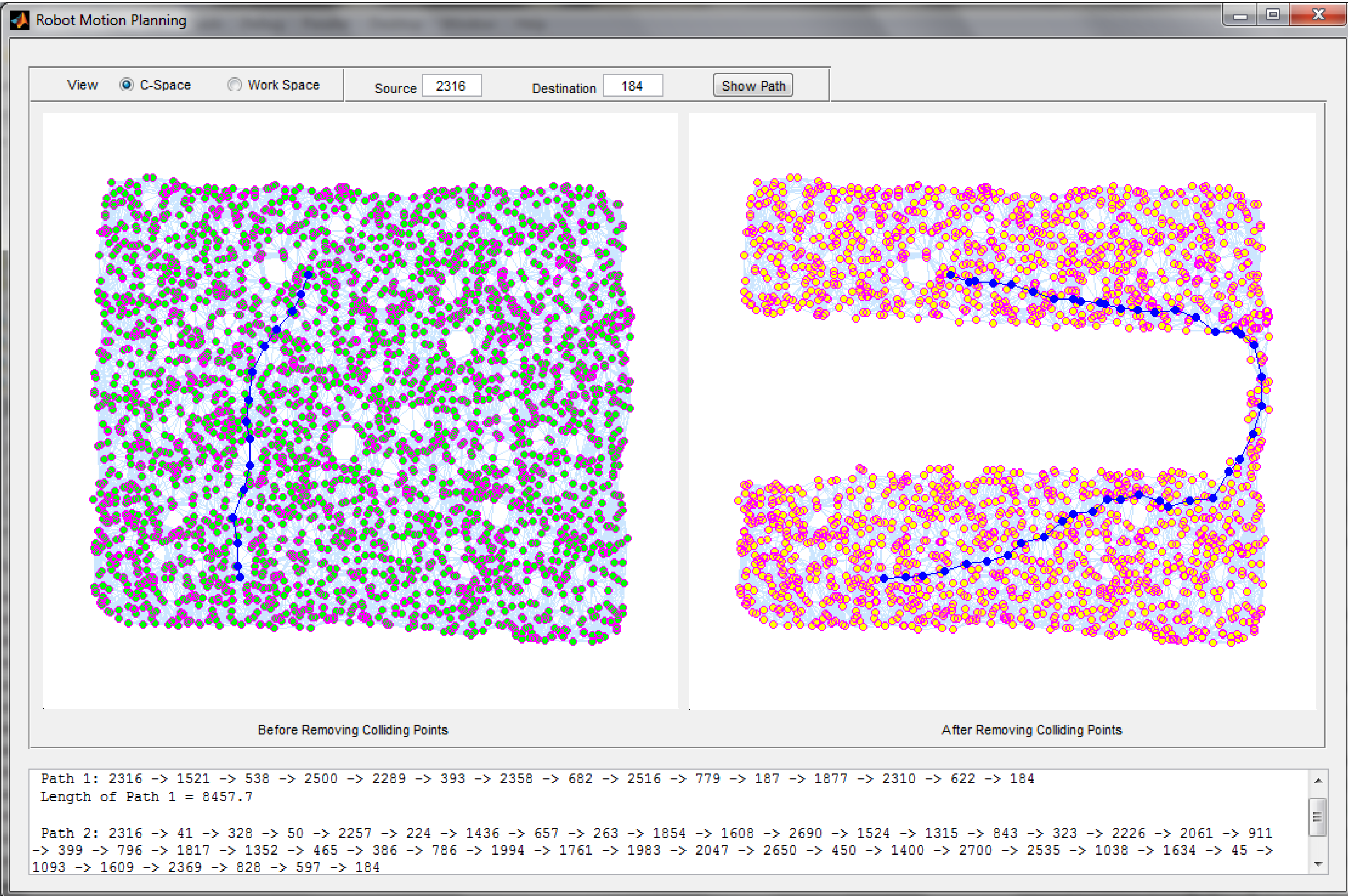
Residual error : disk robot

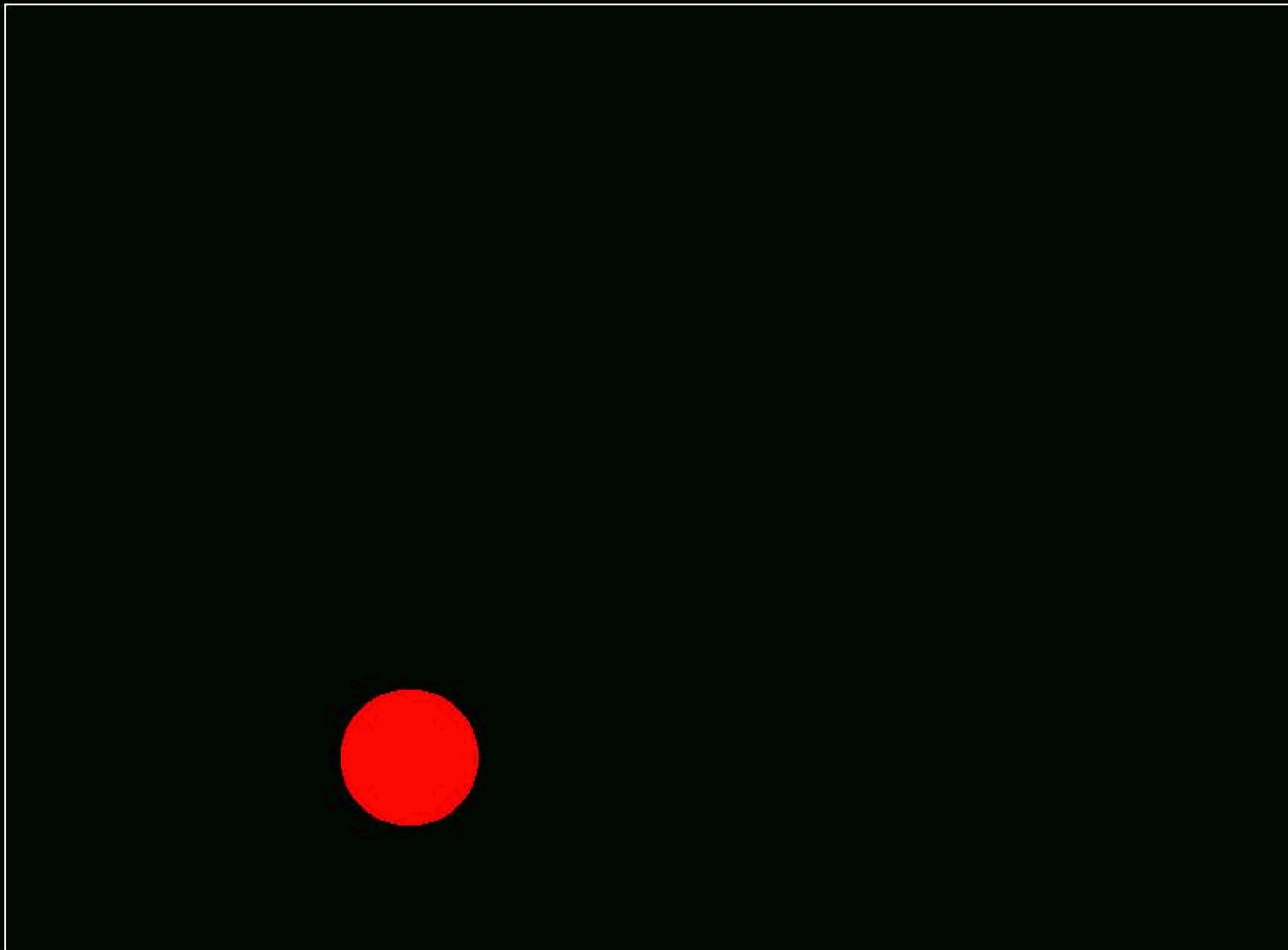


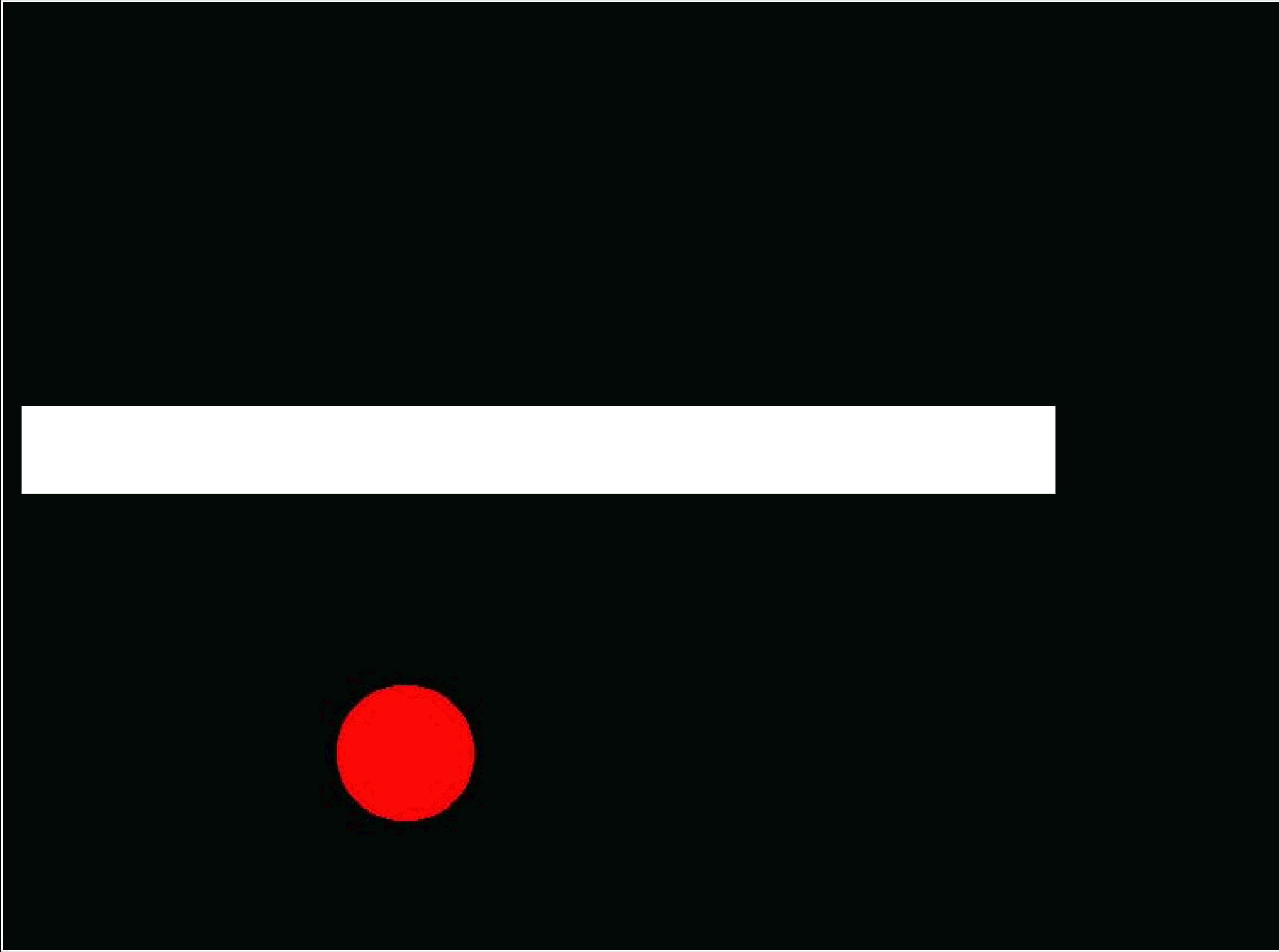
Robot Motion Planning



Path Planning Interface

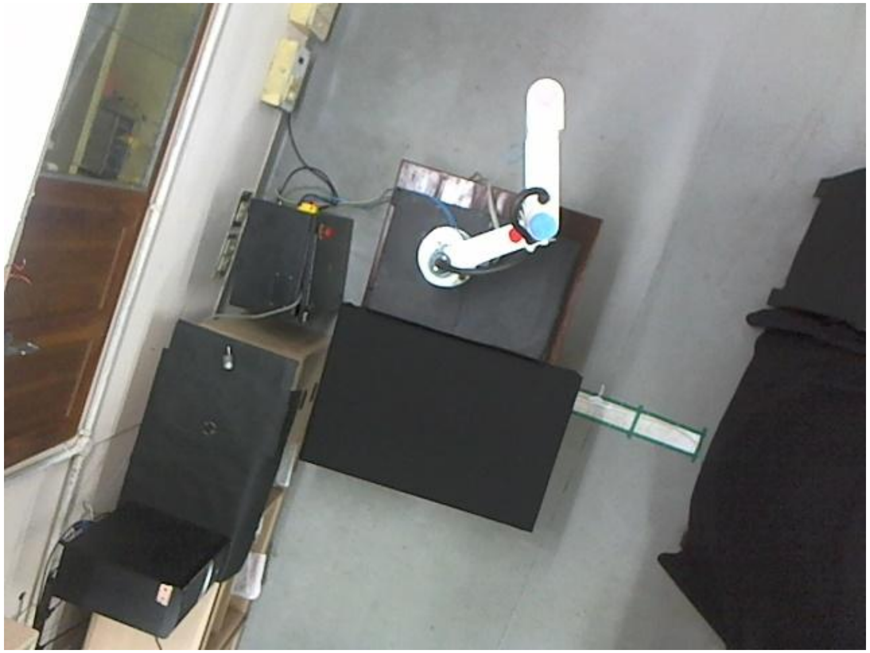




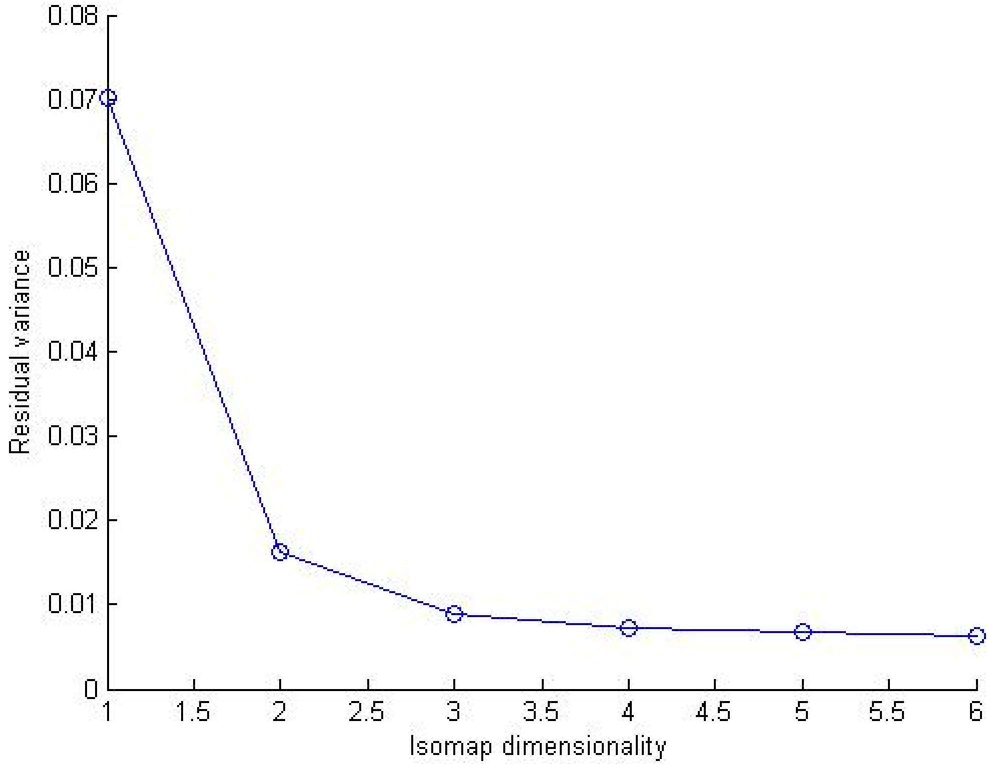


Real robots

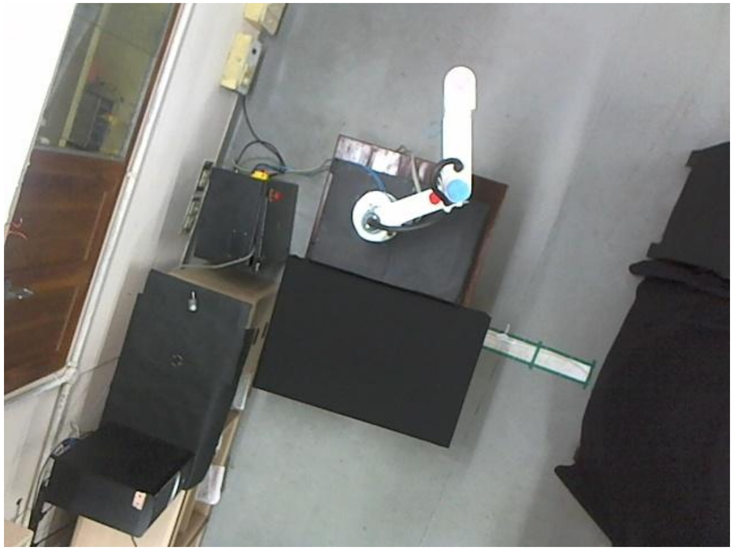
SCARA arm



SCARA arm : degrees of freedom

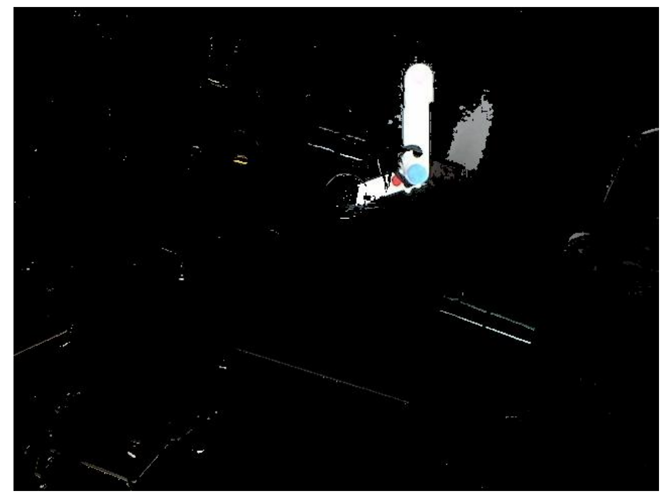


Background Subtraction : Robot



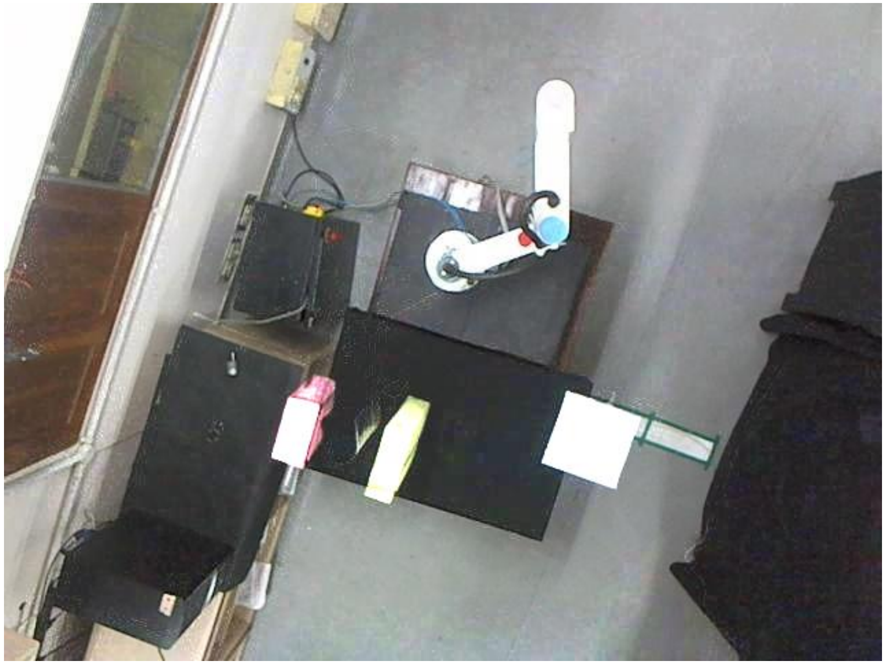
image

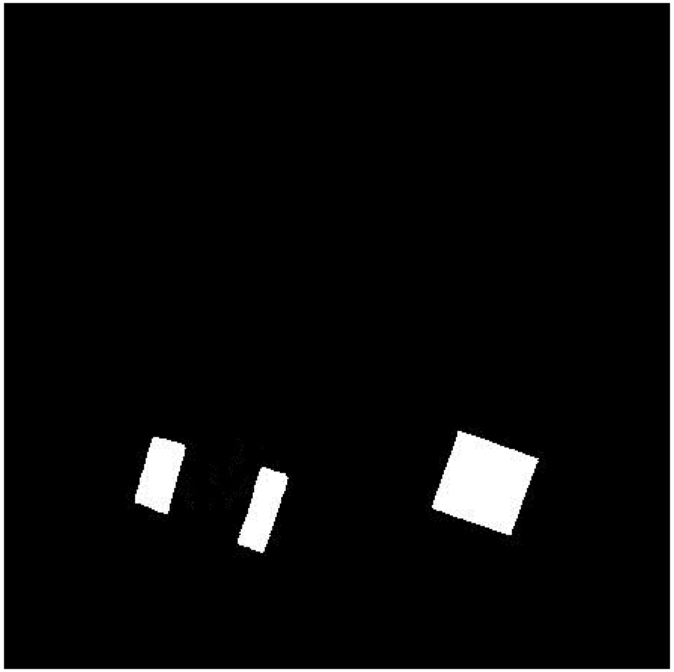
foreground (moving part)



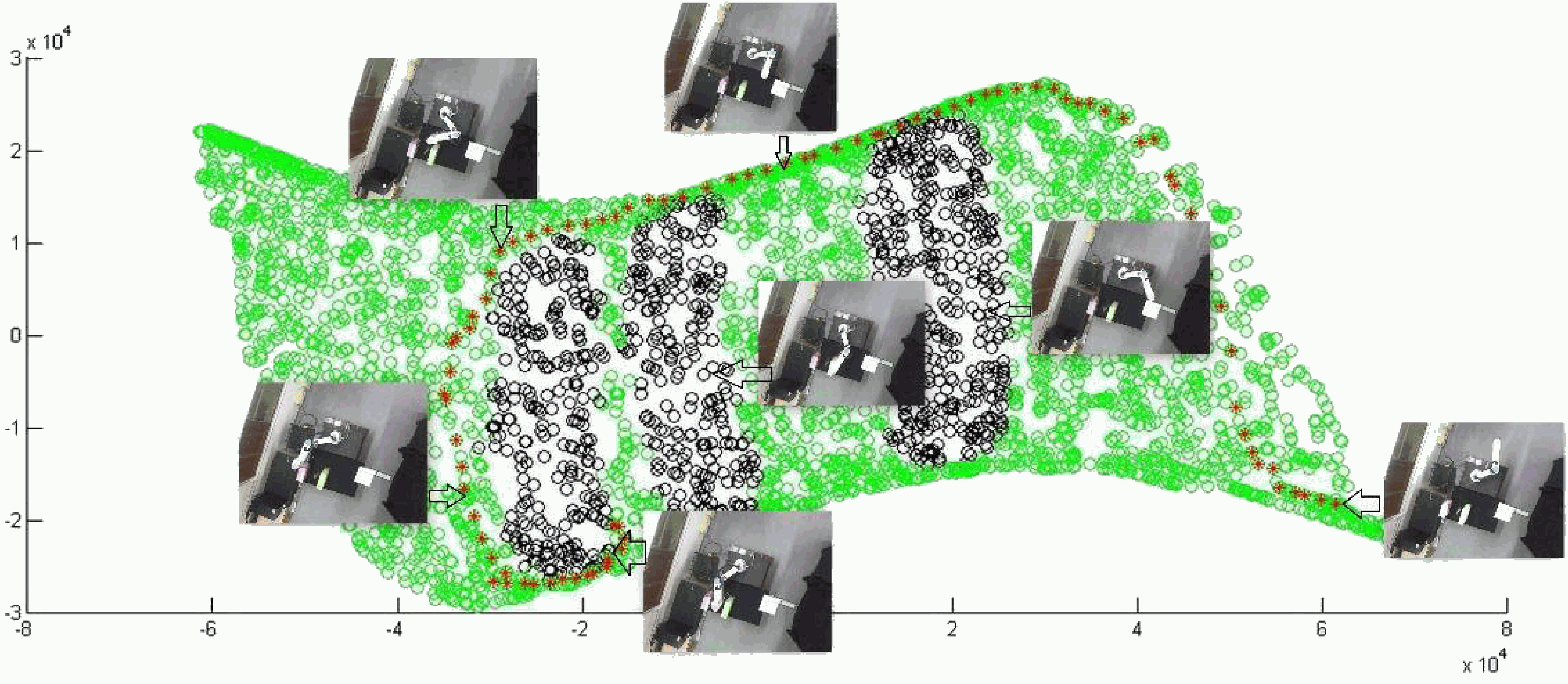
learned background

Background Subtraction : Obstacle





Visual Configuration Space



Application to Graphics

Head Motion

[M. S. Ramaiah, Ankit Vijay, Geetika Sharma, Amitabha Mukerjee IEEE VR-13]

Minimal Commitment Language Acquisition

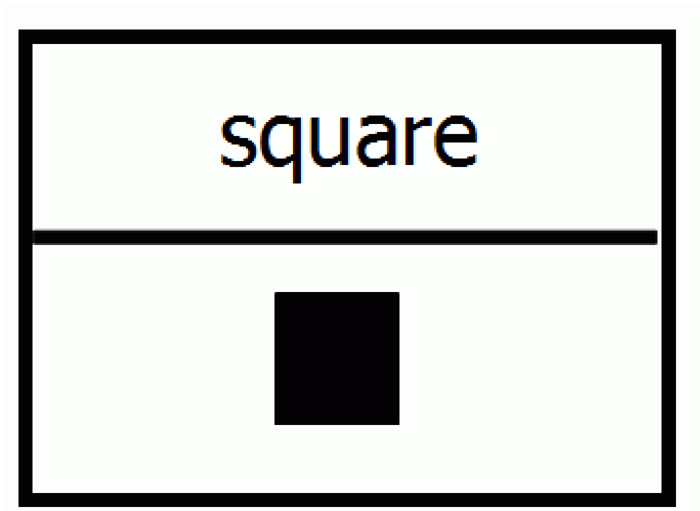
Previous Work:
Unsupervised Semantics

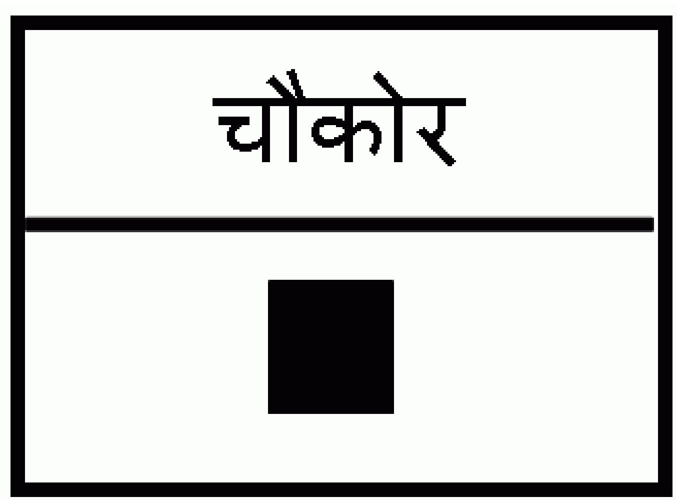
- single word or phrase learning   [no grammar]
  - Hand-coded propositional (T/F) semantics
    - [plunkett etal 92]
    - [regier 96] (prepositions)
    - [steels 03] [roy/reiter 05]   [caza/knott 12]
  - Supervised Learning of semantics
    - [kate/mooney 06] : set of predicates are known
    - [yu/ballard 07] : semantics = scene-region

- Unsupervised Semantic Acquisition :
  “right” granularity for concepts; dynamic predicates

What is a Symbol?

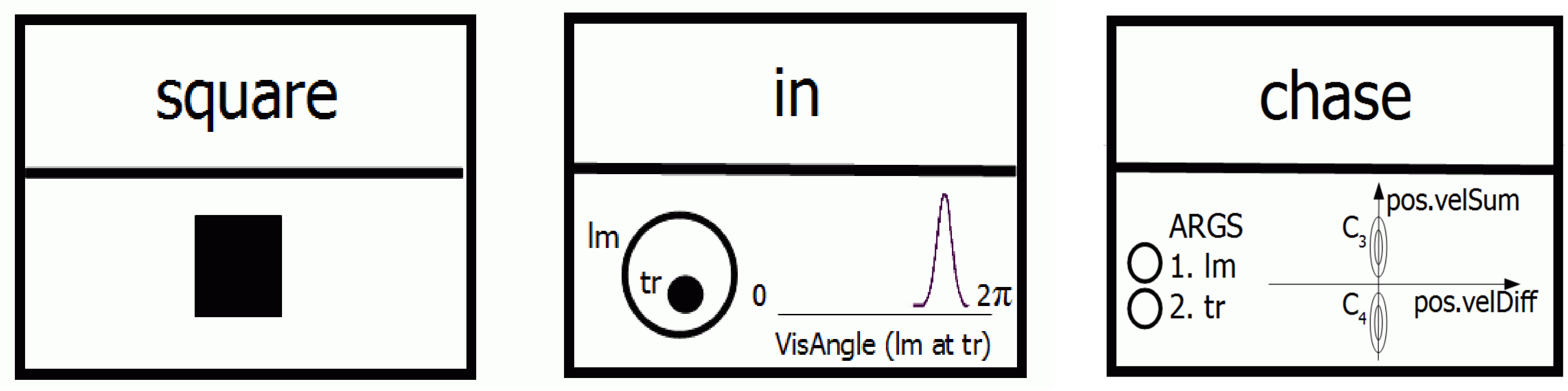
- grounded **lexicon**:





Lexicon

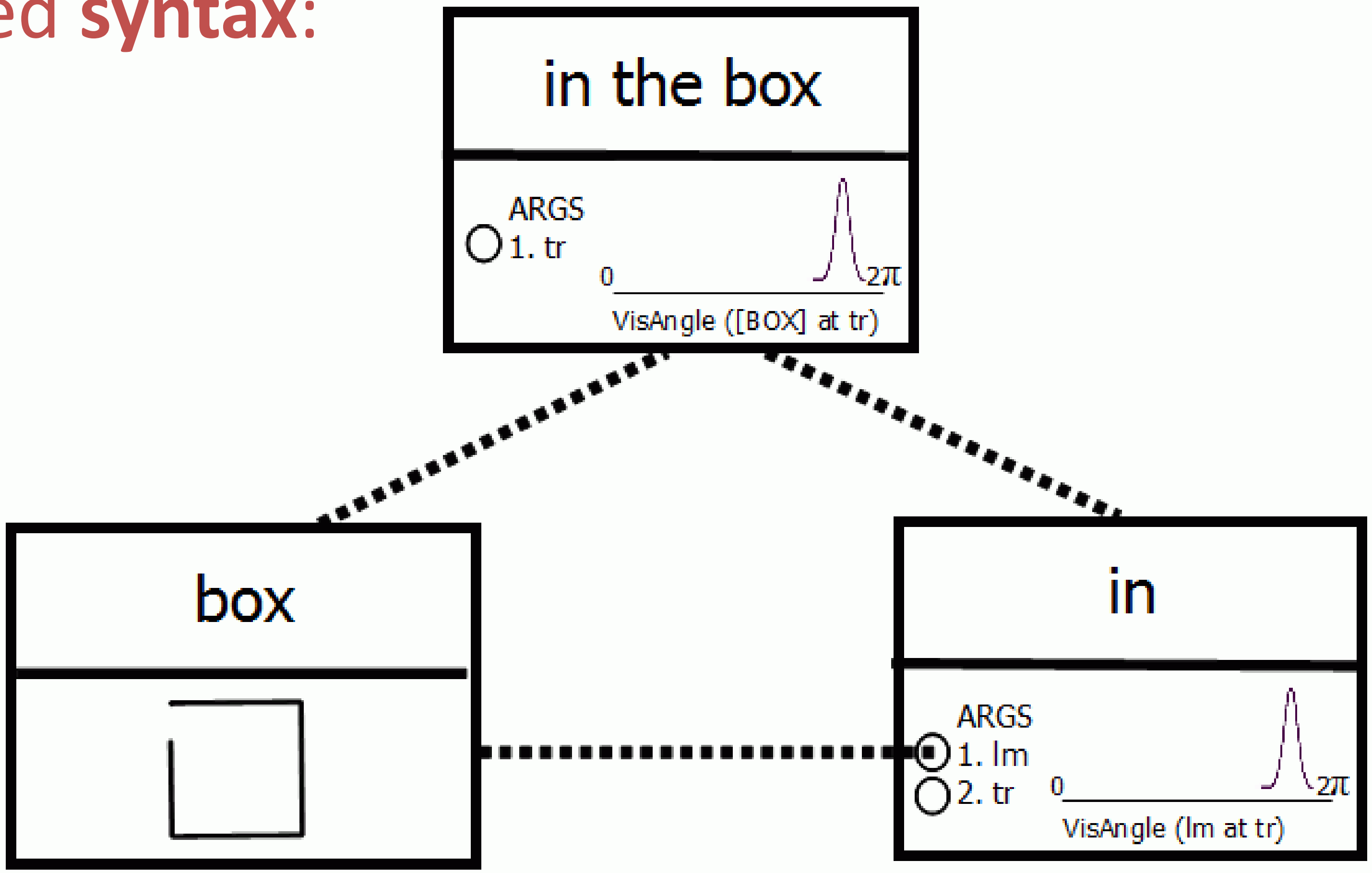
- grounded **lexicon**:



- semantic pole : perceptual patterns (image schemas)
  → probabilistic predicate + arguments

Grammar

- grounded **syntax**:



Minimal Commitment

- minimize prior knowledge in agent:
  - preference: minimize description lengths
    → inventory of machine learning algorithms
  - no knowledge of grammar – no POS tags, no syntactic structure
  - no knowledge of domain
- **bootstrapping** stage:
  - semantic schemas come first
  - language regularities later

POS categories are discovered

$$\begin{bmatrix} ball \\ block \\ box \\ circle \\ square \end{bmatrix} \begin{bmatrix} in \\ inside \\ into \end{bmatrix} \begin{bmatrix} chases \\ pushes \\ corners \\ the \end{bmatrix} \begin{bmatrix} big \\ large \\ little \\ the \end{bmatrix}$$

Acquisition Domains

Previous Work: Unsupervised Semantics

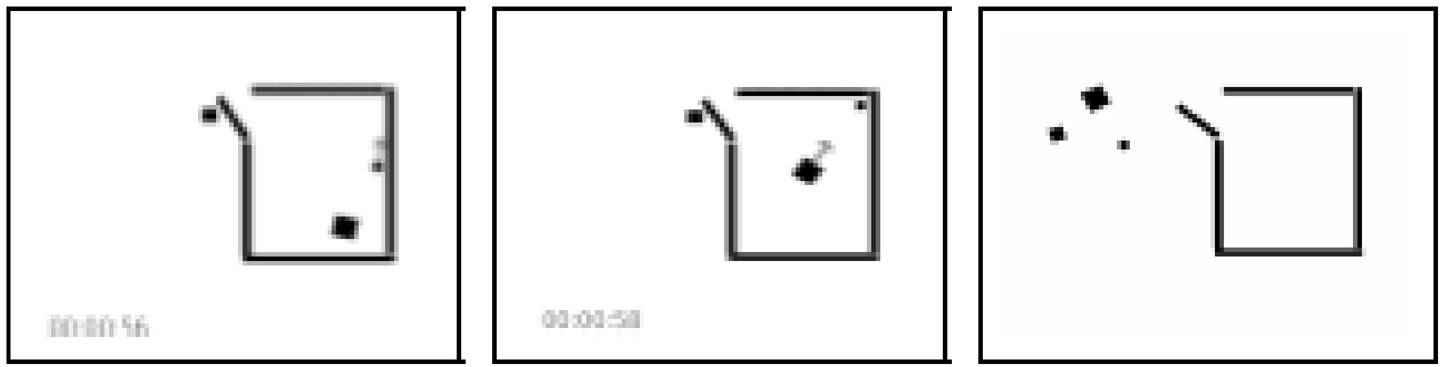
- single word or phrase learning [no grammar]
  - Hand-coded propositional (T/F) semantics
    - [plunkett etal 92]
    - [regier 96] (prepositions)
    - [steels 03] [roy/reiter 05] [caza/knott 12]
  - Supervised Learning of semantics
    - [kate/mooney 06] : set of predicates are known
    - [yu/ballard 07] : semantics = scene-region
- Unsupervised Semantic Acquisition :
  “right” granularity for concepts; dynamic predicates

Previous Work: Grammar

- Grammar learning:
  - Grammatical categories:
    - [redington etal 98] (RNN)
    - [wang / mintz 07] (frequent frame)
  - Grammar induction : Structure is known
    - No semantics:
      - [marino etal 07] [solan edelman 05]
    - Propositional semantics
      - [kwiatkowski zettlemoyer 10] (SVM)
      - [kim/mooney 12] (altered visual input)

Language Acquisition : Domain 1

- Perceptual input



- Discovery Targets:
  - semantics: objects, 2-agent actions, relations
  - lexicon : nominal, transitive verbs, preposition
  - lexical categories: N VT P Adj
  - constructions: PP VP S
  - sense extension (metaphor)   [nayak/mukerjee (AAAI-12)]

Language Acquisition : Domain 2

- Perceptual input

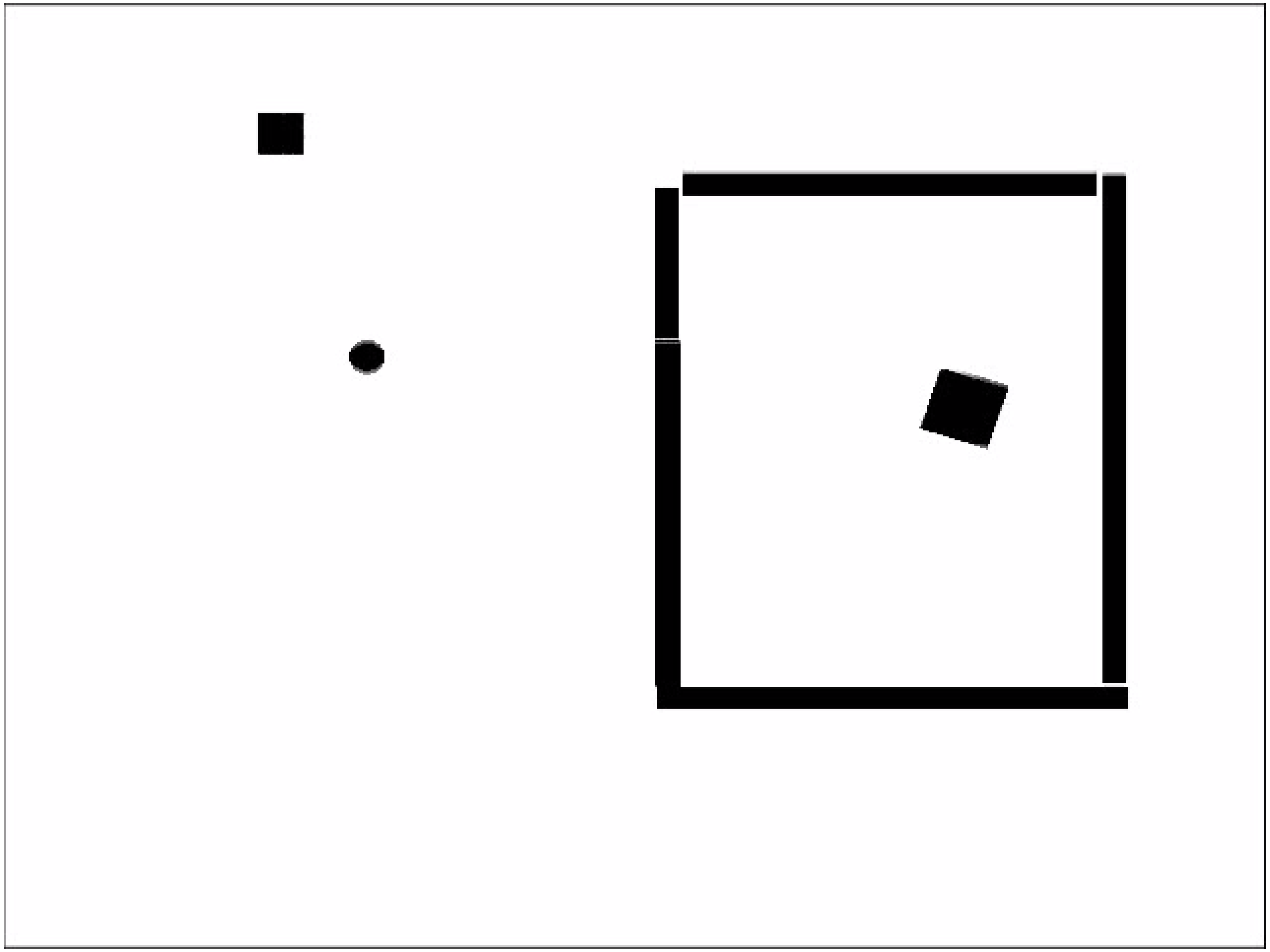
- Discovery Targets:
    - semantics: object categories, motion categories

Language Acquisition : Domain 2

- object categories

- Discovery Targets:
  - semantics: object categories, motion categories
  - lexicon : word boundaries, nominals, intransitive verbs
  - construction: intransitive VP

Video Fragment



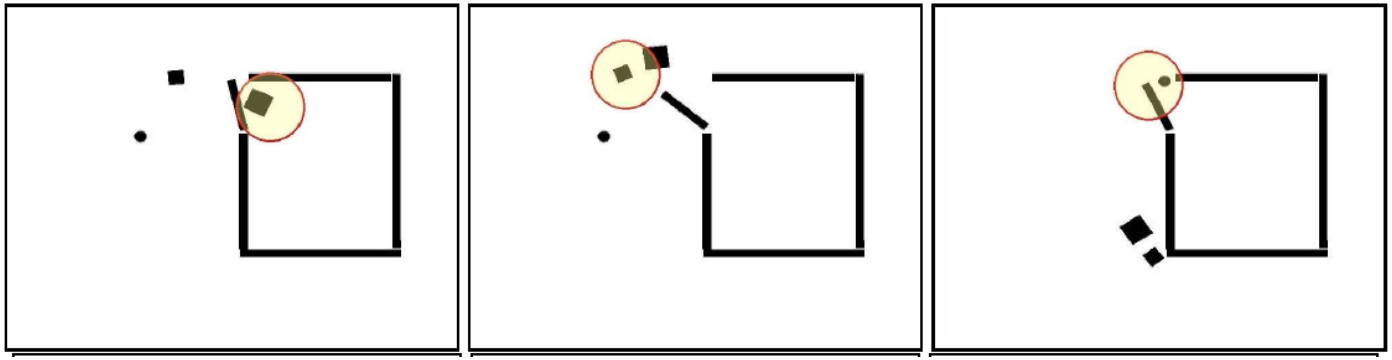
Discovering Language

- Perceptual structure discovery:
    - Given perceptual space *W* discover set of structures Γ that partition it into patterns relevant to agents goals.
    - Elements $\gamma \in \Gamma$ constitute a hierarchy; structures learned earlier are used for more complex patterns
- Linguistic Structure Discovery
    - Given set of sentences formed from words w $\in L$, discover set of subsequences Λ that result in a more compact description of the structure
    - Elements $\lambda \in \Lambda$ constitute a hierarchy, leaf nodes (POS) are subsets of L

Semantics First:
Objects / Nominals

Language Grounding: Entity/Object

- object = coherent salient region in perceptual space
    - object view schema [white maruti 800 from camera 1]
    - object schema [white maruti 800]
    - object category schema [car]
- bottom-up dynamic attention



Language – Meaning Associaction

- Relative Association (bayesian)

$$P(\gamma_j|\lambda_i) = \frac{P(\lambda_i|\gamma_j)P(\gamma_j)}{P(\lambda_i)} \propto \frac{P(\lambda_i|\gamma_j)}{P(\lambda_i)}$$

- Mutual association (contribution to M.I.)

$$P(\lambda_i,\gamma_j)\log\frac{P(\lambda_i,\gamma_j)}{P(\lambda_i)P(\gamma_j)}$$

$$I(\Gamma,\Lambda) = \sum_i\sum_j P(\lambda_i,\gamma_j)\log\frac{P(\lambda_i,\gamma_j)}{P(\lambda_i)P(\gamma_j)}$$

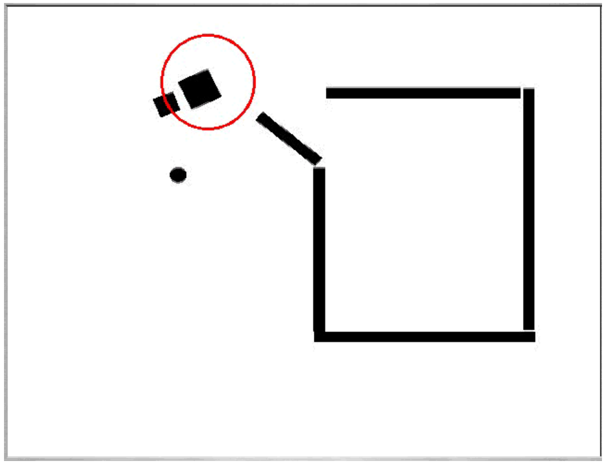
Language Grounding: Nominals

[BS]			[SS]			[C]		
word(s)	A_{ij}^{rel}	A_{ij}^{mut}	word(s)	A_{ij}^{rel}	A_{ij}^{mut}	word(s)	A_{ij}^{rel}	A_{ij}^{mut}
square	0.70	1.41	little	0.66	0.79	circle	0.79	2.11
big	0.89	1.11	small	0.72	0.63	square	0.41	1.54
box	0.69	0.78	square	0.46	1.12	little	0.68	1.22
the big	0.87	0.71	small square	0.93	0.53	the little	0.71	0.81
big square	0.94	0.75	little square	0.89	0.46	little circle	0.91	0.60
large square	0.86	0.15	the little	0.70	0.54	the big	0.48	0.61

Perceptual Discovery : Actions : Verbs

Perceptual Discovery: 2-agent actions

- Consider every pair of objects A,B
  A : attended to object (tr)
  B : other object (landmark, lm).



- 2 features suffice:

relative-velocity and relative position

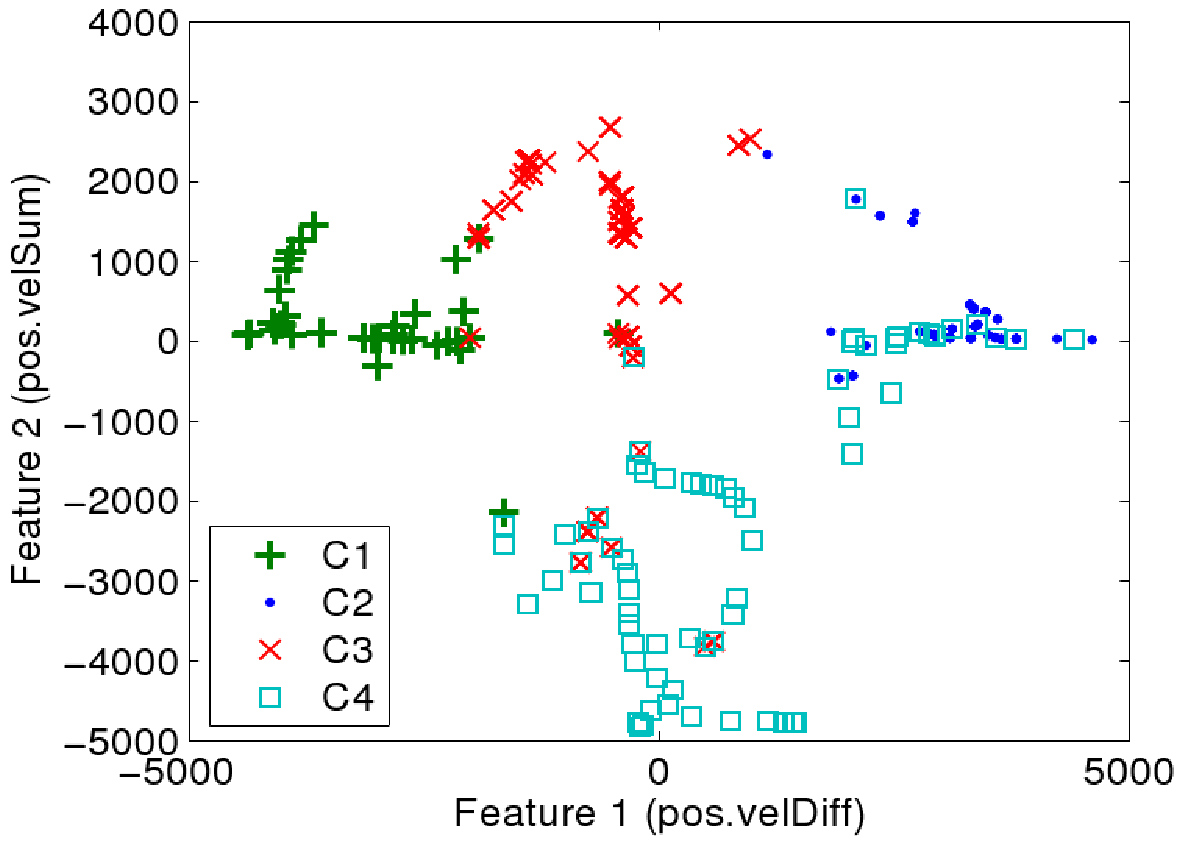
$$pos{\cdot}velDiff : (\vec{x}_B - \vec{x}_A) \cdot (\vec{v}_B - \vec{v}_A)$$

relative pose and the sum of the velocities

$$pos{\cdot}velSum : (\vec{x}_B - \vec{x}_A) \cdot (\vec{v}_B + \vec{v}_A)$$

Perceptual Discovery: 2-agent actions

- Static time-shots of feature space trajectories



Switching the in-Focus agent

- Human Labels *(CC, MA, Chase)* → Ground Truth
- Label Vs Cluster assigned

	C_1	C_2	C_3	C_4	Total	%	TCA
CC	399	6	10	29	444	90	
MA	16	311	5	48	380	82	84
Chase	21	59	149	154	383	79	

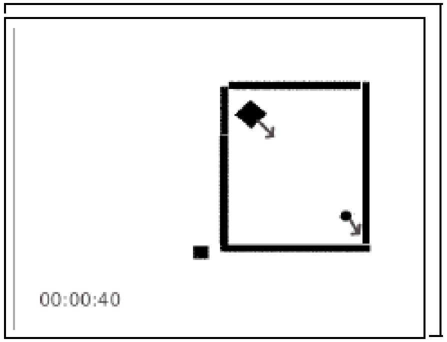
Number of Clusters from MNG = 4 when *Edge Aging* = 30 (0.9 prob)

CC: Come-Closer (C1), MA: Move Away (C2), C_3 & C_4 : Chase

Chase sub-categories:

Chase_*RO-chases-LO*: C_3 →

Chase_*LO-chases-RO*: C_4 →



Hierarchy in Concept Space

- More clusters → Reveals category hierarchy:

Come-Closer
- Come-Closer-RO-static
- Come-Closer-LO-static
- Come-Closer-both-moving

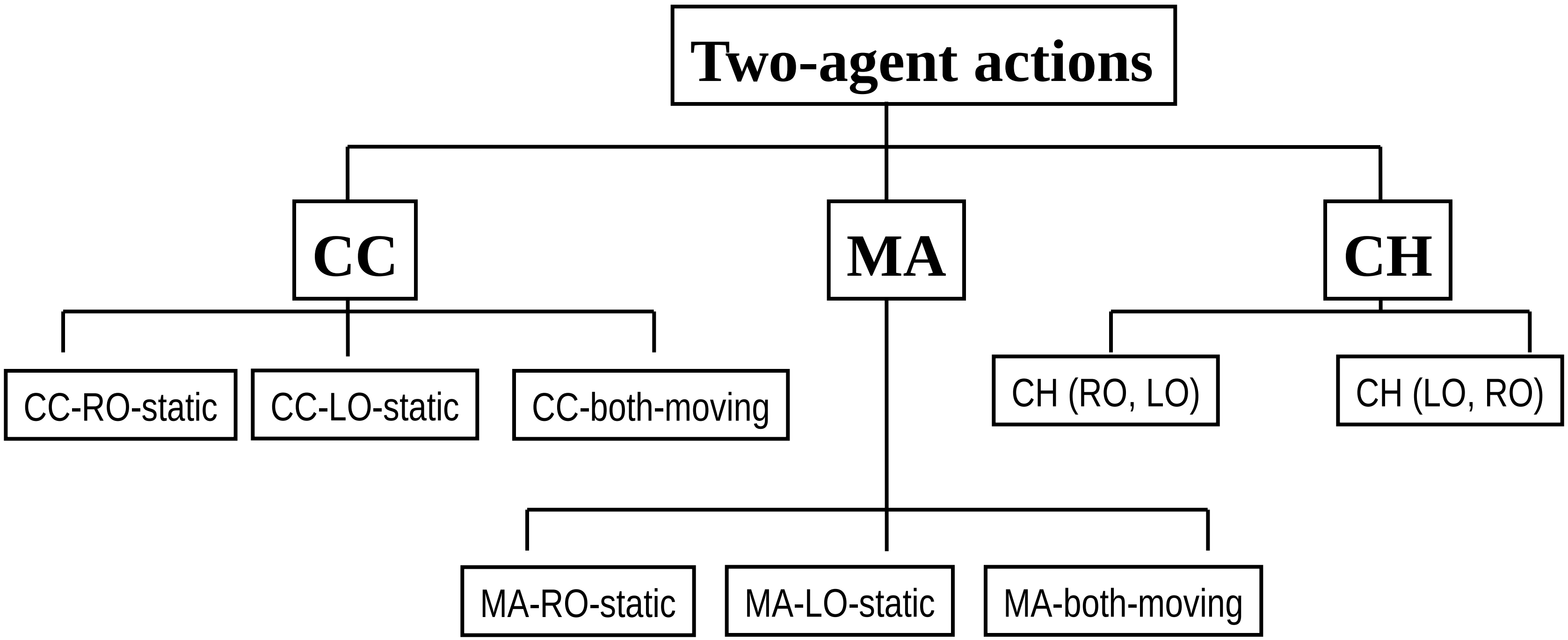
- Similarly: Move-Away : 3 subclasses

Number of Clusters from MNG = 8 when *Edge Aging* = 16

	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
CC	201	3	9	20	189	21	1	0
MA	8	126	4	45	9	1	181	6
Chase	1	9	142	151	13	9	32	26

C_1, C_5, C_6 : sub-classes of *Come-Closer*; C_2, C_7, C_8 :of *Move-Away*

Two agent action ontology



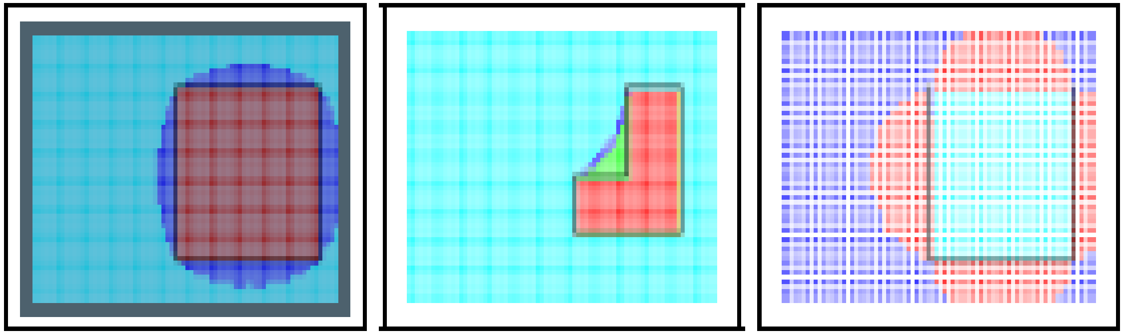
Learning verbs

CLUSTER 1 (Come-Close)		CLUSTER 2 (Move-Away)		CLUSTER 3 (Chase)		CLUSTER 4 (Chase)	
ONE WORD LONG LINGUISTIC LABELS(MONOGRAMS)							
corner	0.077	away	0.069	chase	0.671	chase	0.429
move	0.055	move	0.055	other	0.185	after	0.112
attack	0.042	chase	0.049	around	0.183	out	0.033
TWO WORD LONG LINGUISTIC LABELS(BIGRAMS)							
each other	0.086	move away	0.111	chase around	0.306	chase after	0.218
move toward	0.065	go into	0.035	each other	0.227	just chase	0.060
toward each	0.065	into with	0.035	chase each	0.198	chase out	0.058
THREE WORD LONG LINGUISTIC LABELS(TRIGRAMS)							
move toward each	0.182	go into with	0.099	chase each other	0.558	just chase out	0.142
toward each other	0.182	run away out	0.051	start run away	0.132	run away out	0.047
move close together	0.114	scare in corner	0.032	begin to move	0.127	to go after	0.031

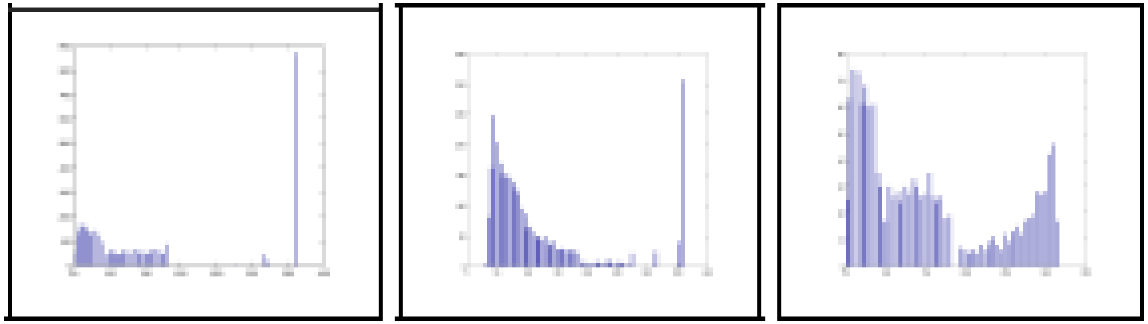
Discovering Containment Relations : Prepositions

Clustering spatial relations

Feature Commitment:
Visual angle subtended
at trajector by landmark

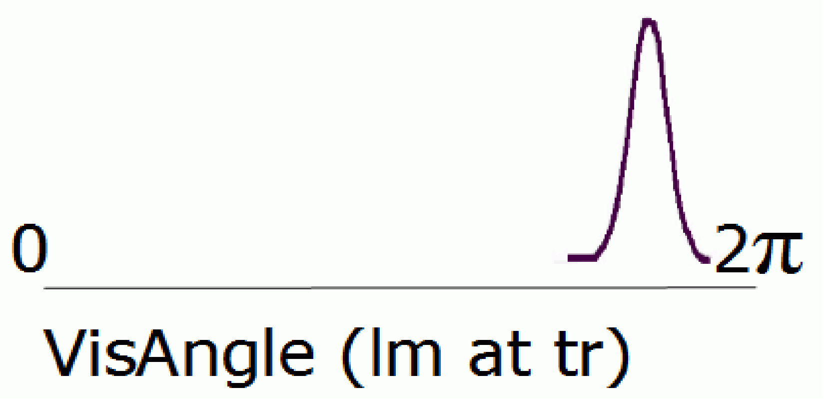


Histogram of visual
subtended angle
for the 3 shapes

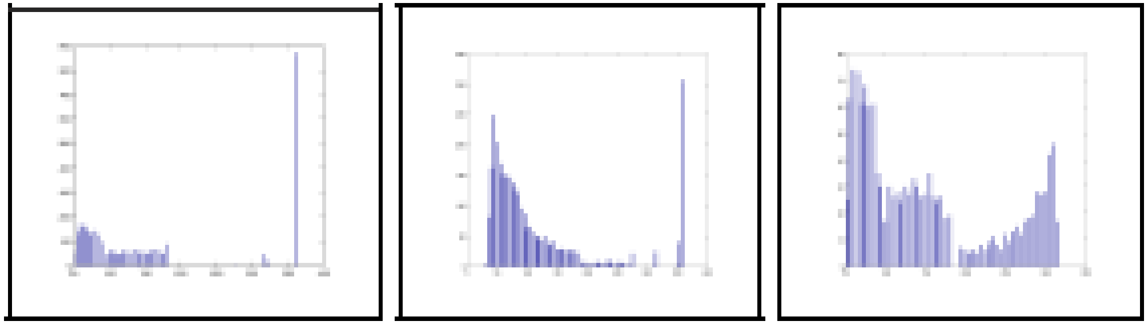


Clustering spatial relations

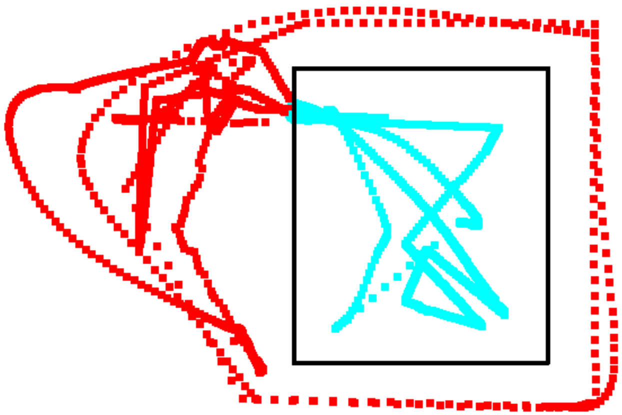
IN cluster
(emergent)

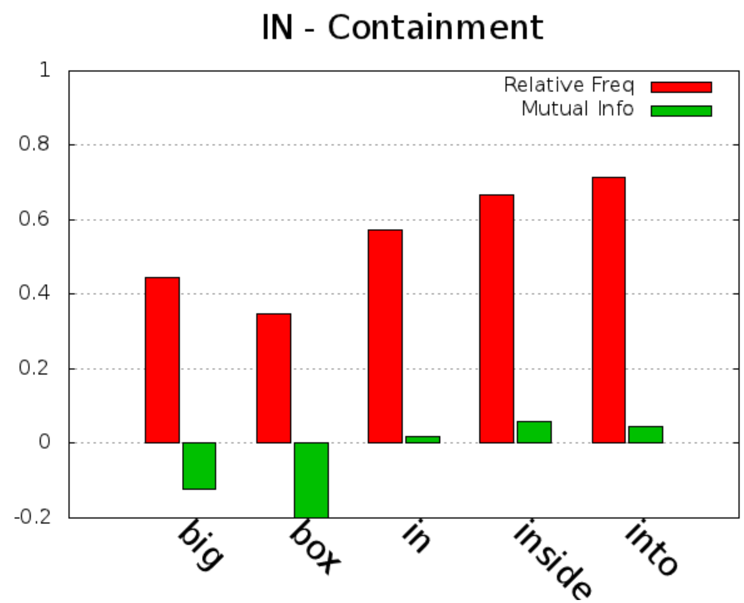


Histogram of visual
subtended angle
for the 3 shapes



Words for motions ending in / out





IN	A_{ij}^{rel}	A_{ij}^{mut}	INTO	A_{ij}^{rel}	A_{ij}^{mut}	OUT OF	A_{ij}^{rel}	A_{ij}^{mut}
inside	0.79	11.78	into	0.82	6.98	out	0.65	5.71
into	0.90	9.43	inside	0.53	1.03	leaves	1.00	4.16
in	0.61	4.16	enters	1.00	4.85	exits	1.00	3.46

Syntax discovery and Semantic Association

Syntax Discovery

- Syntactic discovery:
    - Given input text, attempt to find graph that results in minimizing the description length
    - Relational Graph RDS: patterns as nodes; edges as transitions
    - Attempt to edit RDS to detect significant patterns
    - Equivalence classes emerge at the nodes

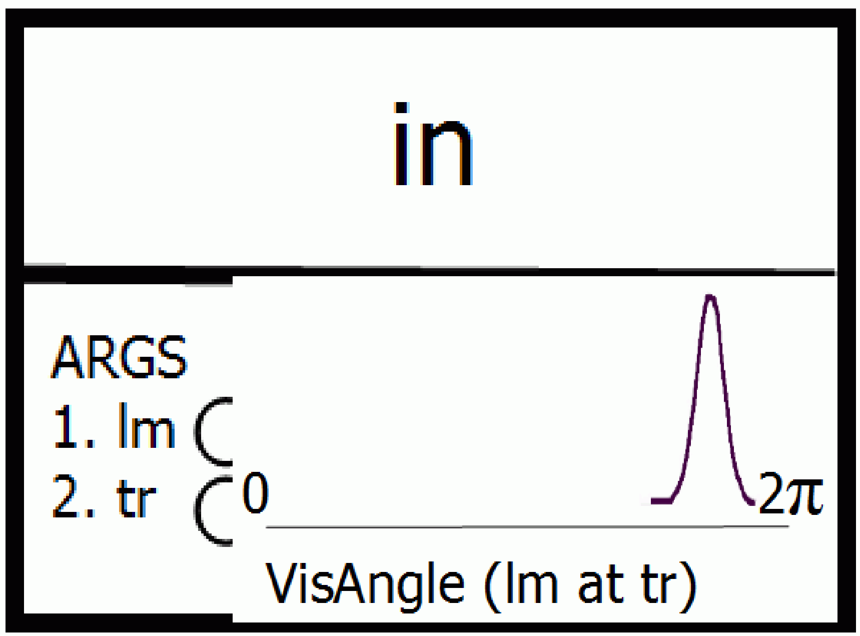
$$\left[\left[\begin{array}{c} in \\ inside \\ into \end{array}\right] \rightarrow the\right] \rightarrow box$$

ADIOS [solan / edelman 05]

Computing the Image Schema

Our reflective baby
has discovered:
"in" = label corresponding to this
image schema

Hence: symbol for [IN] is



.

(note: this is an early, very basic,
low-confidence characterization

Language Structures : Verbs

$$1.\ \begin{bmatrix} \textit{the} \rightarrow \begin{bmatrix} \textit{big} \\ \textit{large} \end{bmatrix} \rightarrow \textit{square} \\ \textit{the} \rightarrow \textit{square} \end{bmatrix} \rightarrow \begin{bmatrix} \textit{scares} \\ \textit{approaches} \\ \textit{chases} \end{bmatrix} \rightarrow \begin{bmatrix} \textit{the} \rightarrow \begin{bmatrix} \textit{small} \\ \textit{little} \end{bmatrix} \end{bmatrix}$$

$$2.\ \begin{bmatrix} \textit{the} \rightarrow \begin{bmatrix} \textit{ball} \\ \textit{box} \\ \textit{door} \\ \textit{square} \end{bmatrix} \\ \textit{circle} \\ \textit{it} \end{bmatrix} \rightarrow \begin{bmatrix} \textit{moved} \\ \textit{moves} \\ \textit{runs} \end{bmatrix}$$

ADIOS [solan / edelman 05]

Hindi Acquisition: Word learning

[BS]			[SS]			[C]			[IN]		
word(s)	A^{rel}_{ij}	A^{m}_{ij}	word(s)	A^{rel}_{ij}	A^{m}_{ij}	word(s)	A^{rel}_{ij}	A^{m}_{ij}	word(s)	A^{rel}_{ij}	A^{m}_{ij}
बक्सा baksA/box	.77	.37	बक्सा baksA/box	.62	.44	गोला golA/ball	.83	.54	अन्दर andar/in	.80	1.30
बडा(badA/ big) बक्सा	.85	.18	छोटा(chota/ small) बक्सा	.90	.25	बक्से के(ke/–)	.63	.27	बाहर (bA-har/out)	.78	.73

Incipient Syntax

$$\begin{bmatrix} \text{डब्बे(dabbA/box)} \\ \text{बक्से(bakse/box)} \end{bmatrix} \rightarrow \begin{matrix}\text{के}\\ \text{(ke/–)}\end{matrix} \rightarrow \begin{bmatrix} \begin{matrix}\text{बाहर}\\ \text{(bAhar/out)}\end{matrix} & \begin{bmatrix}\text{बाहर(bAhar/out)}\\ \text{आ(aa/come)}\\ \text{भाग(bhAg/run)}\end{bmatrix} & \begin{matrix}\text{जाता}\\ \text{(jAtA/goes)}\end{matrix}\end{bmatrix}$$

Scaling up?

- Domain-specific grammar
- After learning several such grammars for different domains, how to merge?

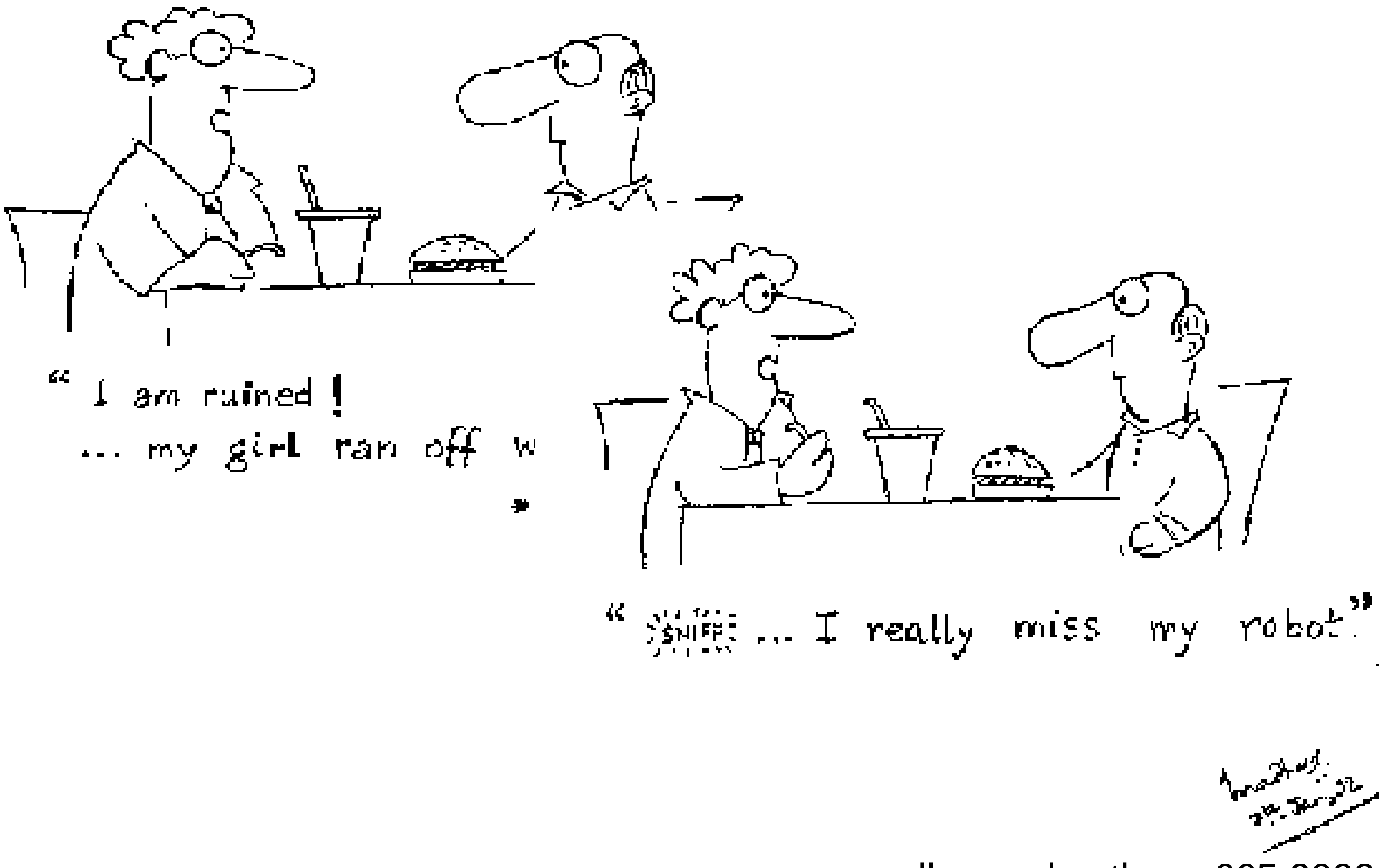
Conclusion

The power of Latent Discovery

- Data is not randomly drawn
- Discovering implicit strucgture in data →
  - Tacit Knowledge
- Much work remains in scaling up
  - How to merge diverse domains?
- **Learning** to plan motions

  \- Learn low-dimensional representations of motion
- Learning **Language** / **Vision**

Humans and Robots



madhur ambastha cs665 2002