Associative Memories and Applications: New Results

Humberto Sossa
Pattern Recognition Laboratory
Center for Computing Research - IPN
Av. Juan de Dios Bátiz esquina con M. Othón de Mendizábal
Colonia Nueva Industrial Vallejo
México, D. F. 07738, MEXICO

Outline

1. Presentation.
2. Introduction
3. Foundations
4. Examples of Associative Memories
5. Foundations of MEDMEMs
6. Numerical examples
7. Case of a general fundamental set
8. Differences with other models
9. Some applications
10. Conclusions and present research

Presentation

National Polytechnic Institute

200,000 people!

Schools

Research Centers

Center for Computing Research

600 people

Center for Computing Research

Research Unity in Computing Engineering	Research Unity in Computer Sciences
High Performance Computing Laboratory	Artificial Intelligence Laboratory
Digital Systems Laboratory	Natural Language Laboratory
Electronics Laboratory	Geo-processing Laboratory
Automatic Control Laboratory	Software Engineering Laboratory
Real Time Systems Laboratory	Information Systems Laboratory
Numerical Analysis Laboratory	**Pattern Recognition and Image Processing Laboratory**

Center for Computing Research

Pattern Recognition and Image Processing Laboratory

(40 people)

Researchers:

Dr. Humberto Sossa.

Dr. Edgardo Manuel Felipe.

Dr. Juan Luis Díaz de León.

Dr. Cornelio Yáñez.

M. Sc. Ricardo Barrón.

Plus Ph. D. and M. Sc. Students.

Areas of interest:

- Mathematical morphology.
- Pattern recognition.
- Associative memories.
- Invariants.
- Automatic object modeling.
- Object recognition.
- Medical image processing.
- Visual servoing.

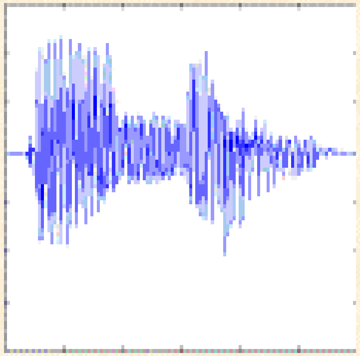
Introduction (1)

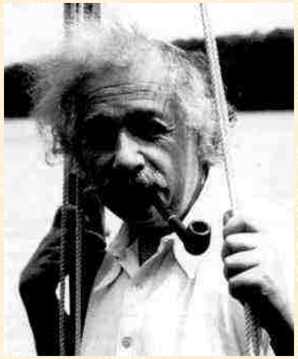
Two important problems in computer vision are:

- pattern classification, and

- pattern recall.

Examples of patterns:





Pattern classification: Given a set S of patterns and set of labels, assign an appropriate label to each $\mathbf{x} \in S$.

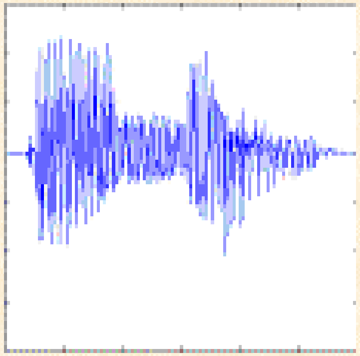
Introduction (2)

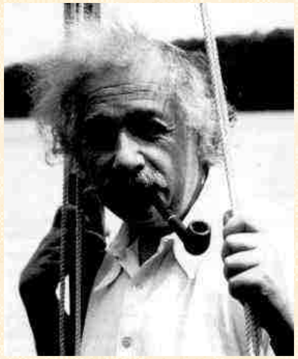
Two important problems in computer vision are:

- pattern classification, and

- pattern recall.

Examples of patterns:





Pattern classification: Given a set S of patterns and set L of labels, assign an appropriate label l to each $\mathbf{x} \in S$.

Pattern recall: Given a set S of patterns, recall a given pattern $\mathbf{y}$ given its key pattern $\mathbf{x}$ or a distorted version of it $\tilde{\mathbf{x}}$.

Introduction (3)

Approaches for pattern classification:

- Statistical approach
- Syntactical or structural approach
- Neural network approach
- Logical combinational approach
- Fuzzy logical approach

Introduction (4)

Approaches for pattern recall or pattern reconstruction:

- Neural networks ⇨ Associative memories

Introduction (5)

An associative memory, what is it?

Introduction (6)

An AM, What is it?

It is a device useful to associate patterns.

It is an input-output device that associates input patterns (**keys**) and output patterns (**patterns to be recalled or reconstructed**):

$$\mathbf{x} \longrightarrow \boxed{\mathbf{M}} \longrightarrow \mathbf{y}$$

Introduction (7)

Example: Build a device able to memorize and the recall the following three associations:

10010 ⟶ 00101

01100 ⟶ 01010

10001 ⟶ 10100

Three main questions:

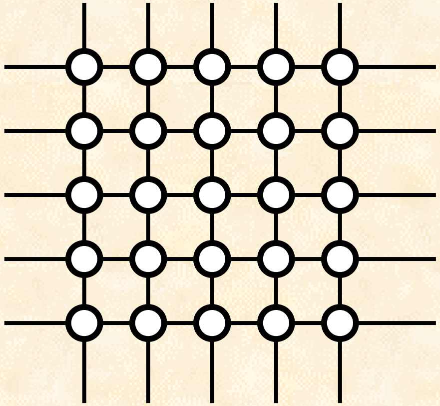
1. Structure of the memory?

2. How the patterns are saved to memory?

3. How the patterns are recalled from memory?

Introduction (8)

1. Structure: We begin with a 5x5 matrix full of 0's:



Full of 0's because we do not have yet any thing to recall!

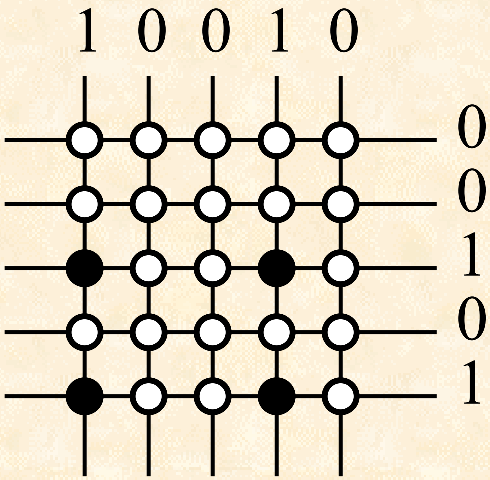
Introduction (9)

2. Learning the associations:

$10010 \longrightarrow 00101$

$01100 \longrightarrow 01010$

$10001 \longrightarrow 10100$



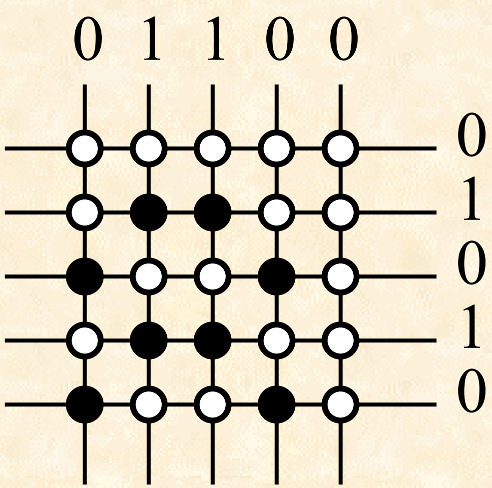
Introduction (10)

2. Learning the associations:

10010 → 00101

01100 → 01010

10001 → 10100



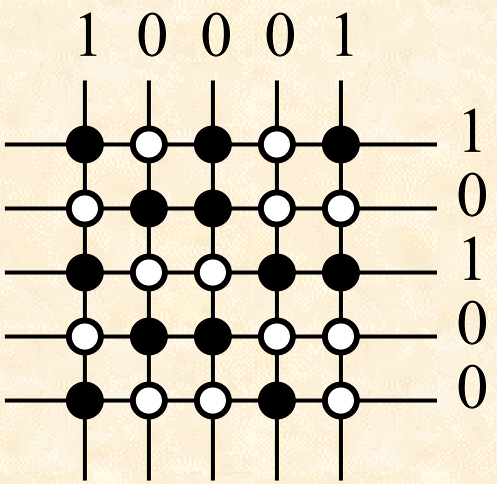
Introduction (11)

2. Learning the associations:

10010 → 00101

01100 → 01010

10001 → 10100

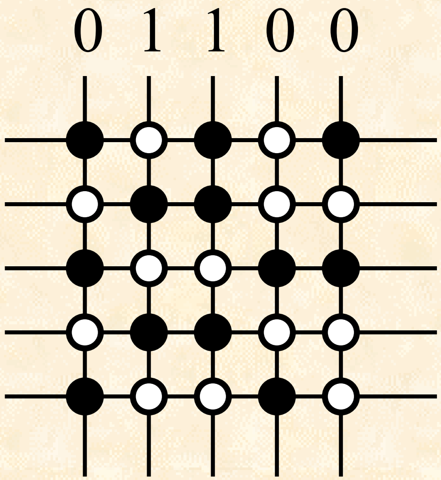


The information of the three associations is recorded in the same arrangement!

Introduction (12)

10010 → 00101
01100 → 01010
10001 → 10100

3. Recall of a pattern:



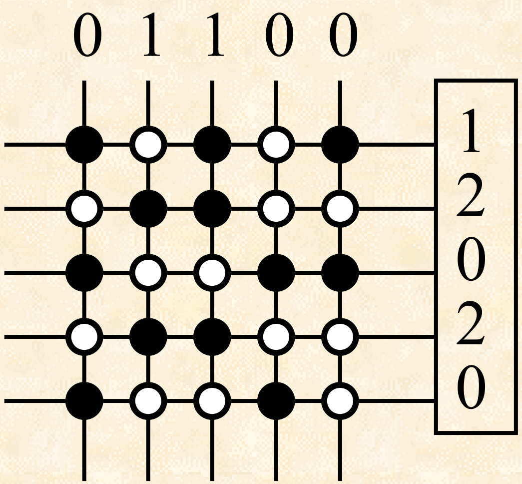
Introduction (13)

$10010 \longrightarrow 00101$

$01100 \longrightarrow 01010$

$10001 \longrightarrow 10100$

3. Recall of a pattern:



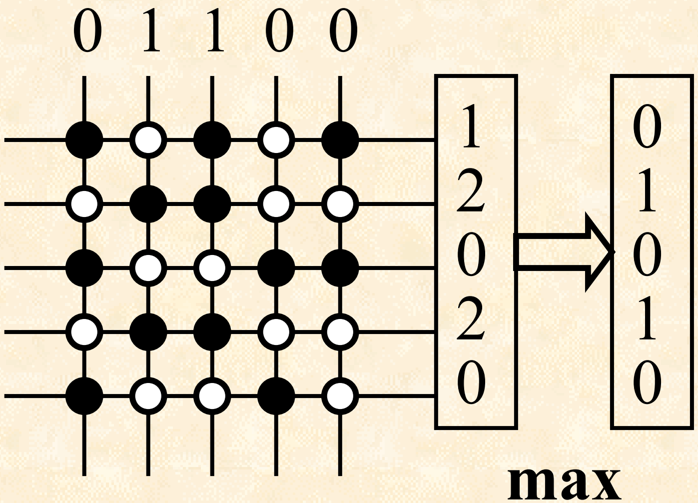
Introduction (14)

10010 ⟶ 00101

01100 ⟶ 01010

10001 ⟶ 10100

3. Recall of a pattern:

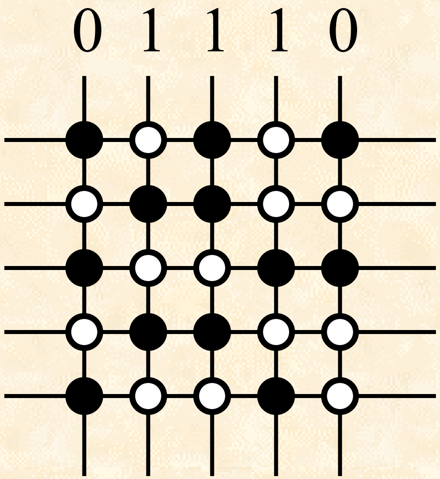


Take the **max** and convert it to 1,
the remaining to 0.

Introduction (15)

10010 → 00101
01100 → 01010
10001 → 10100

3. Recalling with noise:

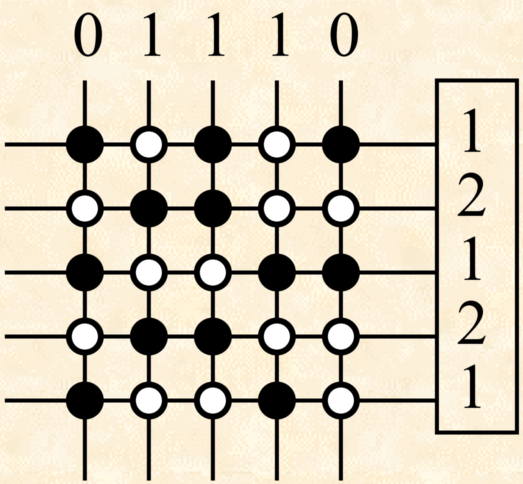


Take the **max** and convert it to 1,
the remaining to 0.

Introduction (16)

10010 → 00101
01100 → 01010
10001 → 10100

3. Recalling with noise:



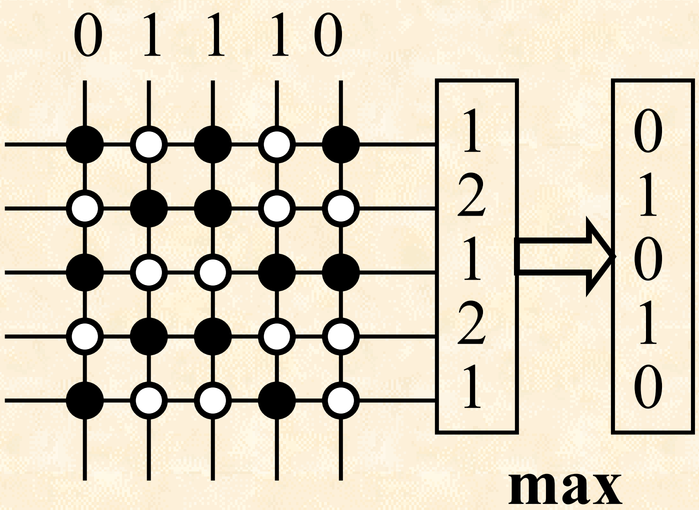
Introduction (16)

10010 → 00101

01100 → 01010

10001 → 10100

3. Recalling with noise:



Take the **max** and convert it to 1,
the remaining to 0.

Foundations (1)

A pattern is represented as:

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \qquad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}$$

Components can binary numbers, integers, reals,…

Foundations (2)

Each pattern **x** forms an association with a corresponding output pattern **y**.

An **association** between input pattern **x** and output pattern **y** is denoted as (**x**,**y**).

For *k* integer and positive, the corresponding association will be denoted as $\left(\mathbf{x}^k, \mathbf{y}^k\right)$.

The associative memory **M** is represented by a matrix whose *ij*-th component is m_{ij}.

M is generated from a finite a priori set of known associations, known as the *fundamental set of associations*, or simply the *fundamental set* (FS).

Foundations (3)

If ξ is an index, the fundamental set (FS) is represented as:

$$\left\{\left(\mathbf{x}^{\xi}, \mathbf{y}^{\xi}\right) \mid \xi = 1,2,\ldots, p\right\}$$

with p the cardinality of the set.

The patterns that form the fundamental set are called *fundamental patterns*.

If it holds that $\mathbf{x}^{\xi} = \mathbf{y}^{\xi} \; \forall \, \xi \in \{1,2,\ldots p\}$ **M** is auto-associative.

otherwise it is hetero-associative.

Foundations (4)

A distorted version of a pattern **x** to be recalled will be denoted as: $\tilde{\mathbf{x}}$

If when feeding a distorted version of $\mathbf{x}^w$ with $w \in \{1,2,\ldots,p\}$

to an associative memory **M**, it happens that the output corresponds exactly to the associated pattern $\mathbf{y}^w$, we say that recalling is perfect.

If this hold for all *w*, **M** has perfect recall.

Foundations (5)

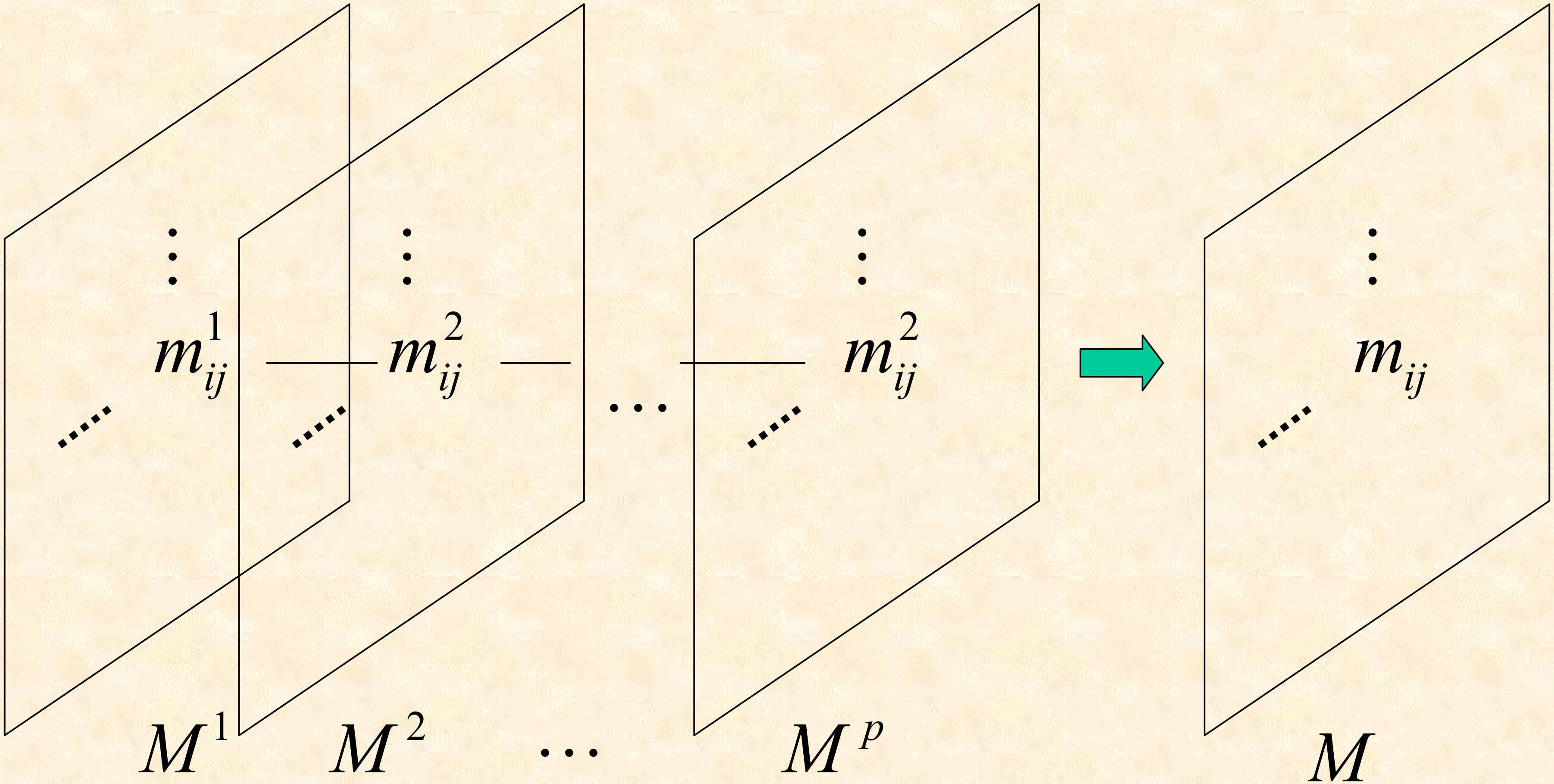
Learning:

$$
\begin{array}{c|ccccc|}
 & x_1^k & x_2^k & \cdots & x_j^k & \cdots & x_n^k \\
\hline
y_1^k & & & & & & \\
y_2^k & & & & \vdots & & \\
\vdots & & & & & & \\
y_i^k & & \cdots & & m_{ij}^k & \cdots & \\
\vdots & & & & \vdots & & \\
y_m^k & & & & & & \\
\hline
\end{array}
$$

$$M^k$$

Foundations (6)

Learning:



Foundations (7)

Pattern recalling:

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \Rightarrow \boxed{M} \Rightarrow \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{pmatrix}$$

Features of an associative memory:

They are very simple devices. They have one single layer.

Training is in one epoch. No convergence is required.

Conditions for perfect recall can be easily derived.

Examples of models of AM's:

K. Steinbuch, Die Lermatrix, *Kybernetik*, 1(1):26-45, 1961.

D. Wilshaw et al. Non-holographic associative memory, *Nature* 222:960-962, 1969.

J. A. Anderson, A simple neural network generating an interactive memory, *Mathematical Biosciences*, 14:197-220, 1972.

T. Kohonen, Correlation matrix memories, *IEEE Transactions on Computers*, 21(4):353-359, 1972.

J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, *Proceedings of the National Academy of Sciences*, 79: 2554-2558, 1982.

G. Palm. Neural Assemblies. An Alternative Approach to Artificial Intelligen ce. Springer, Berlin, Heidelberg, New York, 1982.

Examples of models of AM's:

B. Kosko. Bidirectional associative memories. *IEEE Transactions on SMC* 18(1):49-60. 1988.

G. X. Ritter el al. Morphological associative memories, *IEEE Transactions on Neural Networks*, 9:281-293, 1998.

G. X. Ritter, et al. Morphological bi-directional associative memories, *Neural Networks*, 12:851-867, 1999.

C. Yáñez, Associative Memories based on Order Relations and Binary Operators (In Spanish), PhD Thesis, Center for Computing Research, February of 2002.

P. Sussner. Generalizing operations of binary auto-associative morphological memories using fuzzy set theory. *Journal of Mathematical Imaging and Vision*, 19:81-93, 2003.

Examples of models of AM's:

H. Sossa et al. Extended Associative Memories for Recalling Gray Level Patterns. Lecture Notes on Computer Science 3287. Springer Verlag. Pp. 187-194. 2004.

H. Sossa et al. Binary Associative Memories applied to Gray Level Pattern Recalling. Lecture Notes on Artificial Intelligence 3315. Springer Verlag. Pp. 656-666. 2004.

H. Sossa et al. Associative gray-level pattern processing using binary decomposition and a-b memories. To appear in *Neural Processing Letters*. 2005.

Foundations on MEDMEMS (1):

Two kind of associative memories are proposed in:

H. Sossa et al. New Associative Memories to Recall Real-Valued Patterns. Lecture Notes on Computer Science 3287. Springer Verlag. Pp. 195-202. 2004.

H. Sossa et al. Median Associative Memories: New Results. Submitted to CIARP 2005.

One hetero-associative and one auto-associative.

Only hetero-associative case is studied.

Foundations on MEDMEMS (2)

One hetero-associative memory is described. Let us call HAM-memory of type **M**. TRAINING PHASE:

Step 1: For each ξ=1,2,…,p, from each couple $\left(\mathbf{y}^{\xi},\mathbf{x}^{\xi}\right)$

build matrix:

$$\mathbf{M}^{\xi}=\mathbf{y}\lozenge_{\mathrm{A}}\mathbf{x}^{t}=\begin{pmatrix} \mathrm{A}(y_1,x_1) & \mathrm{A}(y_1,x_2) & \cdots & \mathrm{A}(y_1,x_n) \\ \mathrm{A}(y_2,x_1) & \mathrm{A}(y_2,x_2) & \cdots & \mathrm{A}(y_2,x_n) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{A}(y_m,x_1) & \mathrm{A}(y_m,x_2) & \cdots & \mathrm{A}(y_m,x_n) \end{pmatrix}_{m\times n}$$

$$\mathrm{A}\left(y_i^{\xi},x_j^{\xi}\right)=y_i^{\xi}-x_j^{\xi}$$

Step 2: Obtain matrix **M** as: $\mathbf{M}=\underset{\xi=1}{\overset{p}{\mathbf{med}}}\left[\mathbf{M}^{\xi}\right]$

The ij-th component **M** is given as $m_{ij}=\underset{\xi=1}{\overset{p}{\mathbf{med}}}\,\mathrm{A}\left(y_i^{\xi},x_j^{\xi}\right)$

Foundations on MEDMEMS (3)

RECALLING PHASE: We have two cases, i.e.:

Case 1: Recall of a fundamental pattern $\mathbf{y}^w$. A pattern $\mathbf{x}^w$, with $w \in \{1,2,\cdots,p\}$ is presented to the memory **M** and the following operation is done:

$$\mathbf{M}\Diamond_{\mathrm{B}}\mathbf{x}^w$$

The result is a column vector of dimension n, with i-th component given as:

$$\left(\mathbf{M}\Diamond_{\mathrm{B}} x^w\right)_i = \underset{j=1}{\overset{n}{\mathbf{med}}}\,\mathrm{B}\left(m_{ij}, x_j^w\right) \qquad \mathrm{B}\left(y_i^\xi, x_j^\xi\right) = y_i^\xi + x_j^\xi$$

Foundations on MEDMEMS (4)

Case 2: Recall of a pattern from an altered version of its key.

A pattern $\tilde{\mathbf{x}}$ (altered version of a pattern $\mathbf{x}^w$ is presented to the auto-associative memory **M** and the following operation is done:

$$\mathbf{M}\lozenge_{\mathrm{B}}\tilde{\mathbf{x}}$$

Again, the result is a column vector of dimension n, with i-th component given as:

$$\left(\mathbf{M}\lozenge_{\mathrm{B}}\tilde{\mathbf{x}}\right)_i = \underset{j=1}{\overset{n}{\mathbf{med}}}\,\mathrm{B}\left(m_{ij},\tilde{x}_j\right)$$

$$\mathrm{B}\left(y_i^{\xi},x_j^{\xi}\right) = y_i^{\xi} + x_j^{\xi}$$

Foundations on MEDMEMS (6)

The following proposition provides the conditions for perfect recall of a pattern of the FS:

Proposition 1. Let $\left\{\left(\mathbf{x}^{\alpha}, \mathbf{y}^{\alpha}\right) \mid \alpha = 1,2,\ldots,p\right\}$ with $\mathbf{x}^{\alpha} \in \mathbf{R}^{n}, \mathbf{y}^{\alpha} \in \mathbf{R}^{m}$ the fundamental set of an AS-memory **M** and let $\left(\mathbf{x}^{\gamma}, \mathbf{y}^{\gamma}\right)$ an arbitrary fundamental couple with $\gamma = 1, \cdots, p$

If $\operatorname*{\mathbf{med}}_{j=1}^{n} \varepsilon_{ij} = 0$, $i = 1, \cdots, m$, $\varepsilon_{ij} = m_{ij} - \mathrm{A}\left(y_i^{\gamma}, x_j^{\gamma}\right)$ then

$$\left(\mathbf{M} \Diamond_{\mathrm{B}} \mathbf{x}^{\gamma}\right)_i = \mathbf{y}_i^{\gamma}, i = 1 \ldots m$$

Foundations on MEDMEMS (7)

More restricted conditions are given by the following (special case):

Corollary 1. Let $\{(\mathbf{x}^{\alpha}, \mathbf{y}^{\alpha}) \mid \alpha = 1,2,\ldots,p\}$, $\mathbf{x}^{\alpha} \in \mathbf{R}^{n}$, $\mathbf{y}^{\alpha} \in \mathbf{R}^{m}$

An AS-median memory **M** has perfect recall if for all $\alpha = 1,\cdots,p$

$$\mathbf{M}^{\alpha} = \mathbf{M}.$$

Foundations on MEDMEMS (8)

The following proposition provides the conditions for perfect recall of a pattern of the FS in terms of an altered version of it:

Proposition 2. Let $\left\{\left(\mathbf{x}^{\alpha}, \mathbf{y}^{\alpha}\right) \mid \alpha = 1,2,\ldots,p\right\}$, $\mathbf{x}^{\alpha} \in \mathbf{R}^{n}$, $\mathbf{y}^{\alpha} \in \mathbf{R}^{m}$

a FS with perfect recall.

Let $\eta^{\alpha} \in \mathbf{R}^{n}$ a pattern of mixed noise.

An AS-memory **M** has perfect recall in the presence of mixed noise if this noise is of median zero, this is if

$$\operatorname*{\mathbf{med}}_{j=1}^{n} \eta_{j}^{\alpha} = 0, \forall \alpha$$

Numerical Examples (1)

Example 1. Recalling the FS. Suppose we want to first memorize and then recall the following FS:

$$\mathbf{x}^1 = \begin{pmatrix} 0.1 \\ 0.0 \\ 0.2 \end{pmatrix} \quad \mathbf{y}^1 = \begin{pmatrix} 0.2 \\ 0.3 \\ 0.3 \\ 0.4 \end{pmatrix} \quad \mathbf{x}^2 = \begin{pmatrix} 0.4 \\ 0.3 \\ 0.5 \end{pmatrix} \quad \mathbf{y}^2 = \begin{pmatrix} 0.5 \\ 0.6 \\ 0.6 \\ 0.7 \end{pmatrix} \quad \mathbf{x}^3 = \begin{pmatrix} 0.7 \\ 0.6 \\ 0.8 \end{pmatrix} \quad \mathbf{y}^3 = \begin{pmatrix} 0.8 \\ 0.9 \\ 0.9 \\ 1.0 \end{pmatrix}$$

TRAINING PHASE: We can verify that Proposition 1 and Corollary 1 hold for this example, thus:

$$A\left(y_i^\xi, x_j^\xi\right) = y_i^\xi - x_j^\xi$$

$$\mathbf{M} = \begin{pmatrix} 0.1 & 0.2 & 0.0 \\ 0.2 & 0.3 & 0.1 \\ 0.2 & 0.3 & 0.1 \\ 0.3 & 0.4 & 0.2 \end{pmatrix}$$

A(0.4,0.1)=0.4-0.1=0.3

Numerical Examples (2)

RECALLING PHASE: The whole FS is recalled!

For example: $$\mathrm{B}\left(y_i^{\xi}, x_j^{\xi}\right) = y_i^{\xi} + x_j^{\xi}$$

$$\mathbf{M}\Diamond_{\mathrm{B}}\mathbf{x}^1 = \begin{pmatrix} 0.1 & 0.2 & 0.0 \\ 0.2 & 0.3 & 0.1 \\ 0.2 & 0.3 & 0.1 \\ 0.3 & 0.4 & 0.2 \end{pmatrix} \Diamond_{\mathrm{B}} \begin{pmatrix} 0.1 \\ 0.0 \\ 0.2 \end{pmatrix} = \begin{pmatrix} \mathbf{med}[\mathrm{B}(0.1,0.1), \mathrm{B}(0.2,0.0), \mathrm{B}(0.0,0.2)] \\ \mathbf{med}[\mathrm{B}(0.2,0.1), \mathrm{B}(0.3,0.0), \mathrm{B}(0.1,0.2)] \\ \mathbf{med}[\mathrm{B}(0.2,0.1), \mathrm{B}(0.3,0.0), \mathrm{B}(0.1,0.2)] \\ \mathbf{med}[\mathrm{B}(0.3,0.1), \mathrm{B}(0.4,0.0), \mathrm{B}(0.2,0.2)] \end{pmatrix} = \begin{pmatrix} \mathbf{med}(0.2,0.2,0.2) \\ \mathbf{med}(0.3,0.3,0.3) \\ \mathbf{med}(0.3,0.3,0.3) \\ \mathbf{med}(0.4,0.4,0.4) \end{pmatrix} = \begin{pmatrix} 0.2 \\ 0.3 \\ 0.3 \\ 0.4 \end{pmatrix}$$

$$\mathbf{x}^1 = \begin{pmatrix} 0.1 \\ 0.0 \\ 0.2 \end{pmatrix} \qquad \mathbf{y}^1 = \begin{pmatrix} 0.2 \\ 0.3 \\ 0.3 \\ 0.4 \end{pmatrix}$$

Numerical Examples (3)

If a FS satisfies the conditions imposed by Proposition 1, and the level noise added to a key-pattern $\mathbf{x}^{\alpha}$ of this FS satisfies Proposition 2, then **no matter the level of noise added**, the pattern is perfectly recalled.

Example 2. Suppose we want to recall $\mathbf{y}^1 = \begin{pmatrix} 0.2 \\ 0.3 \\ 0.3 \\ 0.4 \end{pmatrix}$ given the following distorted version of its key:

$$\tilde{\mathbf{x}}^1 = \begin{pmatrix} 9.1 \\ 0.0 \\ -9.2 \end{pmatrix} \longleftrightarrow \mathbf{x}^1 = \begin{pmatrix} 0.1 \\ 0.0 \\ 0.2 \end{pmatrix}$$

The median of the noise added to **x** equals 0:
med(9.0,0.0,-9.4)=
med(-9.4,0.0,9.0)= 0.0.

Note the level of noise introduced.

Perfect recall of the pattern **should** be obtained!!!!

Numerical Examples (4)

RECALLING PHASE: Let us see:

$$\mathbf{M}\Diamond_{B}\mathbf{x}^{1}=\begin{pmatrix}0.1 & 0.2 & 0.0\\ 0.2 & 0.3 & 0.1\\ 0.2 & 0.3 & 0.1\\ 0.3 & 0.4 & 0.2\end{pmatrix}\Diamond_{B}\begin{pmatrix}9.2\\ 0.0\\ -9.2\end{pmatrix}=\begin{pmatrix}\mathbf{med}\left[B(0.1,9.2),B(0.2,0.0),B(0.0,-9.2)\right]\\ \mathbf{med}\left[B(0.2,9.2),B(0.3,0.0),B(0.1,-9.2)\right]\\ \mathbf{med}\left[B(0.2,9.2),B(0.3,0.0),B(0.1,-9.2)\right]\\ \mathbf{med}\left[B(0.3,9.2),B(0.4,0.0),B(0.2,-9.2)\right]\end{pmatrix}=\begin{pmatrix}\mathbf{med}(9.3,\underline{0.2},-9.2)\\ \mathbf{med}(9.4,\underline{0.3},-9.1)\\ \mathbf{med}(9.4,\underline{0.3},-9.1)\\ \mathbf{med}(9.5,\underline{0.4},9.0)\end{pmatrix}=\begin{pmatrix}0.2\\ 0.3\\ 0.3\\ 0.4\end{pmatrix}$$

$$\mathbf{y}^{1}=\begin{pmatrix}0.2\\ 0.3\\ 0.3\\ 0.4\end{pmatrix}$$

Differences with other models (1):

The most similar model is the one proposed in: G. X. Ritter el al. Morphological associative memories, *IEEE Transactions on Neural Networks*, 9:281-293, 1998.

The authors make use of well known **max** and **min** operations on sets.

Ritter et al. define two AS: **M** and **W**. Both operate in the hetero and the auto-associative modes.

Differences with other models (2):

M memories use **max** operation for training and **min** for recalling.

M memories are good with additive noise.

W memories use **min** operation for training and **max** for recalling.

To cope with mixed the so called kernel-method should introduced.

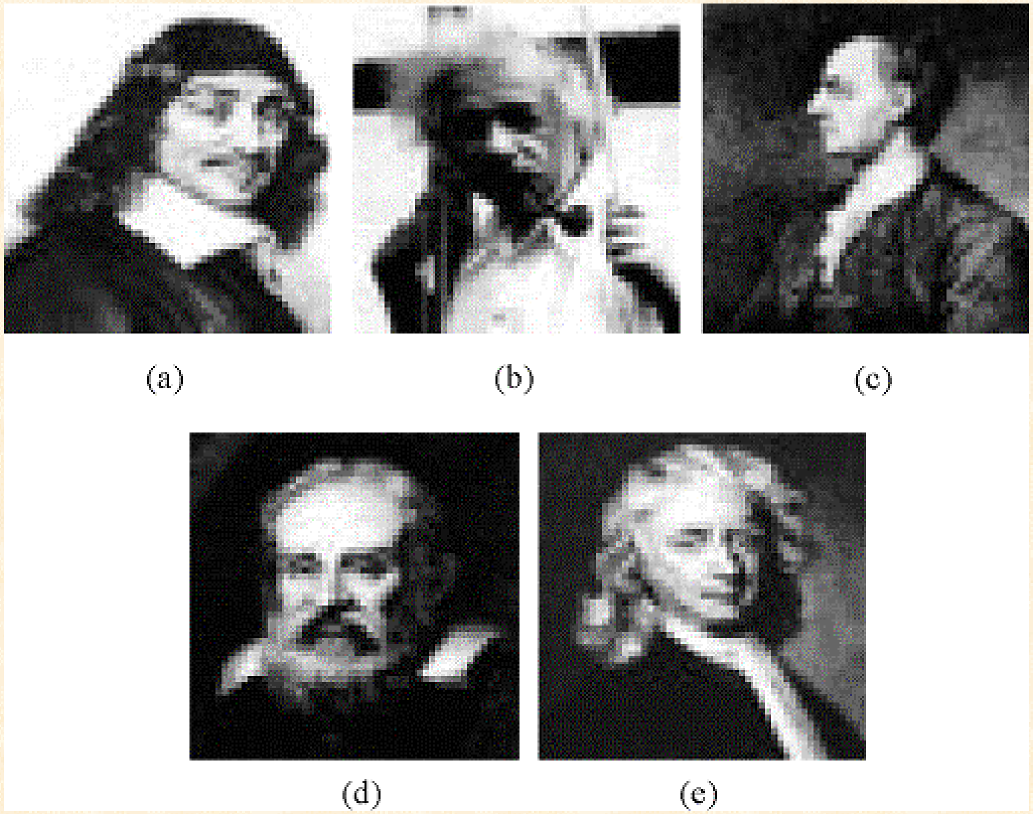
MEDMEMS use **median** operator for training and recalling.

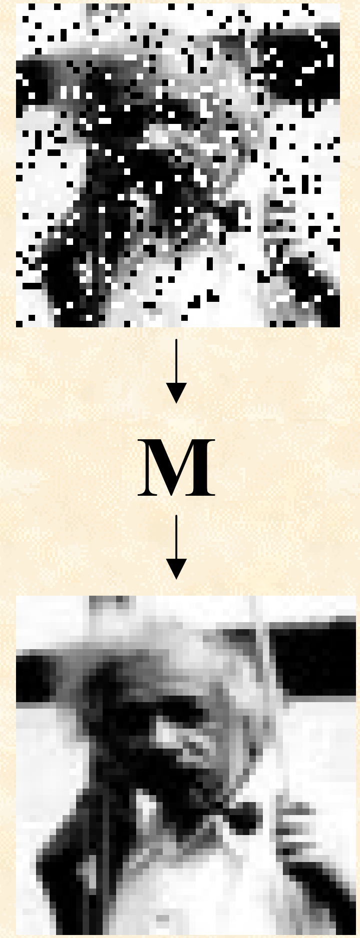
In the general case they need to use <u>differential equalization</u> for perfect recall. (H. Sossa et al. New Associative Memories to Recall Real-Valued Patterns. Lecture Notes on Computer Science 3287. Springer Verlag. Pp. 195-202. 2004).

Some applications of MEDMEMs:

1. Pattern reconstruction.
2. Object classification.
3. Object detection under occlusions.
4. Word reconstruction.

Pattern reconstruction (1):





Pattern reconstruction (2):

Recalling the FS:

Of course the FS is perfectly reconstructed.

Pattern reconstruction (3):

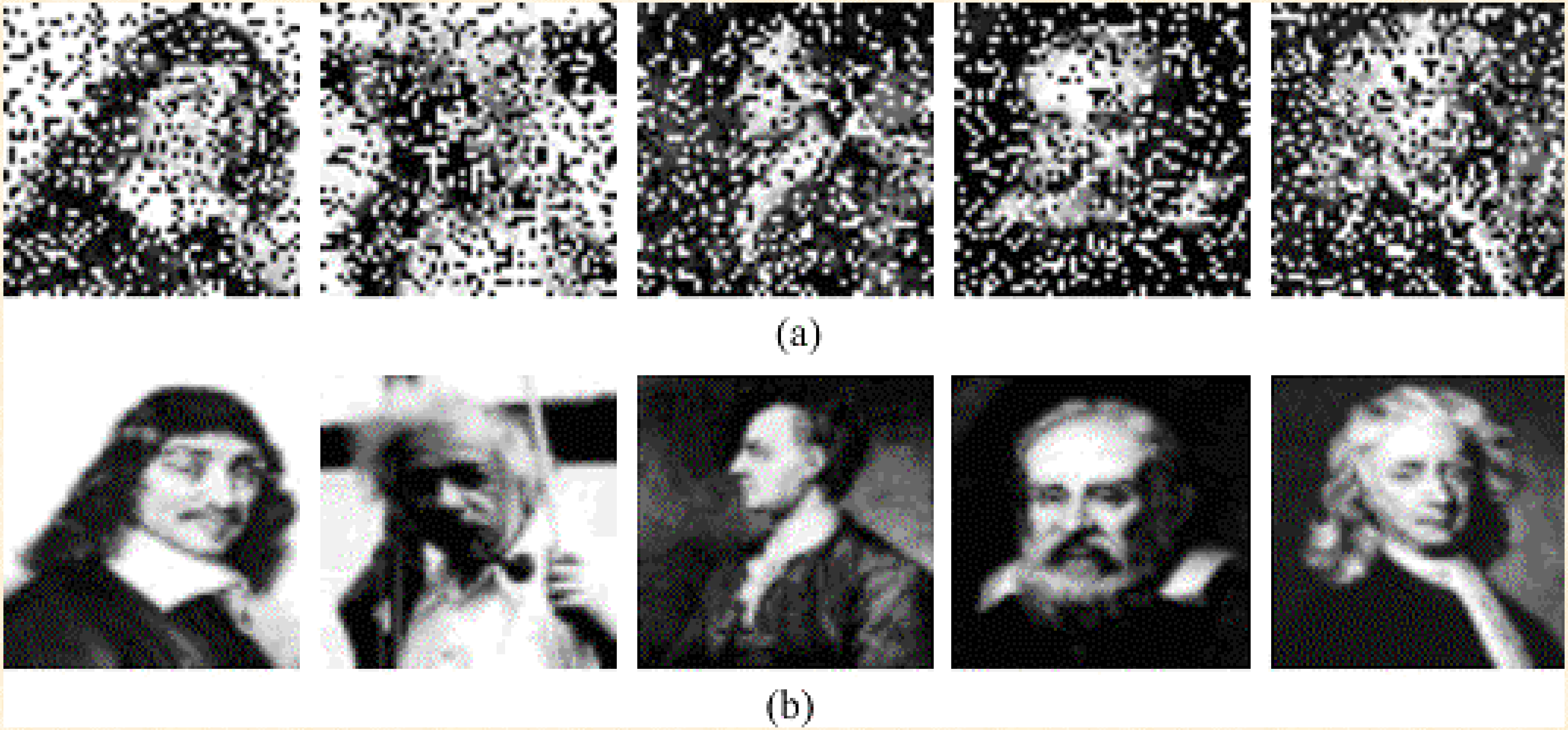
Recalling a pattern of the FS from a noisy version of its key:

Fact: The added noise rarely satisfies Proposition 2.

Proposition 3. Let $\left\{\left(\mathbf{x}^{\alpha},\mathbf{y}^{\alpha}\right) \mid \alpha = 1,2,\ldots,p\right\}$, $\mathbf{x}^{\alpha} \in \mathbf{R}^{n}$, $\mathbf{y}^{\alpha} \in \mathbf{R}^{m}$ a FS. Without lost of generality suppose that is p odd. Thus the associative memory $\mathbf{M} = \mathbf{y}^{\xi} \Diamond_{\mathrm{A}} \left(\mathbf{x}^{\xi}\right)^{T}$

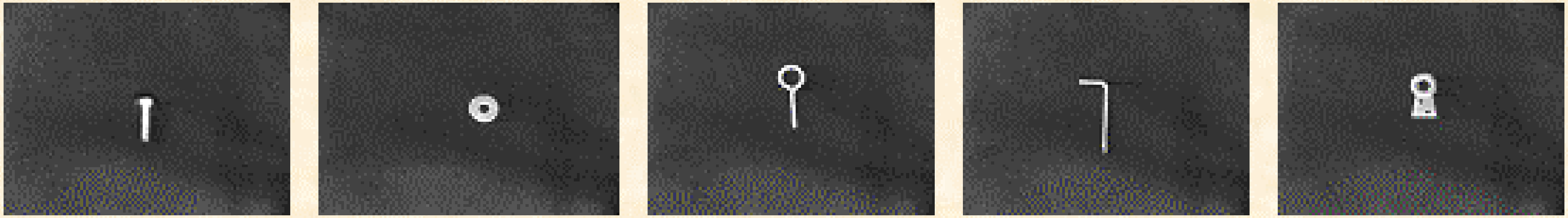
has perfect recall in the presence of noise if less than 50% $\frac{n+1}{2} - 1$ of the elements of any of the input patterns are distorted by mixed noise.

Pattern reconstruction (4):



Object classification (1)

In this section, the proposal is used to recall the class index of an object:



(a) A bolt. (b) A washer. (c) An eyebolt. (d) A hook. (e) A dovetail.

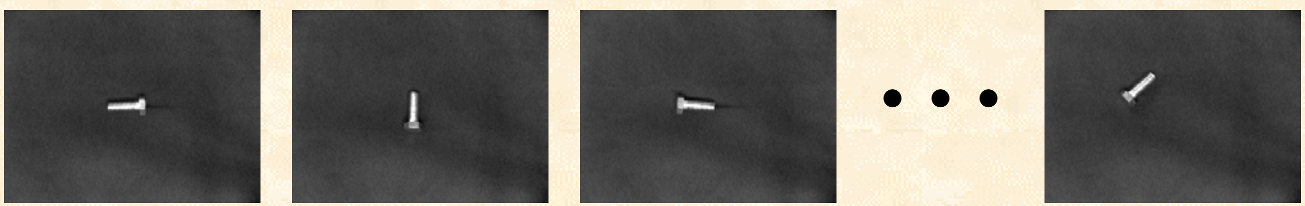
<u>Object classification (2)</u>

<u>BUILDING MATRIX **M**</u>:

Objects <u>were not</u> recognized directly form their images.

They are recognized by invariants description of them.

In both cases 20 images of each object in different positions, rotations and scale changes were used to build matrix **M**.

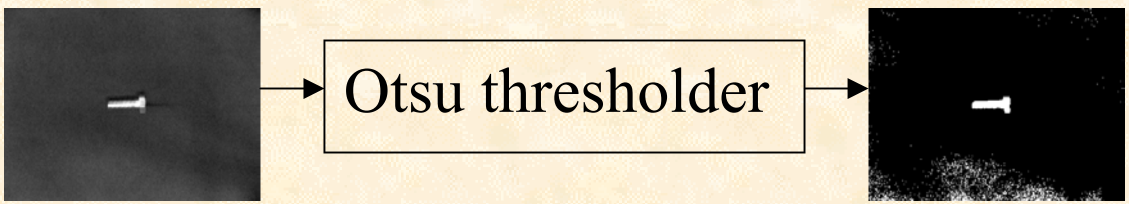


1 2 3 • • • 20

Object classification (3)

BUILDING MATRIX **M**:

To each image of each object a standard thresholder was applied to get its binary version.



Small spurious regions were eliminated form each image by means of a size filter (eliminates regions smaller than a given threshold).

Object classification (4)

To each of the 20 images of each object (class) the first 3 well-known Hu geometric invariants, to translations, rotations and scale changes, were computed:

3 Hu invariants

$$\phi_1 = \mu_{20} + \mu_{02}$$

$$\phi_2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}$$

$$\phi_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2$$

$$\vdots$$

Object classification (5)

$$\mathbf{x}^1 = \begin{pmatrix} 0.4429 \\ 0.1594 \\ 0.0058 \end{pmatrix} \quad \mathbf{y}^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \mathbf{x}^2 = \begin{pmatrix} 0.1896 \\ 5.78E-5 \\ 4.14E-6 \end{pmatrix} \quad \mathbf{y}^2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \mathbf{x}^3 = \begin{pmatrix} 0.7038 \\ 0.2911 \\ 0.1825 \end{pmatrix} \quad \mathbf{y}^3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

$$\mathbf{x}^4 = \begin{pmatrix} 1.1421 \\ 1.5517 \\ 0.8467 \end{pmatrix} \quad \mathbf{y}^4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \quad \mathbf{x}^5 = \begin{pmatrix} 0.2491 \\ 0.0195 \\ 2.41E-5 \end{pmatrix} \quad \mathbf{y}^5 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Object classification (6)

$$\mathbf{M} = \begin{pmatrix} 0.5571 & 0.8406 & 0.9942 \\ -0.4429 & -0.1594 & -0.0058 \\ -0.4429 & -0.1594 & -0.0058 \\ -0.4429 & -0.1594 & -0.0058 \\ -0.4429 & -0.1594 & -0.0058 \end{pmatrix}$$

Object classification (7)

Recalling of a pattern by a corrupted version of its key:

In practical applications the noise added to the values of the patterns do not satisfies Propositions 2 and 3.

To cope with this situation, we propose the following strategy: Index class of an object is recalled as follows:

$$\mathbf{y}^j, j = \arg\min_i \left(\min_{k=1}^{m} \left(\left| y_k - y_k^i \right| \right) \right)$$

Object classification (7)

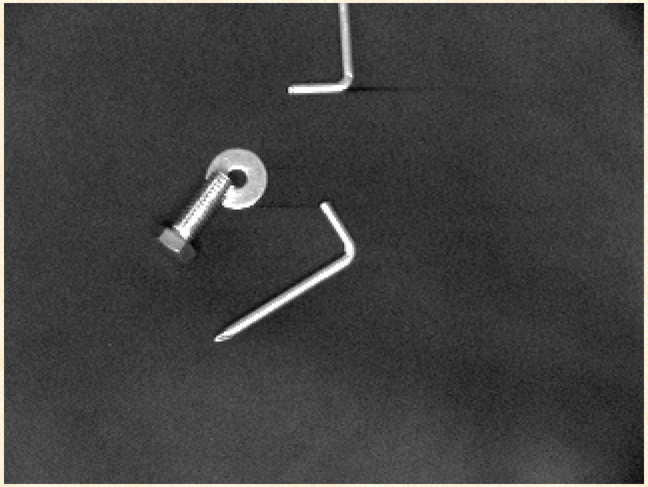
500 images (100 for each object), and different from those used to build matrix **M** were used to measure the efficiency of the proposal. Of course the values of the invariants change.

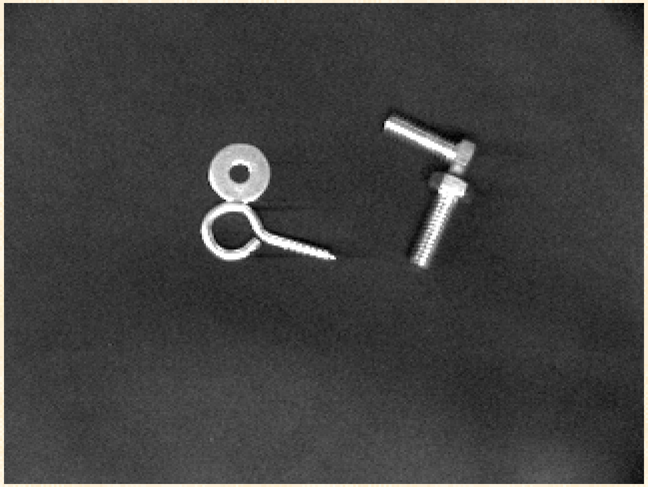
Table 1 shows the recalling results.

	Bolt	Washer	Eyebolt	Hook	Dovetail
Bolt	100%	0	0	0	0
Washer	0	100%	0	0	0
Eyebolt	10%	0	90%	0	0
Hook	0	0	0	100%	0
Dovetail	0	15%	0	0	85%

Object detection under occlusions (1)

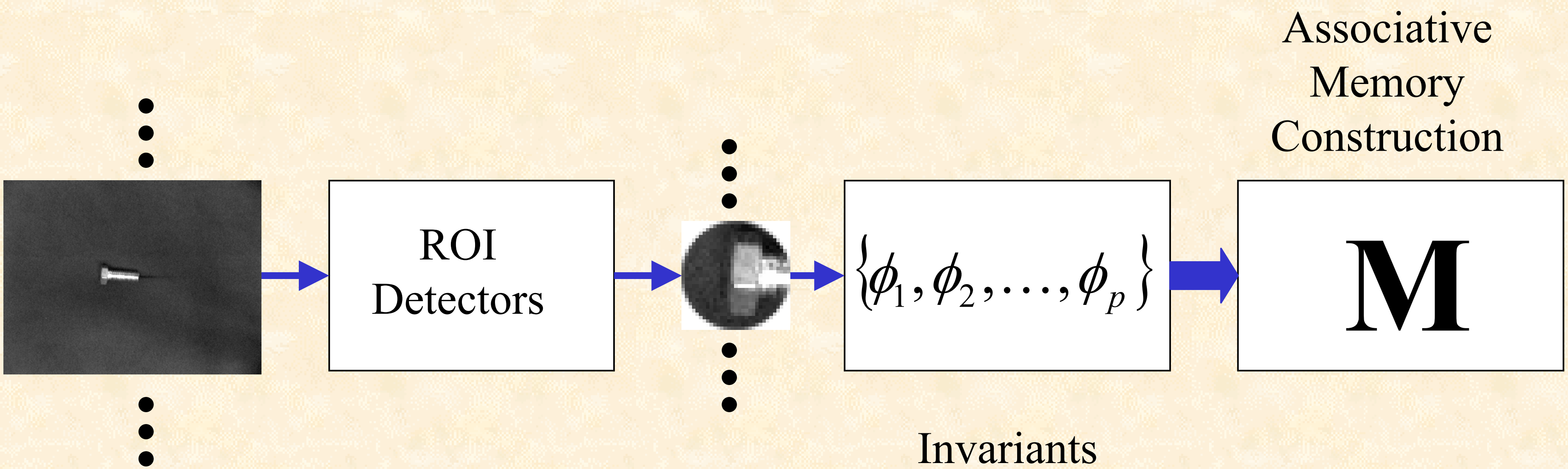
Recognition of objects under occlusions:





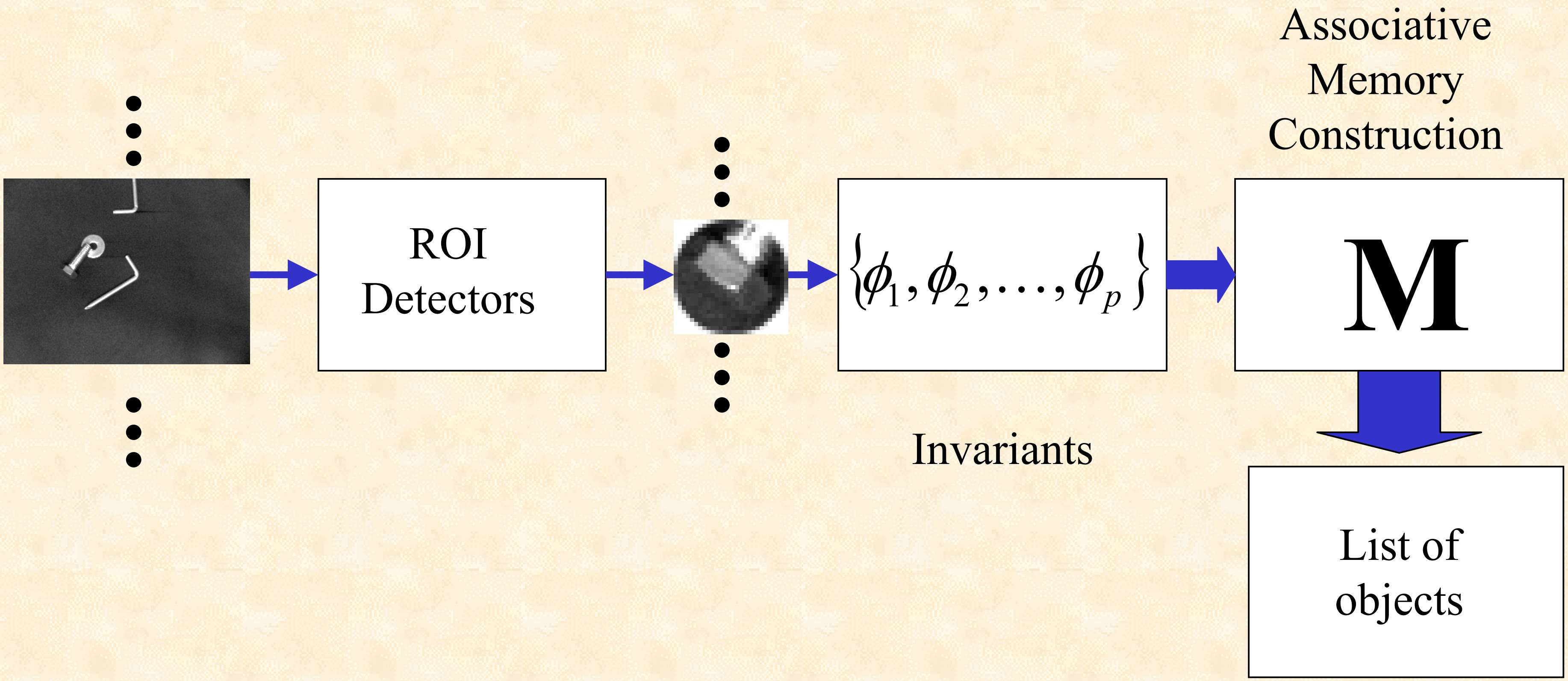
Object detection under occlusions (2)

Recognition of objects under occlusions:



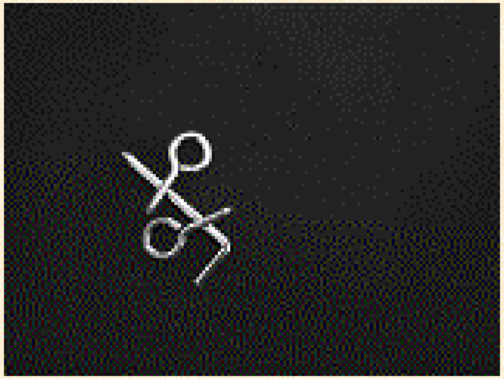
Object detection under occlusions (3)

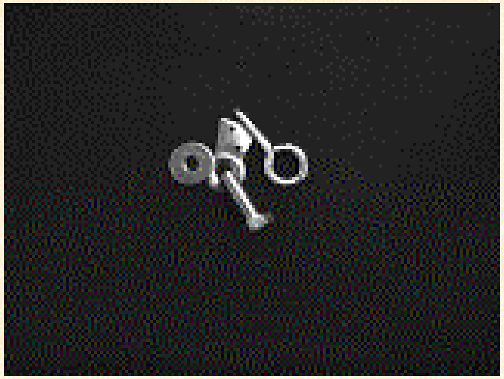
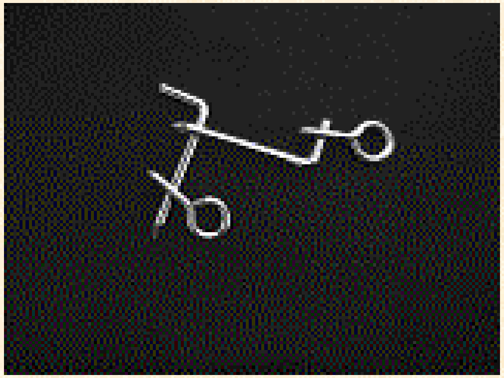
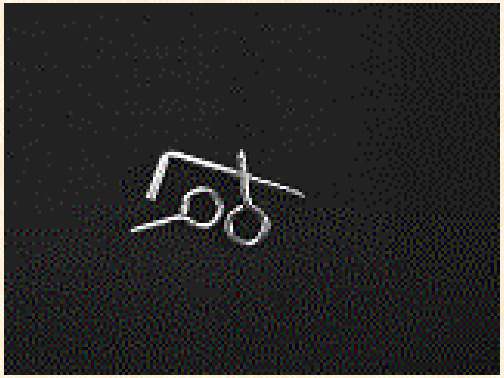
Recognition of objects under occlusions:



Object detection under occlusions (4)

100 images were used to measure the efficiency of the proposal:

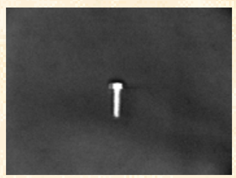
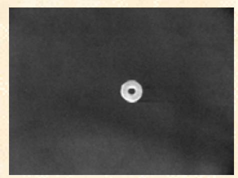
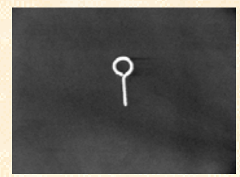
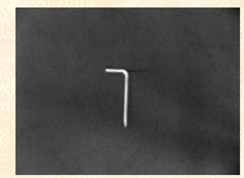
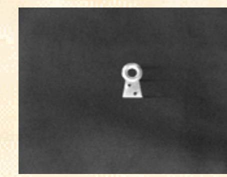


Object detection under occlusions (5)

100 images were used to measure the efficiency of the proposal.

Table 2 shows the recalling results.

					
Porcentages of detection	100%	60%	75%	40%	80%

Word reconstruction (1)

Problem:

ABUECPO?

Word reconstruction (2)

Problem:	A	B	C	E	O	P	U
PUACOEB?	1	2	3	4	5	6	7

Word reconstruction (3)

Problem:

PUACOEB?

A	B	C	E	O	P	U
1	2	3	4	5	6	7

$$\begin{pmatrix} 0 & -2 & 1 & -1 & -3 & -5 & -4 \\ 2 & 0 & 3 & 1 & -1 & -3 & -2 \\ -1 & -2 & 0 & -2 & -4 & -6 & -5 \\ 1 & -1 & 2 & 0 & -2 & -4 & -3 \\ 3 & 1 & 4 & 2 & 0 & -2 & -1 \\ 5 & 3 & 6 & 4 & 2 & 0 & 1 \\ 4 & 2 & 5 & 3 & 1 & -1 & 0 \end{pmatrix}$$

Word reconstruction (4)

Problem:

PUACOEB?

A	B	C	E	O	P	U
1	2	3	4	5	6	7

$$\begin{pmatrix} 0 & -2 & 1 & -1 & -3 & -5 & -4 \\ 2 & 0 & 3 & 1 & -1 & -3 & -2 \\ -1 & -2 & 0 & -2 & -4 & -6 & -5 \\ 1 & -1 & 2 & 0 & -2 & -4 & -3 \\ 3 & 1 & 4 & 2 & 0 & -2 & -1 \\ 5 & 3 & 6 & 4 & 2 & 0 & 1 \\ 4 & 2 & 5 & 3 & 1 & -1 & 0 \end{pmatrix} \Diamond_B = \begin{array}{c} \begin{pmatrix} 6 \\ 7 \\ 1 \\ 3 \\ 5 \\ 4 \\ 2 \end{pmatrix} \end{array} \begin{array}{c} P \\ U \\ A \\ C \\ O \\ E \\ B \end{array}$$

Word reconstruction (5)

Problem: A B C E O P U

PUACOEB? 1 2 3 4 5 6 7

$$\begin{pmatrix} 0 & -2 & 1 & -1 & -3 & -5 & -4 \\ 2 & 0 & 3 & 1 & -1 & -3 & -2 \\ -1 & -2 & 0 & -2 & -4 & -6 & -5 \\ 1 & -1 & 2 & 0 & -2 & -4 & -3 \\ 3 & 1 & 4 & 2 & 0 & -2 & -1 \\ 5 & 3 & 6 & 4 & 2 & 0 & 1 \\ 4 & 2 & 5 & 3 & 1 & -1 & 0 \end{pmatrix} \Diamond_B = \begin{pmatrix} 6 \\ 7 \\ 1 \\ 3 \\ 5 \\ 4 \\ 2 \end{pmatrix} = \begin{pmatrix} med(6,5,2,2,2,-1,-2) \\ med(8,7,4,4,4,1,0) \\ med(5,5,1,1,1,-2,-3) \\ med(7,6,3,3,3,0,-1) \\ med(9,8,5,5,5,2,1) \\ med(11,10,7,7,7,4,3) \\ med(10,9,6,6,6,3,2) \end{pmatrix}$$

Word reconstruction (6)

Problem:

PUACOEB?

A	B	C	E	O	P	U
1	2	3	4	5	6	7

$$\begin{pmatrix} 0 & -2 & 1 & -1 & -3 & -5 & -4 \\ 2 & 0 & 3 & 1 & -1 & -3 & -2 \\ -1 & -2 & 0 & -2 & -4 & -6 & -5 \\ 1 & -1 & 2 & 0 & -2 & -4 & -3 \\ 3 & 1 & 4 & 2 & 0 & -2 & -1 \\ 5 & 3 & 6 & 4 & 2 & 0 & 1 \\ 4 & 2 & 5 & 3 & 1 & -1 & 0 \end{pmatrix} \Diamond_B = \begin{pmatrix} 6 \\ 7 \\ 1 \\ 3 \\ 5 \\ 4 \\ 2 \end{pmatrix} = \begin{pmatrix} med(-2,-1,2,2,2,5,6) \\ med(0,1,4,4,4,7,8) \\ med(-3,-2,1,1,1,5,5) \\ med(-1,0,3,3,3,6,7) \\ med(1,2,5,5,5,8,9) \\ med(3,4,7,7,7,10,11) \\ med(2,3,6,6,6,9,10) \end{pmatrix}$$

Word reconstruction (7)

Problem:	A	B	C	E	O	P	U
PUACOEB?	1	2	3	4	5	6	7

$$
\begin{pmatrix}
med(-2,-1,2,2,2,5,6) \\
med(0,1,4,4,4,7,8) \\
med(-3,-2,1,1,1,5,5) \\
med(-1,0,3,3,3,6,7) \\
med(1,2,5,5,5,8,9) \\
med(3,4,7,7,7,10,11) \\
med(2,3,6,6,6,9,10)
\end{pmatrix}
$$

Word reconstruction (8)

Problem:

PUACOEB?

A	B	C	E	O	P	U
1	2	3	4	5	6	7

$$\begin{pmatrix} med(-2,-1,2,2,2,5,6) \\ med(0,1,4,4,4,7,8) \\ med(-3,-2,1,1,1,5,5) \\ med(-1,0,3,3,3,6,7) \\ med(1,2,5,5,5,8,9) \\ med(3,4,7,7,7,10,11) \\ med(2,3,6,6,6,9,10) \end{pmatrix} = \begin{pmatrix} 2 \\ 4 \\ 1 \\ 3 \\ 5 \\ 7 \\ 6 \end{pmatrix} \quad \begin{matrix} B \\ E \\ A \\ C \\ O \\ U \\ P \end{matrix}$$

BEACOUP

Conclusions

☞ We have described a set of associative memories able to recall patterns altered by mixed noise.

☞ The proposed memories are based on the median operator whidely used in signal and image non-linear filtering.

☞ The use of the same operation for memory construction and pattern recall.

☞ They avoid the use of so called kernels.

☞ We have provided the conditions under which the proposed memories are able to recall patterns either from the fundamental set of from altered versions of them.

☞ We have shown how the proposed memories can be applied in distinct problems, such as: pattern reconstruction (images and words), pattern classification and object detection under occlusions.

Present research (1)

☞ Test with other operators different from the **median** and study their properties.

☞ Look for more conditions for perfect retrieval when the median operator is used for memory training and pattern recall.

☞ Search for other models that do not employ the kernel approach or the median operator to handle mixed noise.

☞ Search for new structures, ways of memorizing and recalling information.

☞ Printed word recognition.

☞ Word reconstruction.

☞ Spell checking.

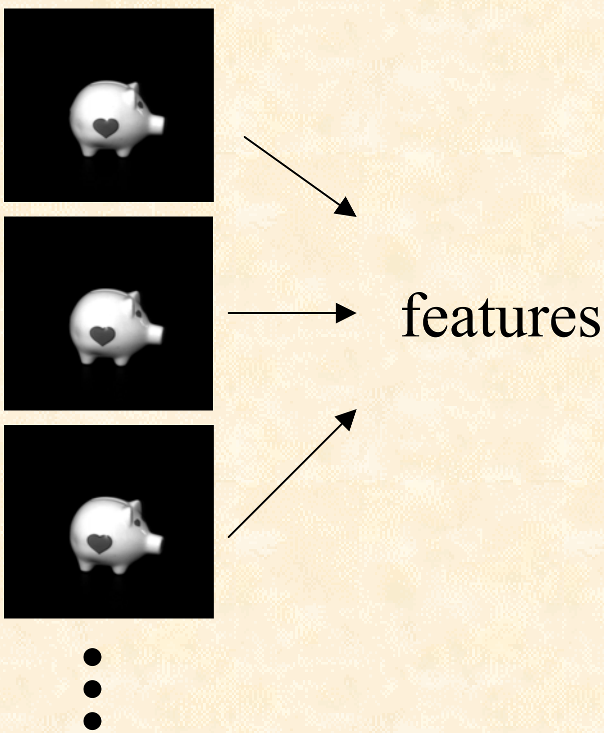
☞ Translation from one language to another one.

☞ Generic object recognition.

Present research (2)

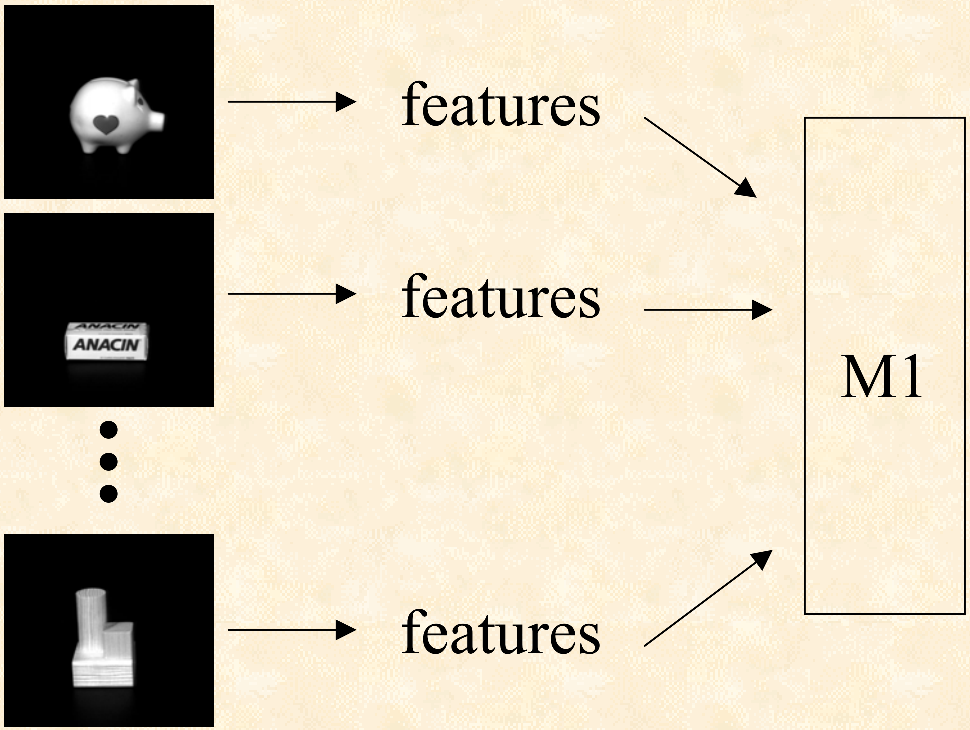
Generic object recognition (training):

From similar views of each object:



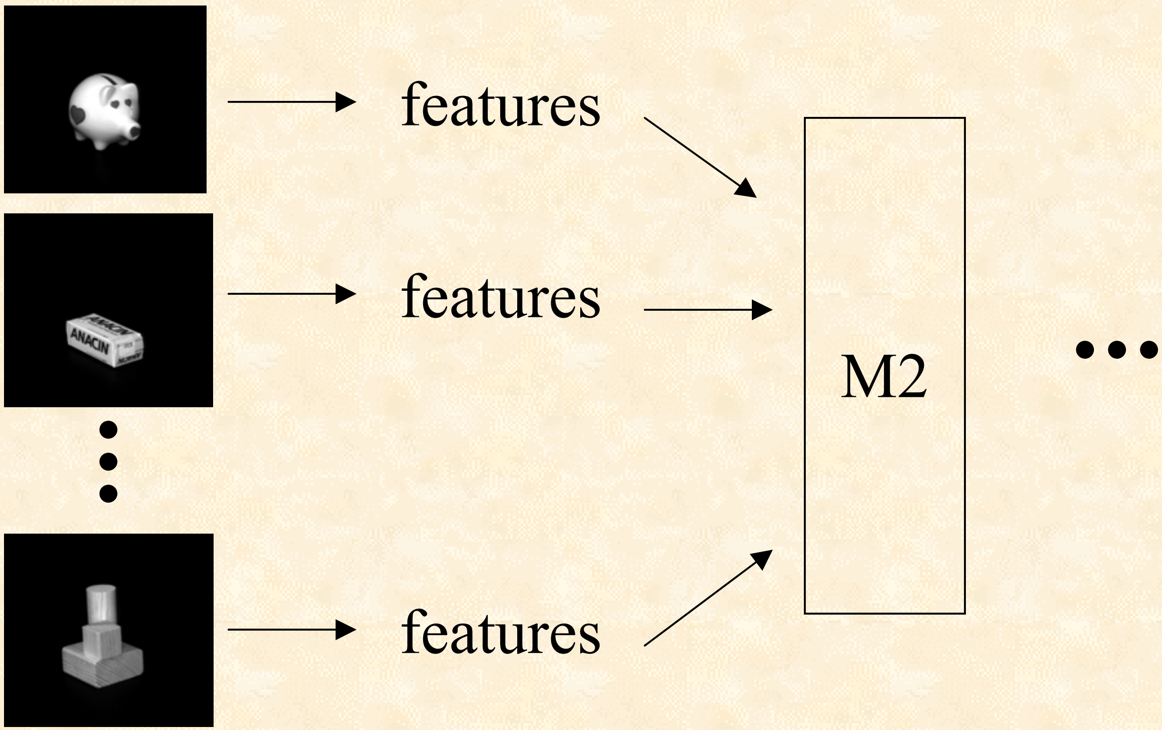
Present research (3)

Generic object recognition (training):



Present research (4)

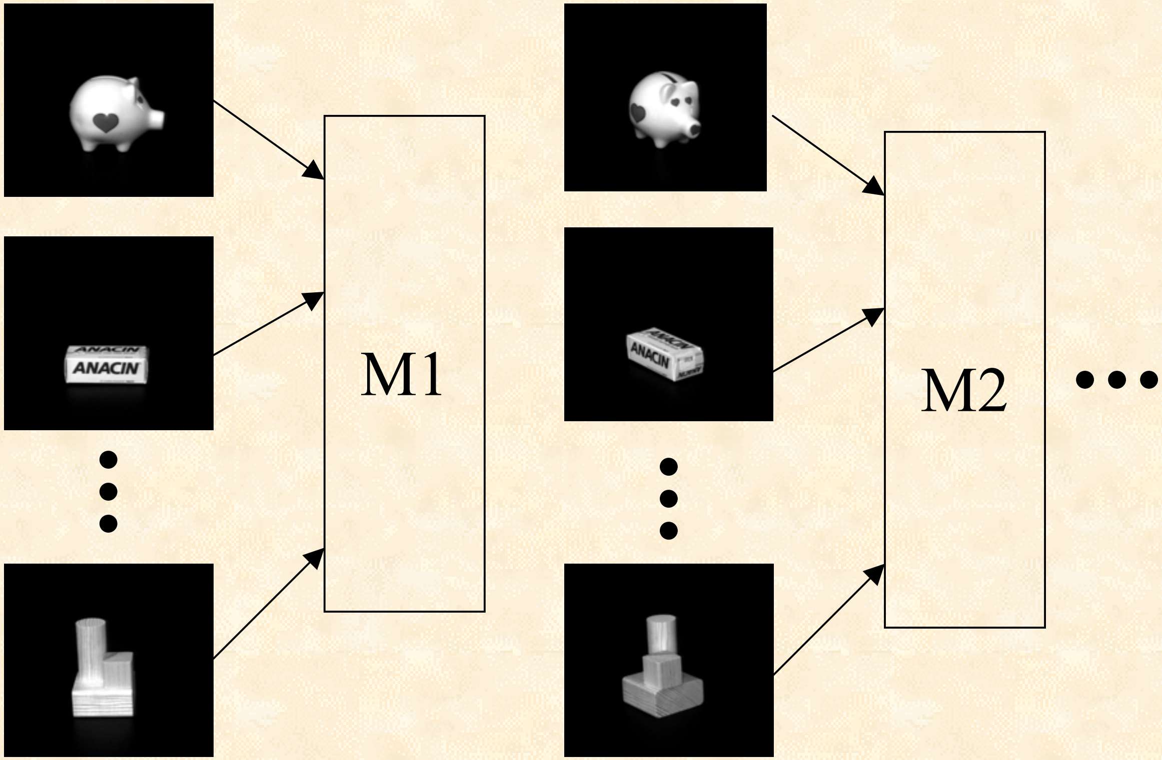
Generic object recognition (training):



Repeat process for more views of objects.

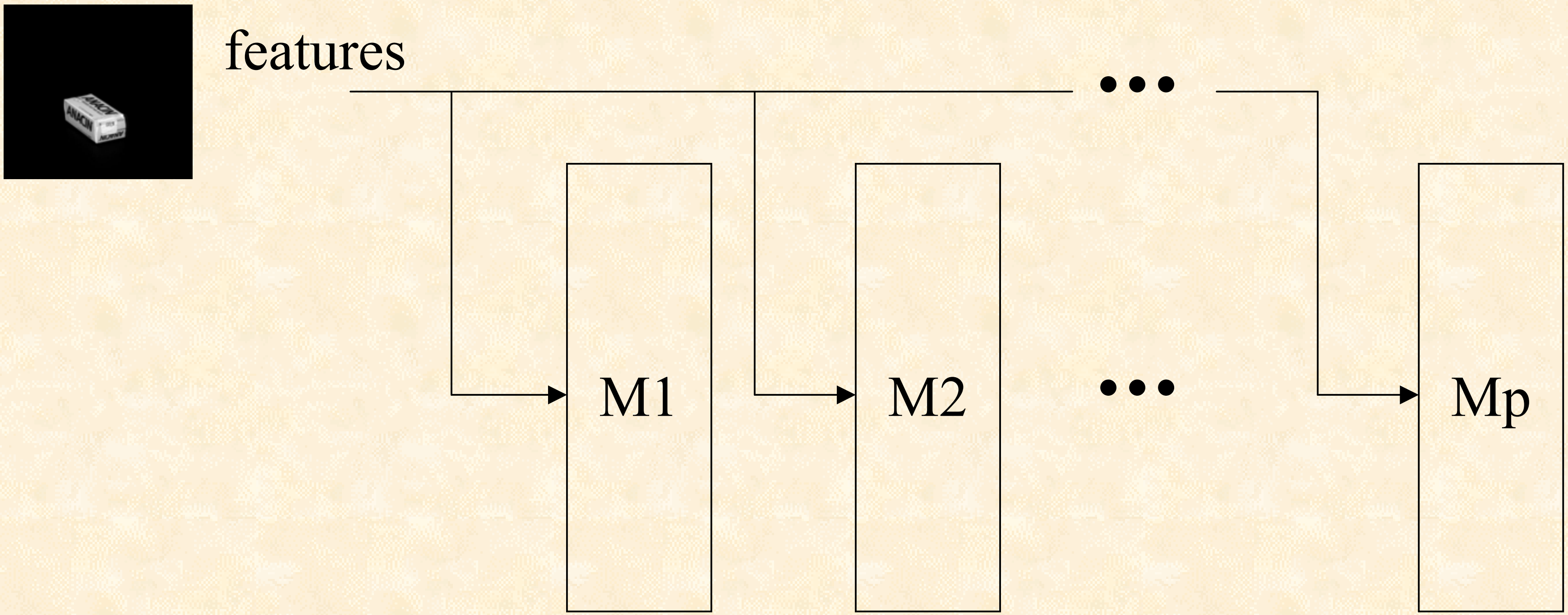
Present research (5)

Generic object recognition (training):



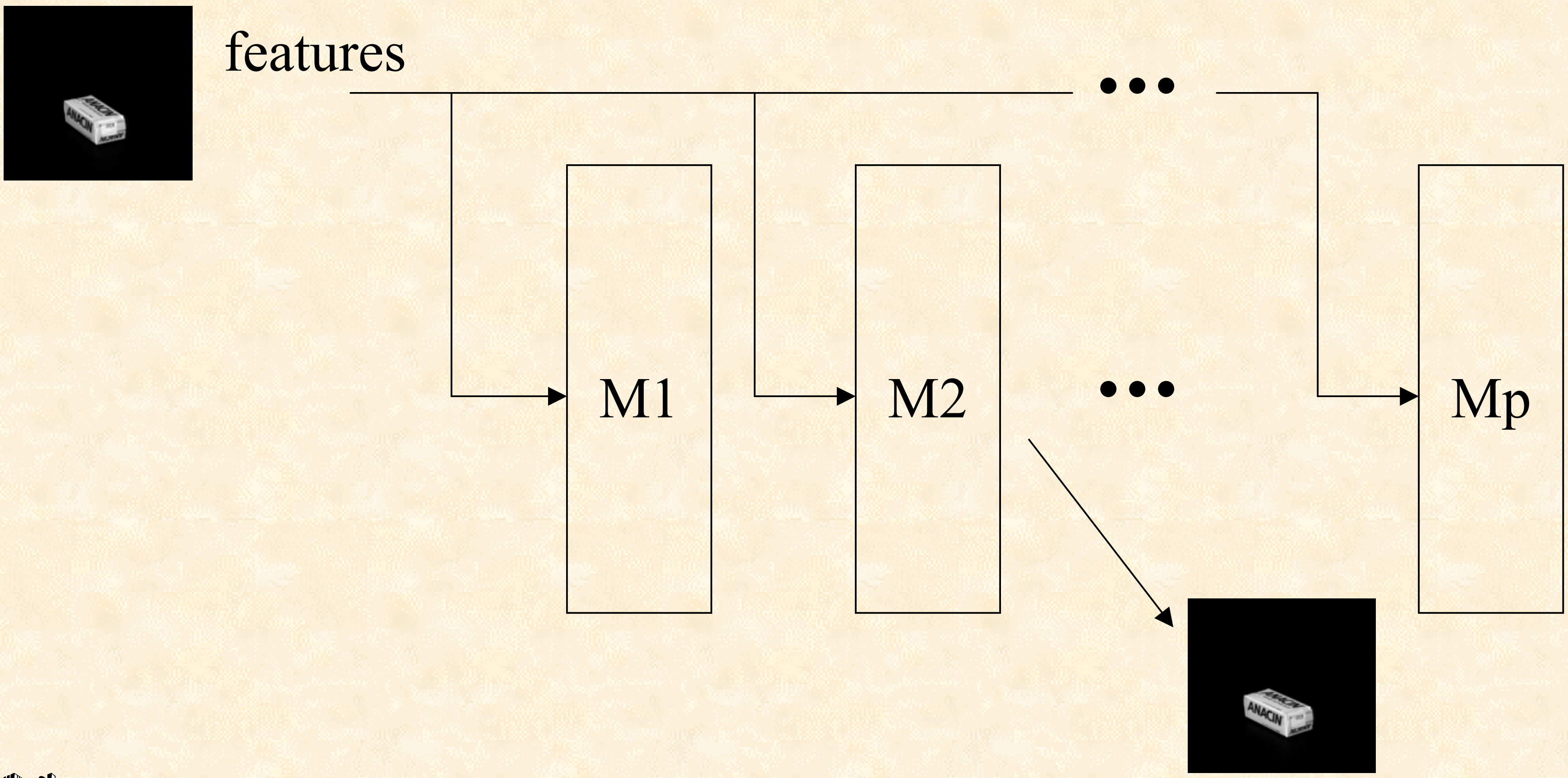
Present research (6)

Generic object recognition (recognition):



Present research (7)

Generic object recognition (recognition):



Present research (8)

We would like to use features obtained from images of the objects.

- PCA, VQ, NMF,…

- Invariants from intensity and range images.

- Features from parts of the objects to tackle occlusion problem.

- Tray with other distances for classification.

THANKS

Humberto Sossa

E-mail: hsossa@cic.ipn.mx