Symmetry and Structural Bootstrapping

Utah: T Henderson
E Cohen
A Joshi
W Wang

NCSU: E Grant

SCHOOL OF COMPUTING

THE UNIVERSITY OF UTAH

Xperience: Oct. 5, 2013

Structural Bootstrapping (from Xperience URL)

SCHOOL OF COMPUTING

a method of **building generative models**

- leverage existing experience
  - to **predict unexplored action effects**, and
  - to **focus the hypothesis space** for learning novel concepts
- A **developmental approach** enables
  - rapid generalization, and
  - acquisition of new knowledge and skills from little additional training data
- shared concepts enable enactive agents to **communicate** effectively with each other and with humans.
- can be **employed at all levels** of cognitive development (e.g. sensorimotor, planning, communication).

rewrite rule

S:=T,Rf,Ro(90)

P:=T,Rf,Ro(2pi/n)

Symbol Strings

Actuation Modes

THE UNIVERSITY OF UTAH

Overall Idea

Sensor signals

1-D

2-D

n-D

Actuator command signals

How to represent, correlate, store and re-use sensorimotor combinations (affordances)?

G-Reps

- Parse 1-D, 2-D, … n-D signals into basic symmetries (represented as **symbols**)
- Build sequences
- Store and index

- Filter by affordance
- Affordances keyed to 3D space group operations (translation, rotation)

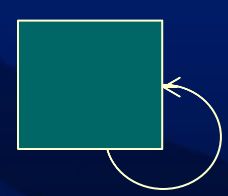
- Each sequence grounded in particulars (shape basis, color, etc.)

Example: Square Shape

Leyton's characterization:

1. Translation symmetry of line (T): gets one side

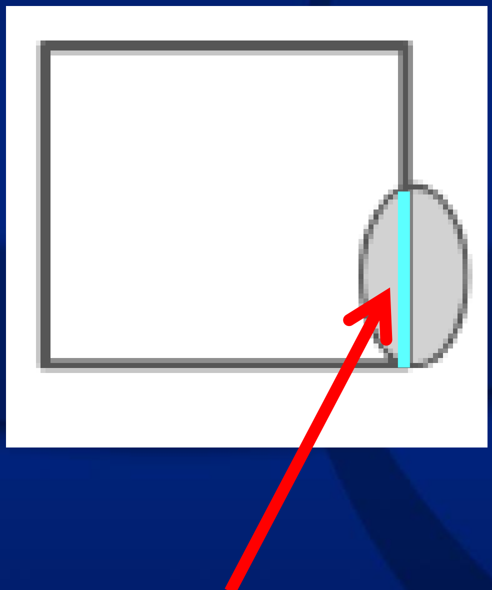
2. Acted on by cyclic (rotation) group (C_4): gets rest



etc.

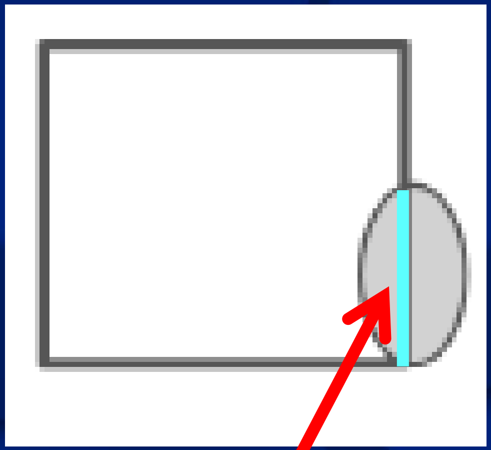
This is a generative model to create shape

G-Reps: Putting Symmetries Together



Shape Basis

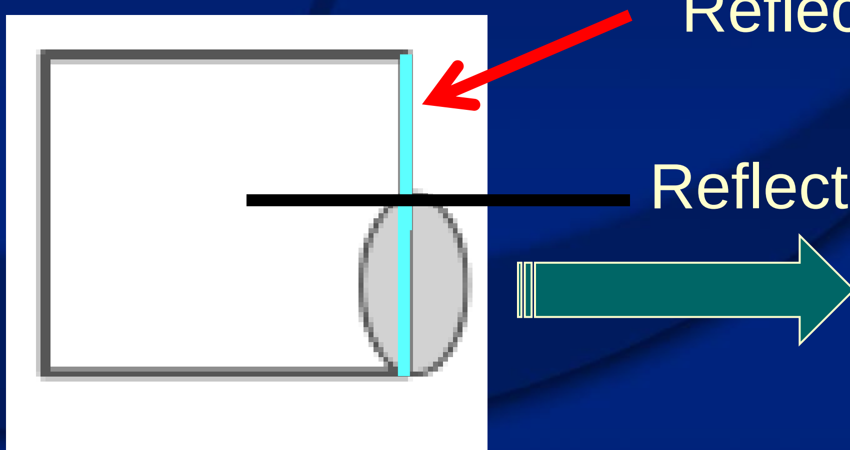
G-Reps: Putting Symmetries Together



Translation Symmetry (T)

Shape Basis

G-Reps: Putting Symmetries Together



Reflection Symmetry (R)

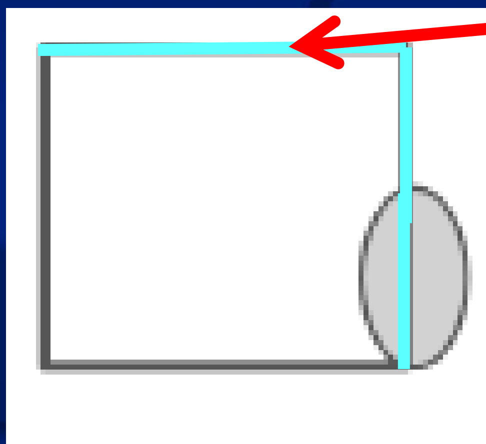
Reflection Axis

Translation Symmetry (T)

G-Rep: T<R

G-Reps

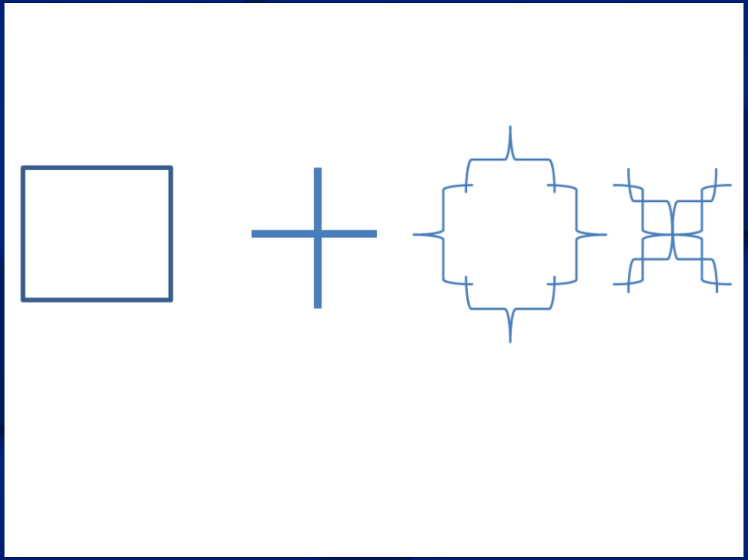
Rotation Symmetry (C)

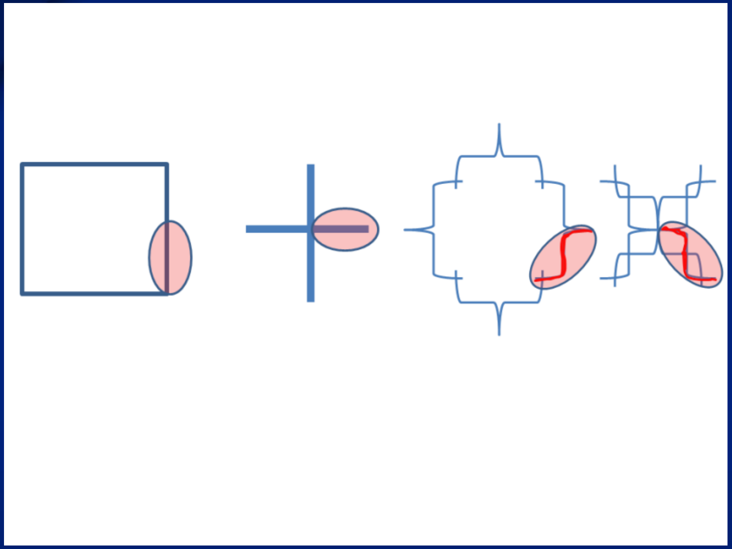


Rotation Axis

G-Rep: T<R<C_4

G-Reps Abstract Shape



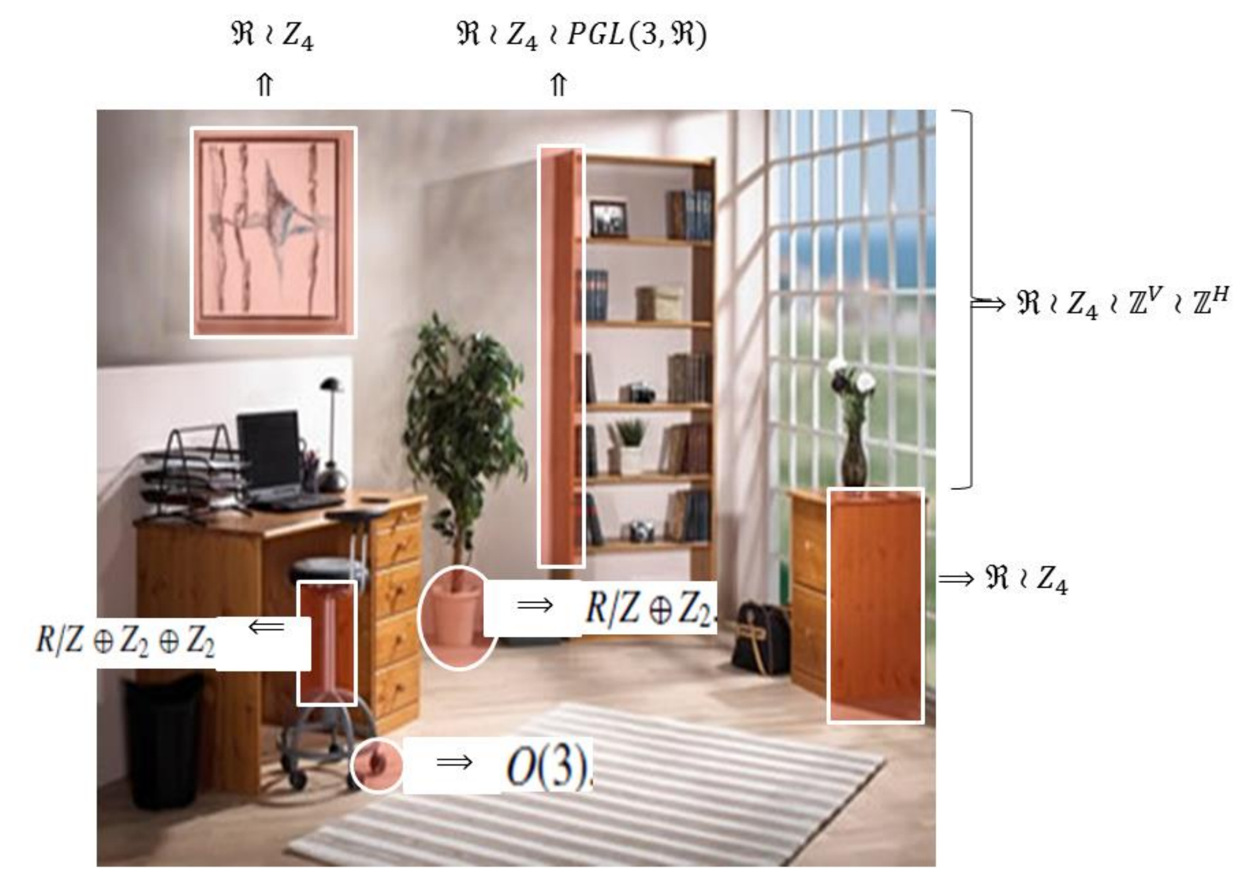


All these shapes have Same G-Rep

But these shapes have different shape basis

THE UNIVERSITY OF UTAH

Group Product Scene Description



G-Rep: Wreath Product

- Symmetry group of regularly branching, rooted trees
- Relabelings that only permute nodes at fixed distance from root.
- Group-based convolutions (e.g., DFT)
- Think of cognitive wreath product as capturing combinatorics of object structure

Wreath Product (cont'd)

The connection between the DFT and group theory is immediate from its definition. Given an input $f = (f(0), \ldots, f(N-1)) \in \mathbb{C}^N$, its DFT is defined as the collection of sums (Fourier coefficients)

$$\hat{f}(k) = \sum_{n=0}^{N-1} f(n) W^{nk} \tag{2.1}$$

where $W = e^{2\pi j/N}$ for $j = \sqrt{-1}$. The N^{th} roots of unity W^{jk} are the values of the irreducible characters (or in this case, equivalently, the irreducible matrix elements) of the cyclic group $\mathbb{Z}/N\mathbb{Z}$. If computed for all $k \in \{0, \ldots, N-1\}$, the expression (2.1) computes the change of basis for the function $f : \mathbb{Z}/N\mathbb{Z} \to \mathbb{C}$ from the basis of delta functions

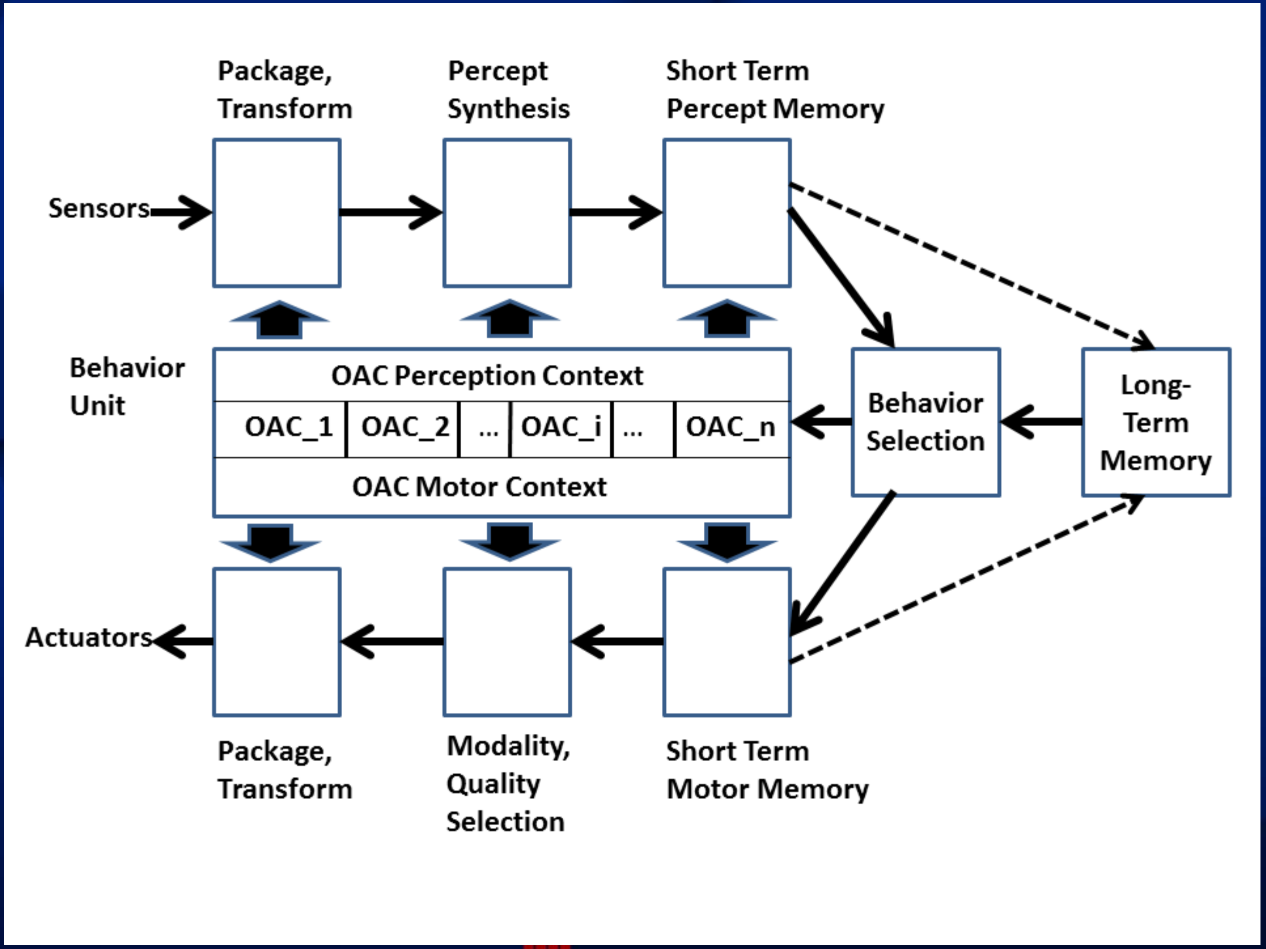
$$\delta_n(k) = \begin{cases} 1 & \text{if } n = k \\ 0 & \text{otherwise} \end{cases}$$

to the basis of irreducible matrix elements

$$\chi_n(k) = W^{nk}.$$

Taken from: "A Wreath Product Group Approach to Signal and Image Processing: Part I – Multi-resolution Analysis," Foote et al. 1999

Current Work: OAC Architecture

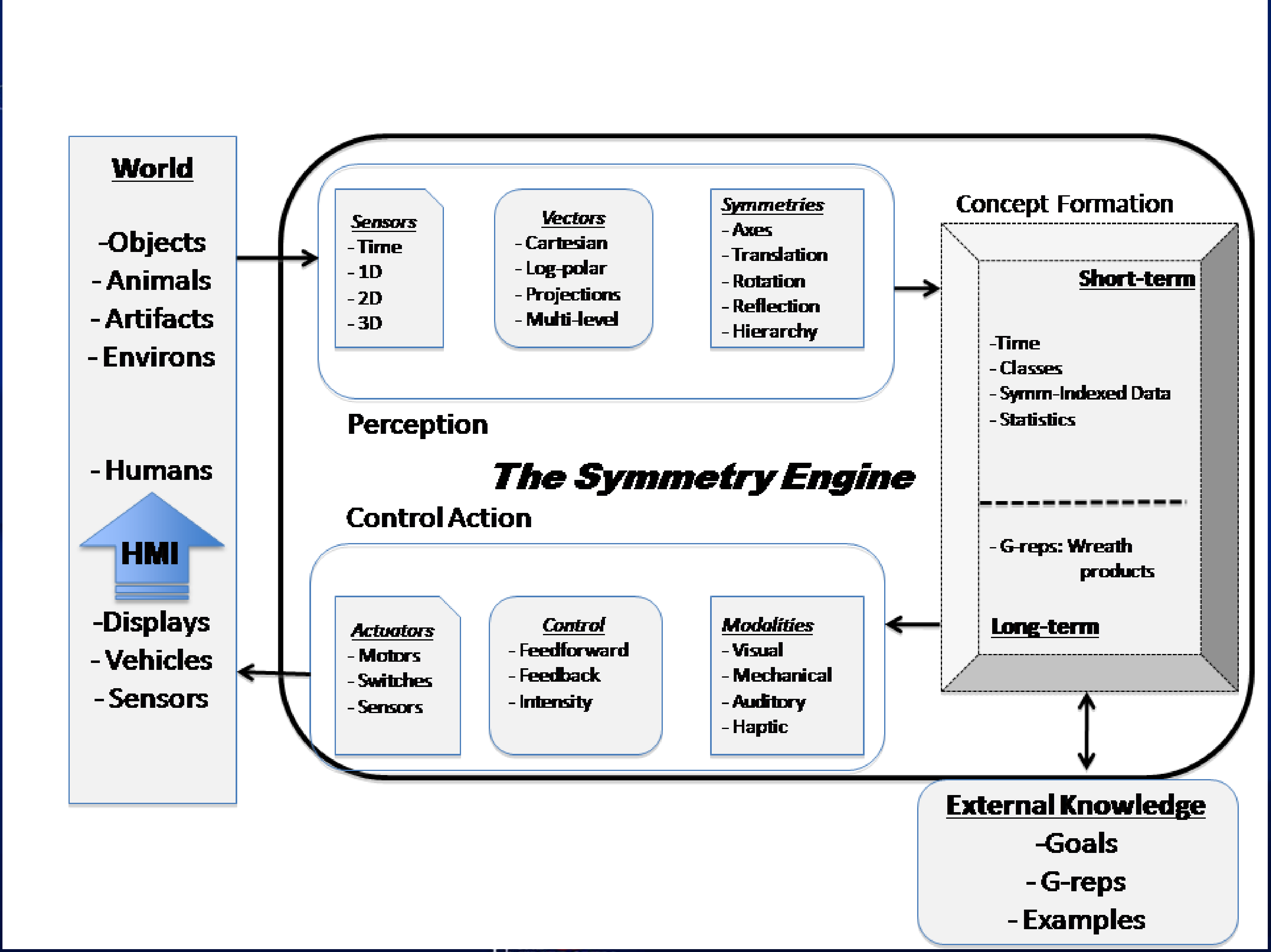


SCHOOL OF COMPUTING

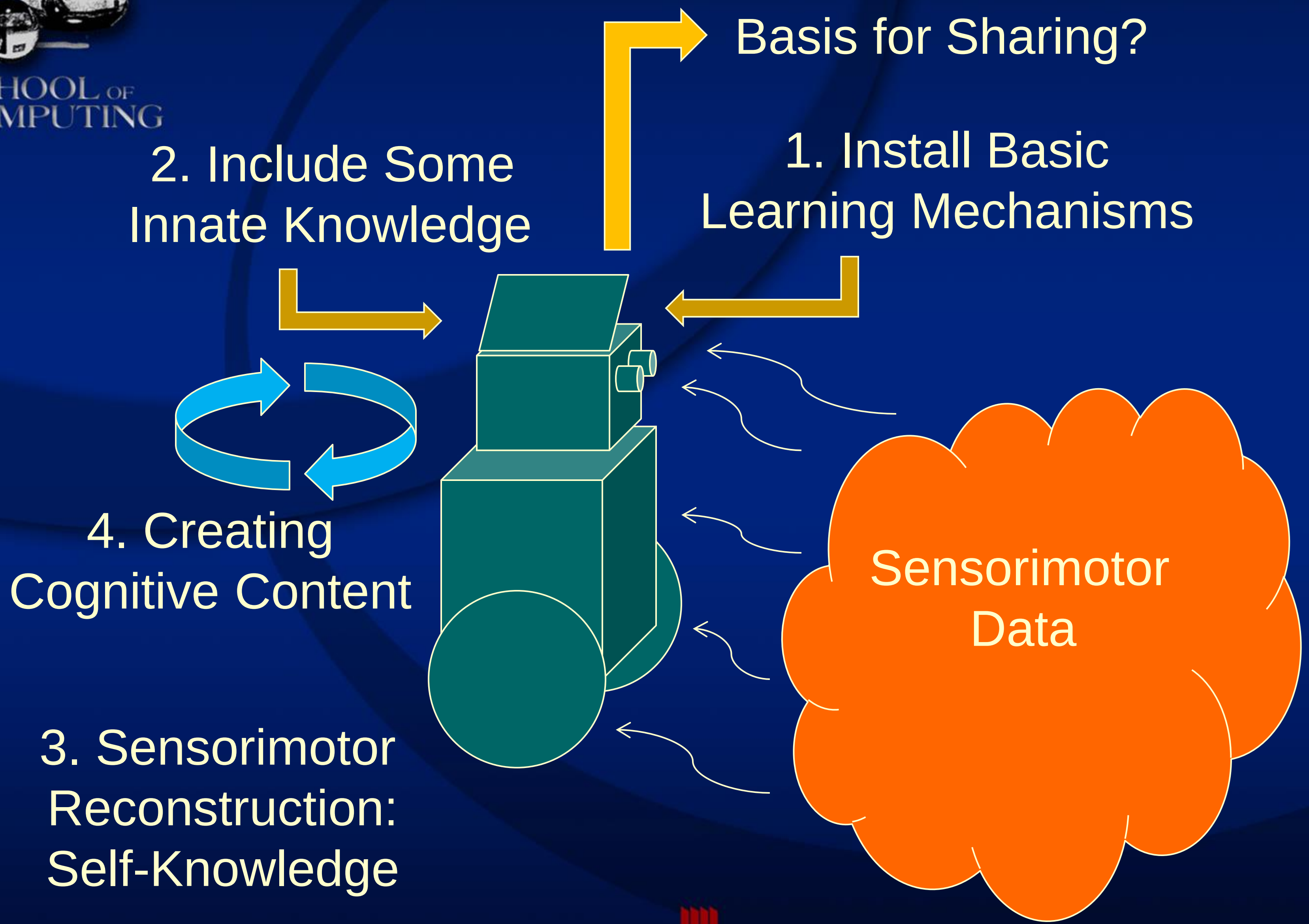
Using Symmetry in OACs

- Symbolic representations of perceptual symmetries
- Symbolic representations of actuation symmetries
- Parameters for symbols
- Context (from position of symbols in string)

Goal: The Symmetry Engine



Issue: Percepts → Knowledge



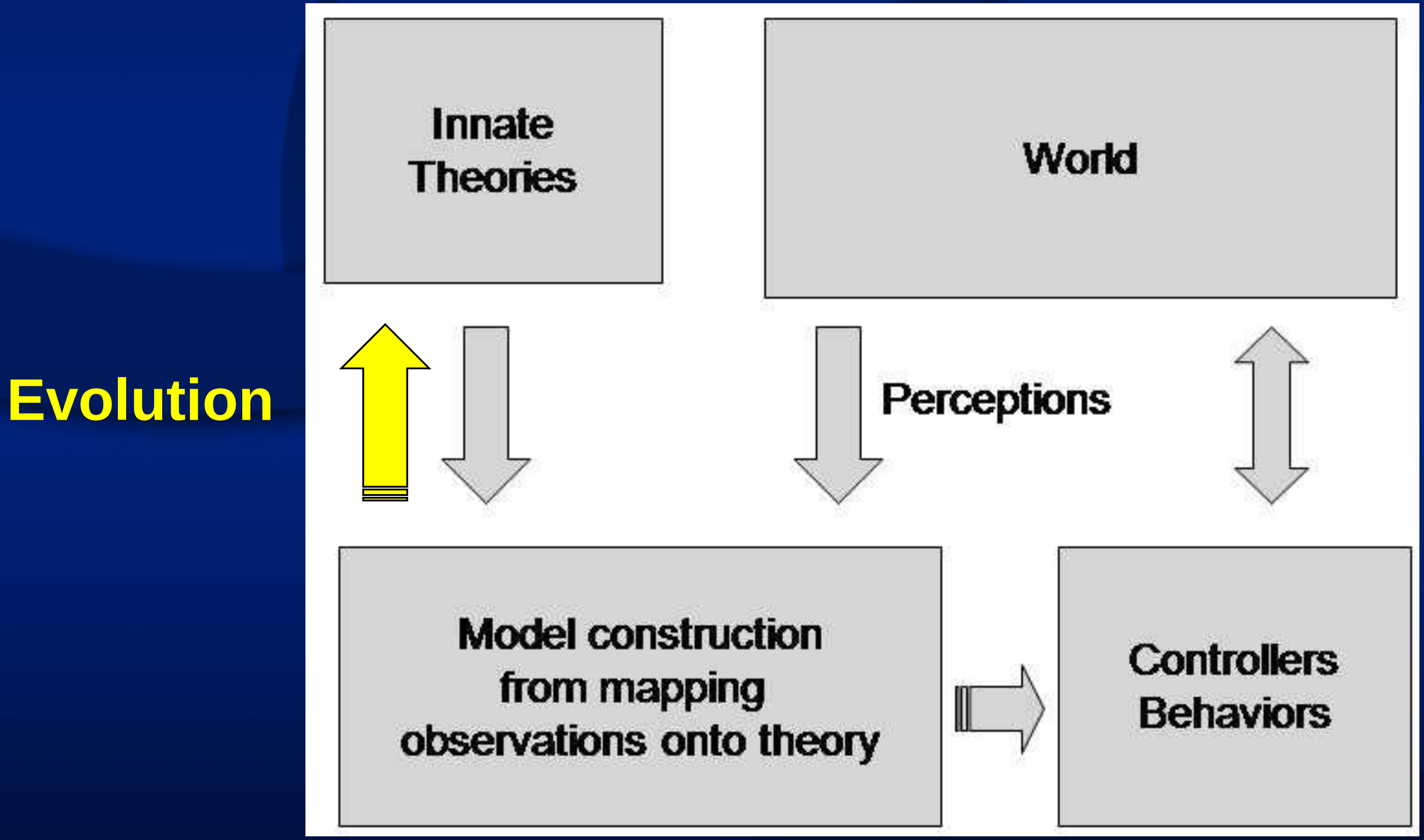
Innate Knowledge Options (see Sloman's work)

Much knowledge, including how to acquire knowledge in domain, is innate

versus

All knowledge derived from experience using general learning methods (e.g., reinforcement learning)

Innate Theories



Does Anybody In Robotics Use Innate Theories?

In fact, almost everybody:

- Subsumption: innate knowledge in HW
- RoboEarth, ARMAR III, SOAR: innate knowledge about sensors, actuators, world objects and actions, language, etc.
- Arny: spline way points for throwing motion

➔ Exceptions: some bootstrapping systems (but even they have some innate things!)

SCHOOL OF COMPUTING

SO, WHAT SHOULD BE INNATE?

Basic Questions

1. What are the sensorimotor grounds of higher cognition?

2. What kinds of mental representations, if any, are necessary as explanatory constructs and which are innate?

➔ I.e., **What can we use to structure (inform) sensorimotor data to obtain useful knowledge?**

Current Approaches

- Ontologies
- Coordinate Frames
- Models (2D and 3D representations of faces, animals, things, …)
- Goals
- Language (both speech processing and synthesis)
- …

A Proposal: Symmetry Theories

A symmetry defines an invariant.

Let's consider a simple example:

Group Theory (important for symmetry!)

E.g., Group Theory

Suppose we have a set of elements, **S**, an operator (+) and the following four axioms:

1. **Closure**: a,b ∈ S ➔ a+b ∈ S
2. **Associativity**: a+(b+c) = (a+b)+c
3. **Identity element**: ∃e ∈ S ∋ ∀a ∈ S (a+e) = a
4. **Inverse element**: ∀a ∈ S ∃a^{-1} ∈ S ∋(a+a^{-1}) = e

Group Theory Example

Consider translations as elements and application of a translation as the operator [called T(a)]:

1. **Closure**: T(a_1)+T(a_2) = T(a_1+a_2)
2. **Associativity**: T(a_1)+T(a_2+a_3) = T(a_1+a_2) + T(a_3)
3. **Identity element**: T(a_1)+T(0) = T(a_1)
4. **Inverse element**: T(a_1)+T($-a_1$) = T(0)

This is then a **model** for group theory.

Group Theory (cont'd)

How is this useful?

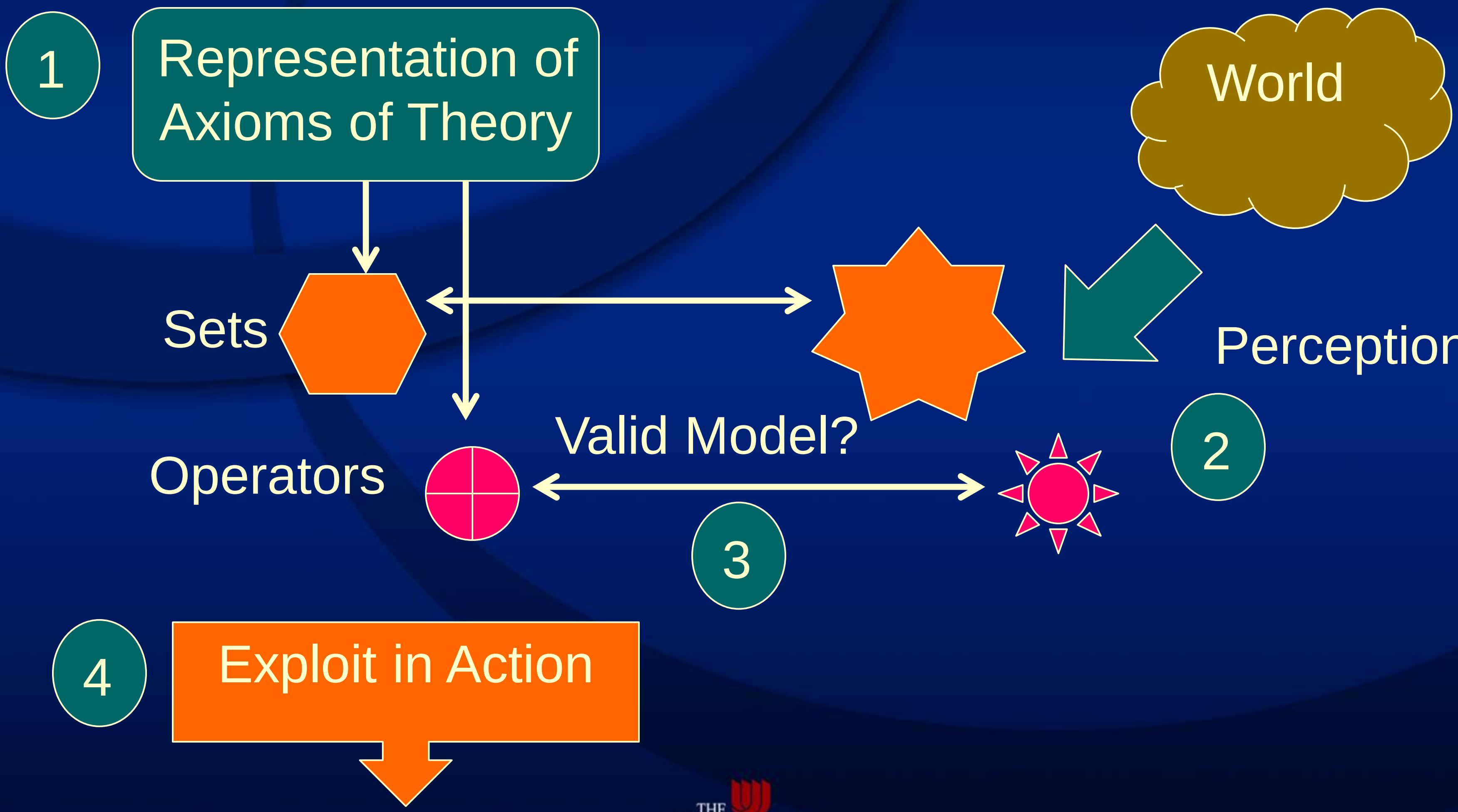
➔If we determine that a specific actuator has this property, then we know it is a prismatic joint and we can exploit this knowledge using the theory

➔ Moreover, we will know that the actuator is similar to other actuators with this property (they are all models of a group)

Symmetry Theory Hypothesis

Semantic cognitive content may be effectively discovered by restricting sensor-actuator solutions to be models of specific symmetry theories intrinsic to the cognitive architecture.

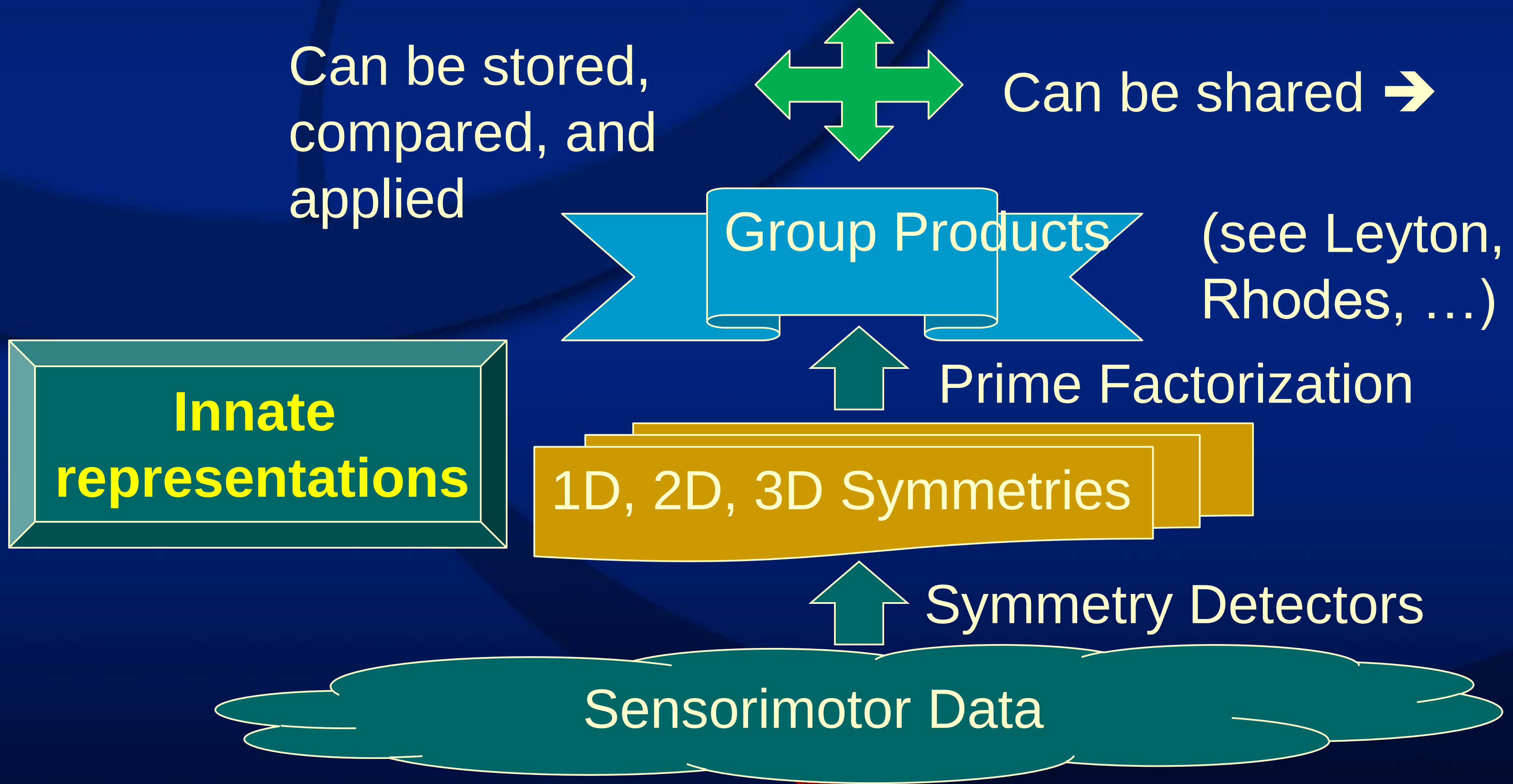
Symmetry Theory Operational Issues



Theory Benefits

1. Similarities between different models can be known because they model the same theory
2. True statements can be known about the model as derived from the theory without any experiential basis by the individual
3. Conflicting models can co-exist because the theories differ (thus, there is not one logical system which is inconsistent)

Symmetry Based Cognition



Symmetry Based Cognition

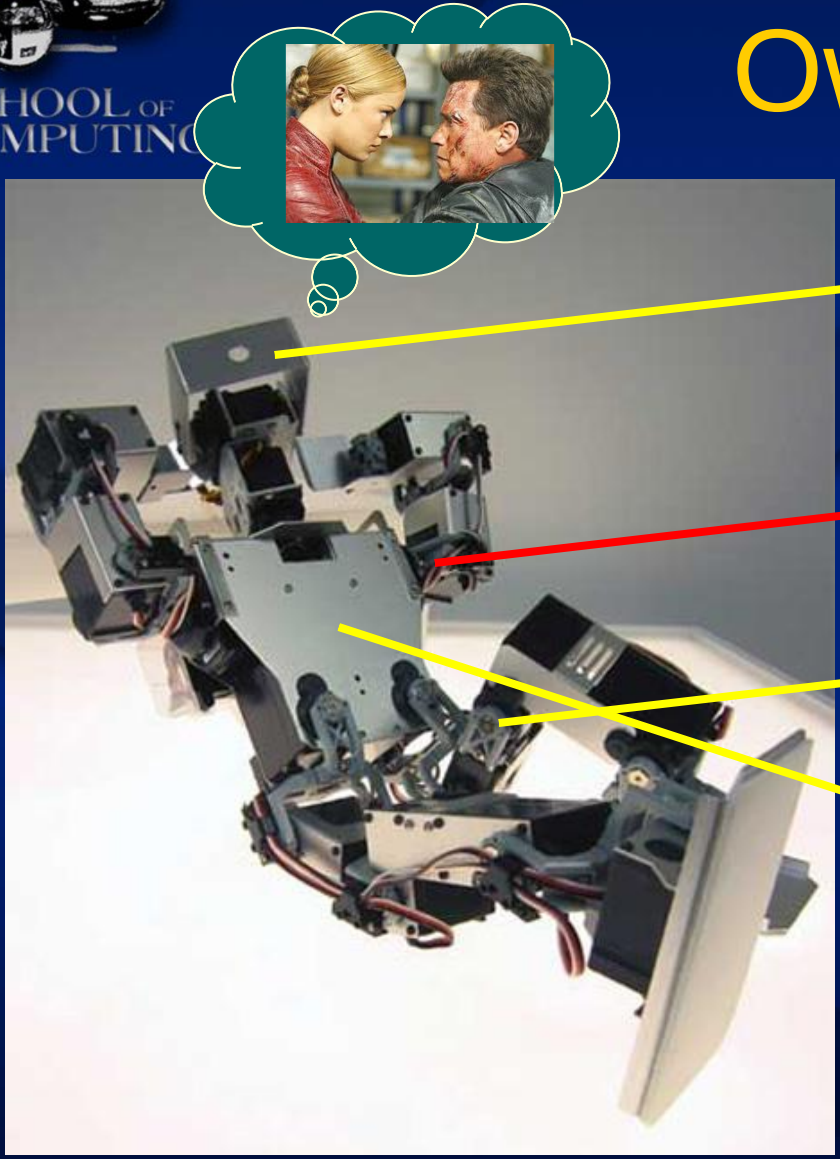
Major points:

- **Built-in** symmetry analysis
- **Symbolic** vocabulary
- **Hierarchical** structure

Phases of Cognition

- Phase I: Self-Discovery
  - Aka: Sensorimotor Reconstruction
- Phase II: Learning from Others
- Phase III: Interaction and Long-Term Cognition

Robot Wakes up & Learns Own Structure



Camera is just a collection of 1D pixel streams

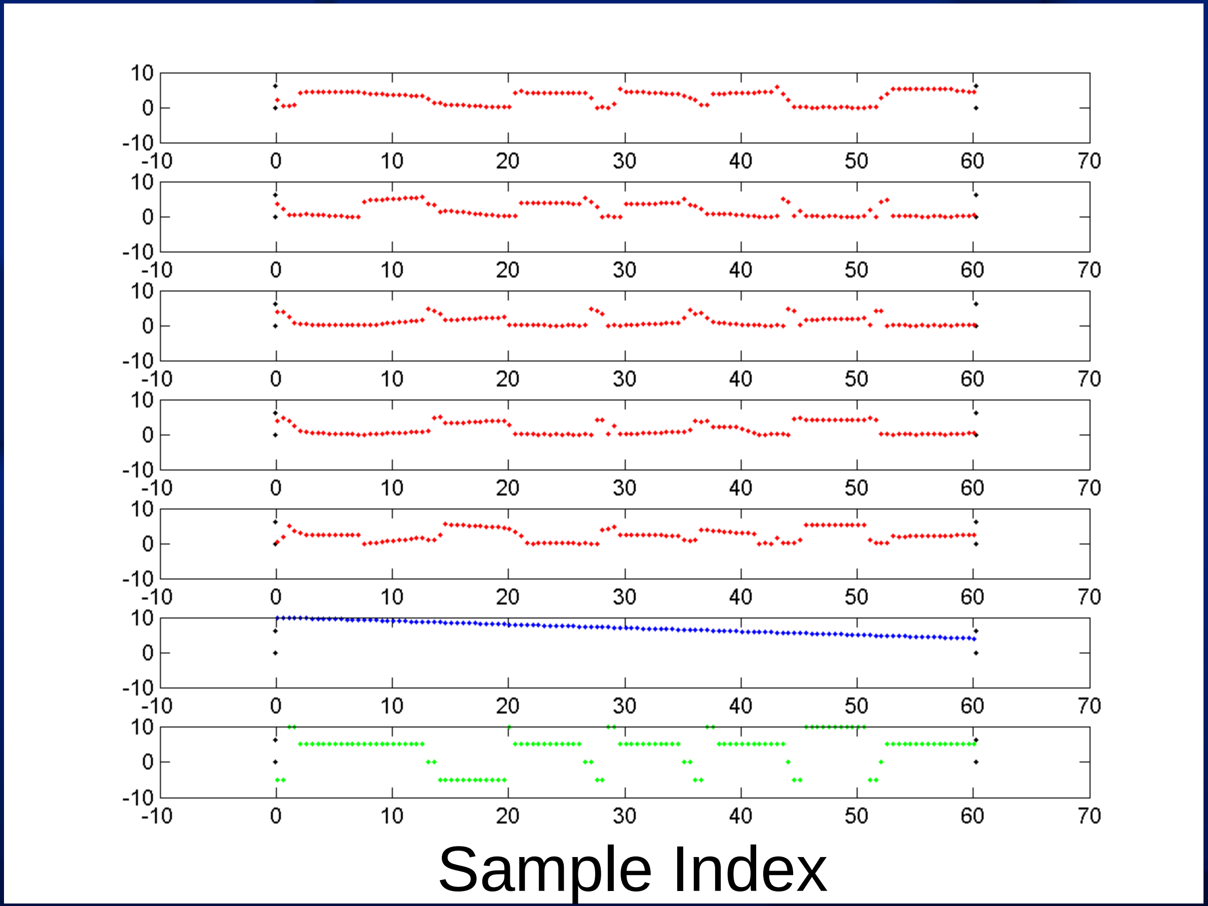
Actuators receive Command streams

Joint sensors deliver 1D streams

Power level is data stream

Goal: Identify similar sensors/ actuators and their relations

Robot Data Streams

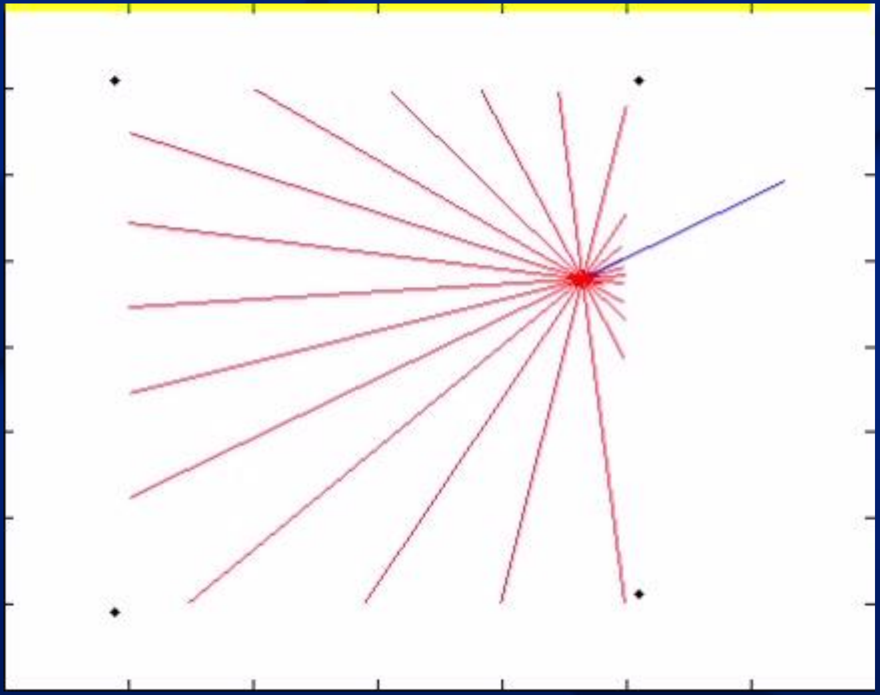


Range Sensors 1,5,10,15,20

Energy Sensor

Direction Sensor

Pierce, Kuipers Scenario



Robot wanders randomly for 5 minutes:

- 4x6 rectangular area
- 20 range sensors
- NSEW direction sensors
- Energy sensor

SCHOOL OF COMPUTING

SYMMETRY ANALYSIS IN SENSORIMOTOR RECONSTRUCTION

Our Approach: Exploit Symmetry

Similarity is basis of symmetry; here, we determine if 1D signals (samples y) are:

- Constant : exactly the same $y(t) = c$
  - Any point maps to any other
- Periodic : $y(t') = y(t)$ where $t' = [1\ T]\begin{bmatrix} t \\ 1 \end{bmatrix}$
  - No point maps to itself

Exploit Symmetry (cont'd)

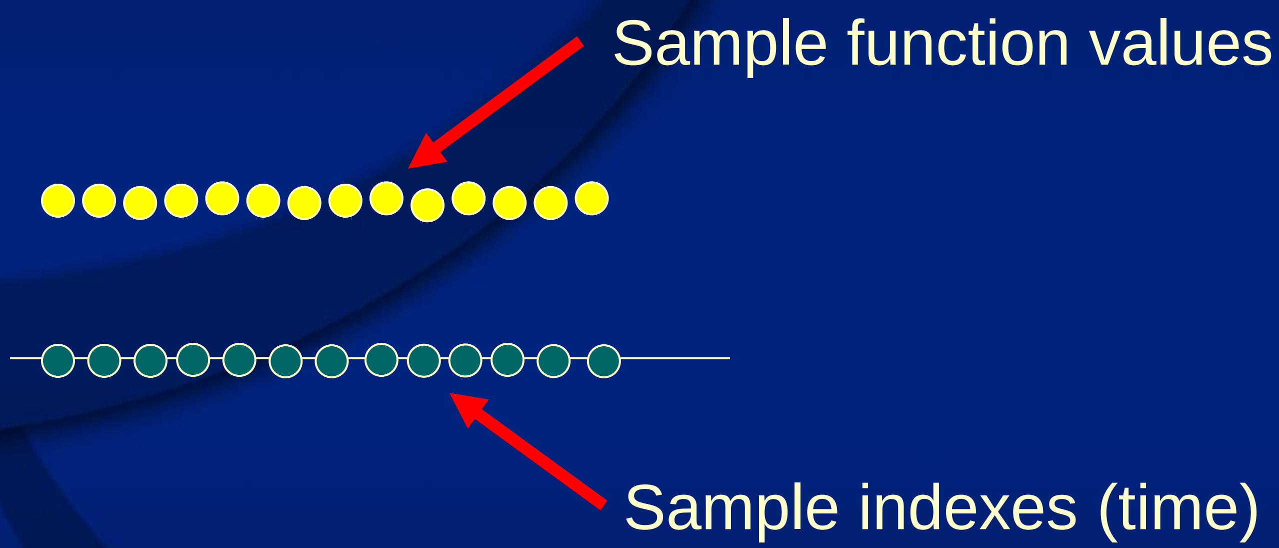
- Reflection : reflection point origin: y(t) = y(-t)
  - Only one point maps to itself
- Linear : y = at + b = [a b]$\begin{bmatrix} t \\ 1 \end{bmatrix}$
  - Take the derivative and find constant signa;
- Gaussian : ~N(μ, σ^2) symmetry of variance or histogram or autocorrelation

THE UNIVERSITY OF UTAH

Symmetries (1D)

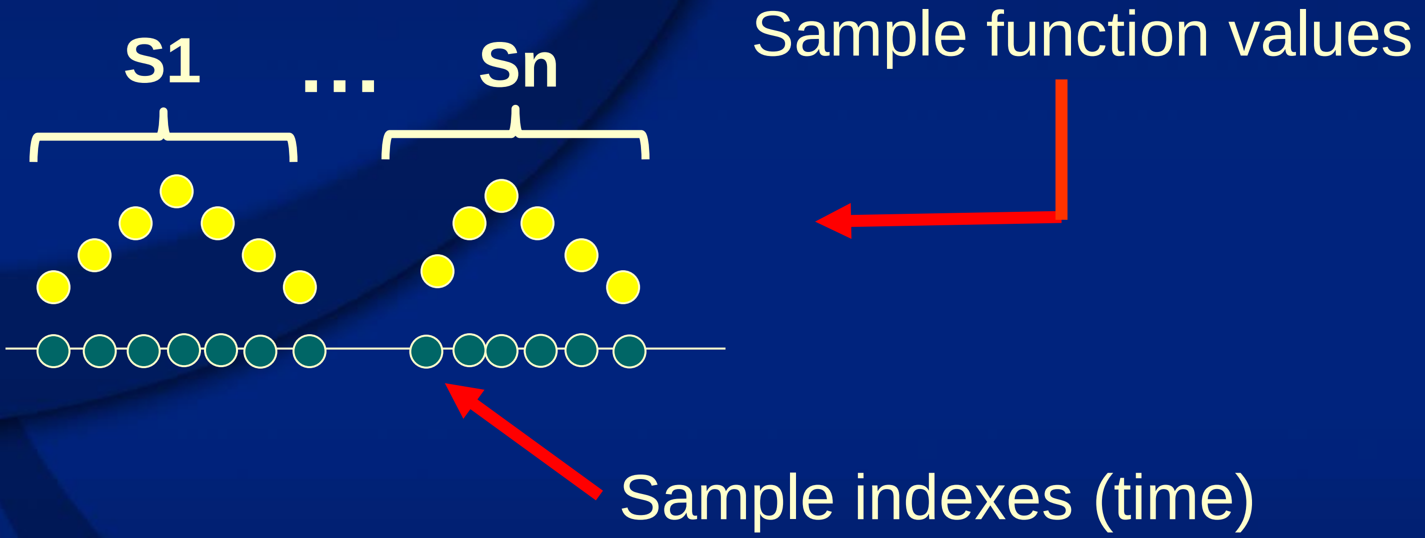
- Constant: **equality** symmetry
- Periodic: (discrete) **translation** symmetry
- Reflection: **reflection** symmetry
- Linear: (continuous) **translation** symmetry
- Gaussian: **reflexive** symmetry on histogram (and on autocorrelation of samples)

Permutations as Symmetries



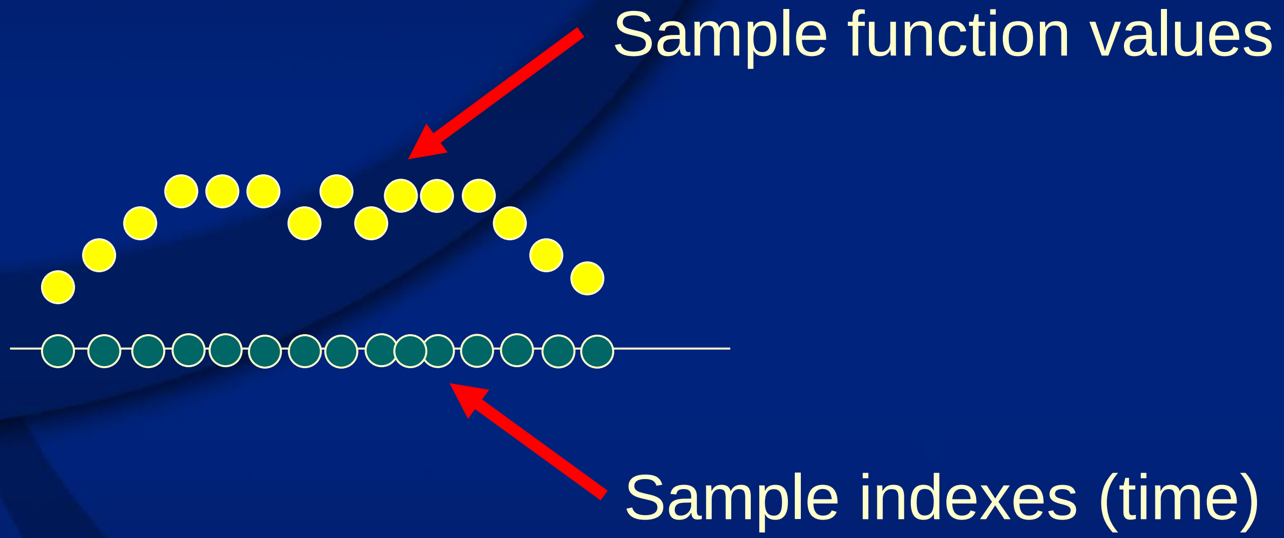
If any permutation results in the same function, then this is a **constant** signal

Permutations as Symmetries



If all permutations of Si are the same function, then this is a **periodic** signal

Permutations as Symmetries

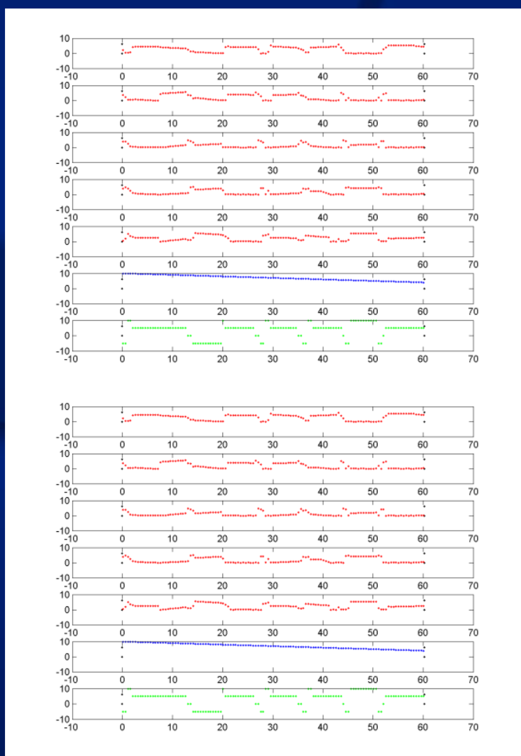


If (1 n)(2 n-1)…((n-1)/2 (n+3)/2) permutation, is the same function, then this is a **reflective** signal

Permutations as Symmetries

- Each of these permutation sets forms a subgroup of the full permutation group on n objects (called Sn)

Symmetries



Gaussian

Linear

Null

Gaussian

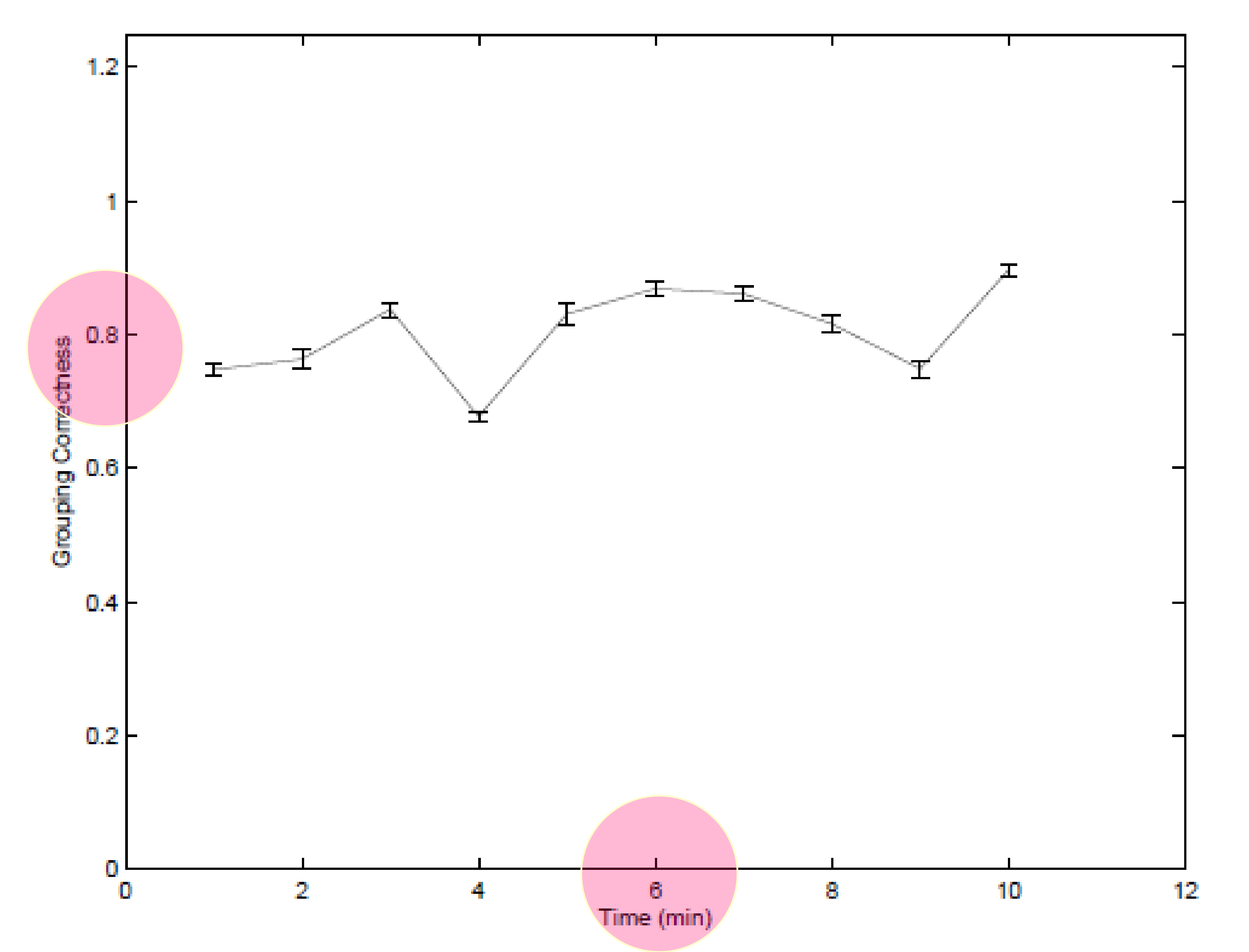
Linear

Null

Grouped by Variance

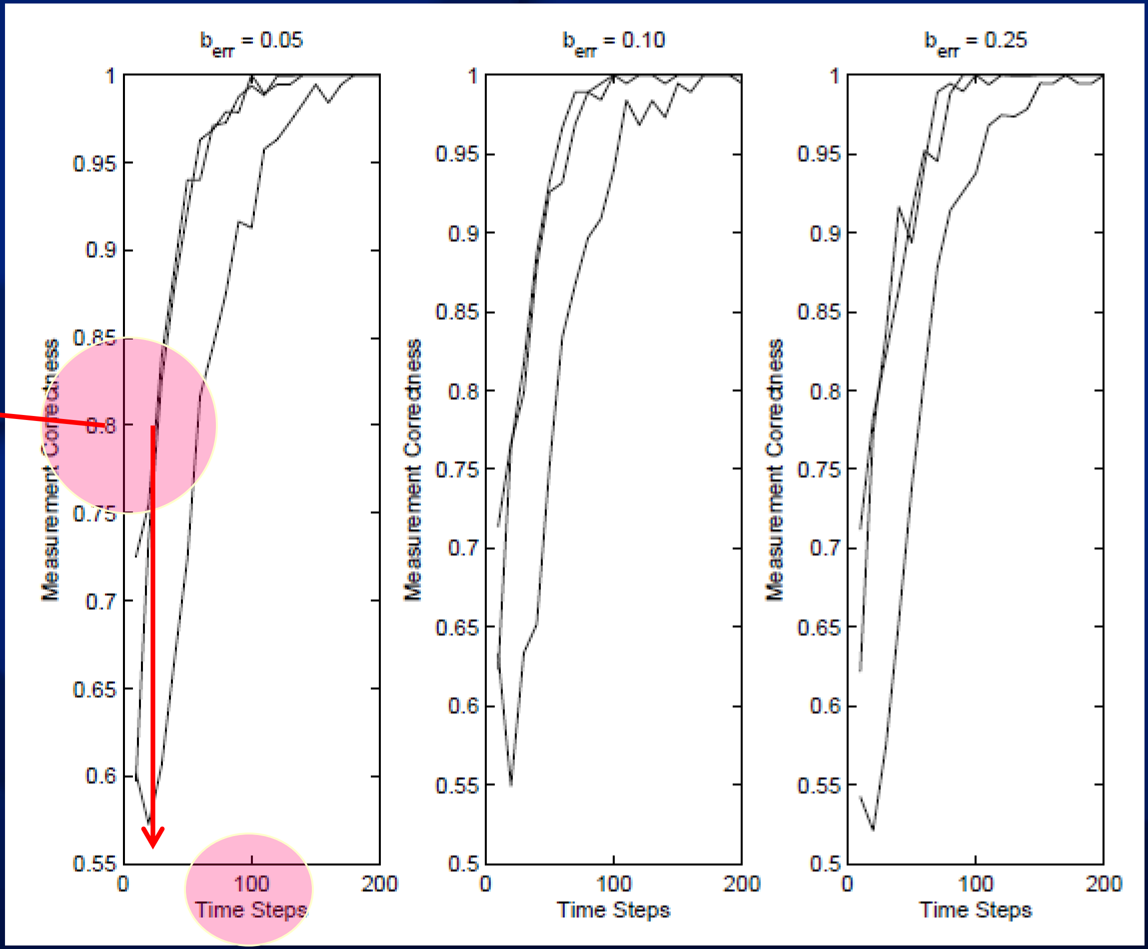
Grouped by Parameters

Pierce Results



Symmetry Based Results

Pierce's grouping success value



Classifies signals as similar, then clusters based on variance (range), etc.

Do Real Pixels Work This Way?

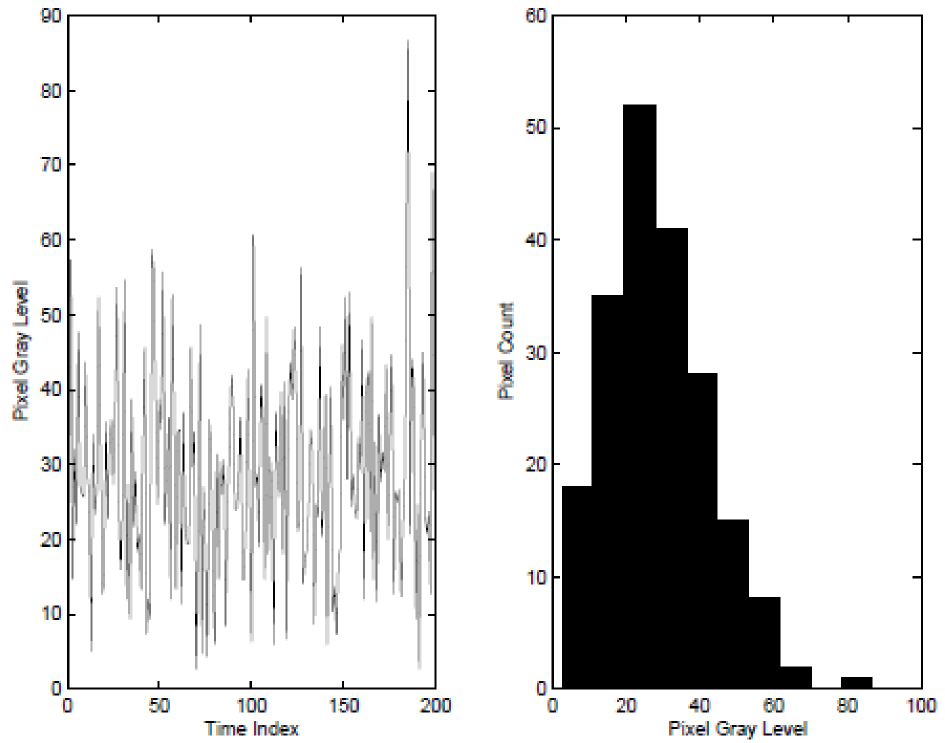


Figure 5: Trace and Histogram of the 200 Pixel Values of the Center Pixel of the Images.

Does Microphone Data Work This Way?

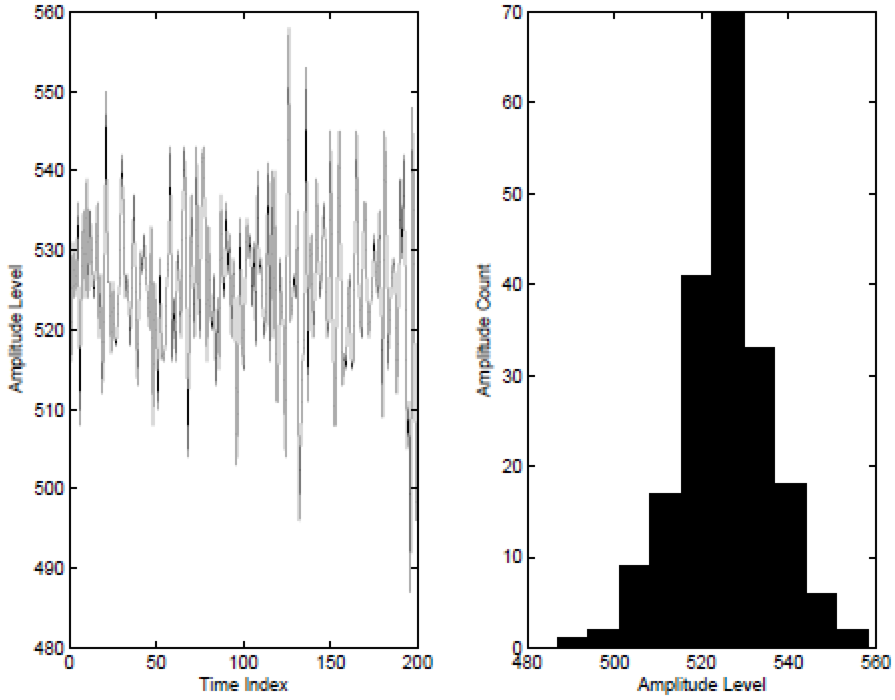


Figure 6: Trace and Histogram of the 200 Amplitude Values of the Microphone Data.

SCHOOL OF COMPUTING

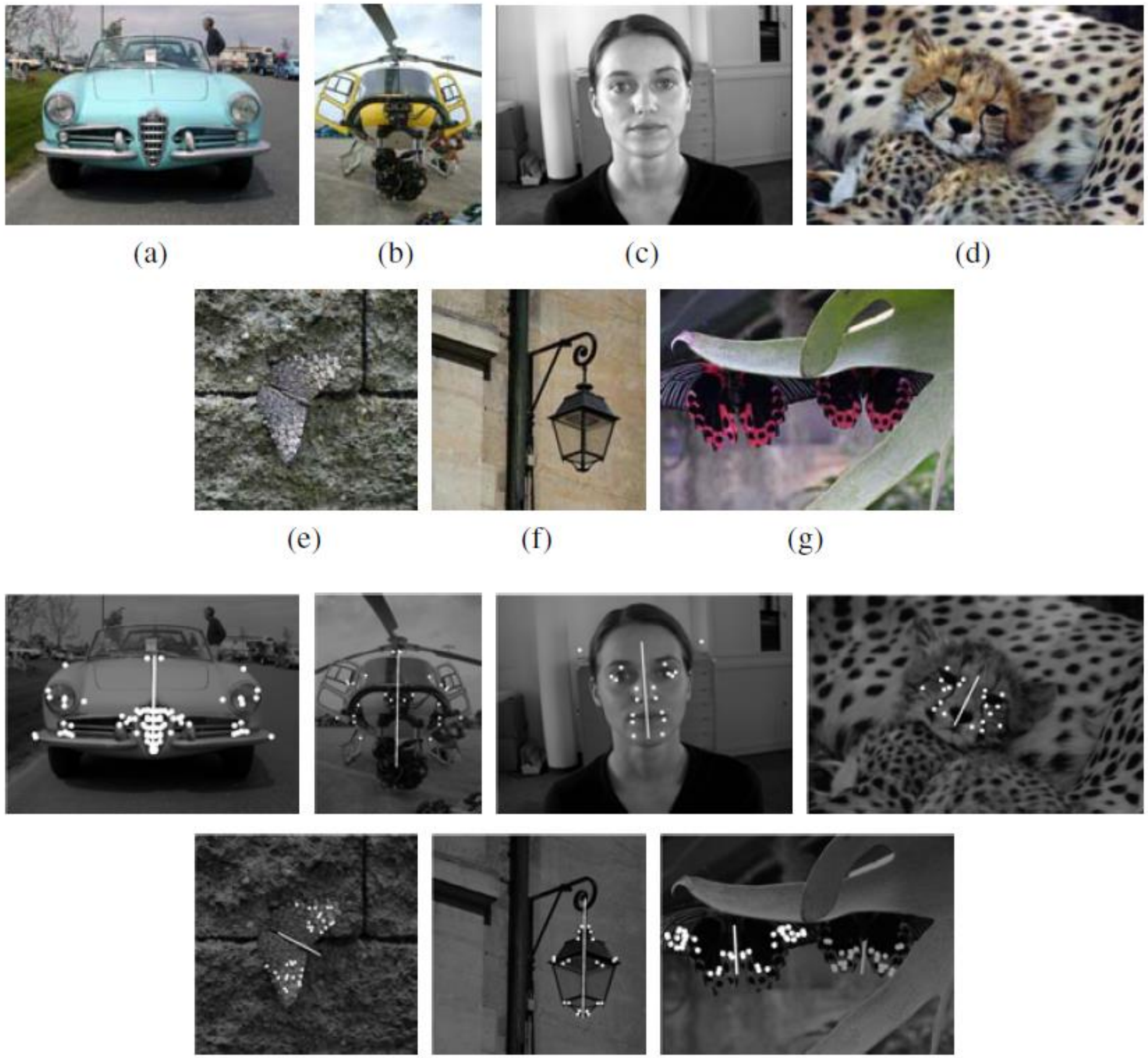
SYMMETRY DETECTION IN COMPUTER VISION (2D)

THE UNIVERSITY OF UTAH

Berner et al. 2009

Figure 1.1: *A 3D scan of a deformed plasticine sculpture with three snail figures. Our algorithm detects symmetries by looking at constellations of surface feature lines. This can be used for detecting deformable symmetries. Left: input data, middle: feature lines overlay, right: detected symmetries.*

Loy and Eklundh, 2006



Original photographs (a) Sandro Menzel, (b) David Martin, (c) BioID face database, (d) Stuart Maxwell, (e) elfintech, (f) Leo Reynolds, (g) ze1, distributed under the Creative Commons Attribution-Non-Commercial-Share-Alike Licence, http://creativecommons.org.

Solomon, 2010

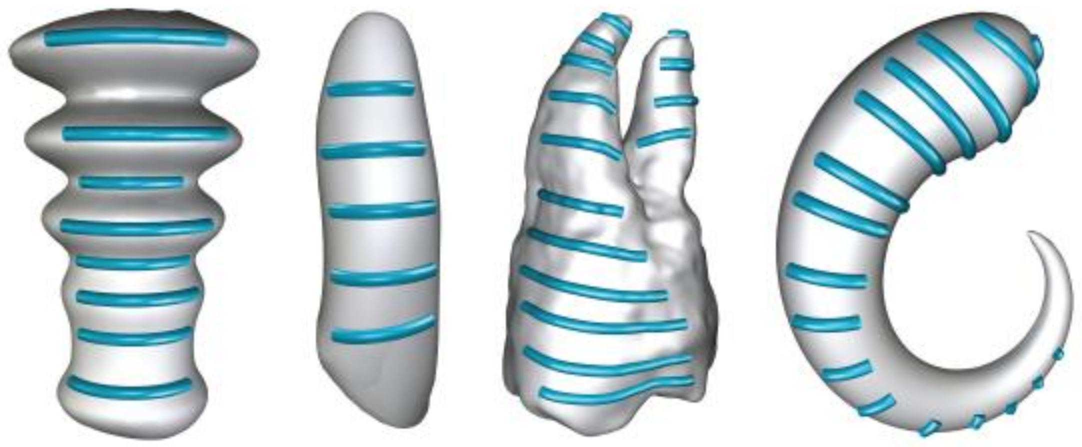


Figure 16: Flows of approximate Killing vector fields on some example surfaces.

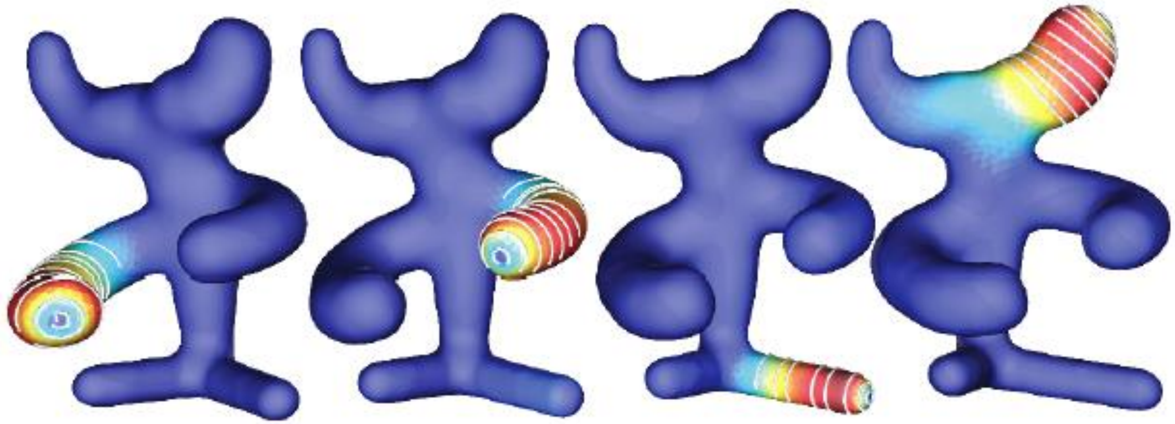


Figure 17: Vector fields corresponding to the four lowest eigenvalues of R; colors indicate the norm of the approximate Killing field displayed.

Liu et al., 2007

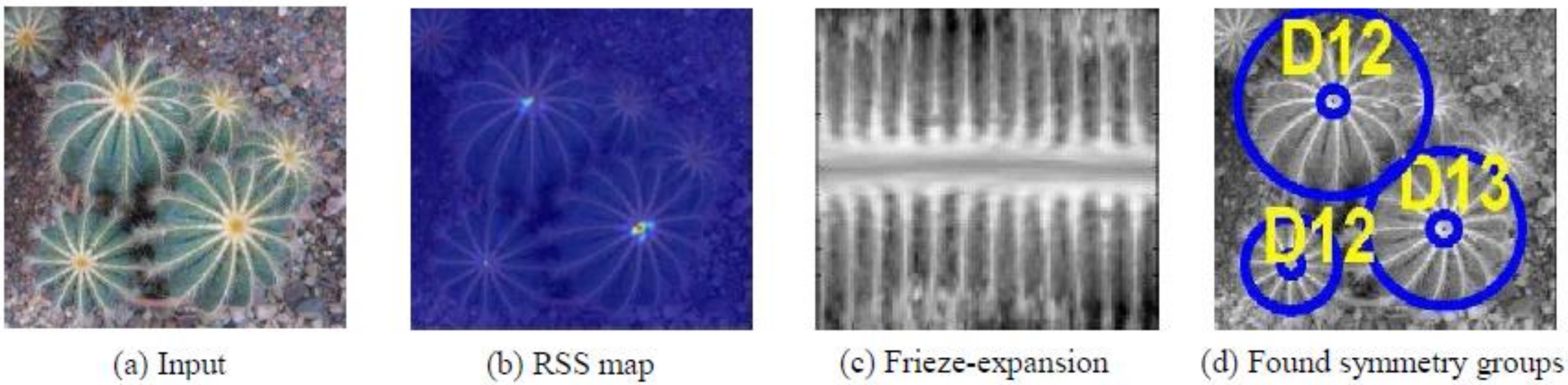


Figure 2: (a) Test image with multiple rotational centers and symmetry types. (b) Rotation Symmetry Strength (RSS) map overlaid on the original image (c) One sample of Frieze-expansion (d) Rotation symmetry group detection result

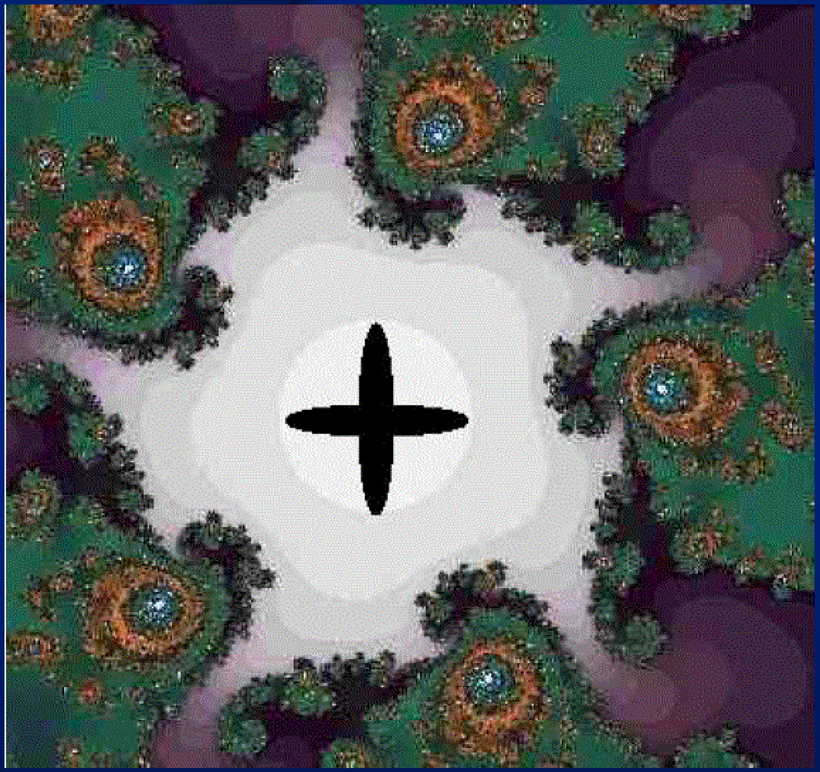
Desired 2D Symmetries

C1: identity map
Cn: rotation symmetry
D1: Single axis bilateral symmetry
Dn: rotation and reflective symmetry
O(2): continuous 2D rotational symmetry

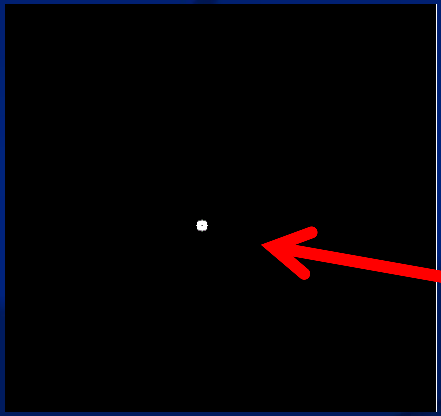
Lee-Liu Symmetry Detection

Example Image:

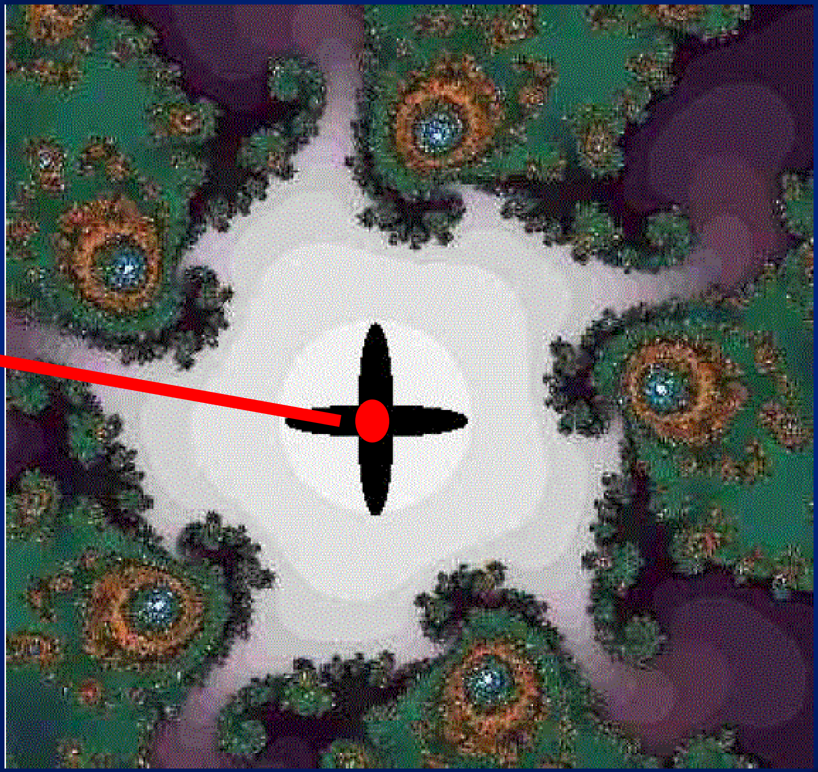
Several embedded symmetries



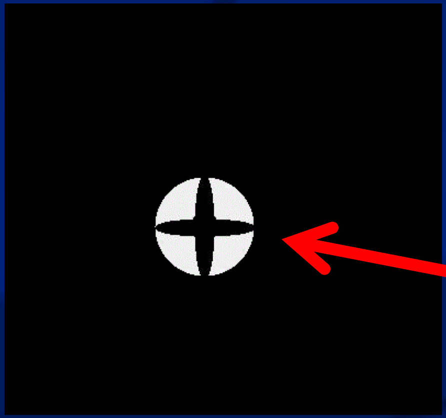
Symmetries



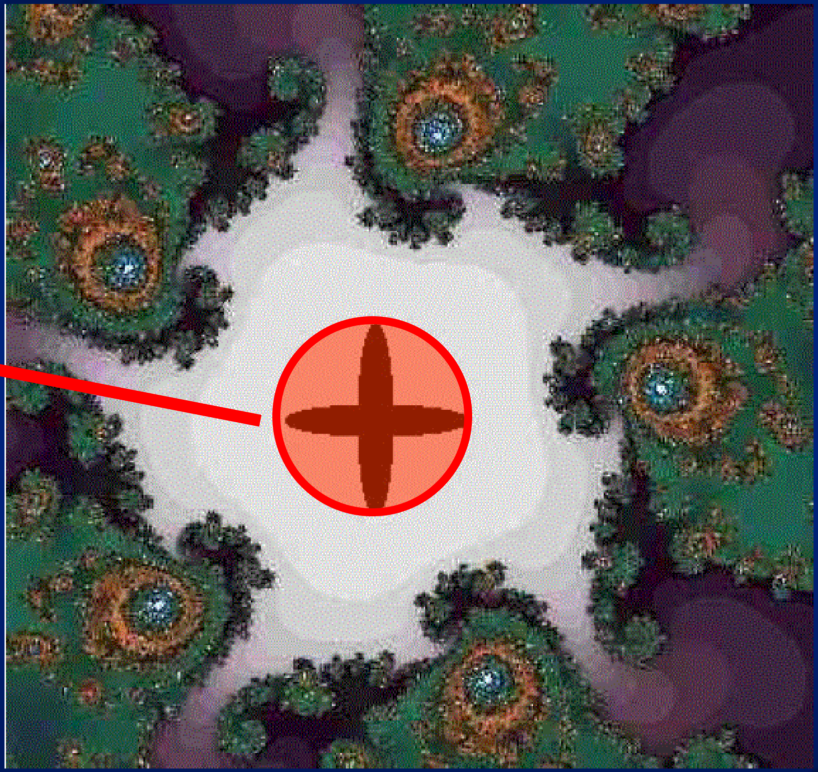
O(2)



Symmetries



D_4



Symmetries

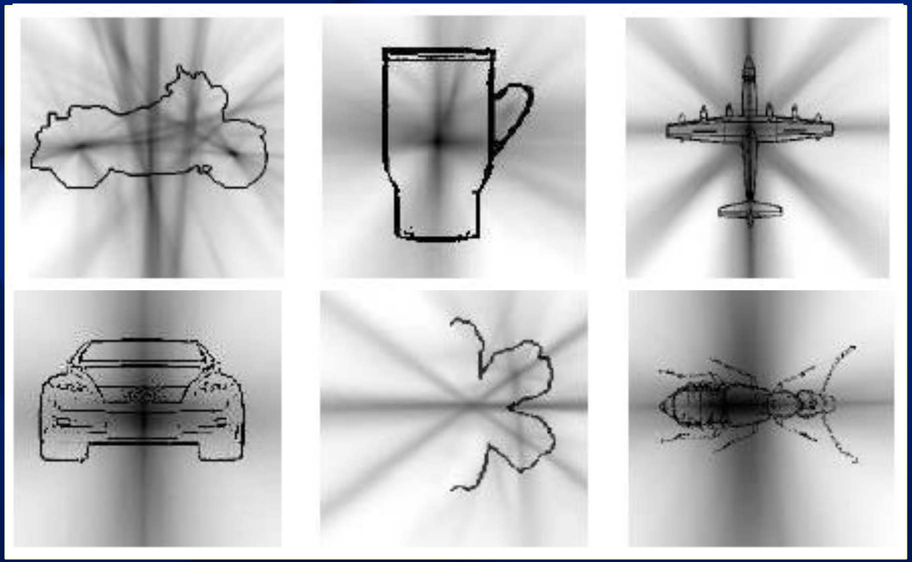
O(2)

Symmetries

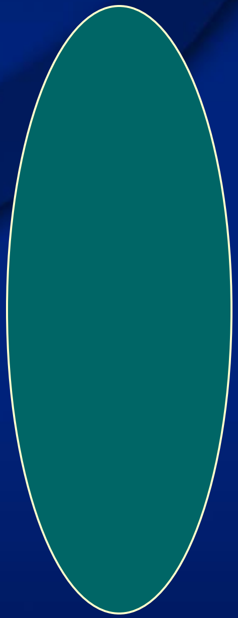
C_5

Podolak et al.

Planar-Reflective Symmetry Transform

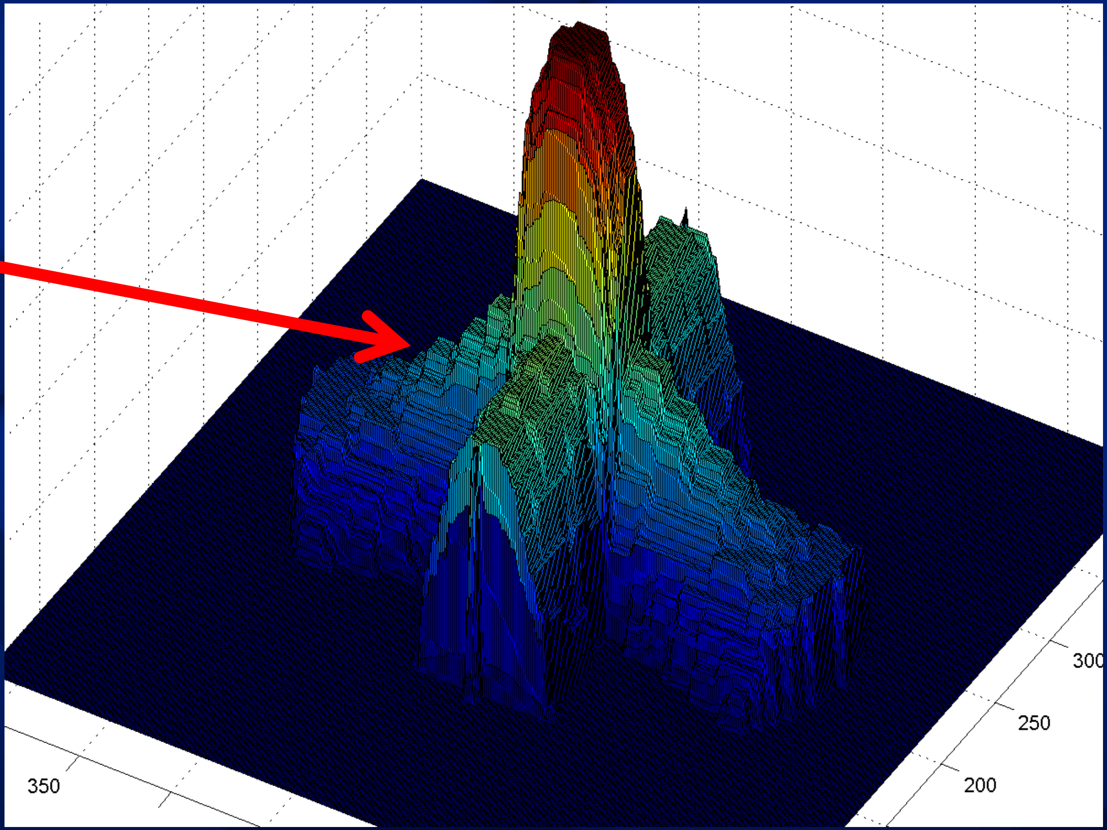


Ellipse Image



PRST for Ellipse Image

Note: Symmetry axes are found

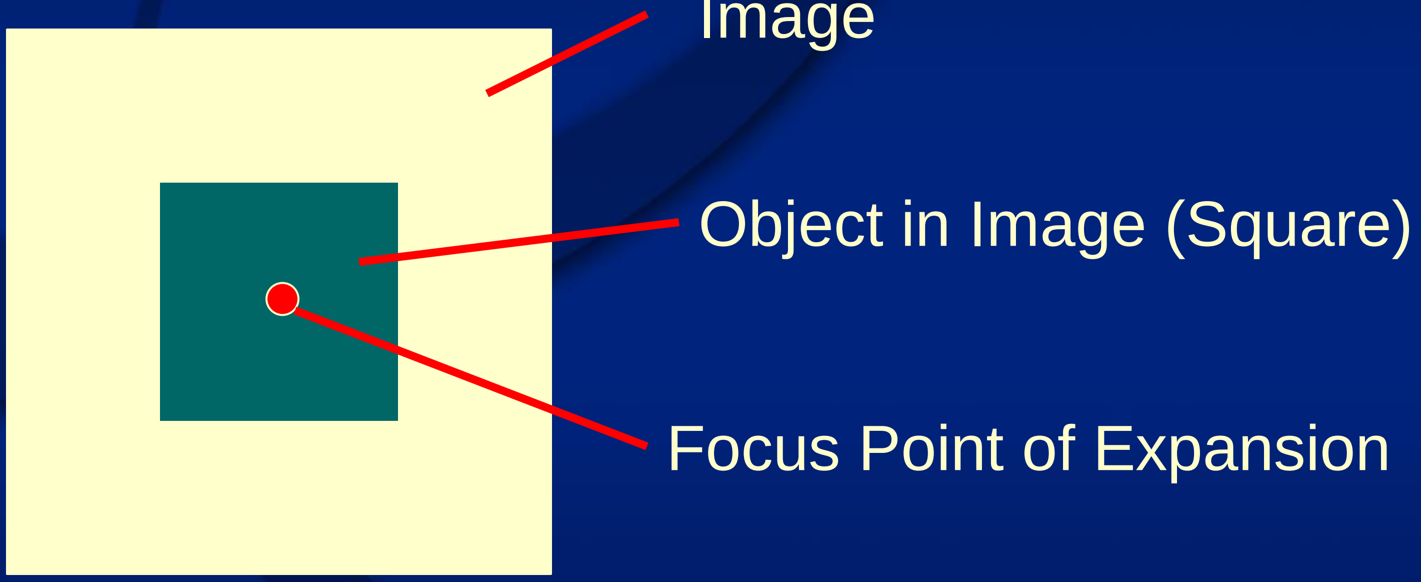


Major Issue

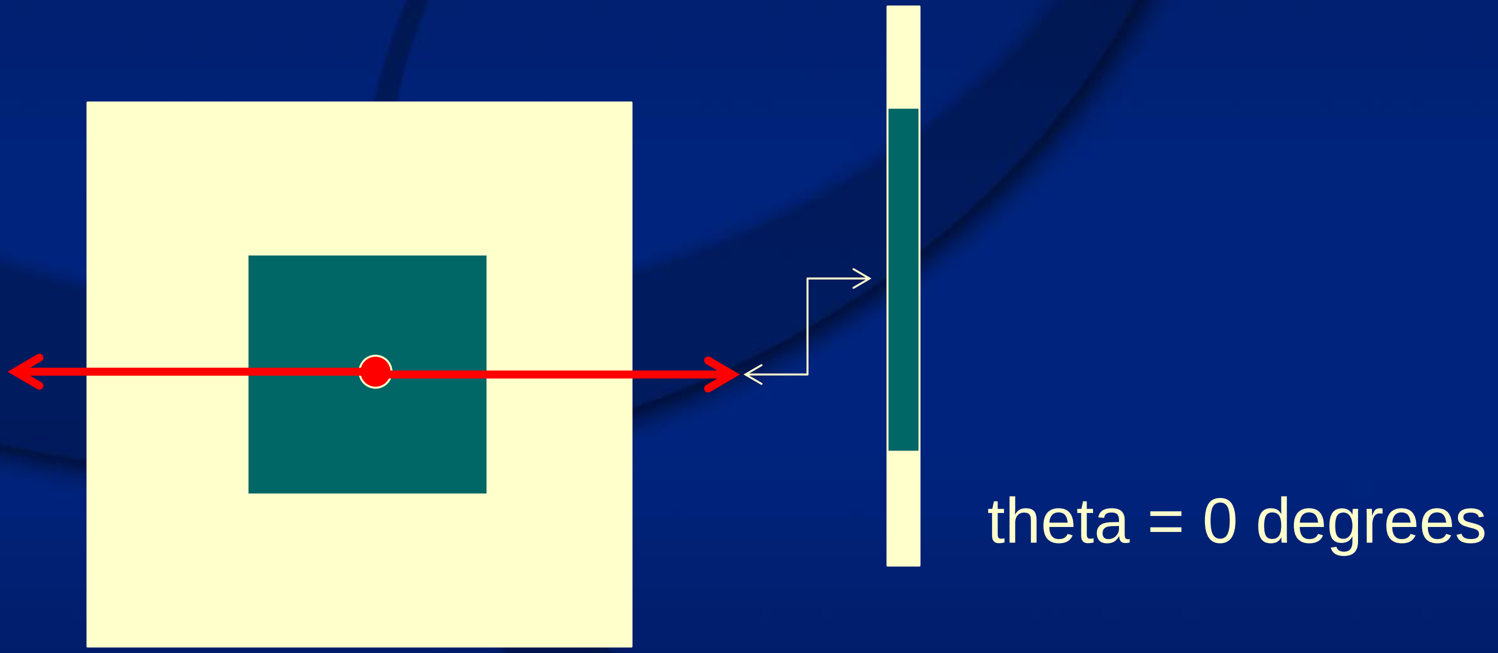
Too slow when applied to every pixel!

➔ So we developed an interest operator

Frieze Expansion Pattern

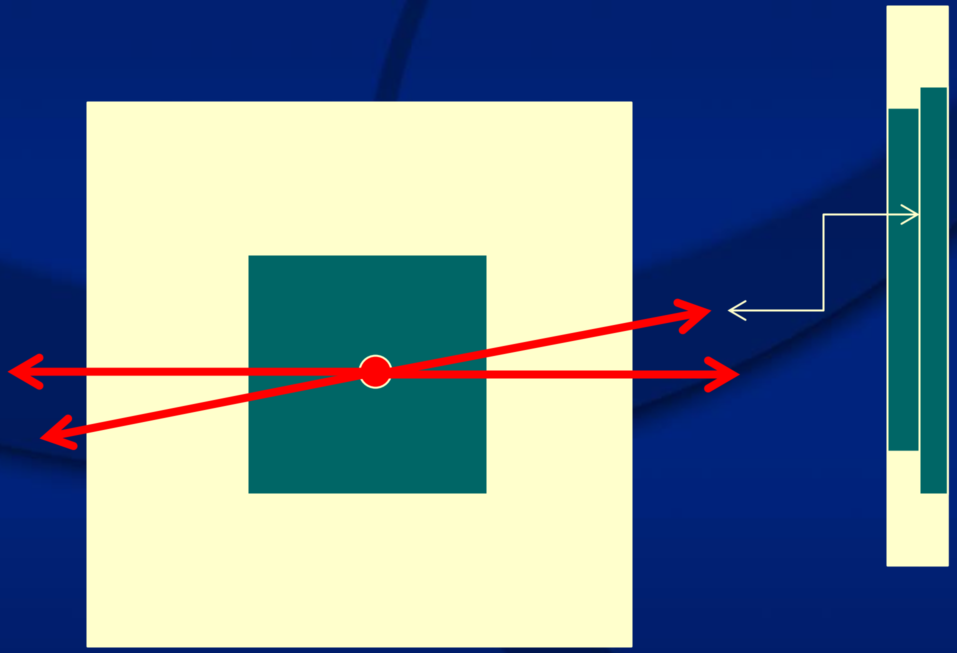


Frieze Expansion Pattern



For each line through the focus point, map the focus point to the middle row, one direction from the middle row up, the other from the middle row down.

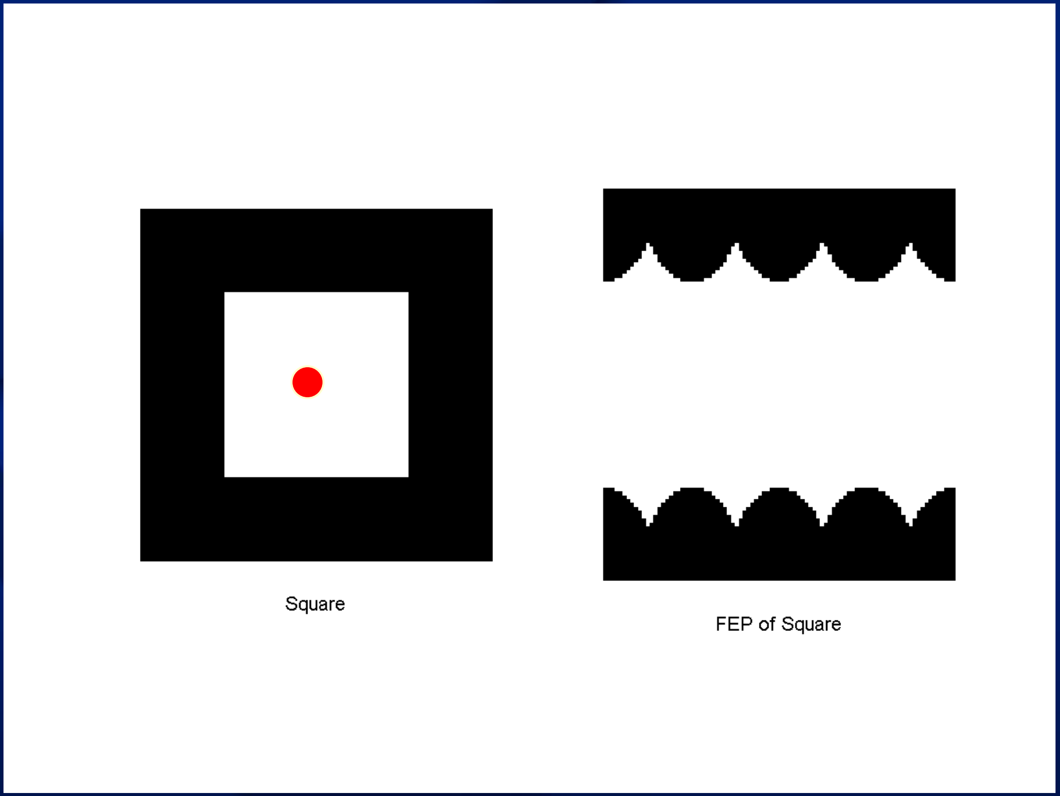
Frieze Expansion Pattern



theta = 10 degrees

For each line through the focus point, map the focus point to the middle row, one direction from the middle row up, the other from the middle row down.

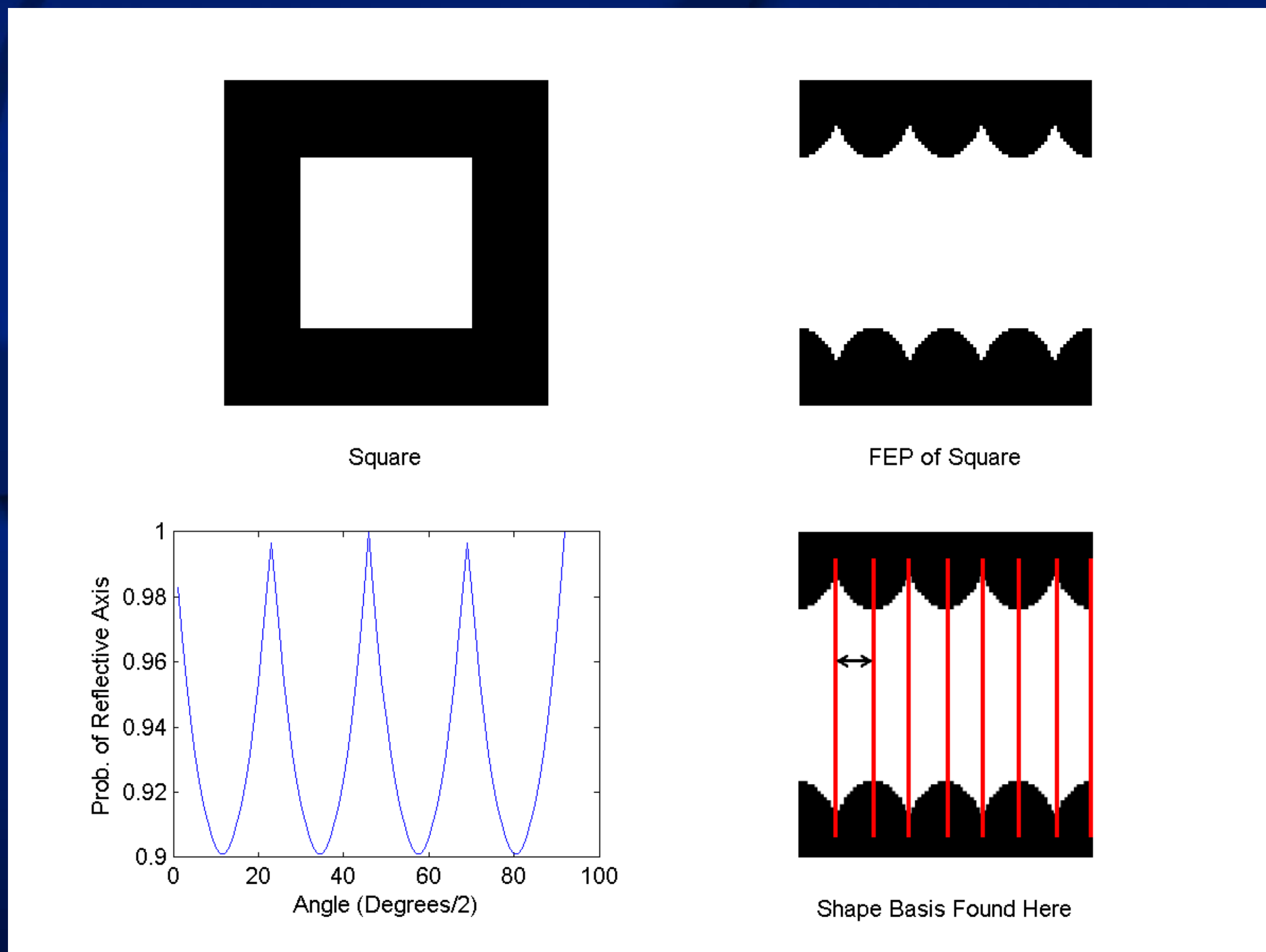
FEP for Square (in white)



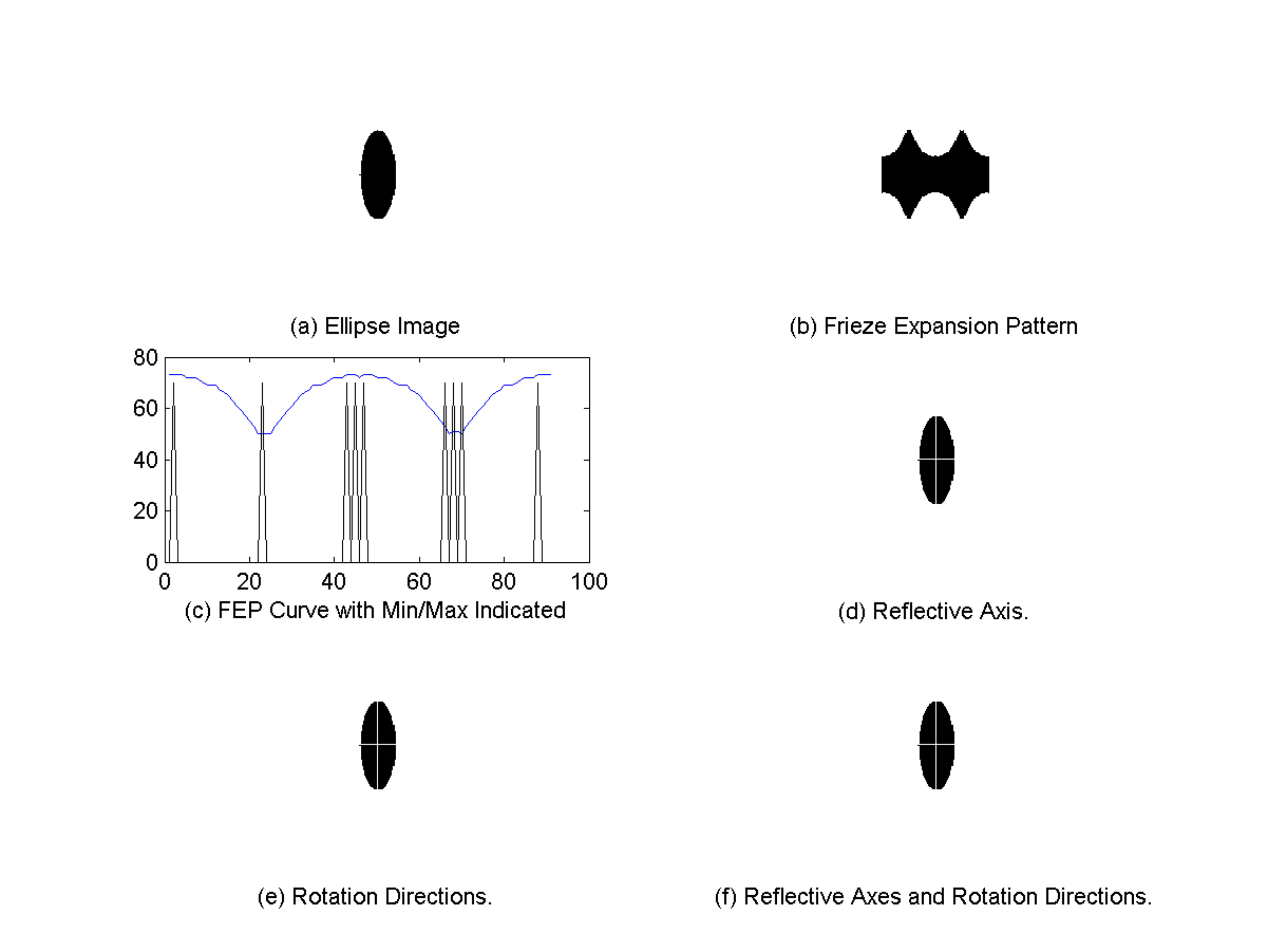
FEP and Symmetries

- Center of mass of shape is on reflective axis for 2D shape
- Rotation symmetries exist if:
  - FEP has reflective axis through middle row
  - Upper half of FEP has translational symmetry
- If reflective symmetries exist:
  - Must occur at max or min of upper half of FEP for object boundary
  - Shape has reflective symmetry about axis

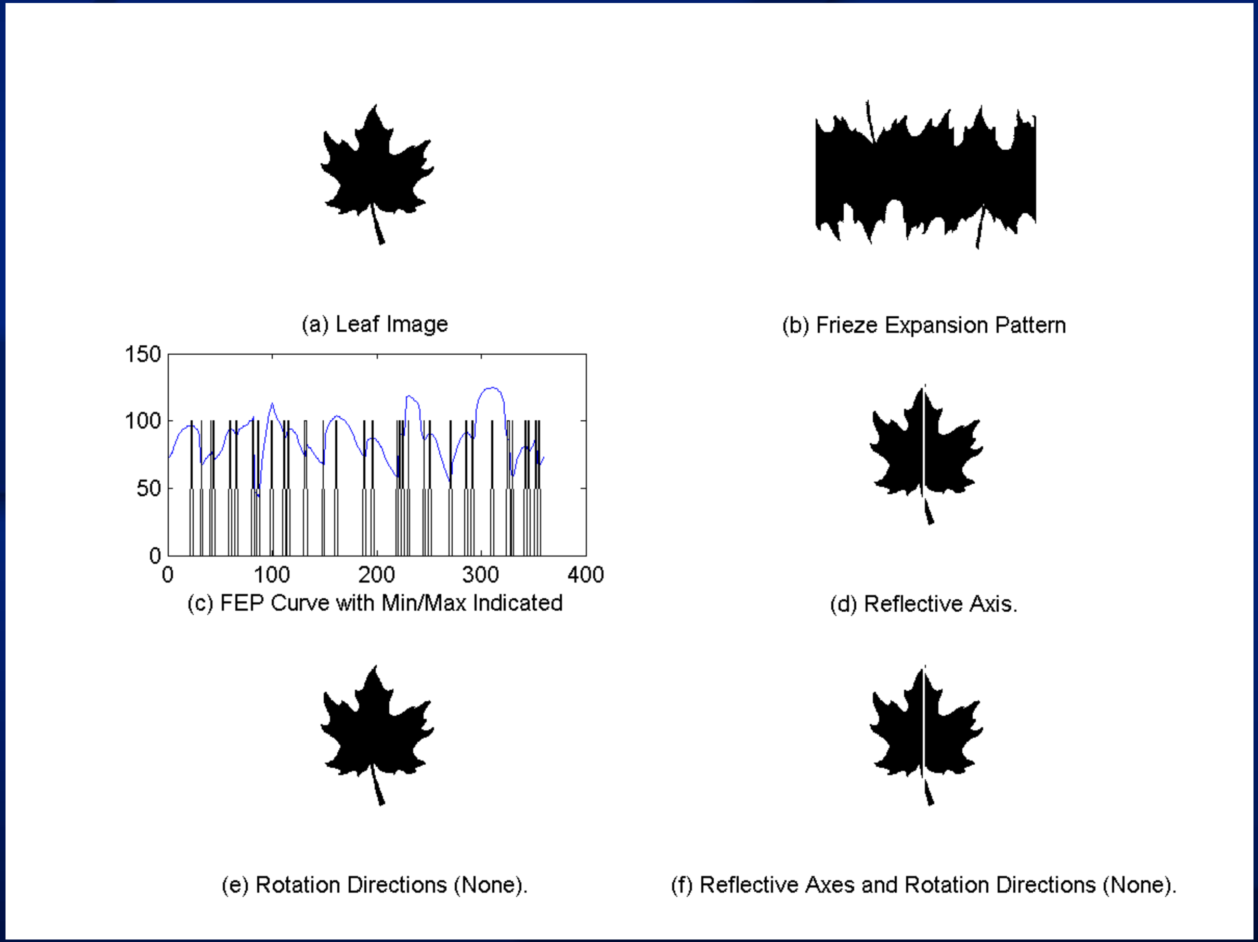
Possible Reflective Axes are Found at Max/Min's



Ellipse Symmetries



Another Example: Leaf



(a) Leaf Image

(b) Frieze Expansion Pattern

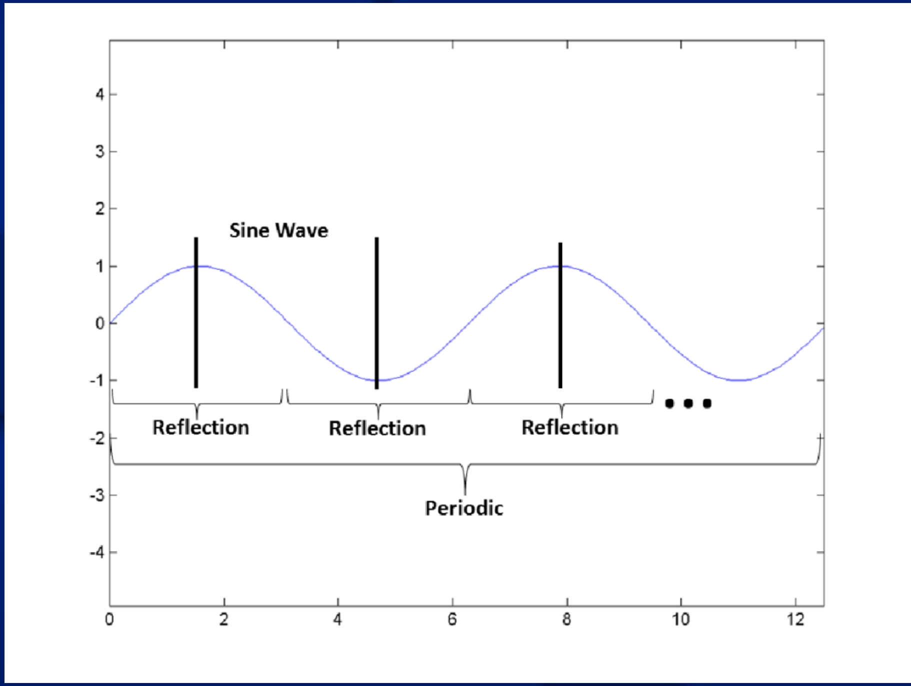
(c) FEP Curve with Min/Max Indicated

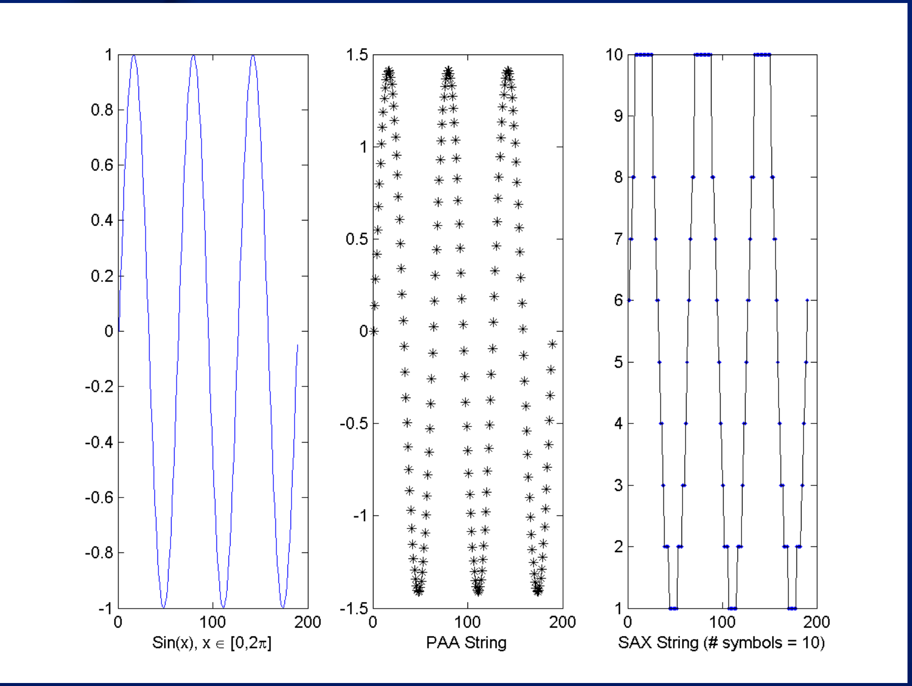
(d) Reflective Axis.

(e) Rotation Directions (None).

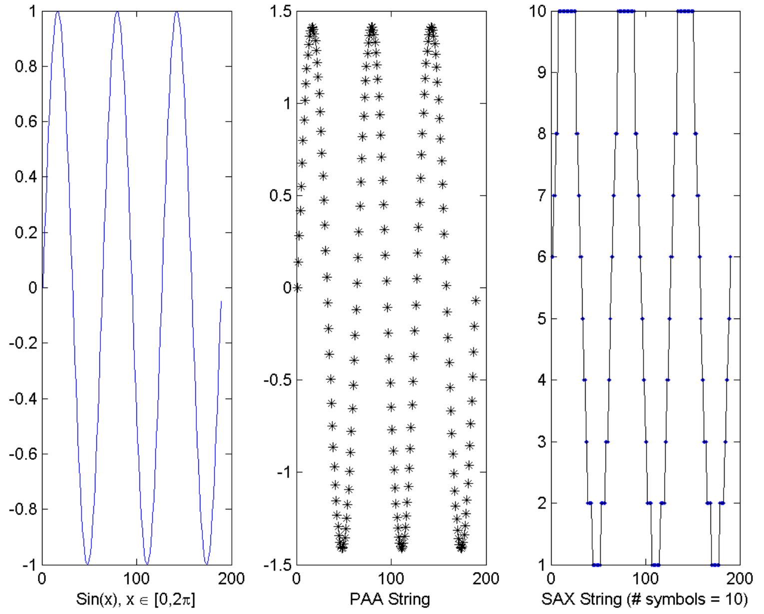
(f) Reflective Axes and Rotation Directions (None).

Symmetry Parsing



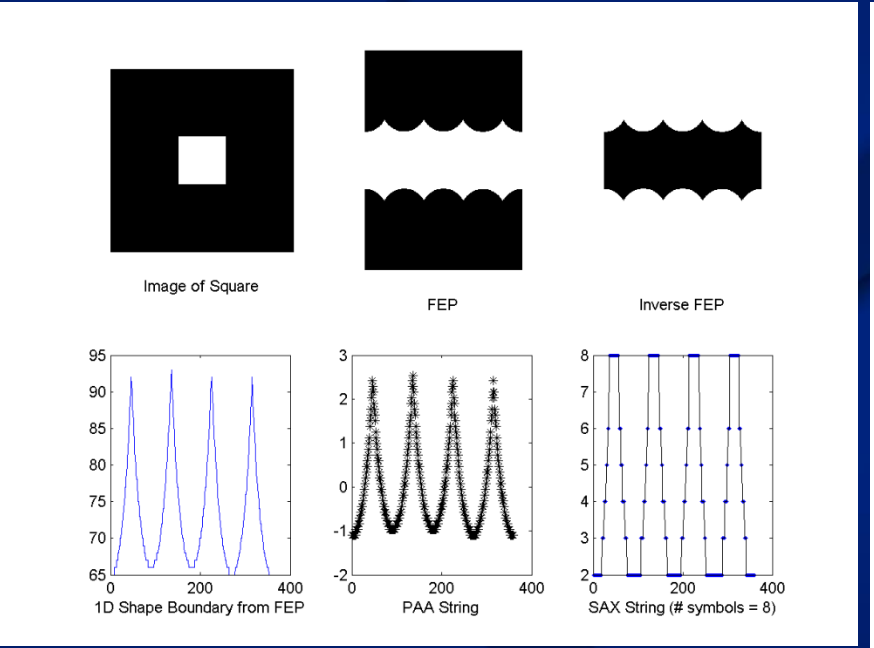


Symmetry Parsing



Symmetry Type	Start Index	End Index	Basic Length	Symmetry Measure	Symmetry Index
periodic	1	189	63	0.7663	
reflective	90	132	21	1.0000	111
reflective	27	69	21	0.9283	48
reflective	2	32	15	0.9239	17
reflective	159	189	15	0.9239	174
reflective	59	101	21	0.7334	80
reflective	121	163	21	0.7334	142

Symmetry Parsing



Symmetry Type	Start Index	End Index	Basic Length	Symmetry Measure	Symmetry Index
periodic	1	189	63	0.7663	
reflective	90	132	21	1.0000	111
reflective	27	69	21	0.9283	48
reflective	2	32	15	0.9239	17
reflective	159	189	15	0.9239	174
reflective	59	101	21	0.7334	80
reflective	121	163	21	0.7334	142

1D Symmetry Attribute Grammar

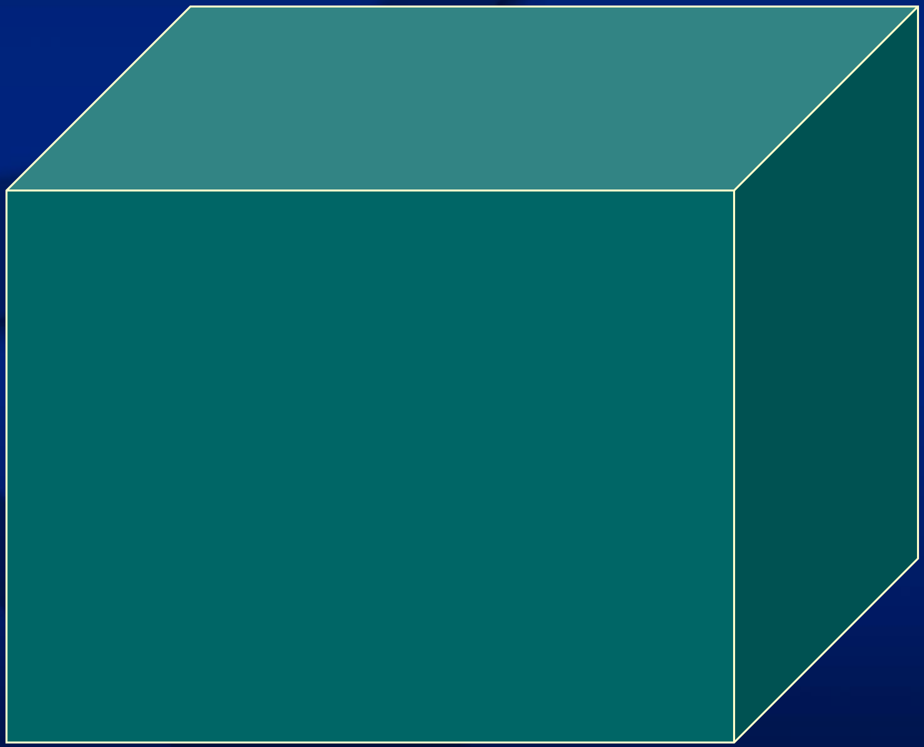
(1) $F \to S^1 S^2 \{\mu_1 == \mu_2\}$
(2) $U \to S^1 S^2 \{\mu_1 < \mu_2\}$
(3) $D \to S^1 S^2 \{\mu_1 > \mu_2\}$
(4) $C \to F$
(5) $C \to CF\{constant(C) == constant(F)\}$
(6) $B \to UD\{slope(U) \approx -slope(D)\}$
(7) $B \to DU\{slope(D) \approx -slope(U)\}$
(8) $W \to$ any permutation
(9) $P \to W^+W^+\{attributes(W^1 \approx attributes(W^2)\}$
(10) $Z \to C \mid B \mid P$
(11) $R \to UZD\{slope(U) \approx -slope(D)\}$
(12) $R \to DZU\{slope(D) \approx -slope(U)\}$
(13) $R \to FZF\{constant(F^1) \approx constant(F^2)\}$
(14) $R \to URD\{slope(U) \approx -slope(D)\}$
(15) $R \to DRU\{slope(D) \approx -slope(U)\}$
(16) $R \to FRF\{constant(F^1) \approx constant(F^2)\}$
(17) $S \to R \mid C \mid P$
(18) $A \to U$
(19) $A \to AU$
(20) $E \to D$
(21) $E \to ED$

Finding Symmetries in 3D using the 3D FEP

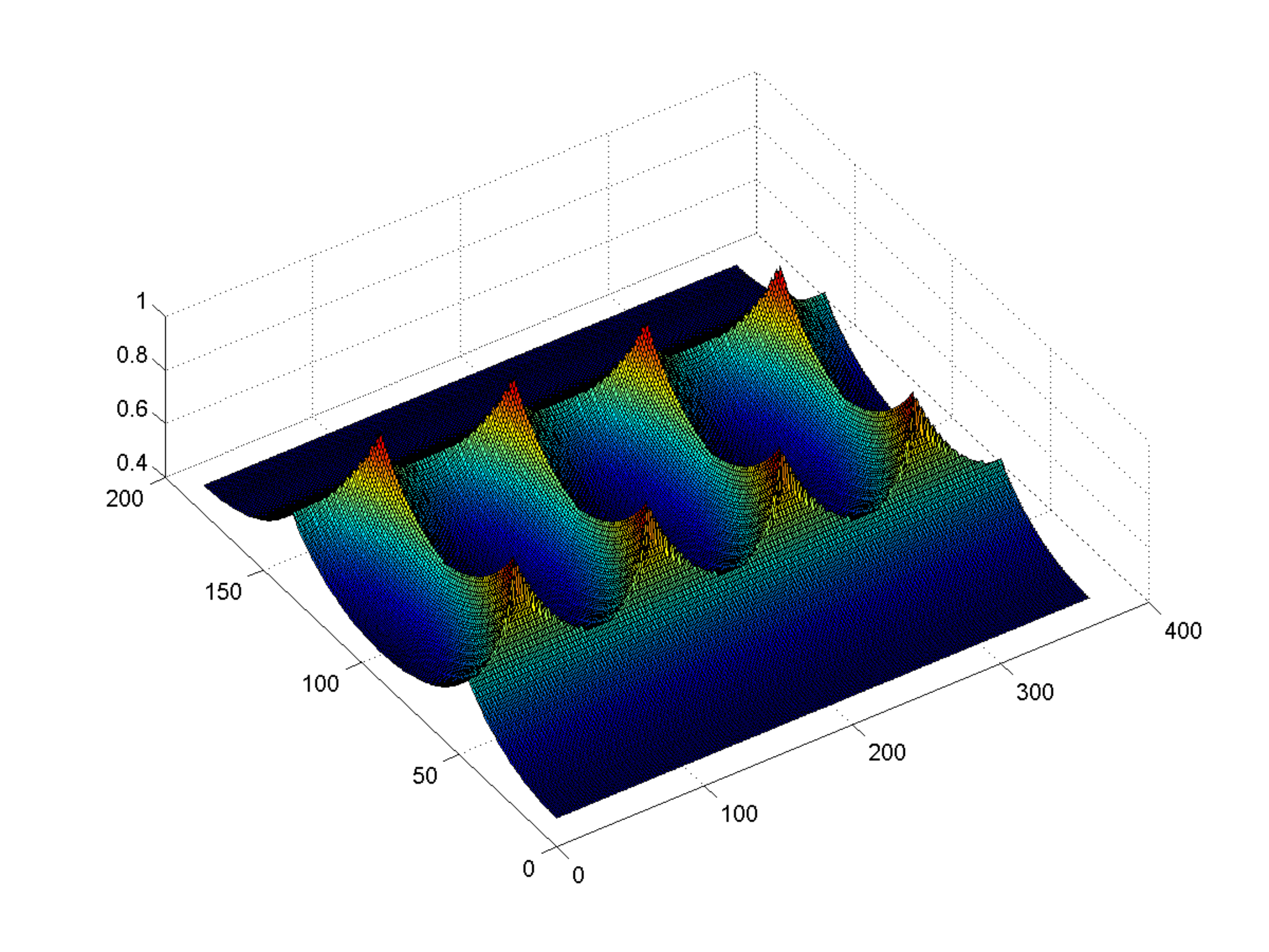
- Assume surfaces are known
  - E.g., Meshes from range data
  - Local density maxima

- Inside object at center of mass,
  - Rotate about x
    - Rotate about z

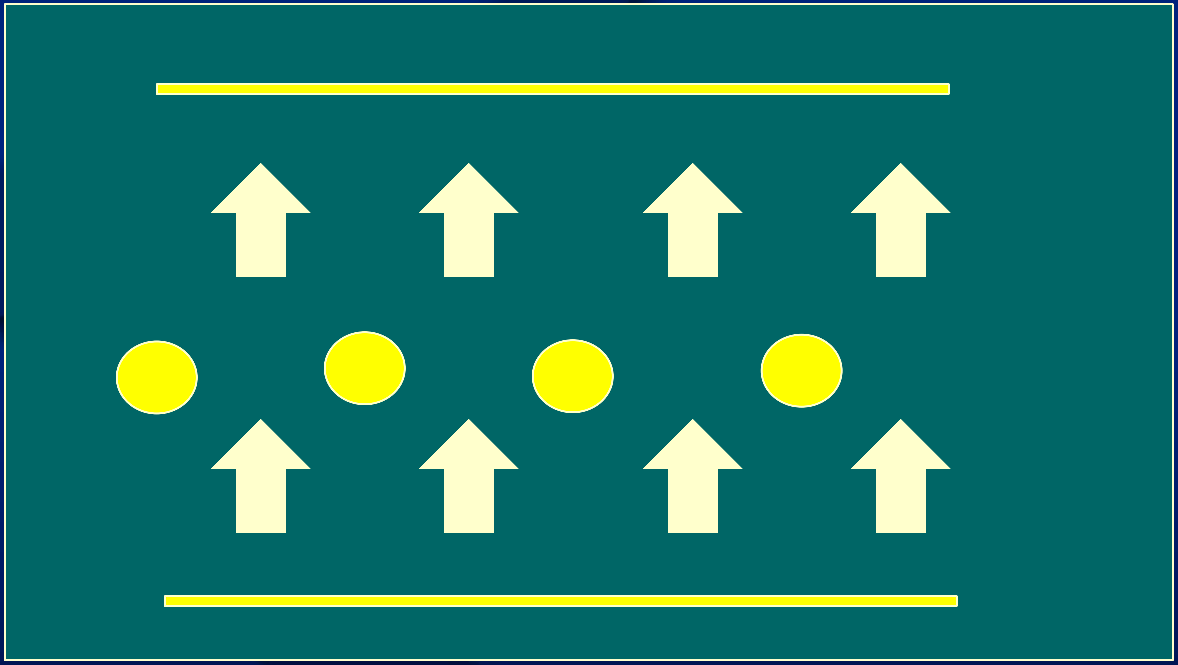
Example: The Cube



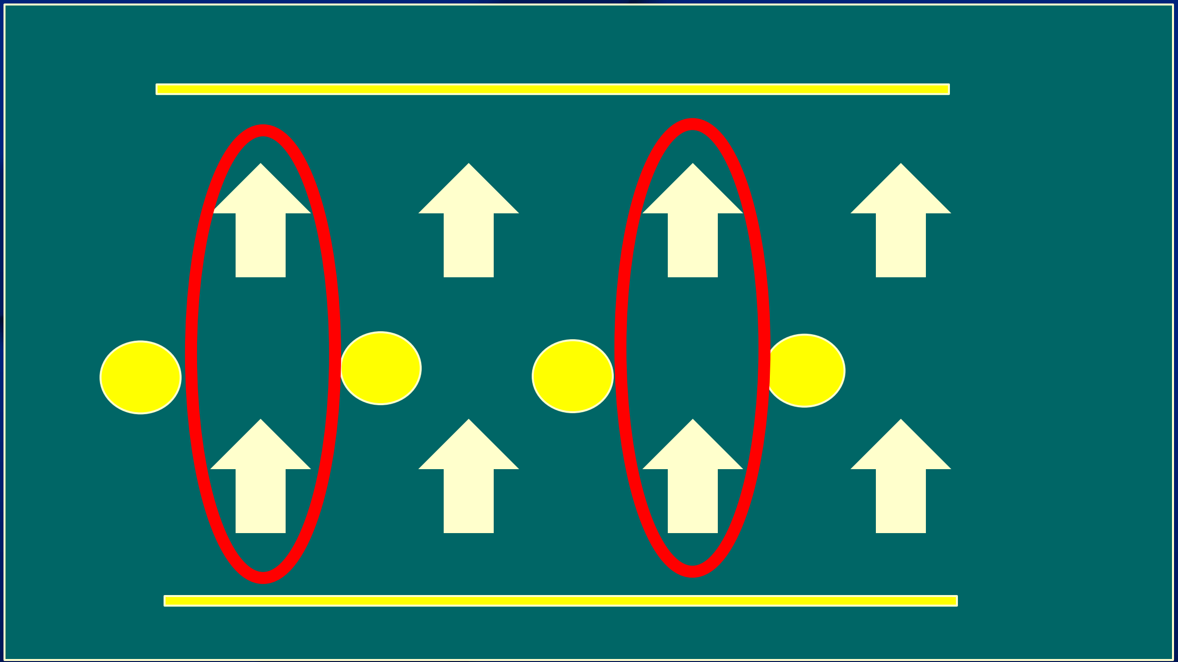
3D FEP Symmetry Detection



Peaks and Pits

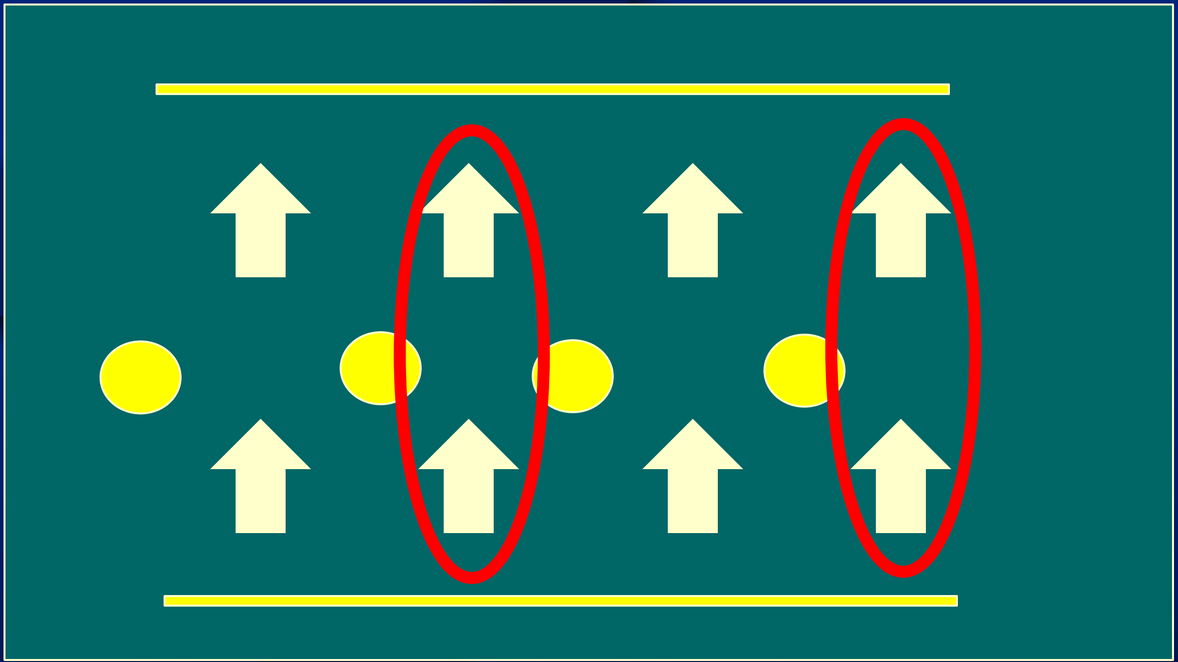


Peaks and Pits: Reflective Symmetry

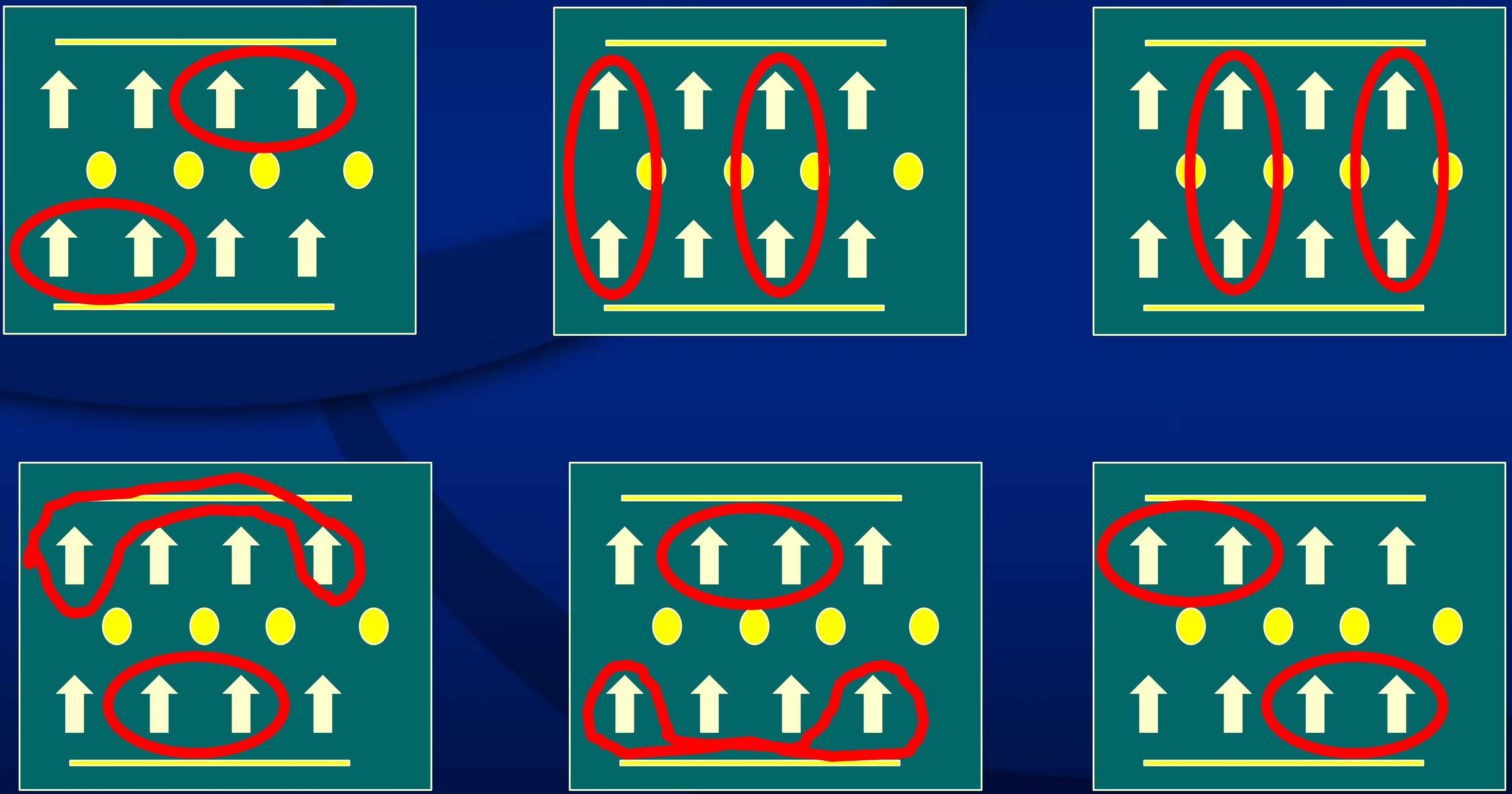


Pairs define a line and the two lines define a plane

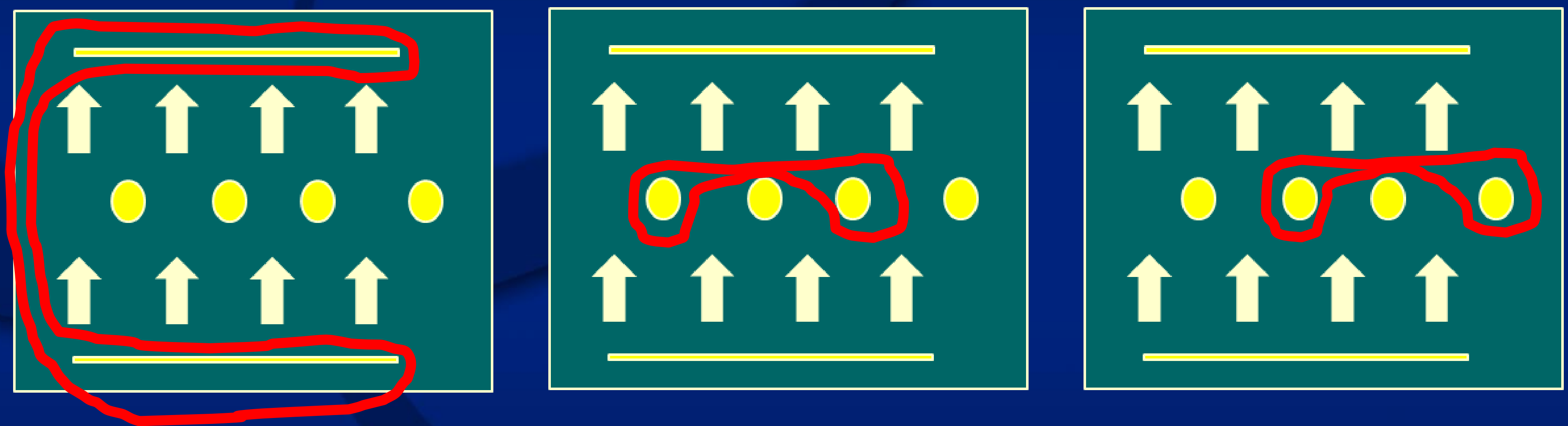
Peaks and Pits: Reflective Symmetry



The 6 Diagonal Reflective Symmetries

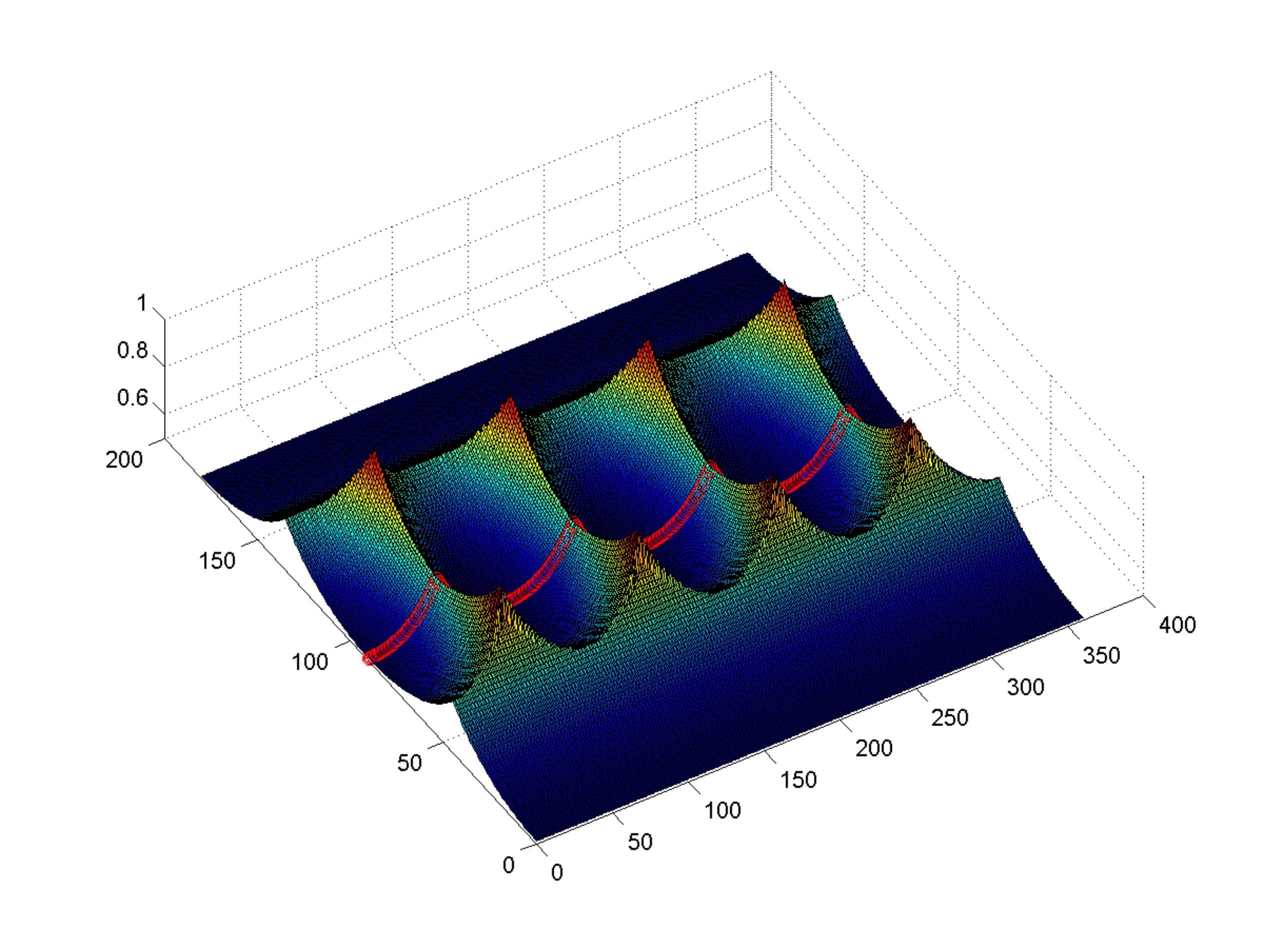


The 3 Orthogonal Reflective Symmetries

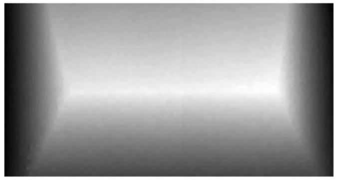


These pairs determine 2 lines which define a plane

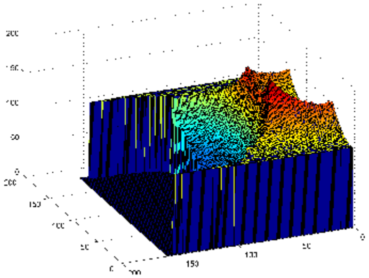
Possible Symmetry Found by Curve Made of Minima



3D Kinect Data of Cube



Two Corner Kinect Data

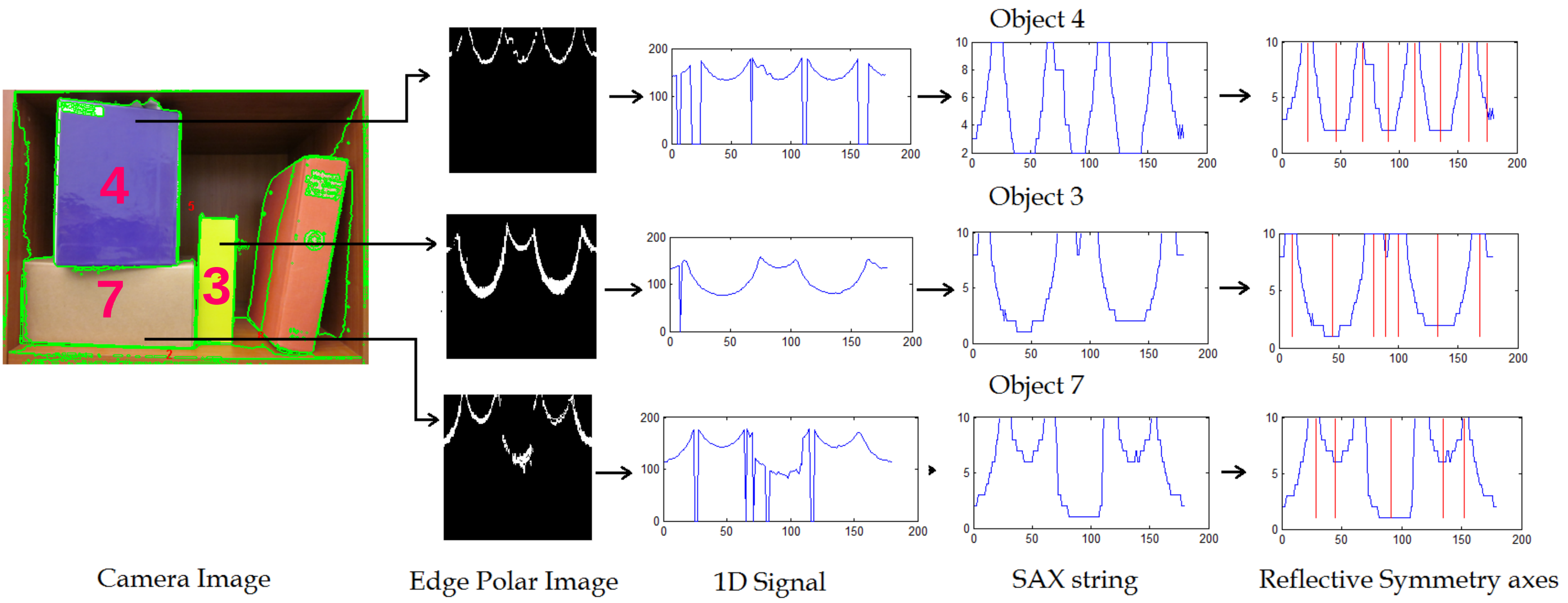


Two Corner Peaks in FEP Data

Some Lemmas; e.g.:

Lemma 1: Given a frieze expansion pattern at the center of mass of a 2D shape with reflective symmetry, then the orientation of the reflective axis corresponds to a min or max of the FEP of the boundary curve.

Concept Formation



Concept Formation

Object 3 Symmetry Type	*Start Index*	*End Index*	*Basic Length*	*Symmetry Measure*
periodic	1	270	90	0.5000
reflective	2	20	9	0.3497
reflective	2	88	43	0.4241
reflective	68	90	11	0.4152
reflective	2	176	87	0.4627
reflective	90	110	10	0.6862
reflective	87	179	46	0.4759
reflective	157	179	11	0.6412

Concept Formation

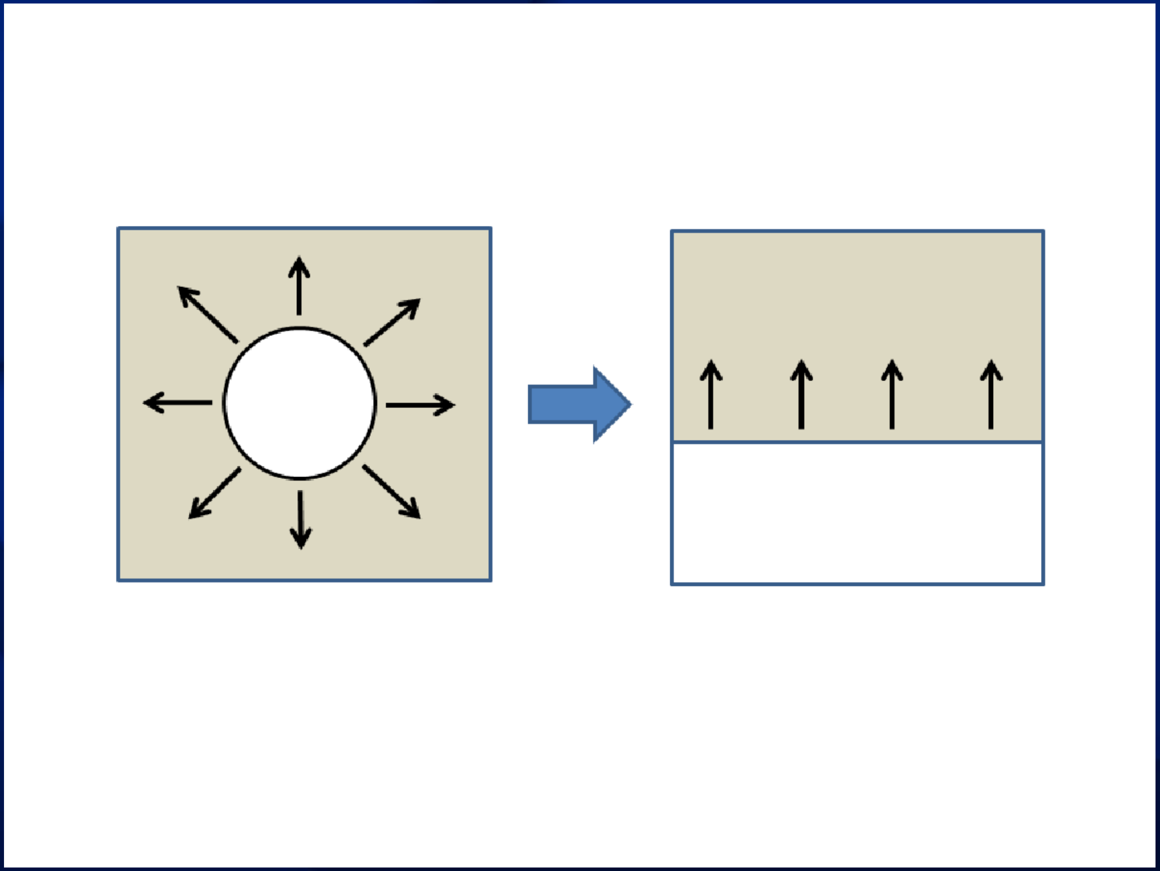
Object 4 Symmetry Type	*Start Index*	*End Index*	*Basic Length*	*Symmetry Measure*
periodic	1	270	90	0.6000
reflective	2.0000	42.0000	20.0000	0.5902
reflective	2.0000	90.0000	44.0000	0.6079
reflective	36.0000	102.0000	33.0000	0.4067
reflective	3.0000	179.0000	88.0000	0.6802
reflective	62.0000	164.0000	51.0000	0.3894
reflective	91.0000	179.0000	44.0000	0.4098
reflective	139.0000	179.0000	20.0000	0.5495
reflective	171.0000	179.0000	4.0000	0.5698

Structural Bootstrapping

- These concept structures allow for tree representations
- There exist group representations that allow for understanding symmetries on trees
- Can substitute symbols at various levels to bootstrap knowledge

Symmetry Bundles

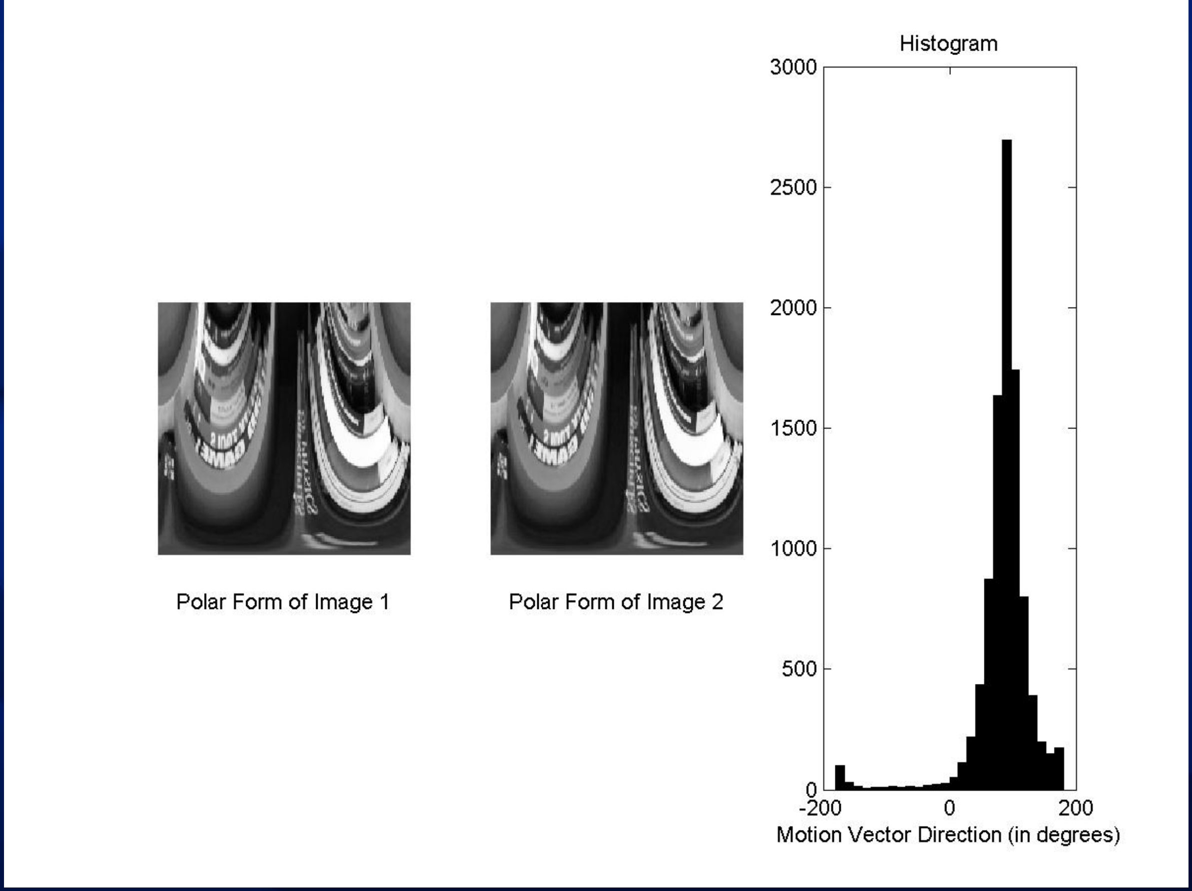
Bundle Definition:
- Actuation
  → Translation
- Perception
  → columnar translation in FEP



Symmetry Bundles

Bundle Detection:

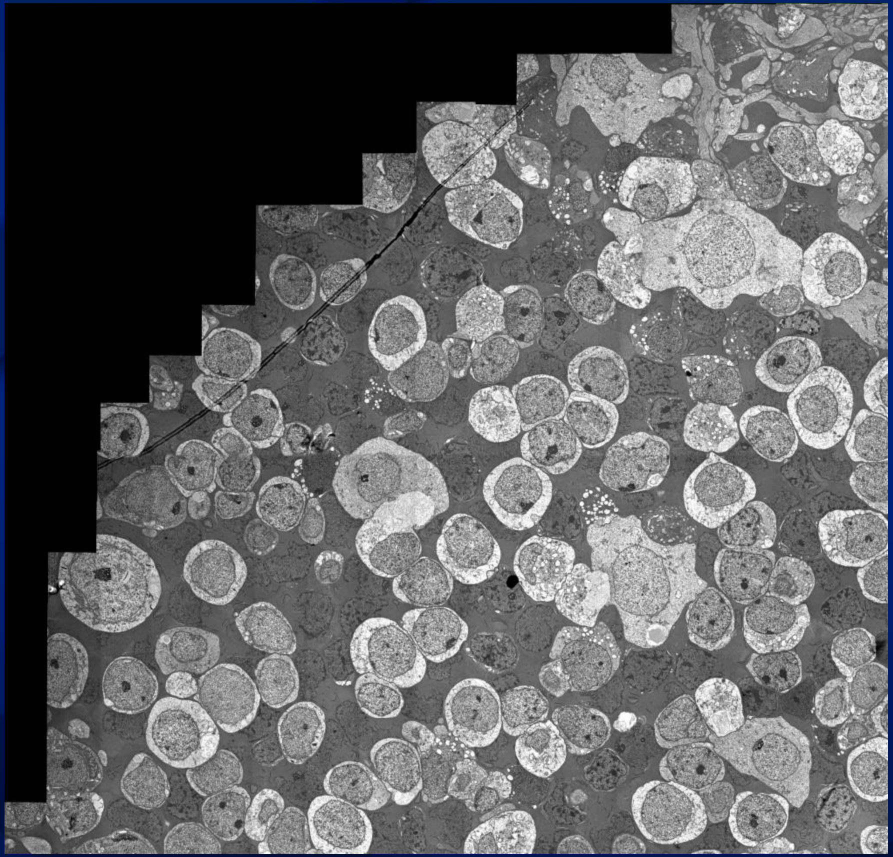
- Actuation
  ➔ Apply signal
- Perception
  ➔ histogram translation direction (90 degrees)



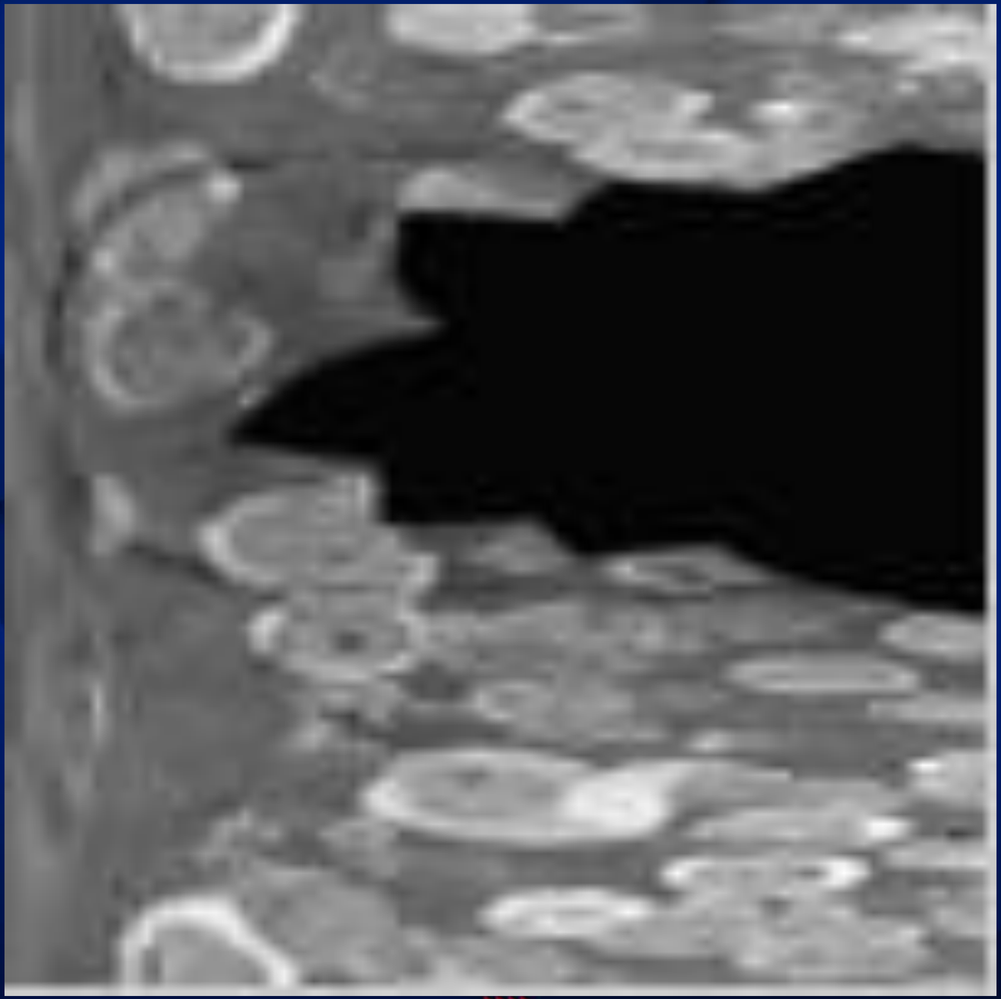
Questions?

Moving through FEP's

Given a section of neurons



Move through to Find Centers



Still Sequence