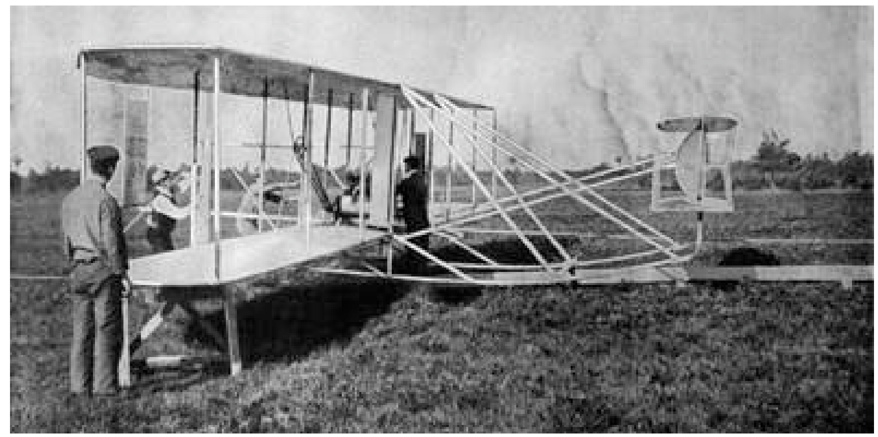
Machine Learning for Sensorimotor Procesing

Daniel D. Lee

PENN

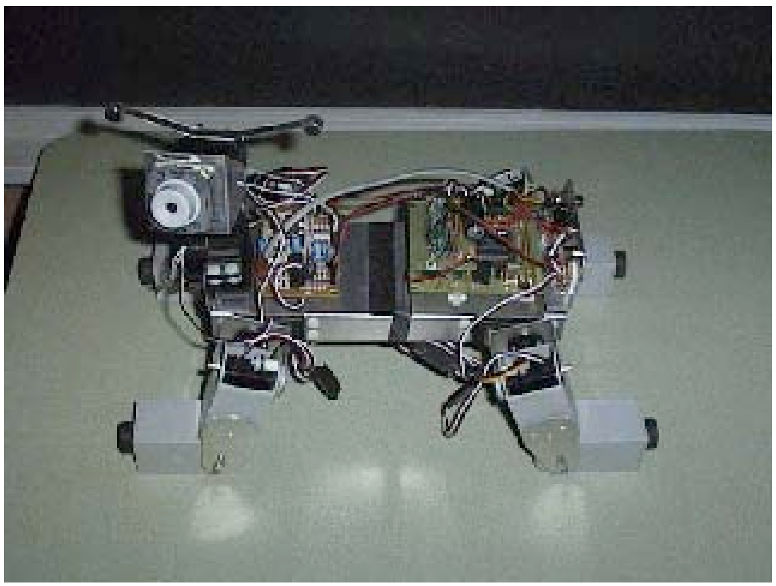
Biological inspiration



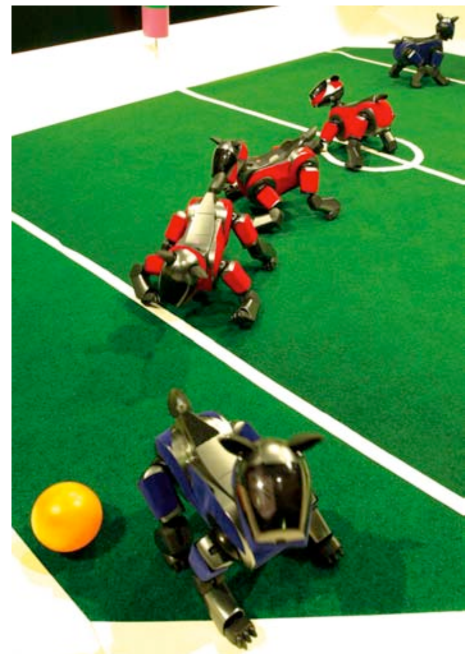
- Biological motivation for the Wright brothers in designing the airplane

Robot dogs

Custom built version

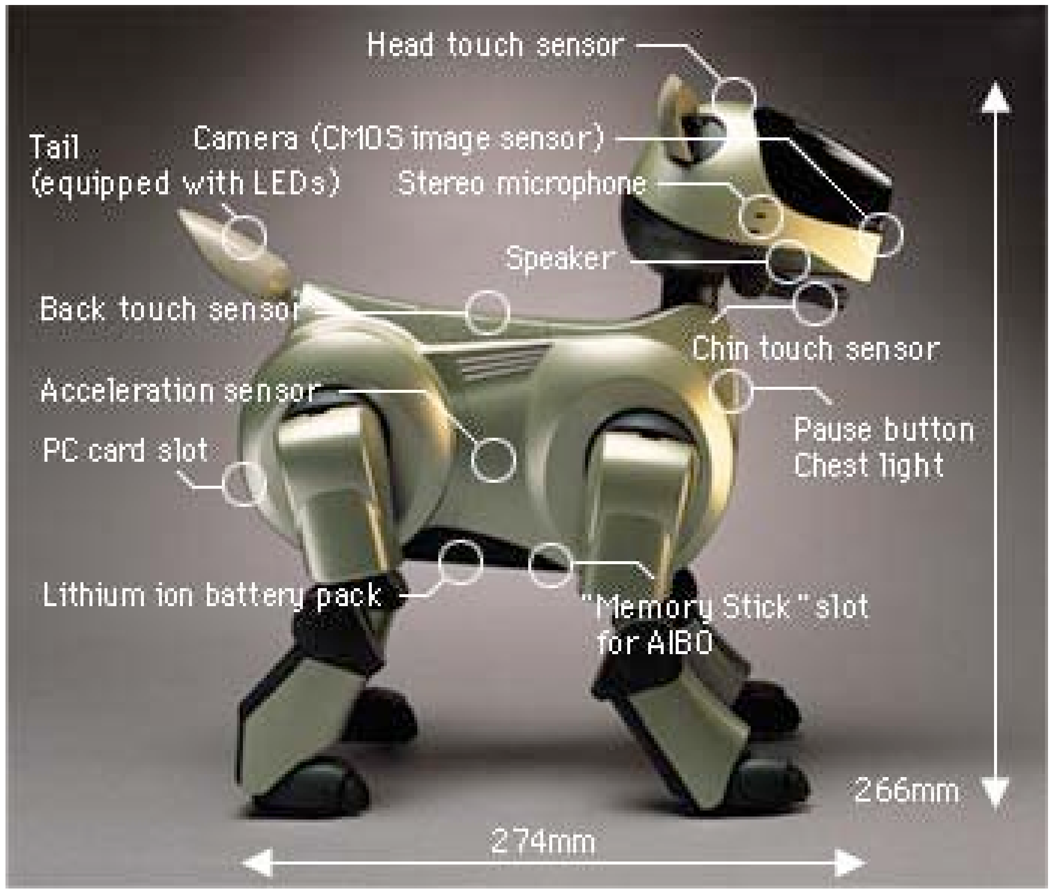


Sony Aibos



- Platforms for testing sensorimotor machine learning algorithms

Robot hardware



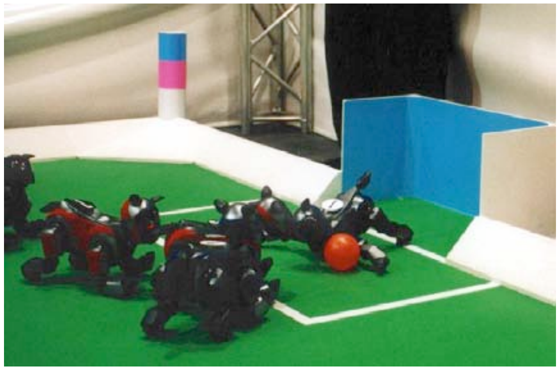
- Wide variety of sensors and actuators.

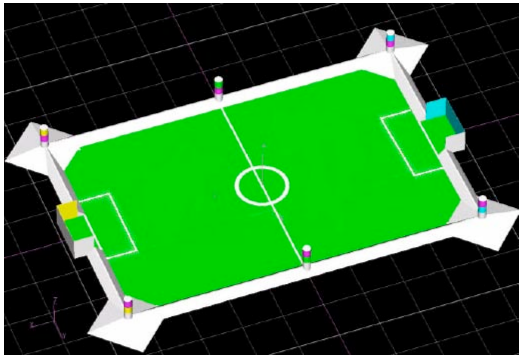
Perception and motor control

- Wide variety of sensors and actuators are readily available

Legged league

- Each team consists of 4 Sony Aibo robot dogs (one is a designated goalie), with WiFi communications.
- Field is 3 by 5 meters, with orange ball and specially colored markers.
- Game played in two halves, each 10 minutes in duration. Teams change uniform color at half-time.
- Human referees govern kick-off formations, holding, penalty area violations, goalie charging, etc.
- Penalty kick shootout in case of ties in elimination round.



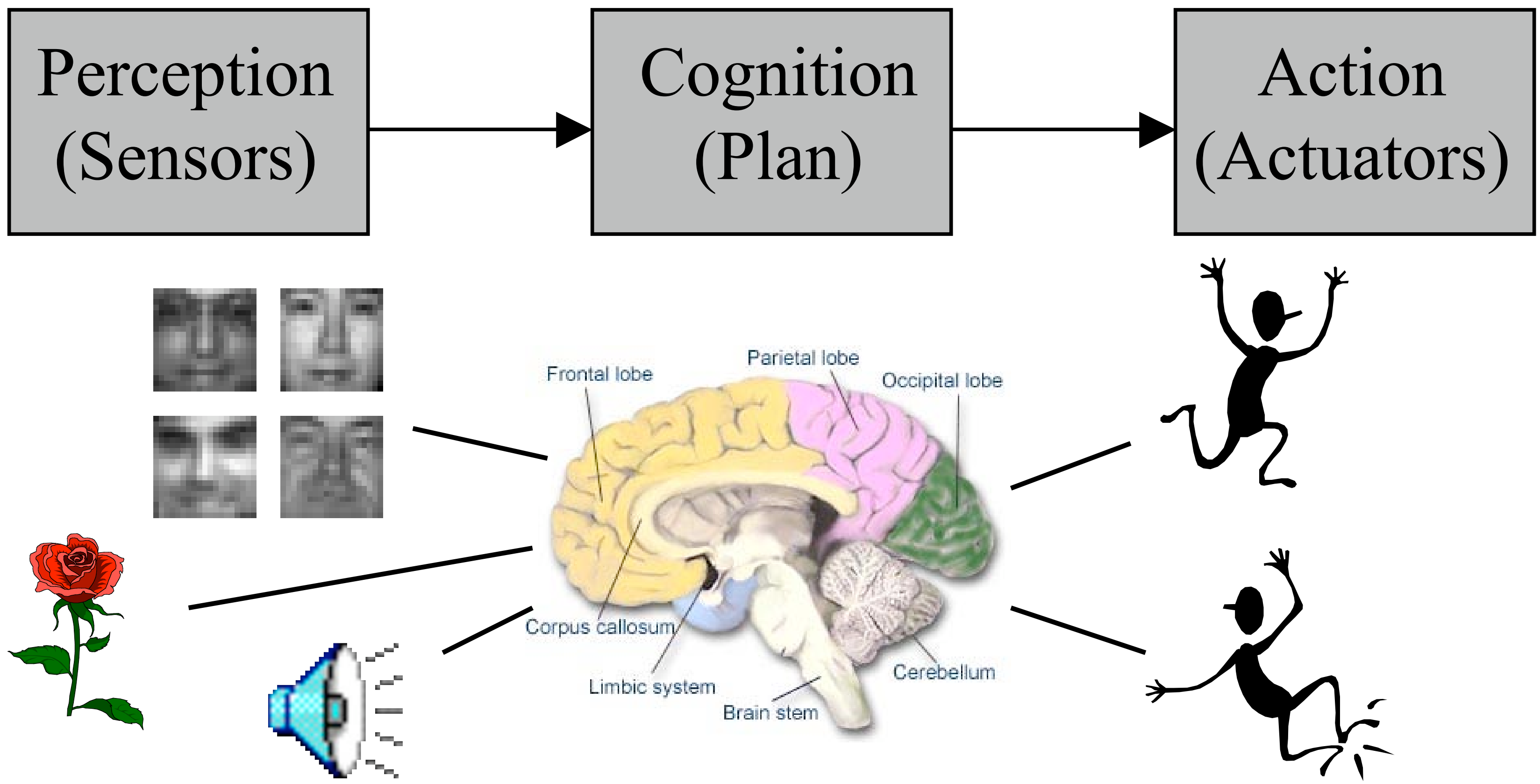


- Recently implemented larger field and wireless communications among robots.

Upennalizers in action

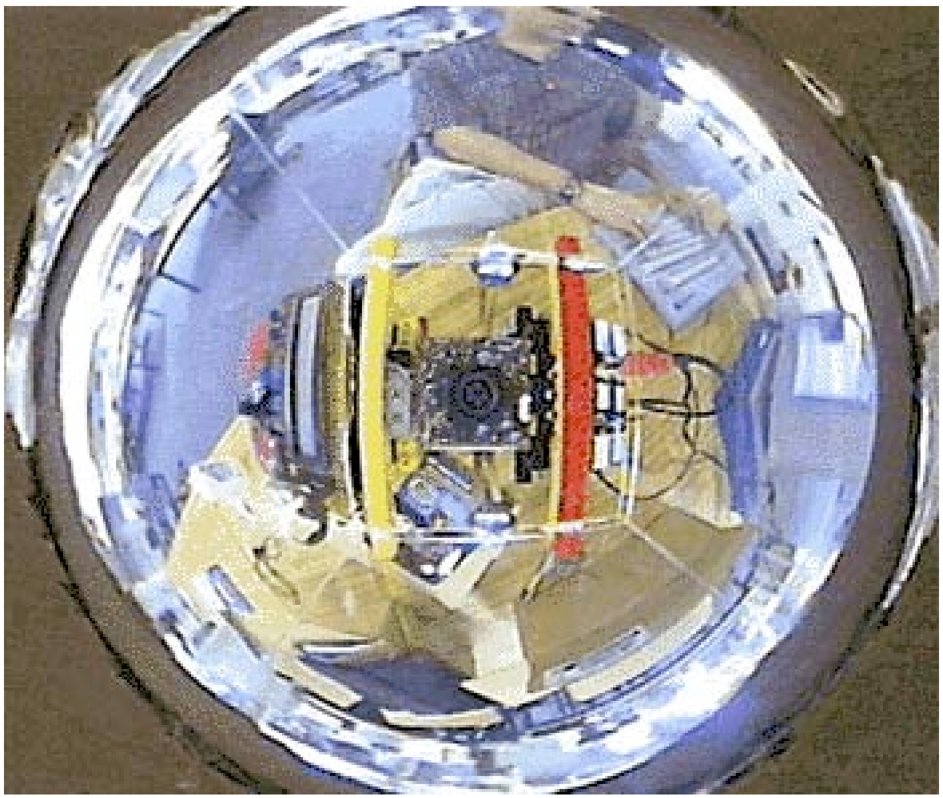
- GOOAAALLL! 2nd place in 2003.

Robot software architecture



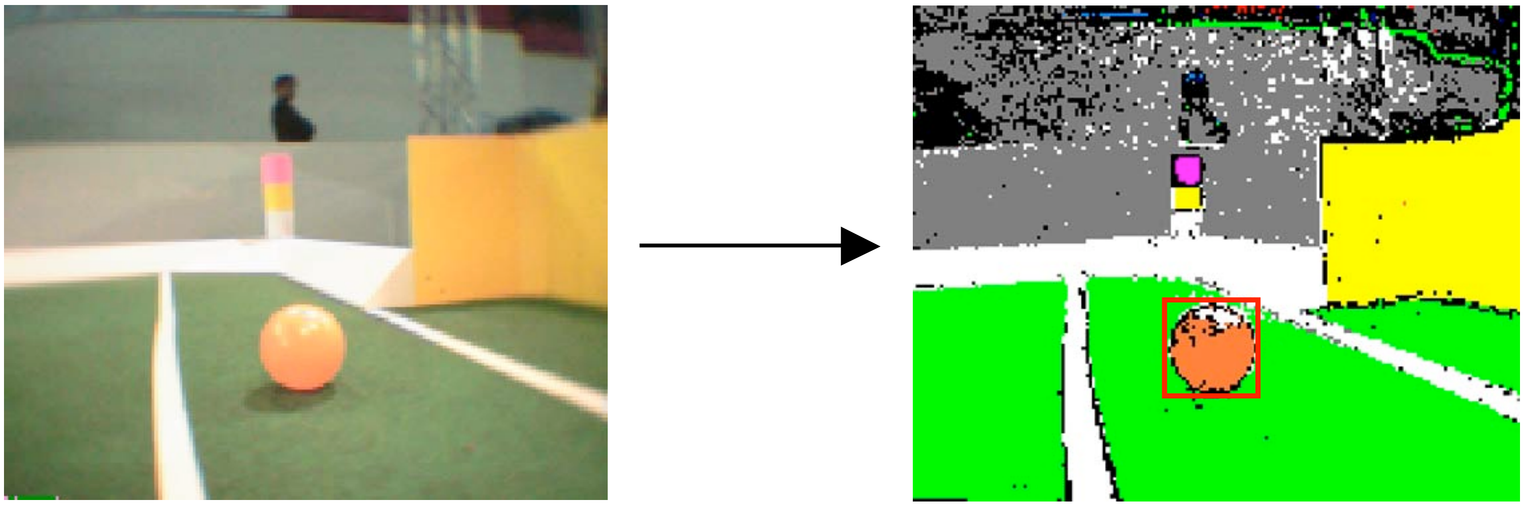
- Sense-Plan-Act cycle.

Perception



View from Penn's omnidirectional camera

Robot vision



Color segmentation: estimate P(Y,Cb,Cr | ORANGE) from training images

Region formation: run length encoding, union find algorithm

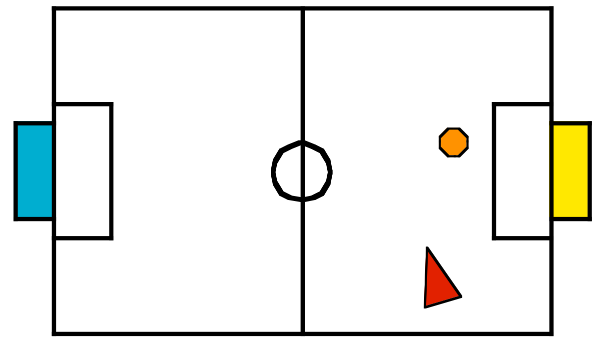
Distance calibration: bounding box size and elevation angle

Camera geometry: transformation from camera to body centered coordinates

- Tracking objects in structured environment at 25 fps

Image reduction

$$\begin{bmatrix} (53,65,27) & (52,67,35) & \cdots \\ (48,68,31) & \ddots & \\ \vdots & & (250,213,196) \end{bmatrix}$$



Deterministic position:
(x_{ball}, y_{ball})

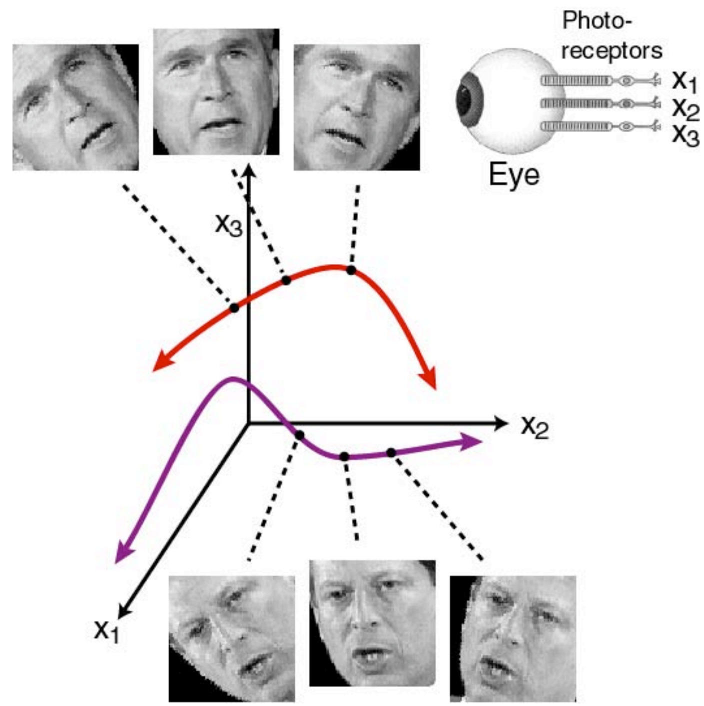
Probabilistic model (Kalman):
$P(x_{ball}, y_{ball})$

- $144 \times 176 \times 3$ RGB image to 2 position coordinates

Image manifolds

$$\begin{bmatrix} 0.0 \\ 0.5 \\ 0.7 \\ 1.0 \\ \vdots \\ 0.8 \\ 0.2 \\ 0.0 \end{bmatrix}$$

Pixel vector



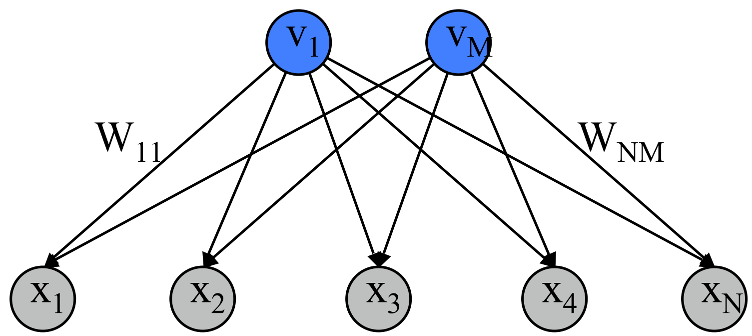
- Variation in pose and illumination give rise to low dimensional manifold structure

Matrix factorization

Data matrix:

$$X = \left[\begin{pmatrix} \vdots \\ \vec{x}_1 \\ \vdots \end{pmatrix} \begin{pmatrix} \vdots \\ \vec{x}_2 \\ \vdots \end{pmatrix} \cdots \right]$$

$$\approx W\,V$$



$$\langle x_i \rangle = \sum_{k=1}^{M} W_{ik} v_k$$

- Learning and inference of hidden units is equivalent to matrix factorization.
- Prior probability distribution of hidden variables.

Linear factorization (PCA)

$$X = \begin{bmatrix} \begin{pmatrix} \vdots \\ \vec{x}_1 \\ \vdots \end{pmatrix} & \begin{pmatrix} \vdots \\ \vec{x}_2 \\ \vdots \end{pmatrix} & \cdots \end{bmatrix}$$

$$X_{ij} \approx \sum_{k=1}^{M} W_{ik} V_{kj}$$

$$\min_{W,V} \| X - WV \|$$

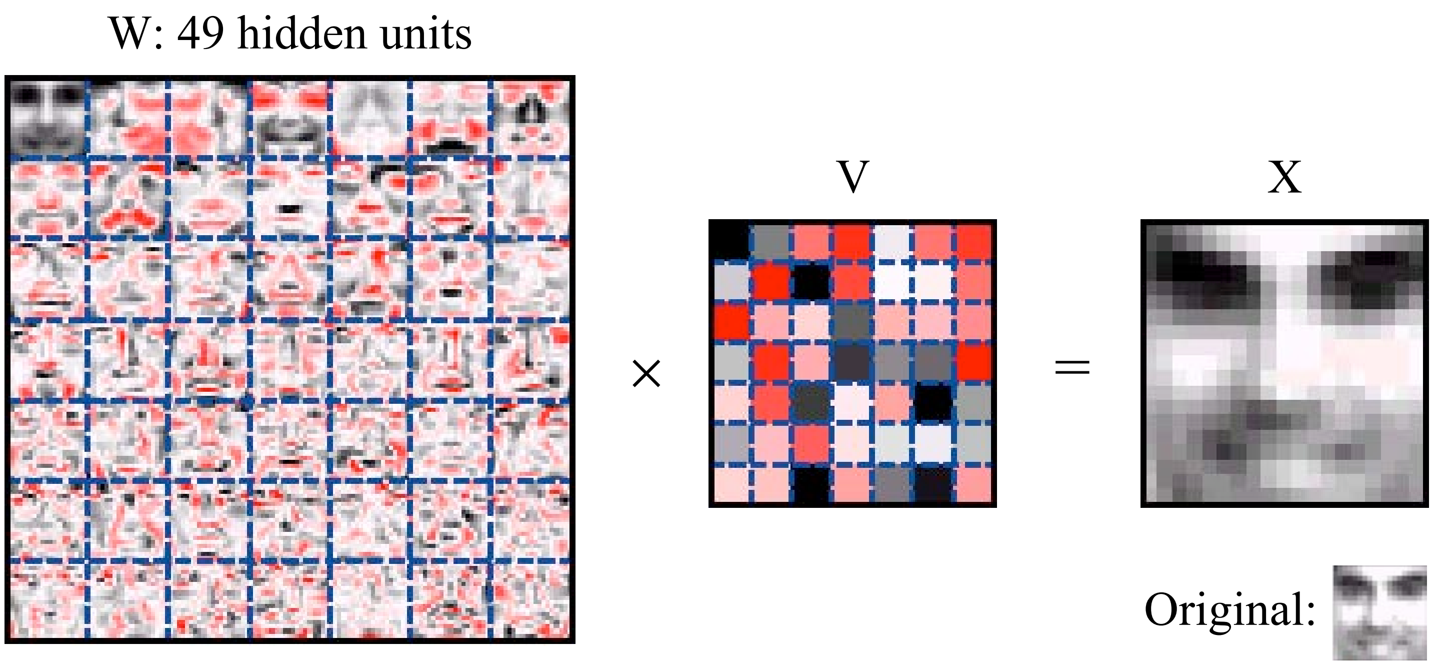
subject to constraints:

$$W^T W = I$$

$$VV^T \text{ diagonal}$$

- Equivalent to singular value decomposition.
- Degeneracy eliminated by constraining W orthonormal, rows of V orthogonal.

PCA representation



- Linear model: representation uses positive and negative combinations to reconstruct original image.

Vector quantization

$$X_{ij} \approx \sum_{k=1}^{M} W_{ik} V_{kj}$$

$$\min_{W,V} \| X - WV \|$$

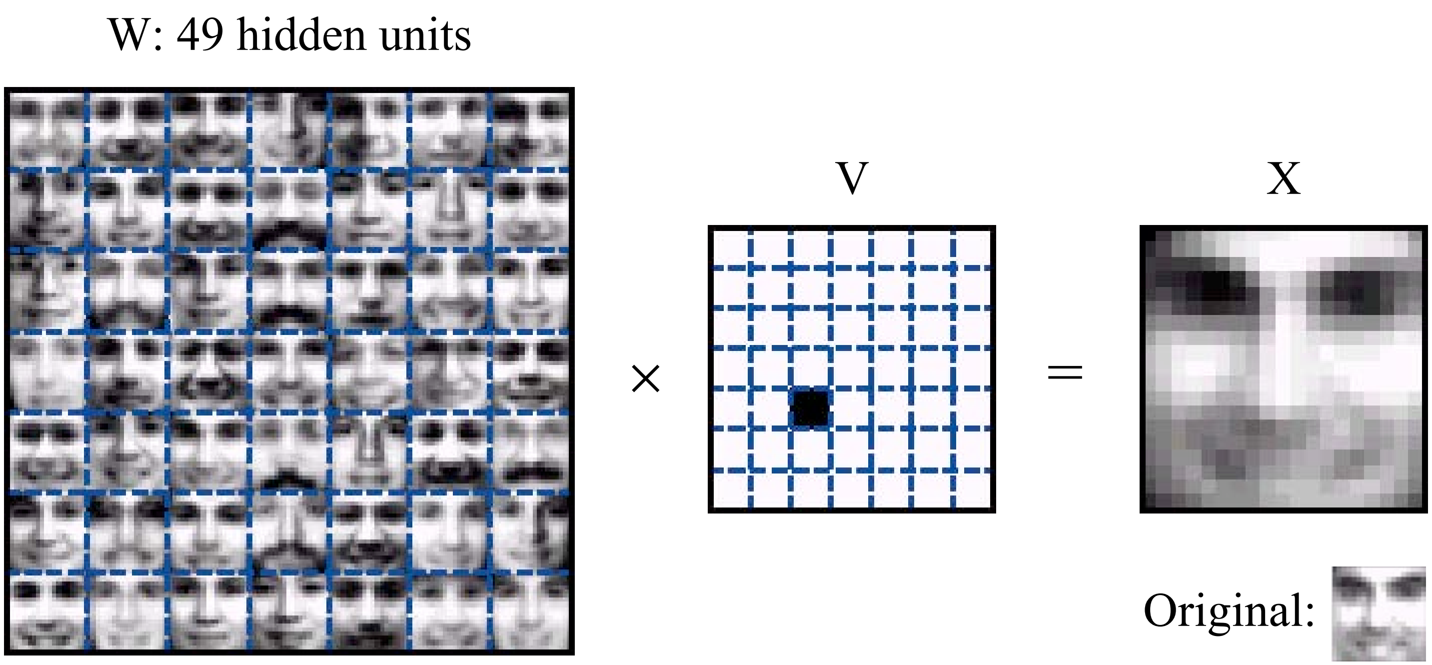
Winner take all constraint:

$$V = \begin{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} & \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} & \cdots \end{bmatrix}$$

Unary vectors

- Columns of V constrained to be unary.
- Winner take all competition between hidden variables in modeling data.

VQ representation



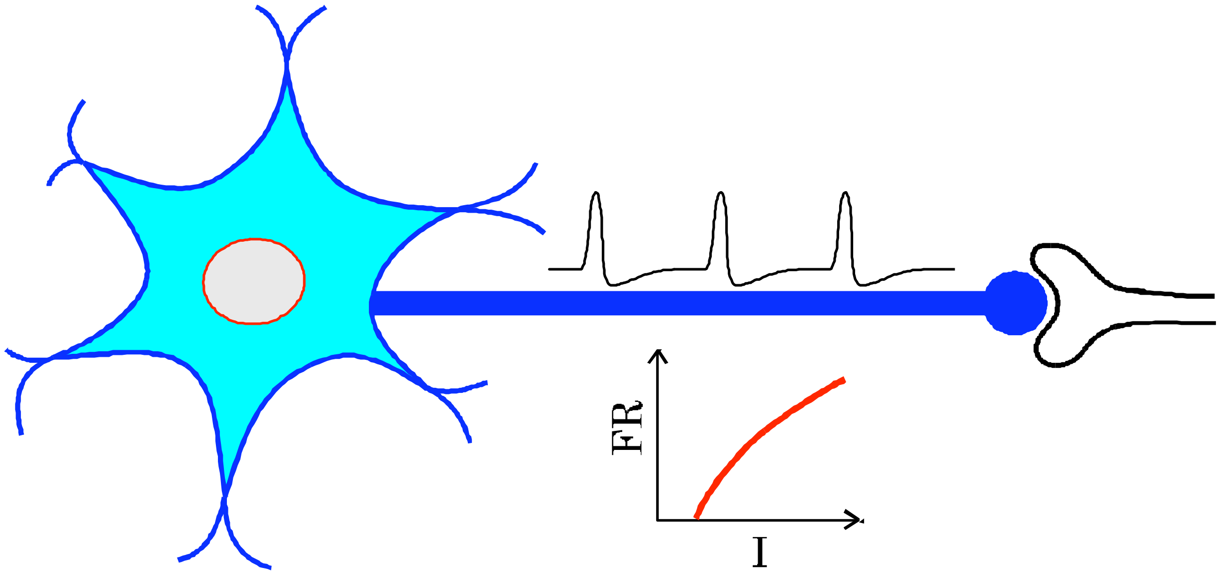
- Weights cluster data into representative prototypes.

Constraints and representations

Model	*Constraints*	*Representation*
VQ	$v_k = \delta_{kk'}$ for some k'	very sparse
PCA	$\vec{v} \cdot \vec{v}' = 0$	distributed

- Hard constraints give rise to very sparse representations, with bases identifiable as prototypes.
- Soft constraints lead to a very distributed representation but with features difficult to interpret.

Biological motivation



- Firing rates of neurons are nonnegative
- Synapses are either excitatory or inhibitory
- What are the functional implications of these nonlinear constraints for learning?

Nonnegative constraints

Hidden Prior:

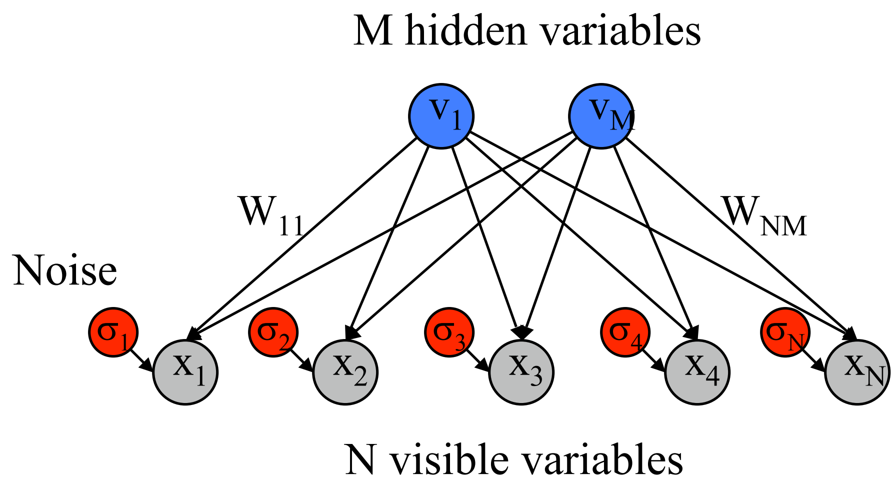
$$P(v) > 0 \quad \text{iff} \quad v \geq 0$$

Excitatory Weights:

$$W_{ij} \geq 0$$

Visible Variables:

$$x_i \approx \sum_{k=1}^{M} W_{ik} v_k$$



- Nonnegative hidden variables and weights.

Nonnegative factorization algorithm

$$\min_{W \geq 0, V \geq 0} F = \sum_{ij} \left| X_{ij} - (WV)_{ij} \right|^2$$

Derivatives:

$$\frac{\partial F}{\partial V_{ij}} = (W^T WV)_{ij} - (W^T V)_{ij}$$

$$\frac{\partial F}{\partial W_{ij}} = (WVV^T)_{ij} - (XV^T)_{ij}$$

Inference:

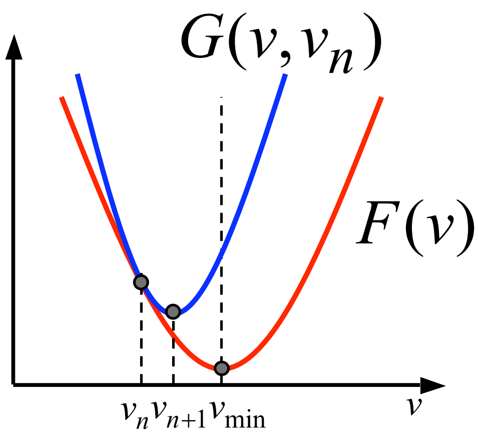
$$V_{ij} \leftarrow V_{ij} \frac{(W^T X)_{ij}}{(W^T WV)_{ij}}$$

Learning:

$$W_{ij} \leftarrow W_{ij} \frac{(XV^T)_{ij}}{(WVV^T)_{ij}}$$

- Multiplicative algorithm can be interpreted as diagonally rescaled gradient descent.

Convergence proof



Upper bound:

$$F(v) \le G(v, v_n)$$

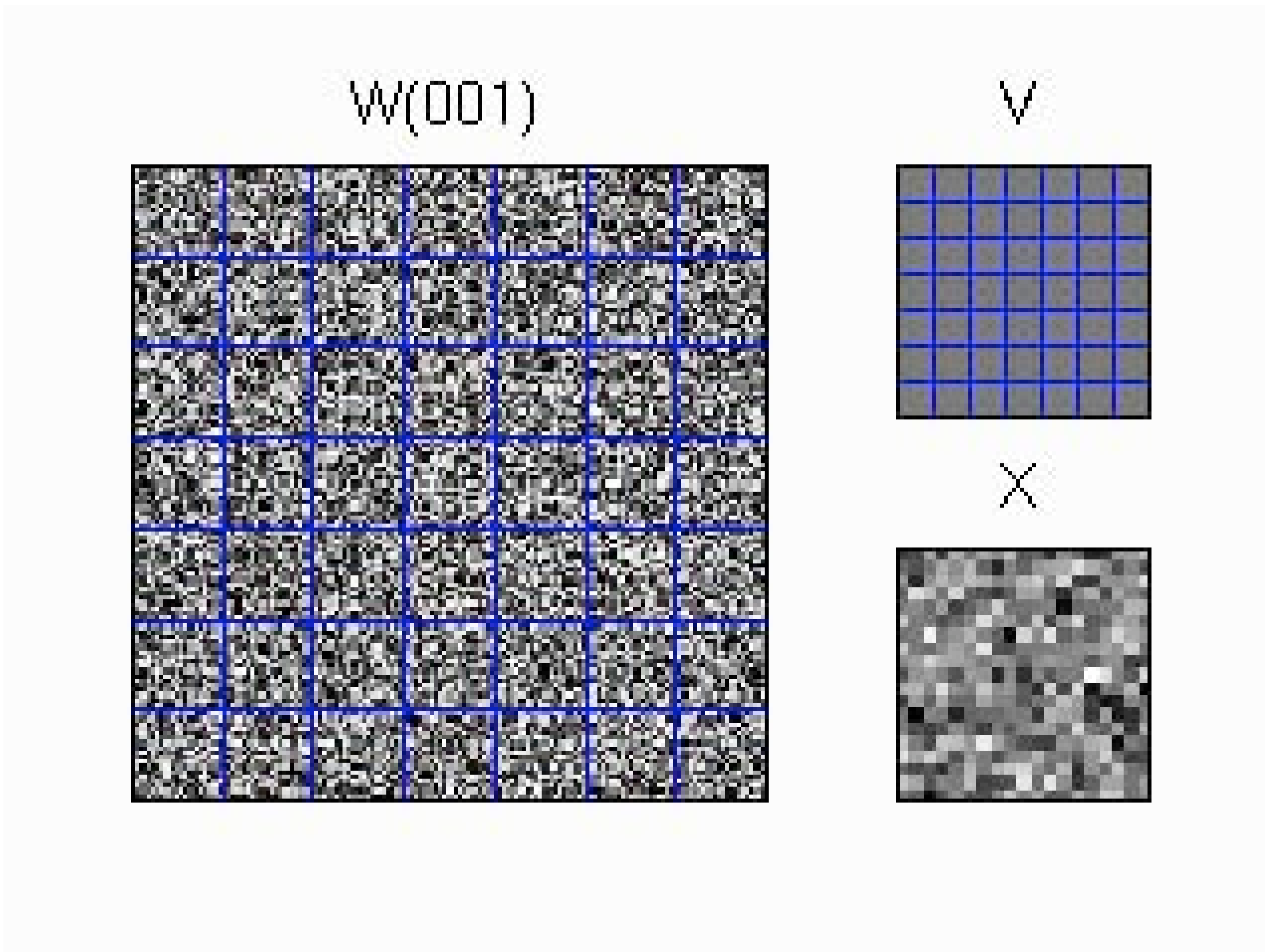
$$F(v_n) = G(v_n, v_n)$$

Update rule: $v_{n+1} = \arg\min_{v} G(v, v_n)$

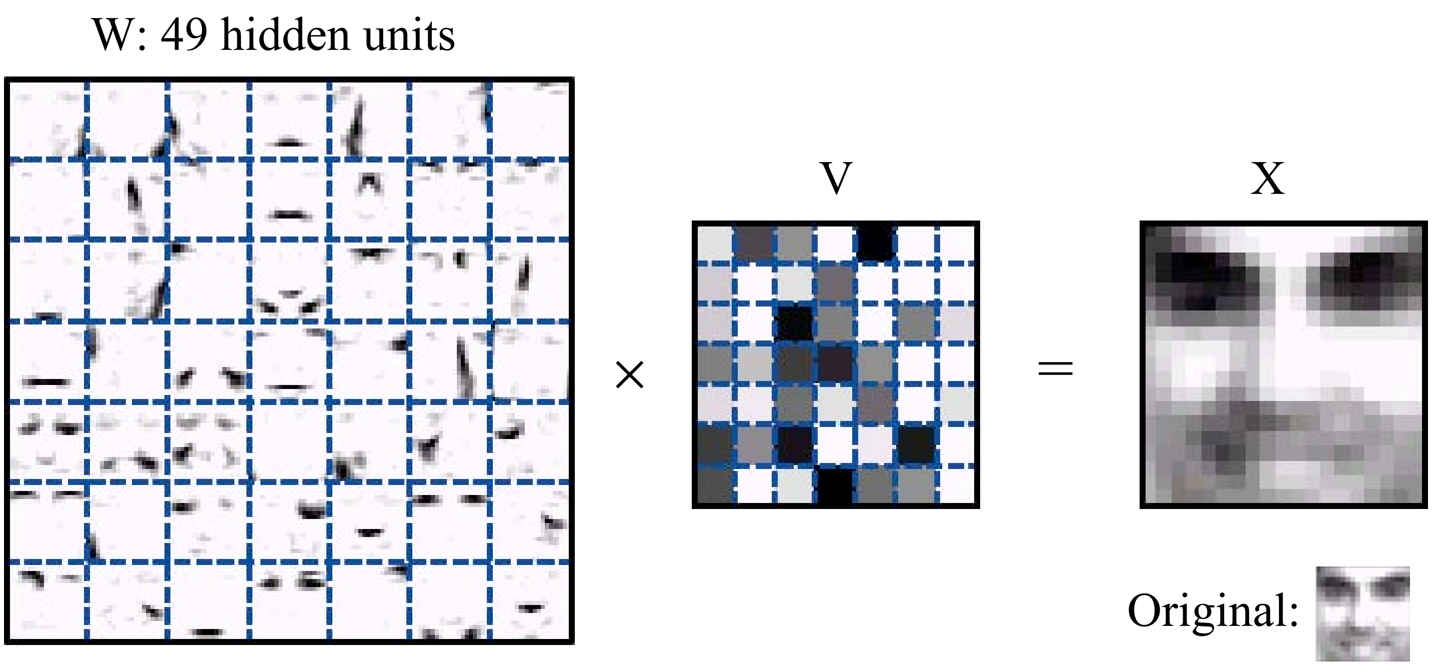
$$\longrightarrow \quad F(v_{n+1}) \le G(v_{n+1}, v_n) \le G(v_n, v_n) = F(v_n)$$

- Minimizing auxiliary function G is guaranteed to monotonically converge to local minimum of F.

NMF learning

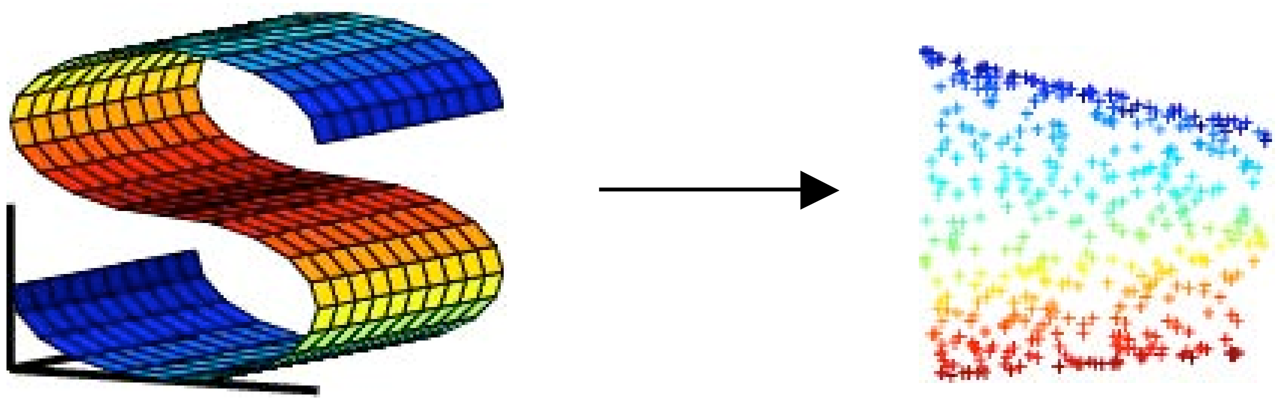


NMF representation



- Learned weights decompose the images into their constituent parts.
- Nonnegative constraint allows only additive combinations.

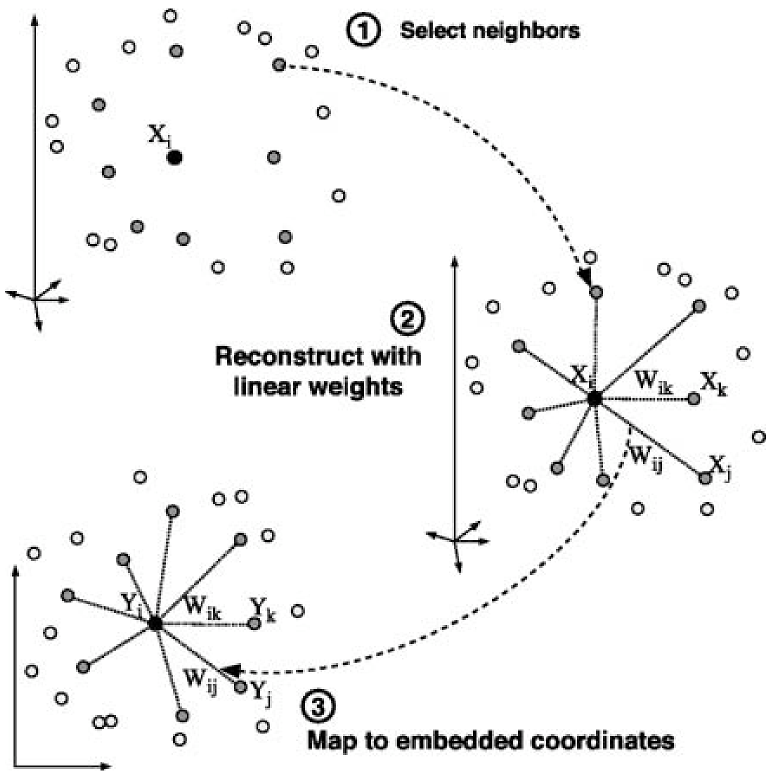
Learning nonlinear manifolds



Kernel PCA, Isomap, LLE, Laplacian Eigenmaps, etc.

- Many recent algorithms for nonlinear manifolds.

Locally linear embedding

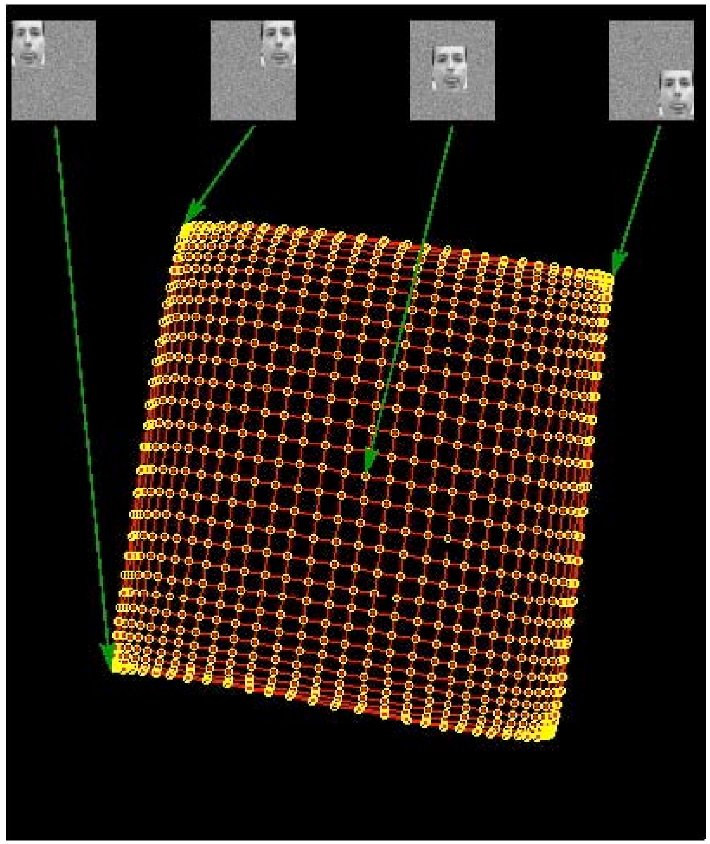


$$E(W) = \sum_i \left| \vec{X}_i - \sum_j W_{ij} \vec{X}_j \right|^2$$

$$\Phi(Y) = \sum_i \left| \vec{Y}_i - \sum_j W_{ij} \vec{Y}_j \right|^2$$

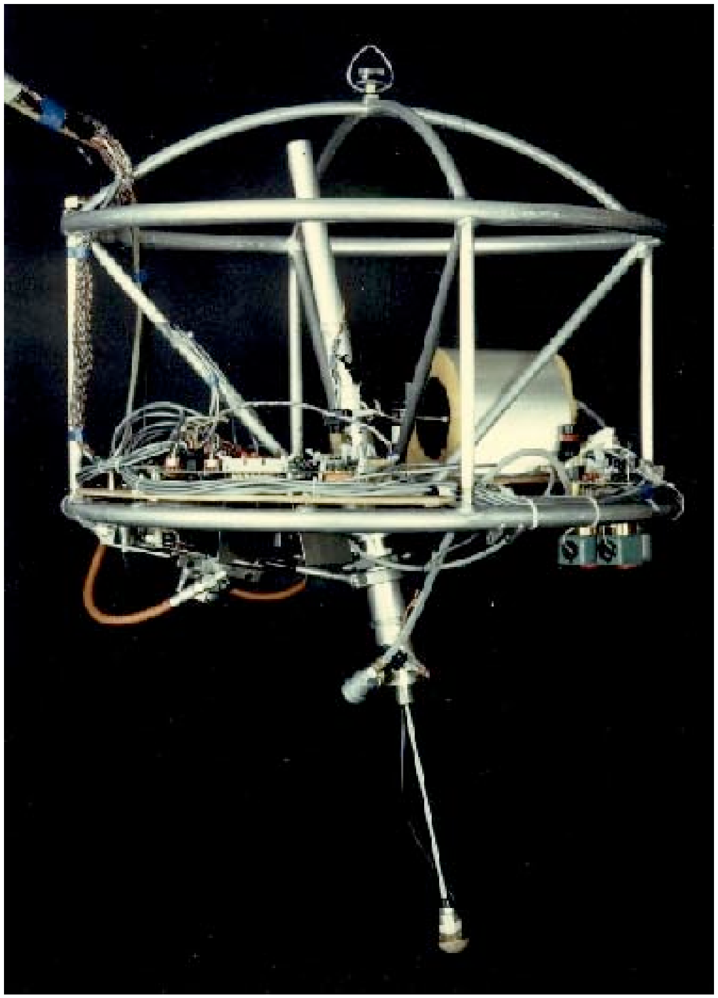
- LLE solves two quadratic optimizations using eigenvector methods (Roweis & Saul).

Translational invariance



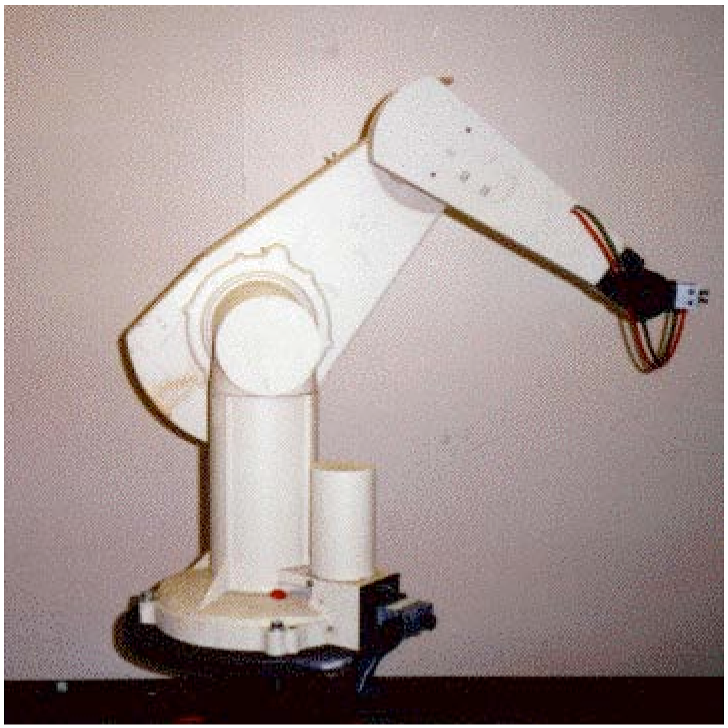
- Application of LLE for translational invariance.

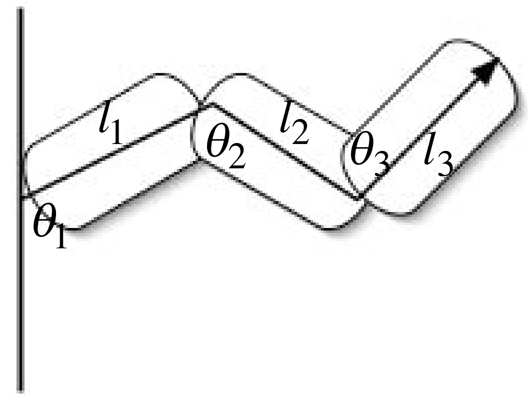
Action



M. Raibert's hopping robot (1983)

Inverse kinematics

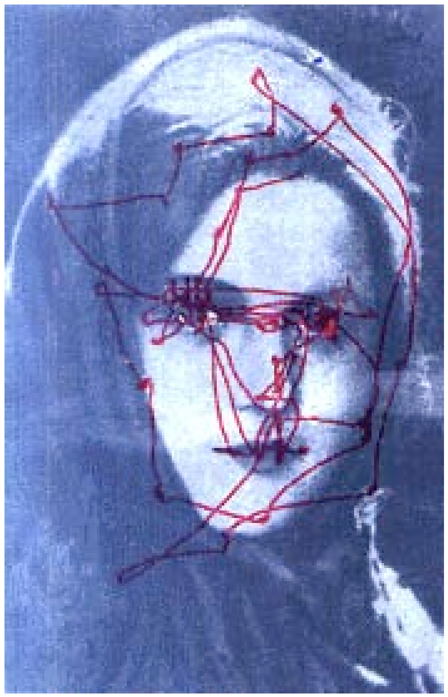




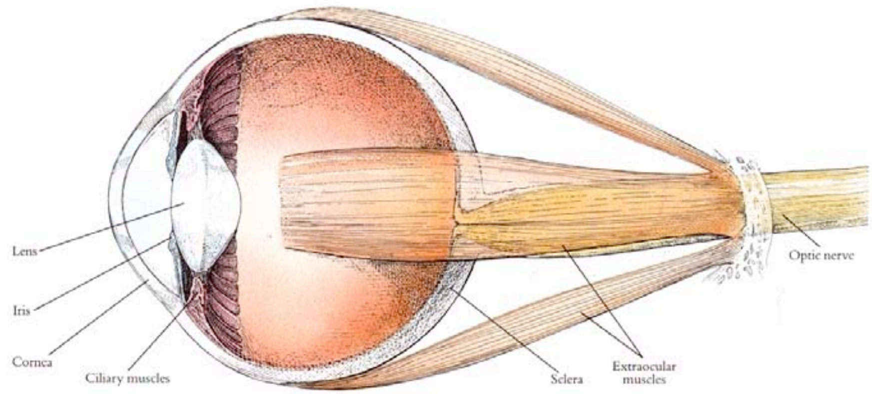
$$H = R(\theta_1) \circ T(l_1) \circ R(\theta_2) \circ T(l_2) \circ R(\theta_3) \circ T(l_3)$$

- Degenerate solutions with many articulators.

Eye movements



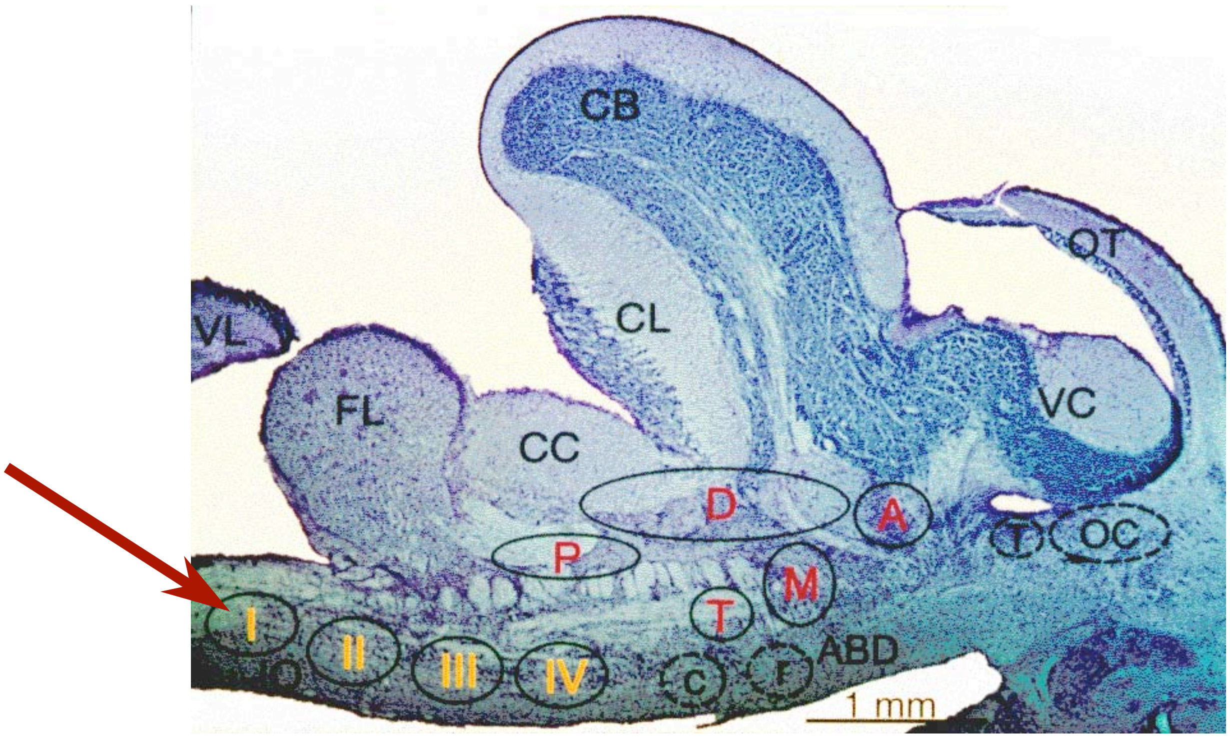
(Yarbus, 1967)



Eye muscles

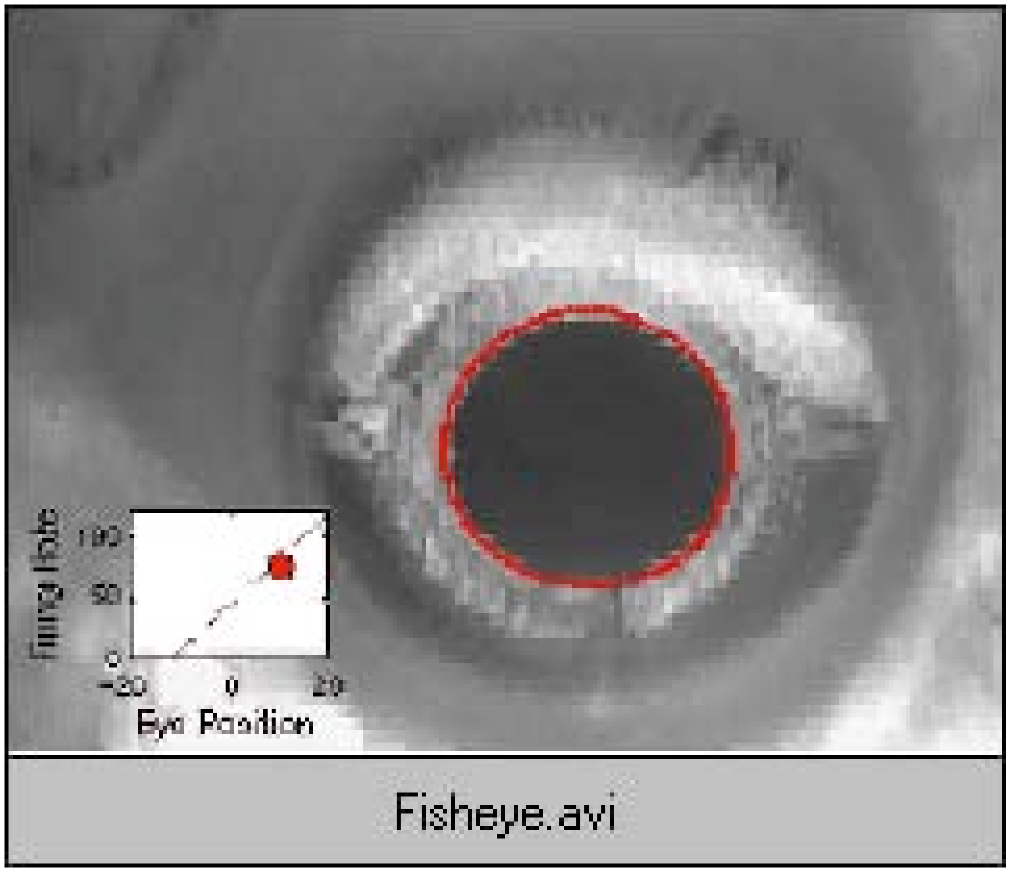
- Fast eye movements to scan visual environment

Neural integrator

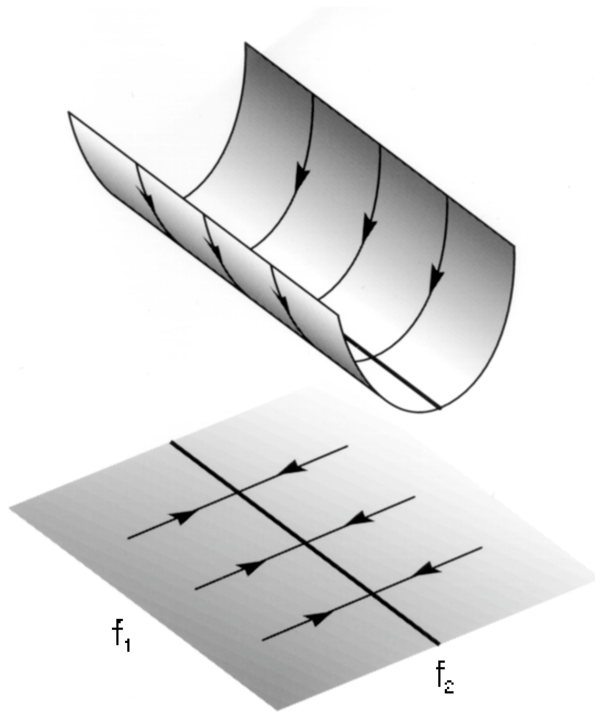


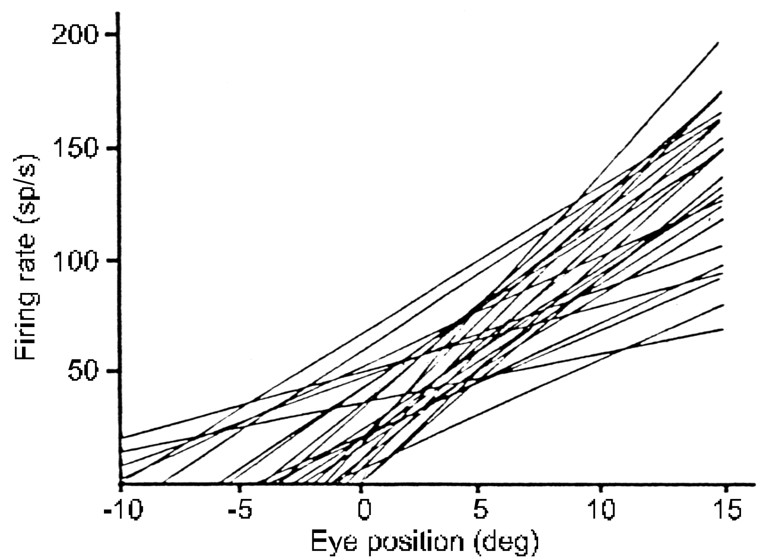
Pastor et al., *PNAS* **91**, 807 (1994)

Control of eye position



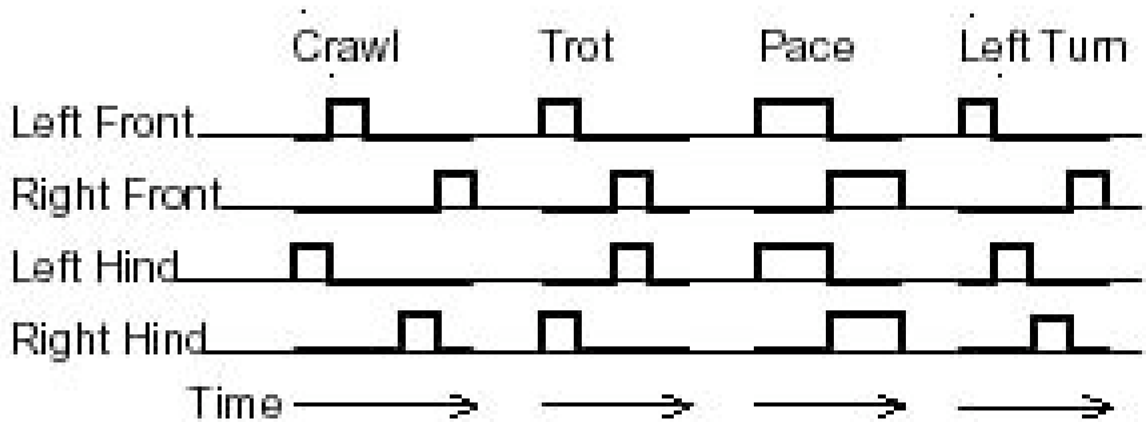
Line attractor

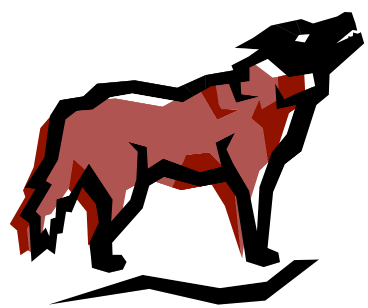




- Low dimensional dynamics for motor control

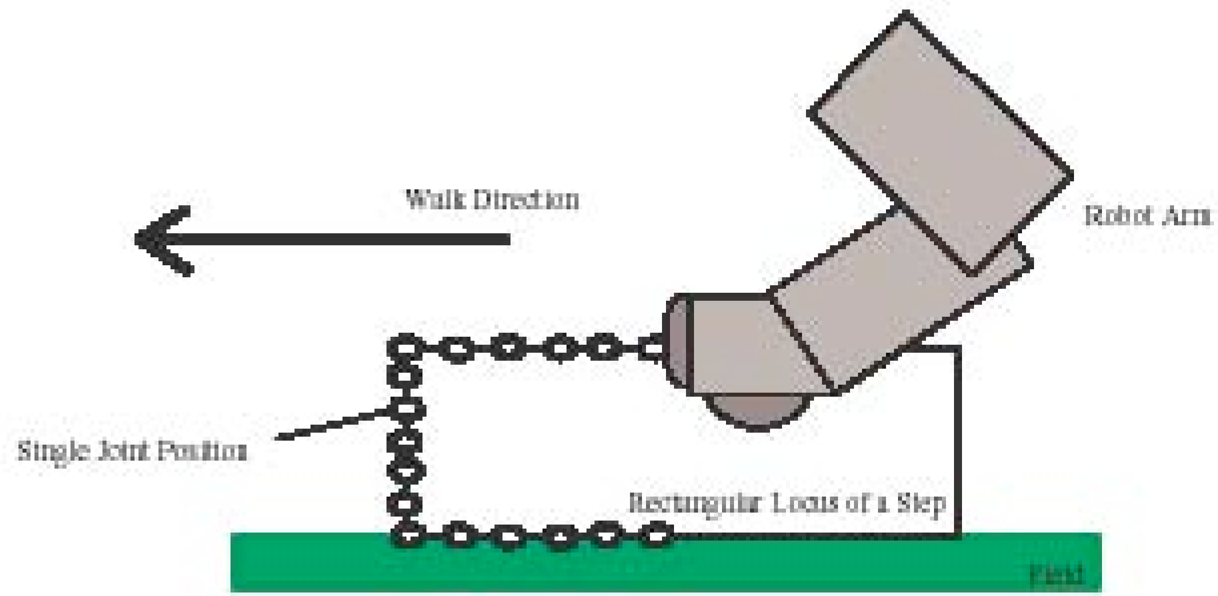
Gaits





- 4-legged animal gaits

Walking



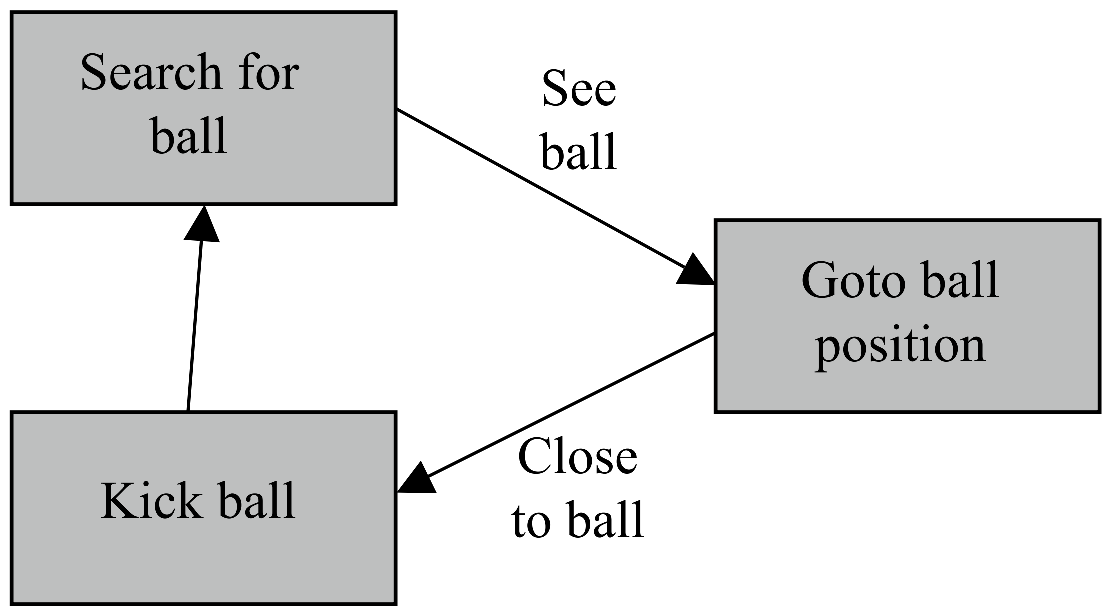
Inverse kinematics to calculate joint angles in shoulder and knee

- Parameters tuned by optimization techniques

Behavior

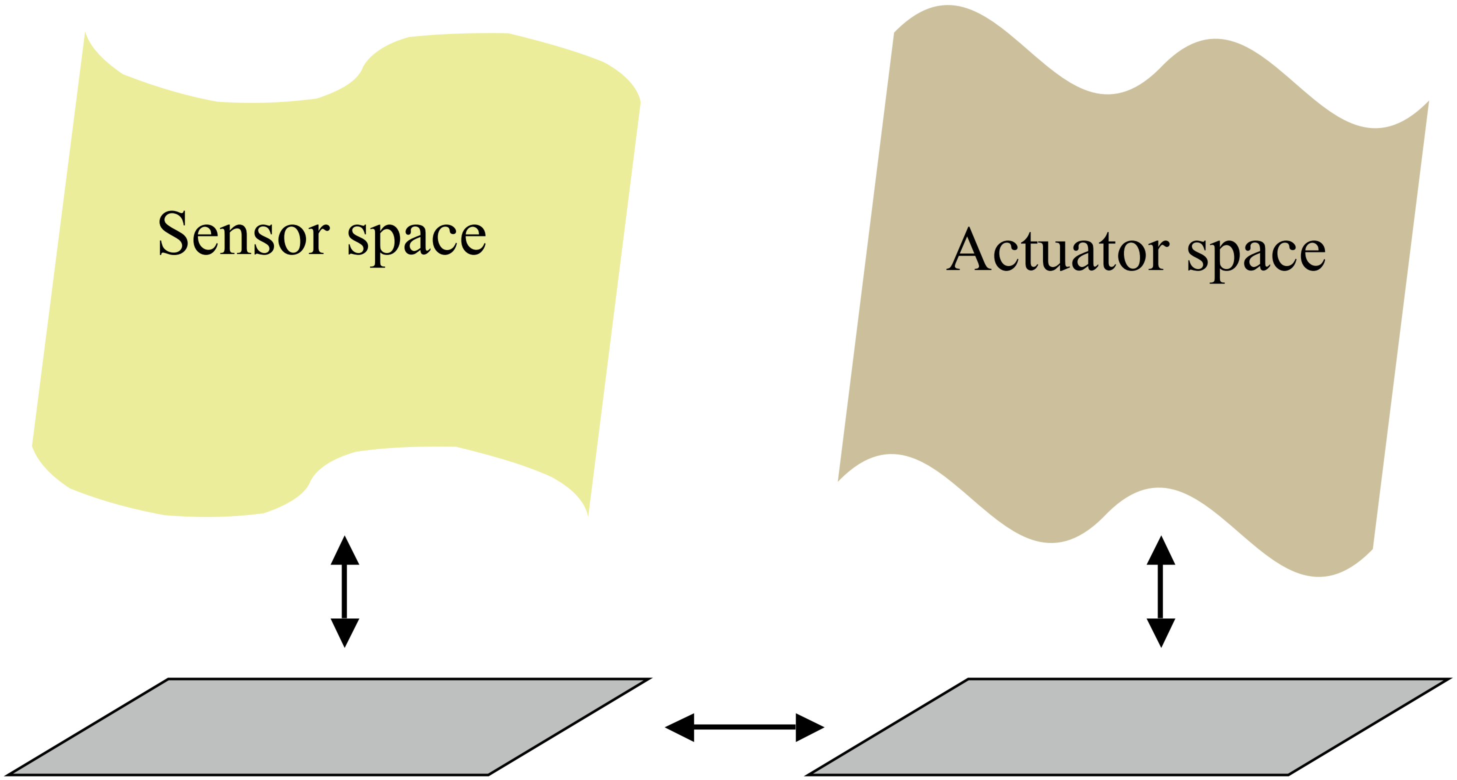
SRI's Shakey (1970)

Finite state machine



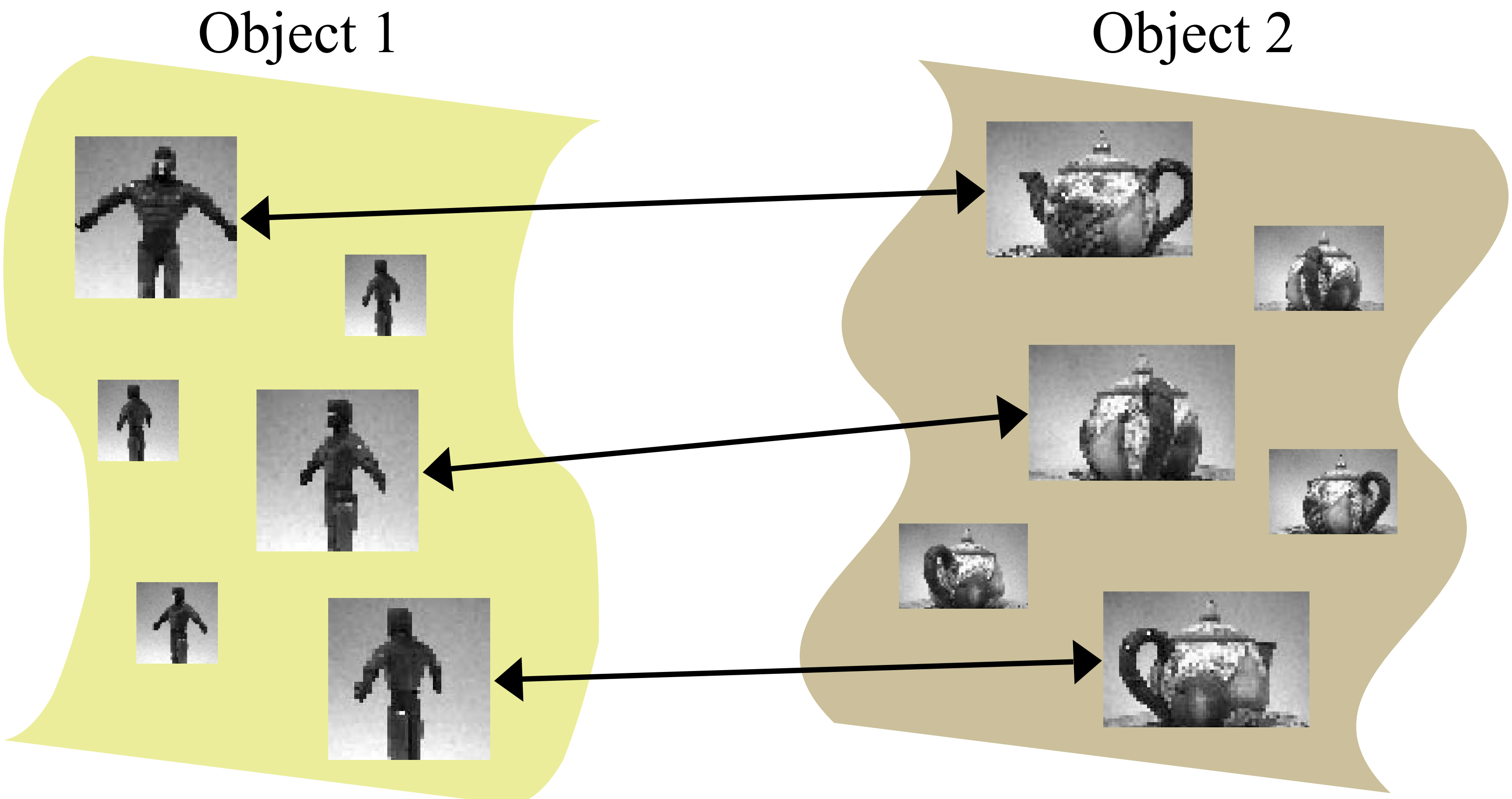
- Event driven state machine.

Sensor-actuator mapping



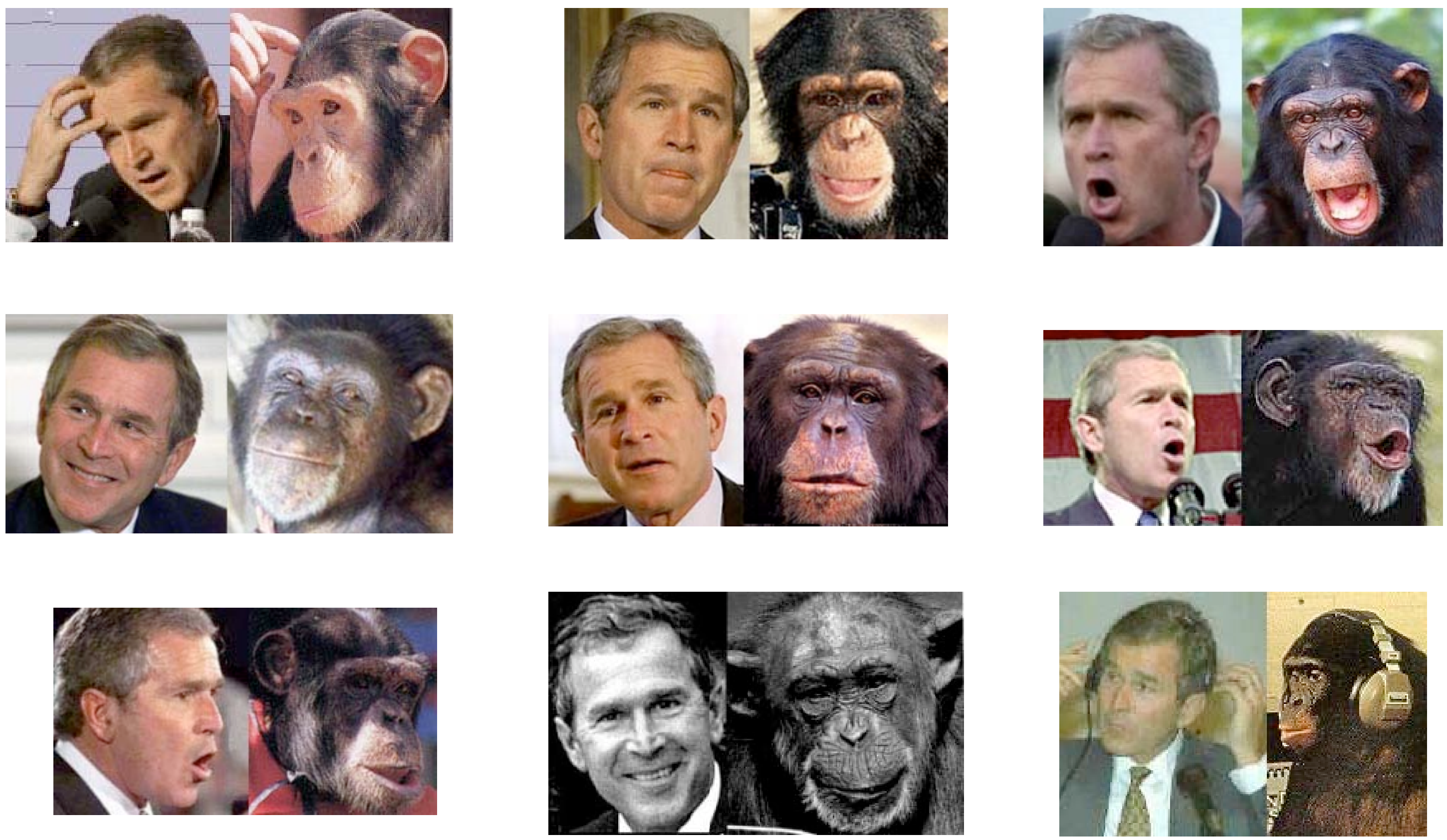
- Construct low dimensional representations for reasoning about sensor stimuli to motor responses

Image correspondences



- Correspondences between images of objects at same pose

Data from the web...



- http://www.bushorchimp.com

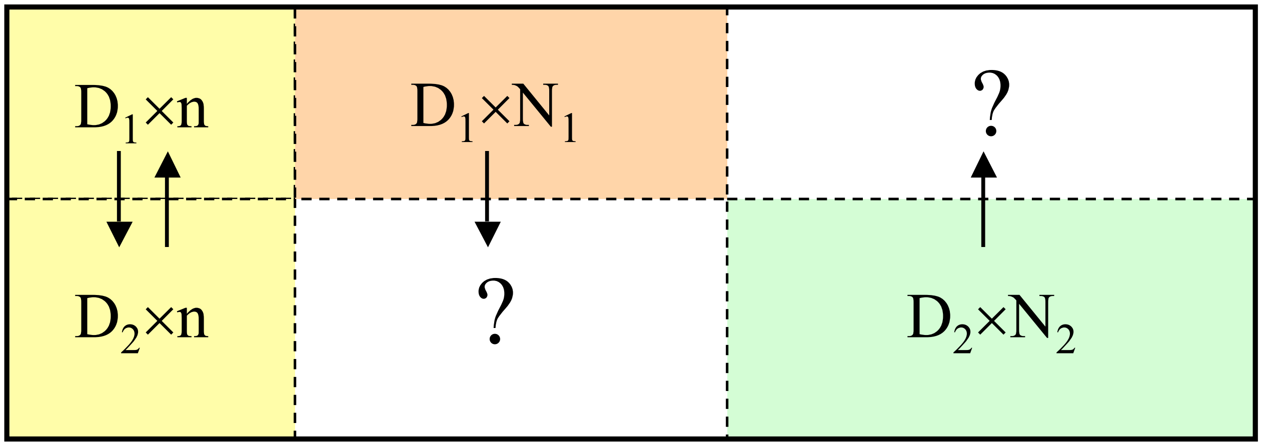
Learning from examples

Given Data (X_1,X_2):

	n labeled correspondences	N_1 examples of object 1 (D_1 dimensions)	N_2 examples of object 2 (D_2 dimensions)
X_1	$D_1 \times n$	$D_1 \times N_1$	?
X_2	$D_2 \times n$	?	$D_2 \times N_2$

- Matrix formulation ($n << N_1, N_2$)

Supervised learning



Training Data

Fill in the blanks:

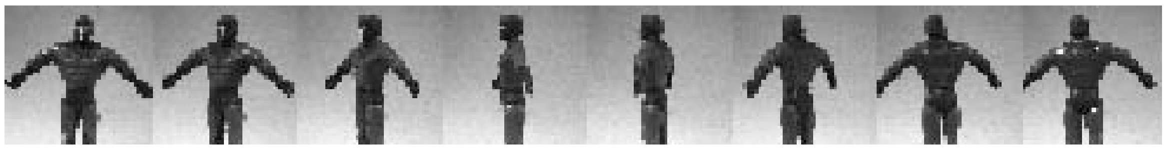
$(D_1 + D_2) \times n$ labeled data

$D_1 \times D_2$ parameters

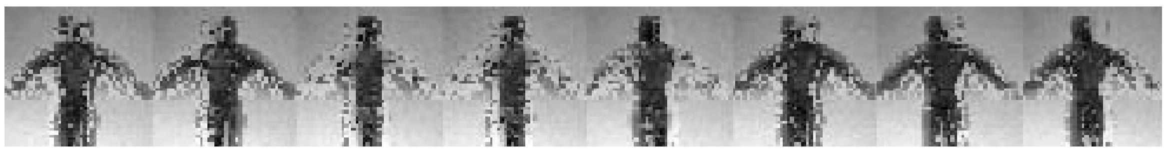
- Problem overfitting with small amount of labeled data

Supervised backprop network

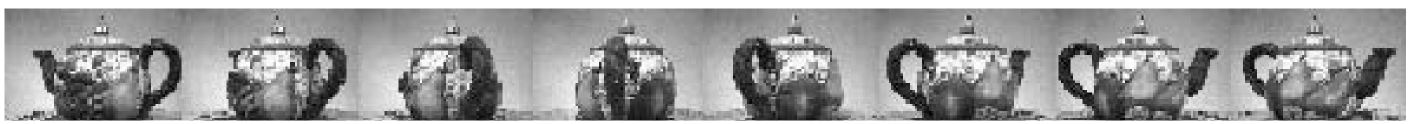
Original:



Reconstruction:



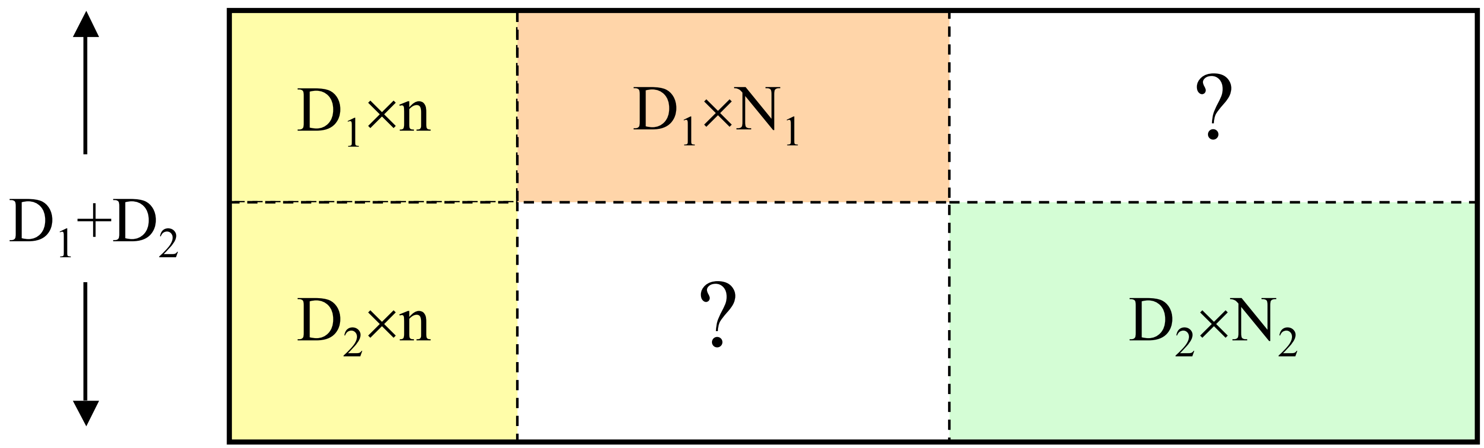
Original:



Reconstruction:

- 15 hidden units, tanh nonlinearity

Missing data



EM algorithm:

Iteratively fills in missing data statistics, reestimates parameters for PCA, factor analysis

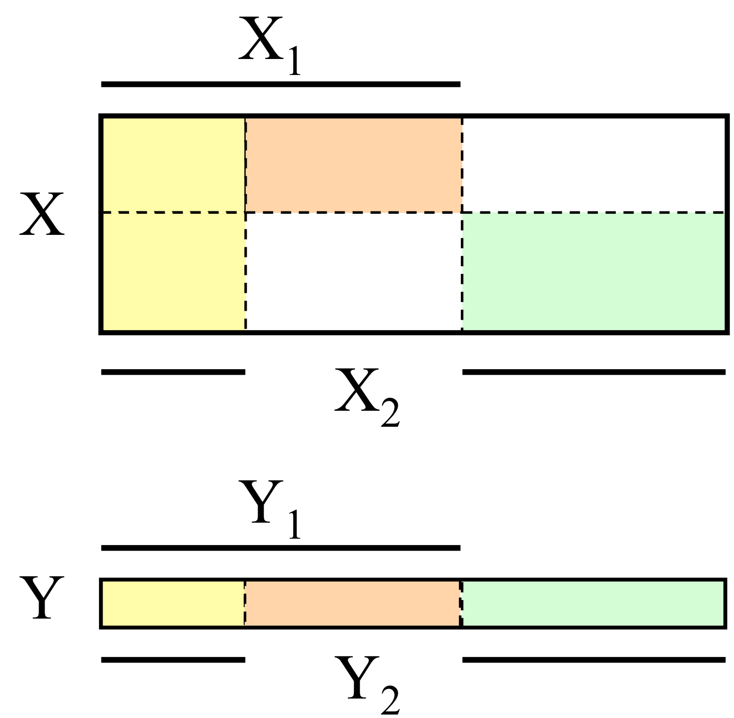
- Treat as missing data problem using EM algorithm

EM algorithm

$$\min_{Y_{ic}=Y_{jc}} \min_{W_i} \sum_i \left| X_i - W_i Y_i \right|^2$$

$$Y_i \leftarrow R Y_i \qquad W_i \leftarrow W_i R^{-1}$$

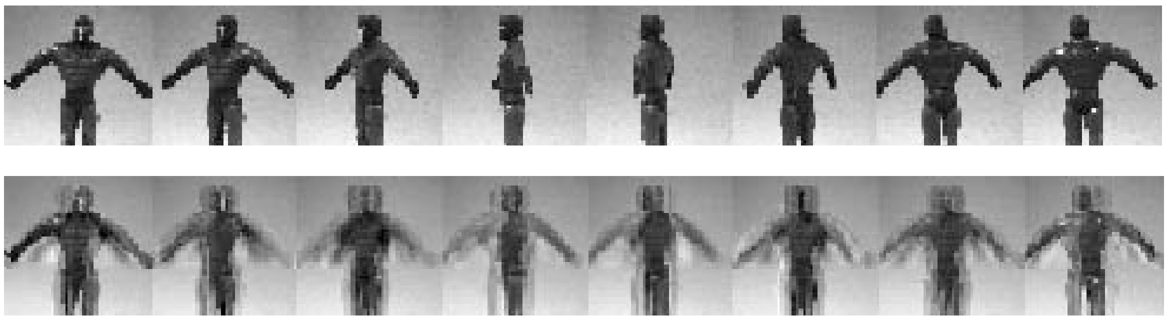
so that $\quad Y^T Y = I$



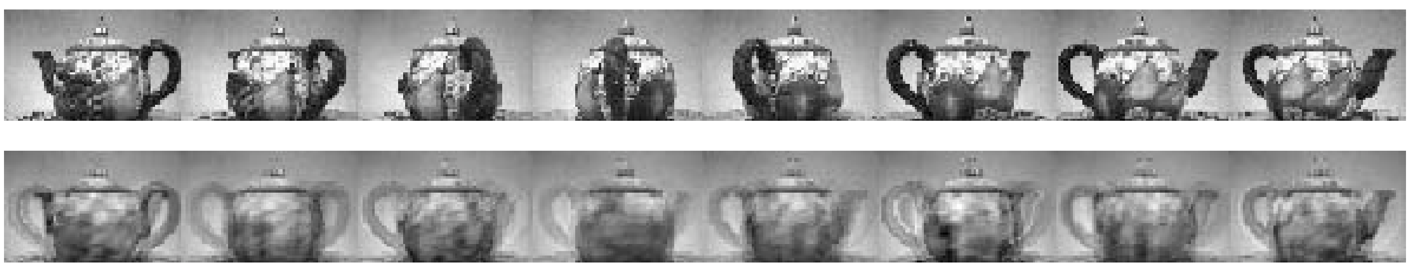
- Alternating minimization of least squares objective function.

PCA with correspondences

Original:



Original:

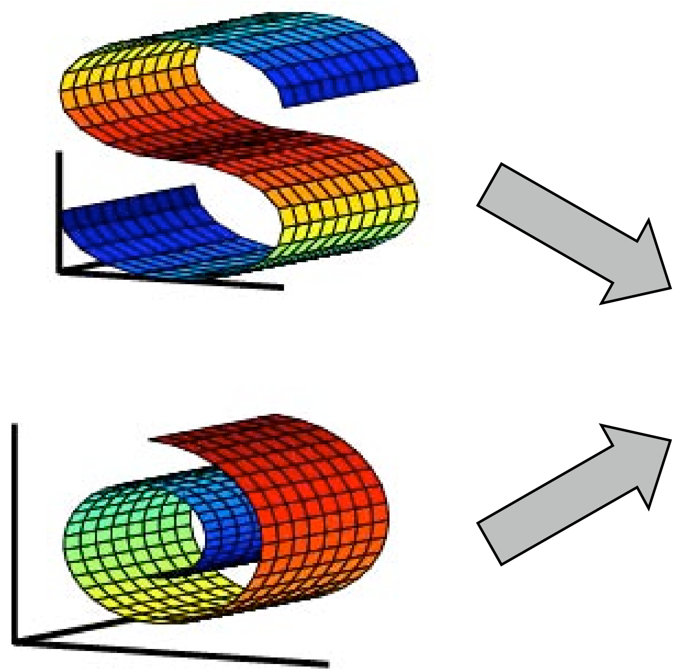


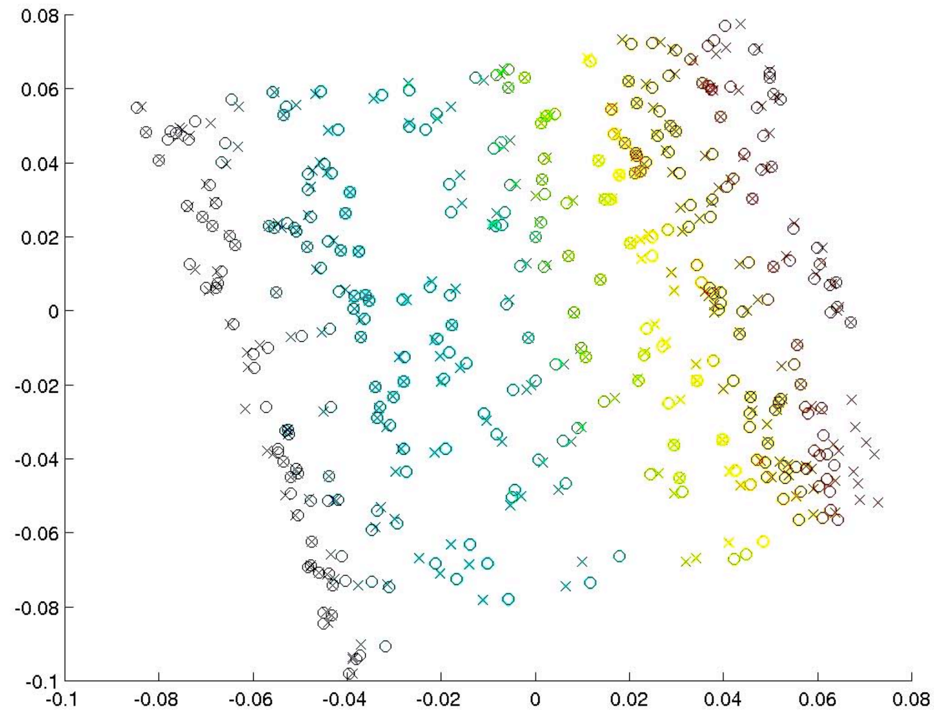
- 15 dimensional subspace, 200 images of each object, 10 in correspondence

Common embedding space

Two input spaces

Common low dimensional space





LLE with correspondences

$$E(W^1) = \sum_i \left| \vec{X}_i^1 - \sum_j W_{ij}^1 \vec{X}_j^1 \right|^2$$

$$E(W^2) = \sum_i \left| \vec{X}_i^2 - \sum_j W_{ij}^2 \vec{X}_j^2 \right|^2$$

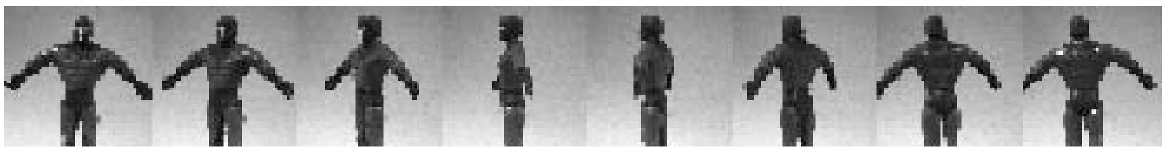
$$\Phi(Y^1, Y^2) = \sum_i \left| \vec{Y}_i^1 - \sum_j W_{ij}^1 \vec{Y}_j^1 \right|^2 + \sum_i \left| \vec{Y}_i^2 - \sum_j W_{ij}^2 \vec{Y}_j^2 \right|^2$$

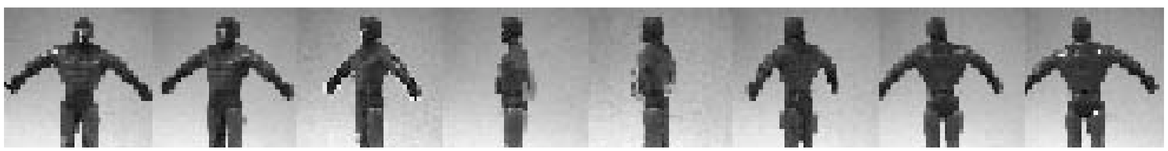
Correspondences: $i \in S_c : \quad \vec{Y}_i^1 = \vec{Y}_i^2$

- Quadratic optimization with constraints is solved with spectral decomposition

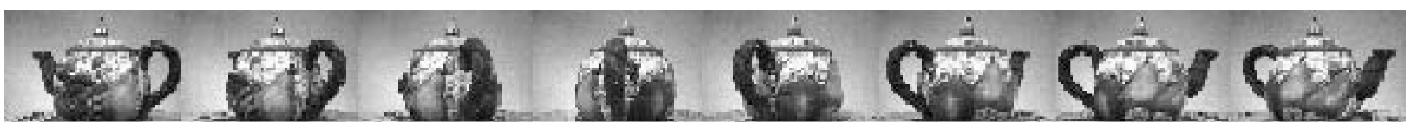
LLE with correspondences

Original:





Original:



- 8 nearest neighbors, 2 dimensional nonlinear manifold

Summary

- Adaptation and learning in biological systems for sensorimotor processing.
- Many sensory and motor activations are described by an underlying manifold structure.
- Development of learning algorithms that can incorporate this low dimensional manifold structure.
- Still much room for improvement…