Identification of Finite Lattice Dynamical Systems

Junior Barrera

BIOINFO-USP

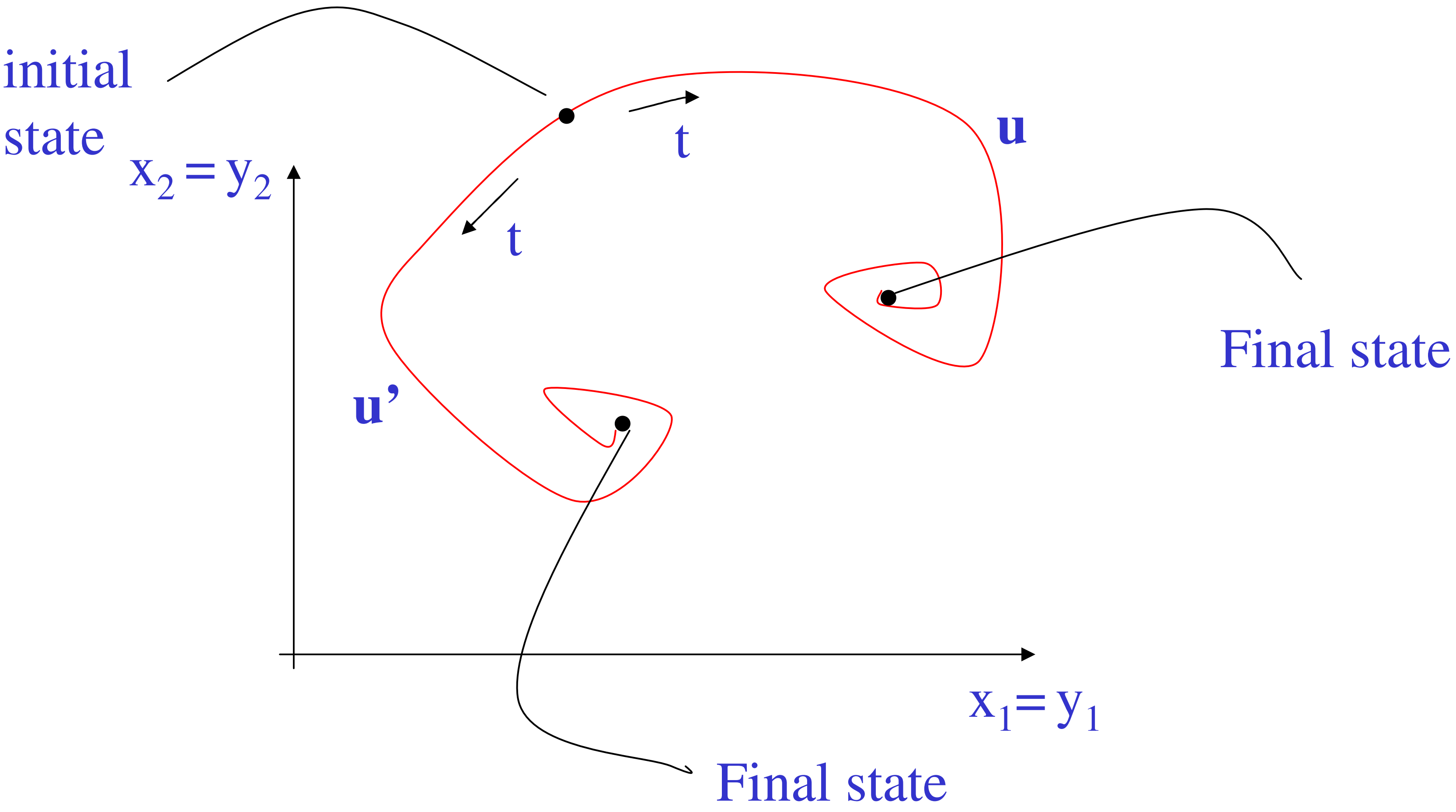
University of São Paulo, Brazil

Outline

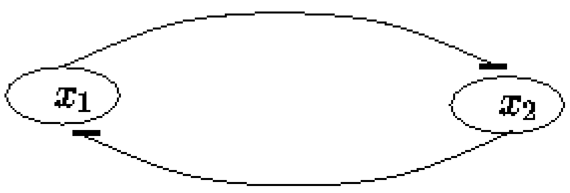
- Introduction
- Lattice operator representation
- Lattice operator design
- Lattice dynamical systems (LDS)
- A simulator for chain dynamical systems
- LDS identification
- System identification examples
- Conclusion

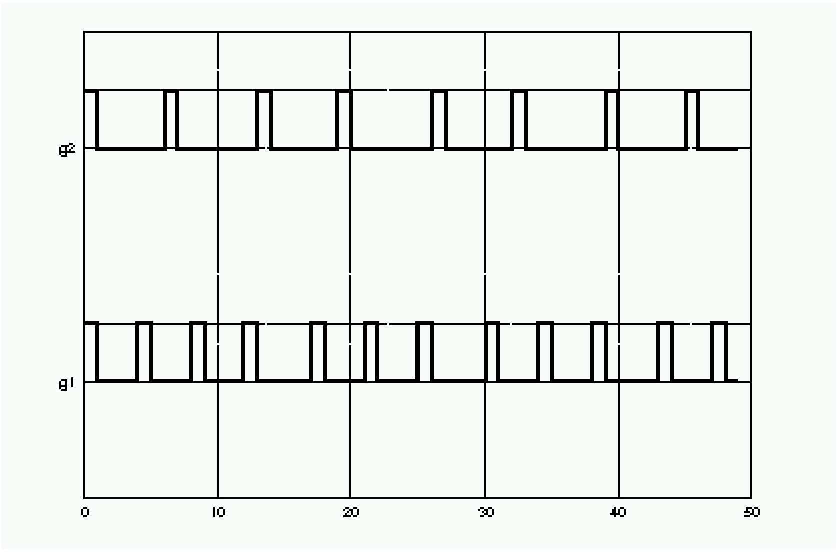
Introduction

Dynamical Systems



Sequential Switching Circuits





$$\phi_1(\mathbf{x}[i-5],\mathbf{x}[i-4],\mathbf{x}[i-3],\mathbf{x}[i-2],\mathbf{x}[i-1],\mathbf{x}[i]) = \overline{x}_1[i-3]\cdot\overline{x}_1[i-2]\cdot\overline{x}_1[i-1]\cdot\overline{x}_2[i-5]\cdot\overline{x}_2[i-3]\cdot\overline{x}_2[i-1]$$

$$\phi_2(\mathbf{x}[i-5],\mathbf{x}[i-4],\mathbf{x}[i-3],\mathbf{x}[i-2],\mathbf{x}[i-1],\mathbf{x}[i]) = \overline{x}_1[i-4]\cdot\overline{x}_2[i-5]\cdot\overline{x}_2[i-4]\cdot\overline{x}_2[i-3]\cdot\overline{x}_2[i-2]\cdot\overline{x}_2[i-1]$$

Mathematical Morphology

- studies operators between complete lattices, what includes switching functions
- lattice operators are decomposed in terms of simple morphological operators: erosion, dilation, anti-erosion, anti-dilation
- Any lattice operator can be decomposed in a canonical morphological representation

Lattice Dynamical Systems

- We present the notion of Lattice Dynamical System (LDS)
- Give a representation for LDSs, based on canonical morphological representations
- Formalize the problem of statistical identification of LDSs,

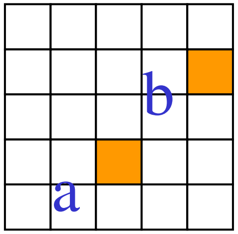
Operator Representation

$$\psi : \mathrm{Fun}[W, L] \;\rightarrow\; L$$

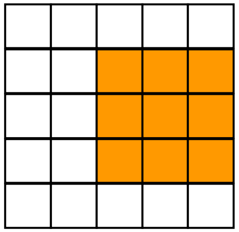
2	0	1	2	2	2
1	0	1	2	2	2
0	-1	1	2	2	2
-1	-1	1	1	1	1
-2	-2	-1	-1	-1	-1
	-2	-1	0	1	2

Intervals

- Let a, $b \in$ Fun[W,L], $a \leq b$ iff $a(x) \leq b(x)$, $x \in W$

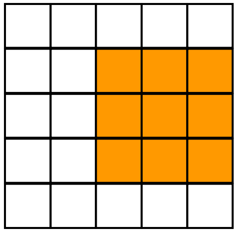
|W| = 2 

- Interval $[a,b] = \{u \in \text{Fun}[W,L]: a \leq u \leq b\}$



Binary Sup-generating

- Sup-generating operator: $\lambda_{a,b}(u) = 1 \Leftrightarrow u \in [a, b]$



[a,b]

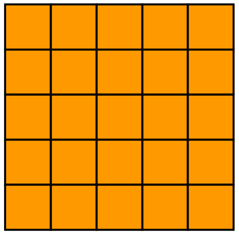
0	0	0	0	0
0	0	1	1	1
0	0	1	1	1
0	0	1	1	1
0	0	0	0	0

$\lambda_{a,b}$

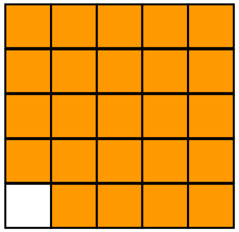
Kernel

Kernel of ψ at y: $K(\psi)(y) = \{u \in \text{Fun}[W,L]: y \leq \psi(u)\}$

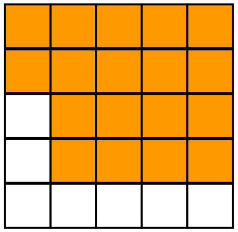
2	0	1	2	2	2
1	0	1	2	2	2
0	-1	1	2	2	2
-1	-1	1	1	1	1
-2	-2	-1	-1	-1	-1
	-2	-1	0	1	2



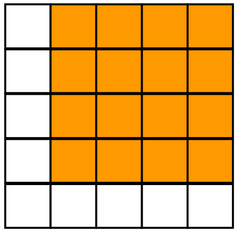
$K(\psi)(-2)$



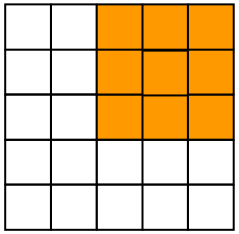
$K(\psi)(-1)$



$K(\psi)(0)$



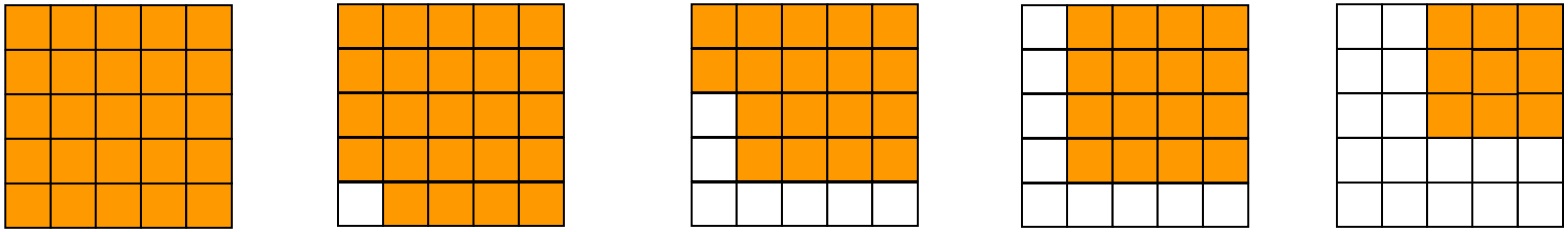
$K(\psi)(1)$



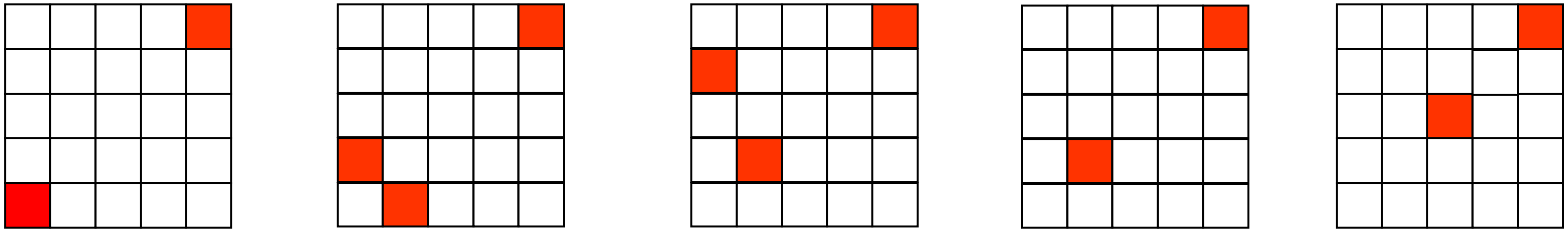
$K(\psi)(2)$

Basis

Basis of ψ at *y*: $B(\psi)$ is the set of maximal intervals contained in $K(\psi)$



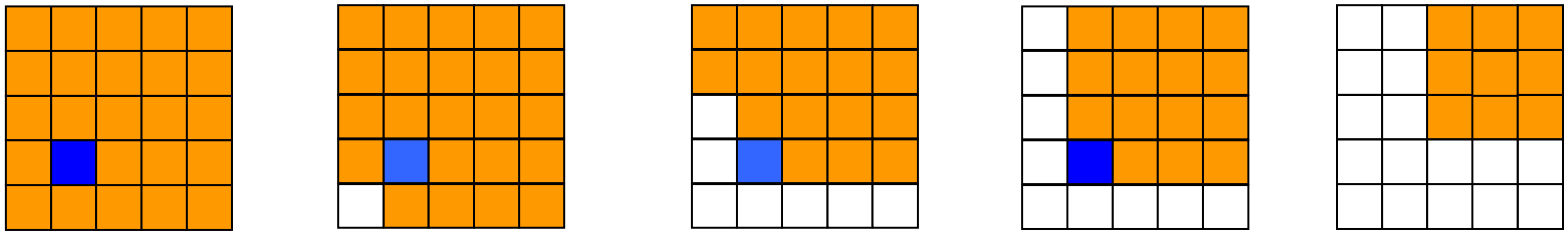
$K(\psi)(-2)$ $K(\psi)(-1)$ $K(\psi)(0)$ $K(\psi)(1)$ $K(\psi)(2)$



$B(\psi)(-2)$ $B(\psi)(-1)$ $B(\psi)(0)$ $B(\psi)(1)$ $B(\psi)(2)$

Canonical Representation

$$\psi(u) = \bigcup\left\{ y \in M : \bigcup\left\{ \lambda_{a,b}(u) : [a,b] \in B(\psi)(y) \right\} = 1 \right\}$$



K(ψ)(-2) K(ψ)(-1) K(ψ)(0) K(ψ)(1) K(ψ)(2)

$\psi(-1,-1) = 1$

Operator Design

Optimization problem

- Design goal is to find a function ψ_{opt} : Fun[W, L] $\rightarrow$ L with minimum error.

- Error (expected loss) of a function :

$$Er(\psi) = \mathrm{E}[l(\psi(X), Y)]$$

X is a random function
Y is a random variable

- Loss function

$$l : L \times L \rightarrow \Re^{+}$$

Estimation problem

- The distribution $P(X,Y)$ is unknown

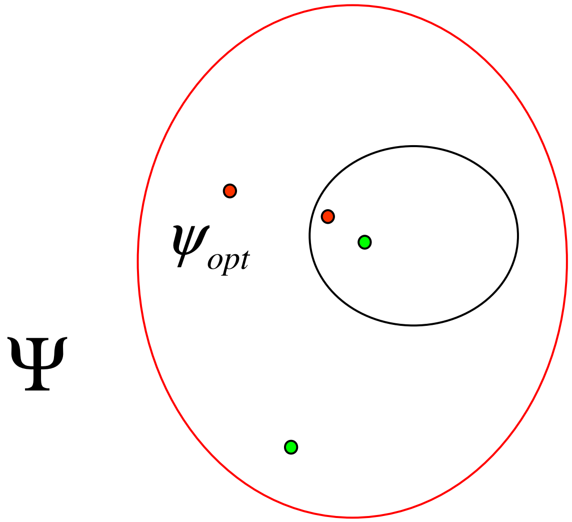
- $P(X,Y)$, $Er(\psi)$ and ψ_{opt} should be estimated from realizations of X and Y.

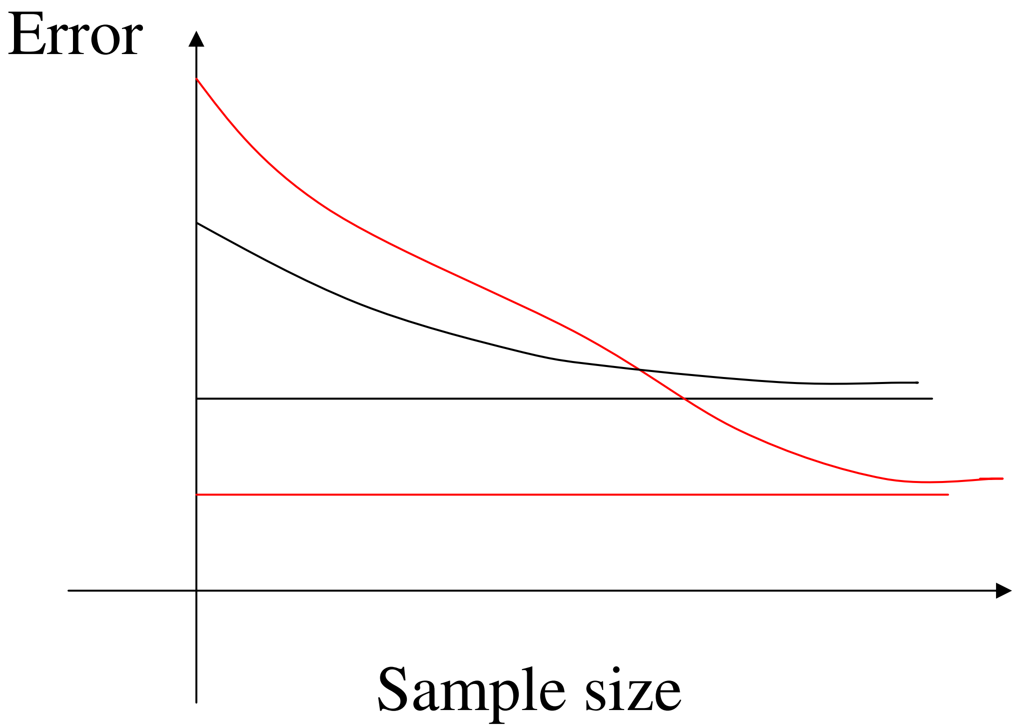
- For $m > m(\varepsilon, \delta)$ examples

$$Pr(|Er(\psi) - Er(\psi_{opt})| < \varepsilon) > 1 - \delta$$

$$\varepsilon, \delta \in (0,1)$$

The constrained estimation problem

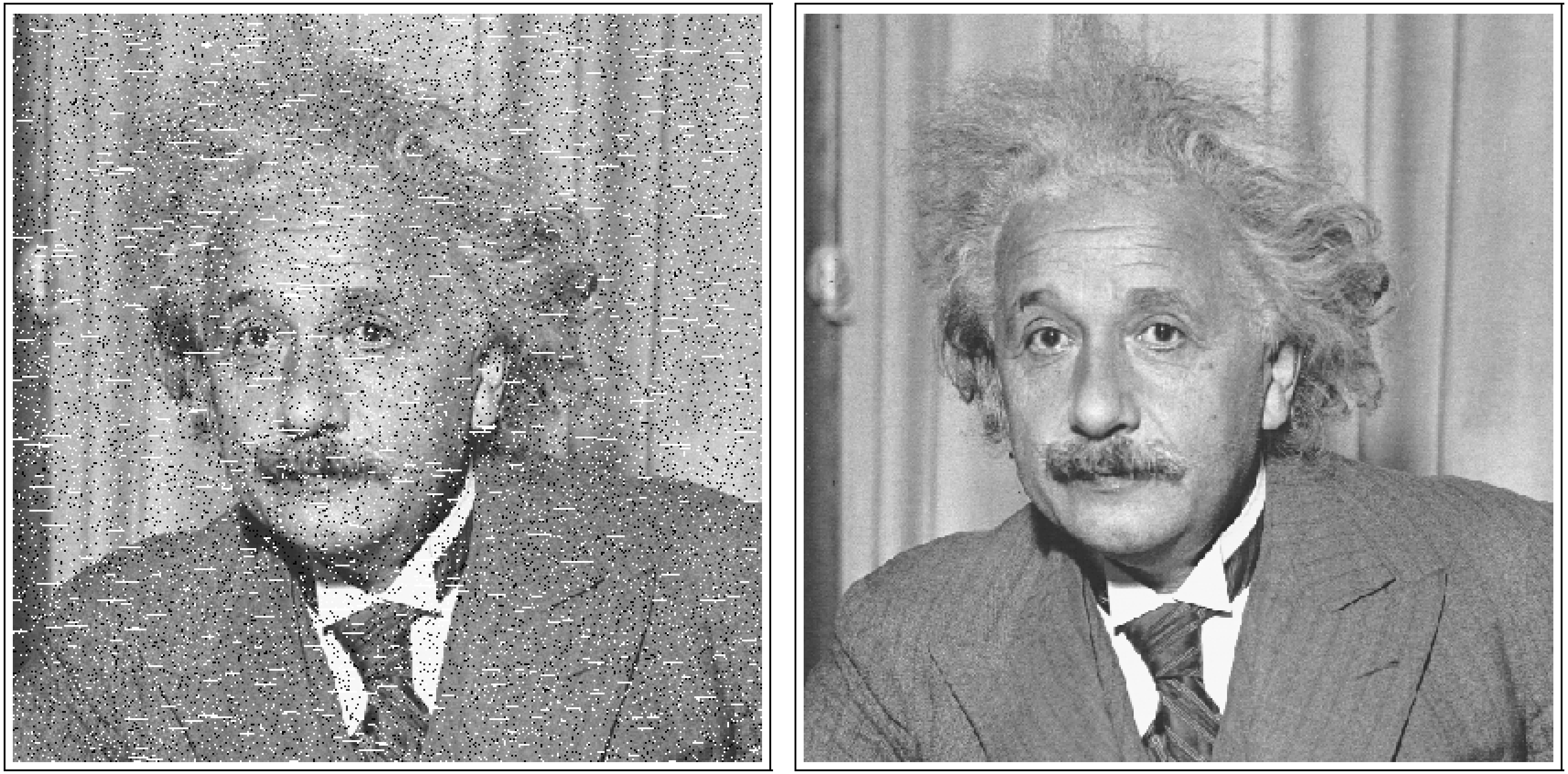




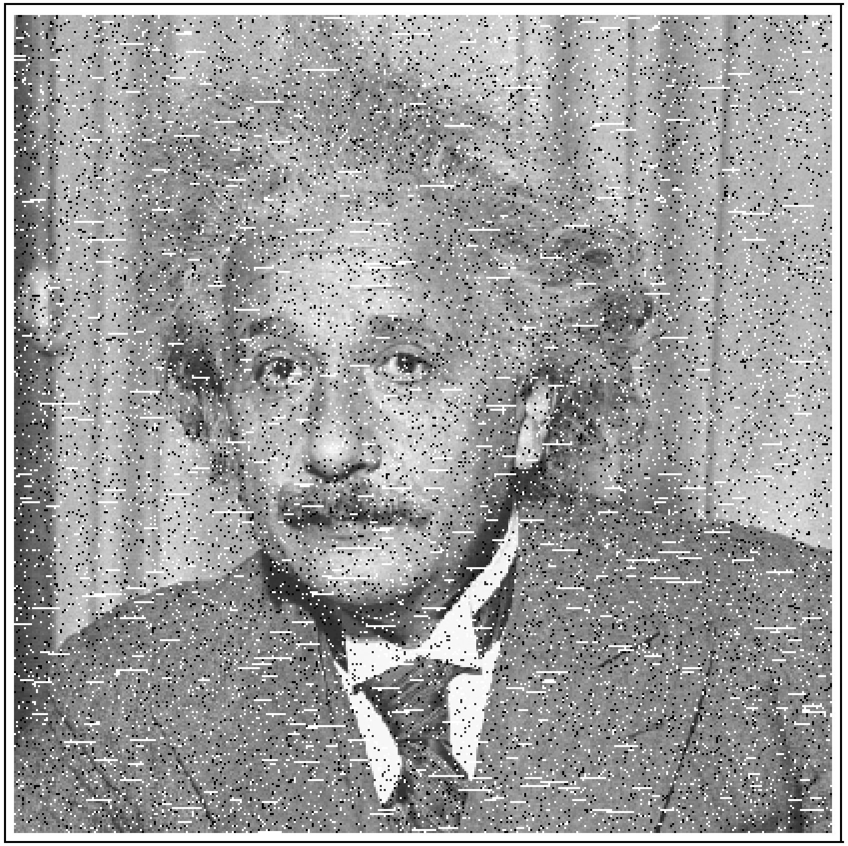
Stack filter

- Spatially translation invariant
- Spatially locally defined
- Increasing
- Commutes with threshold

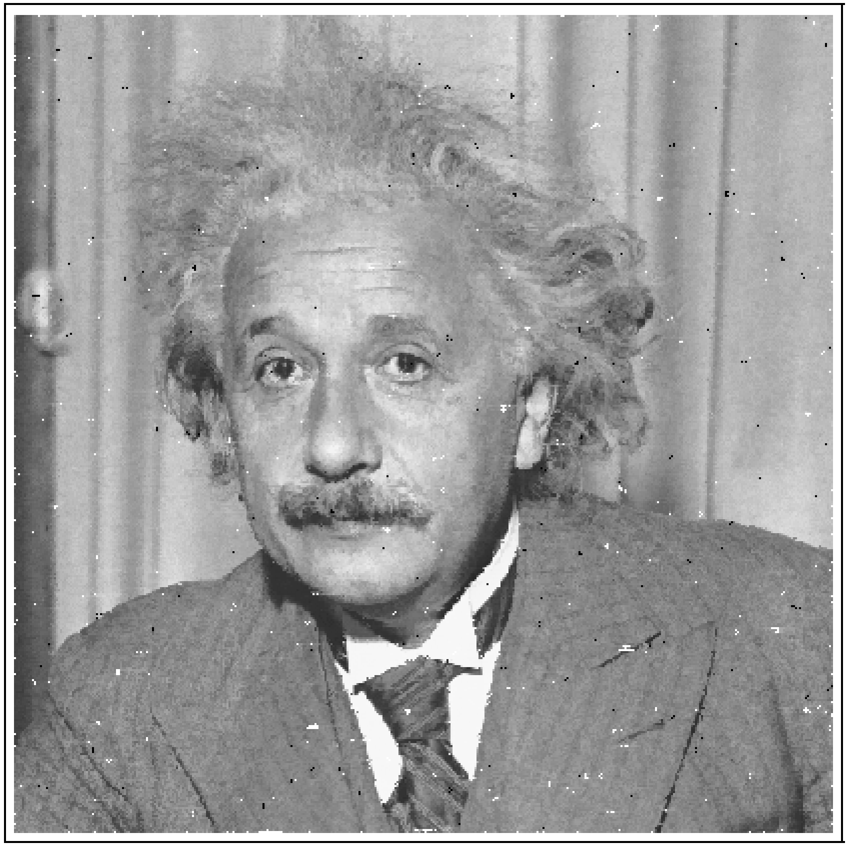
Noise elimination



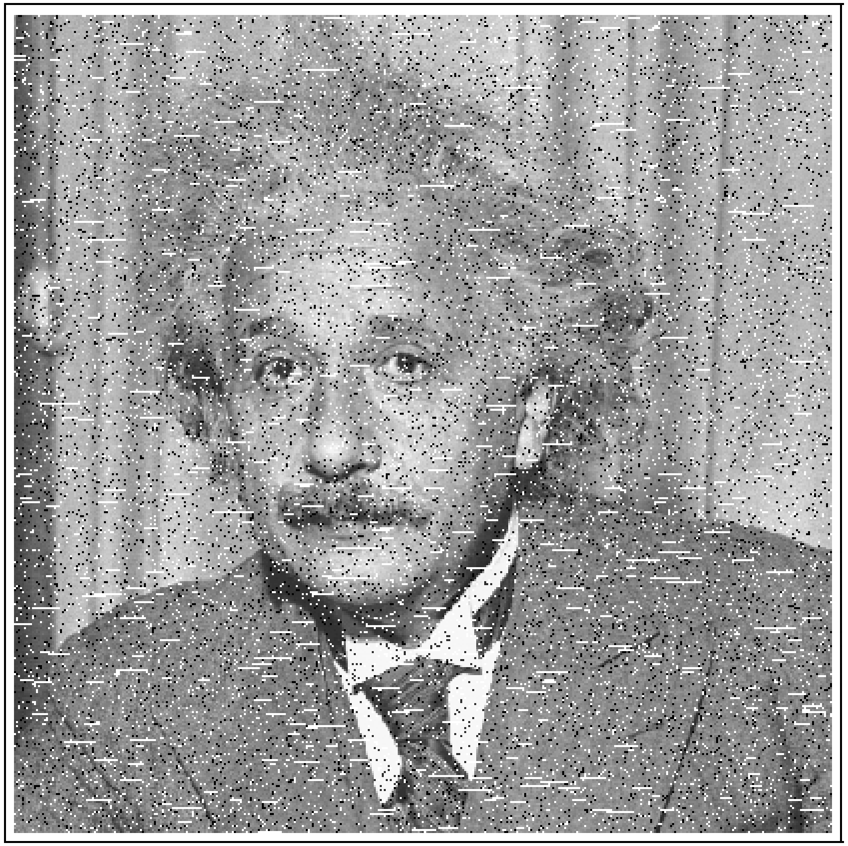
training images



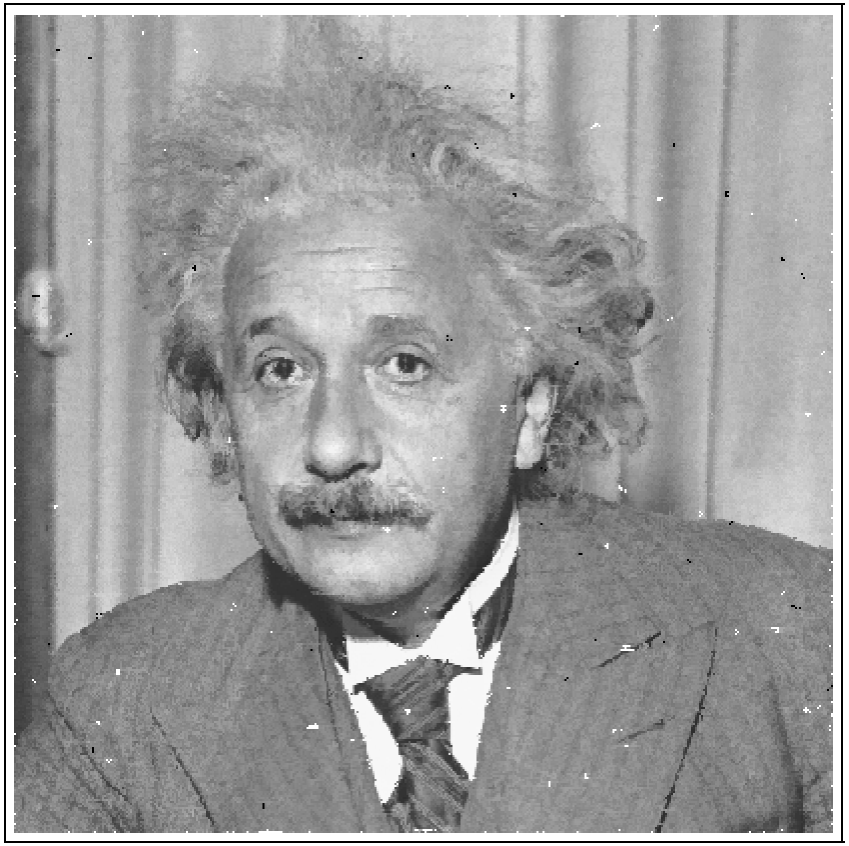
test image



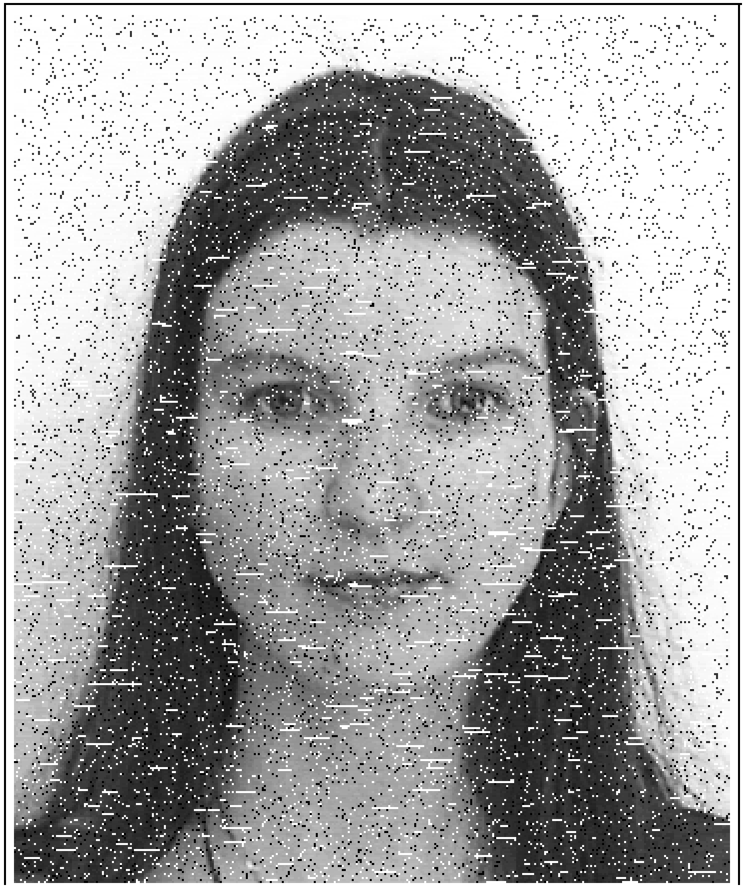
iteration 1



test image

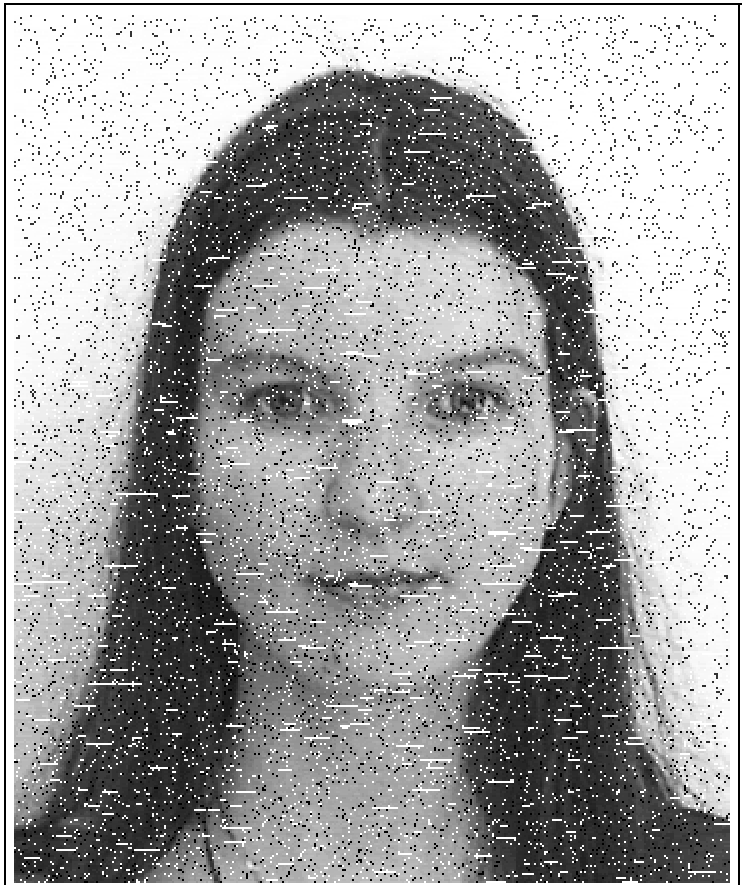


iteration 5



test image

iteration 1



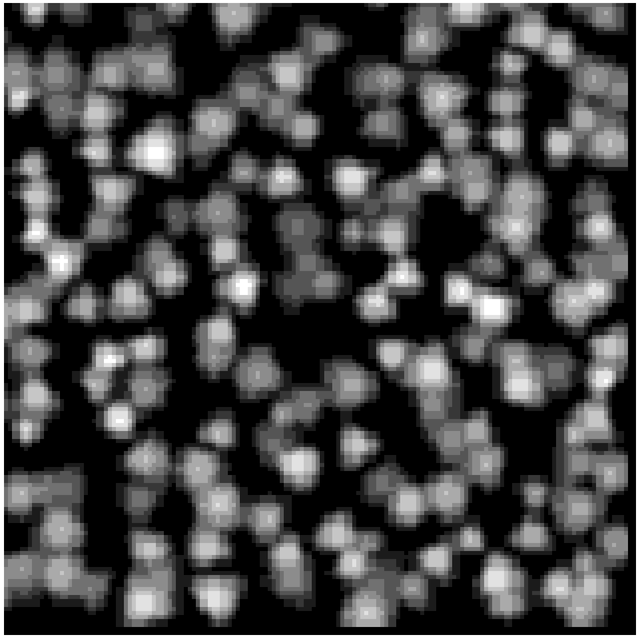
test image

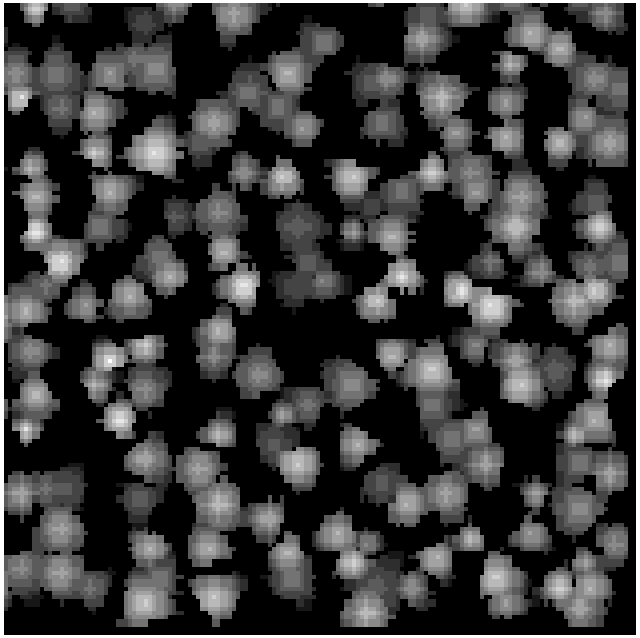
iteration 5

Apertures

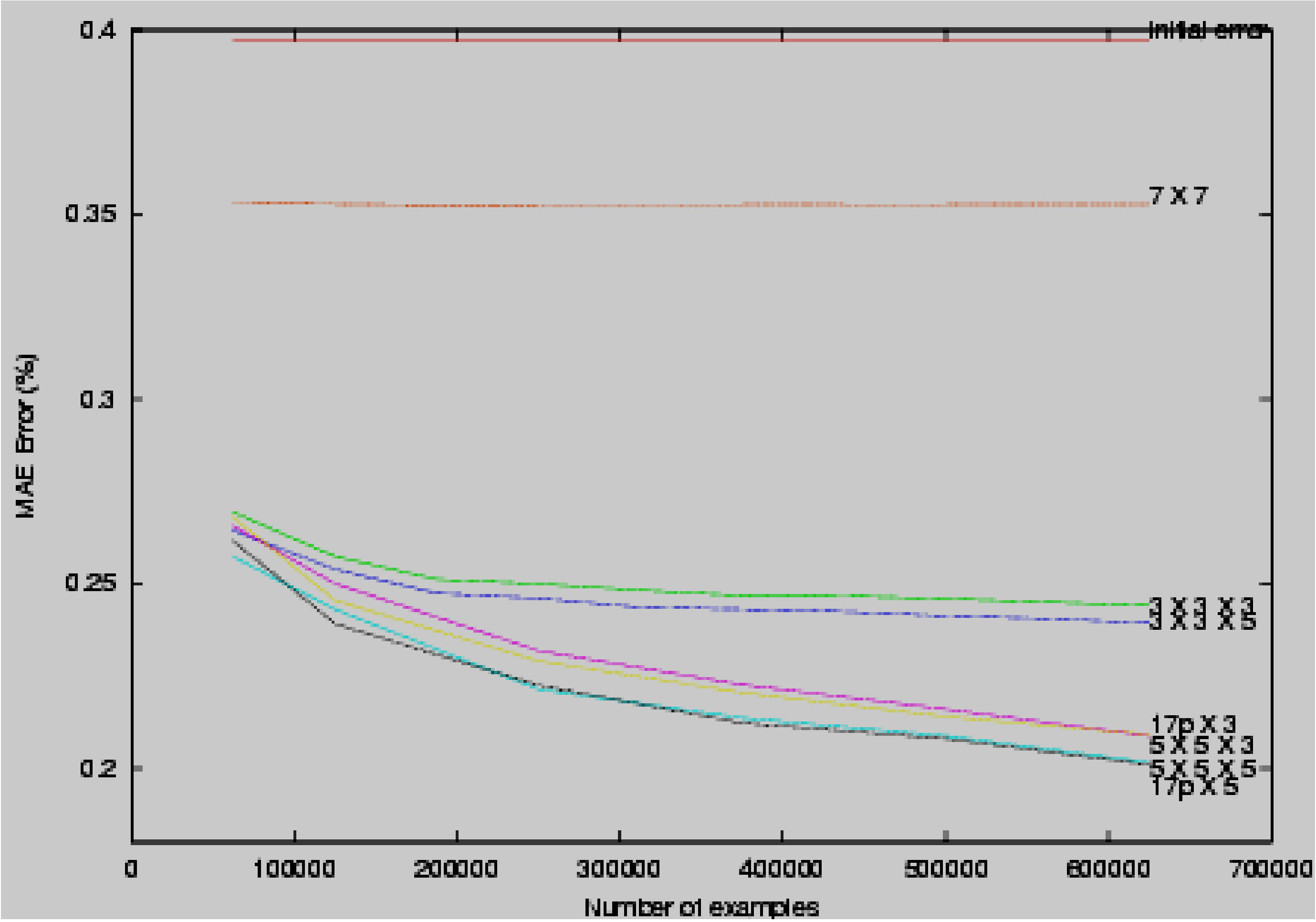
- Spatially translation invariant
- Spatially locally defined
- Range translation invariant
- Range locally defined

Deblurring



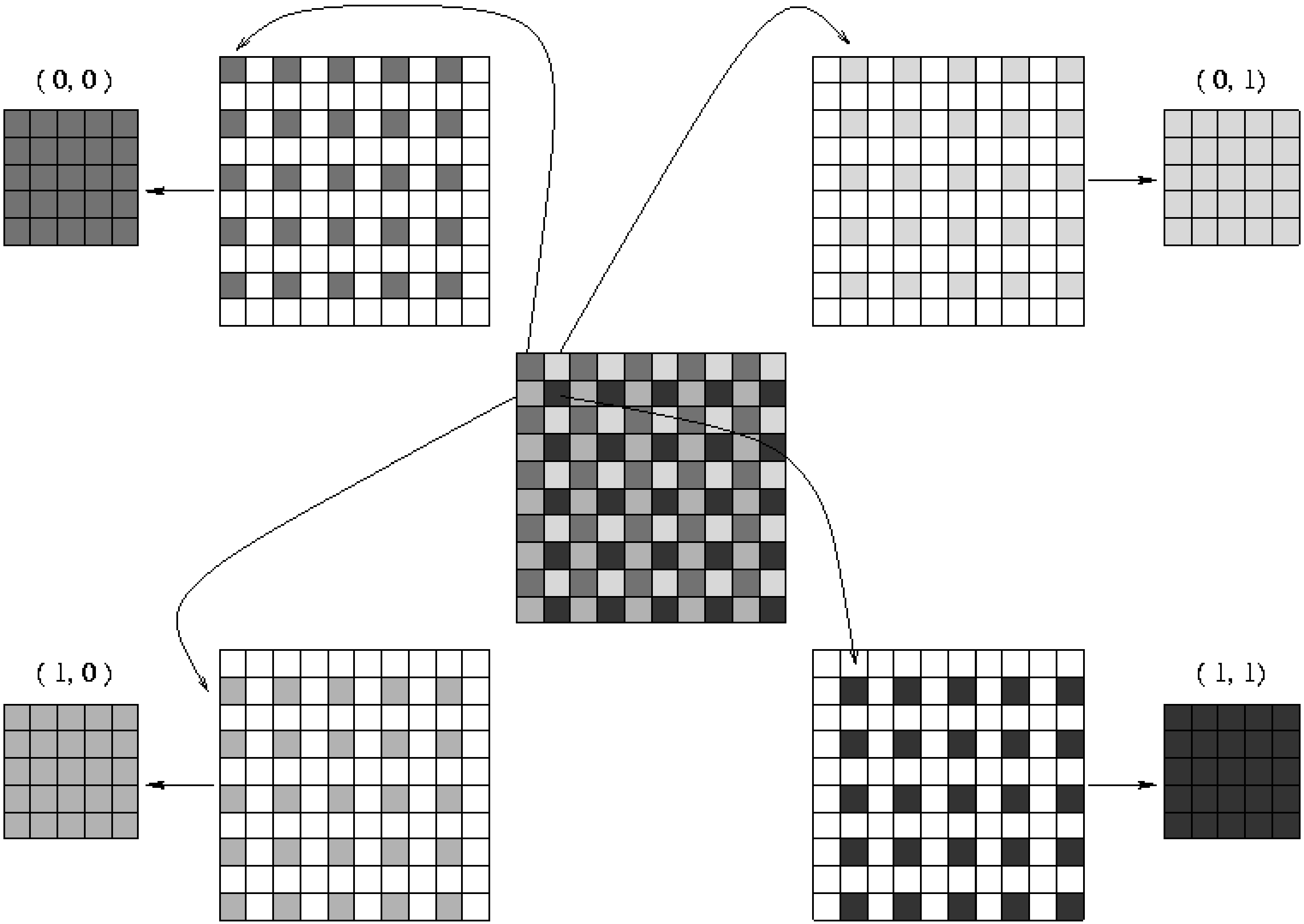


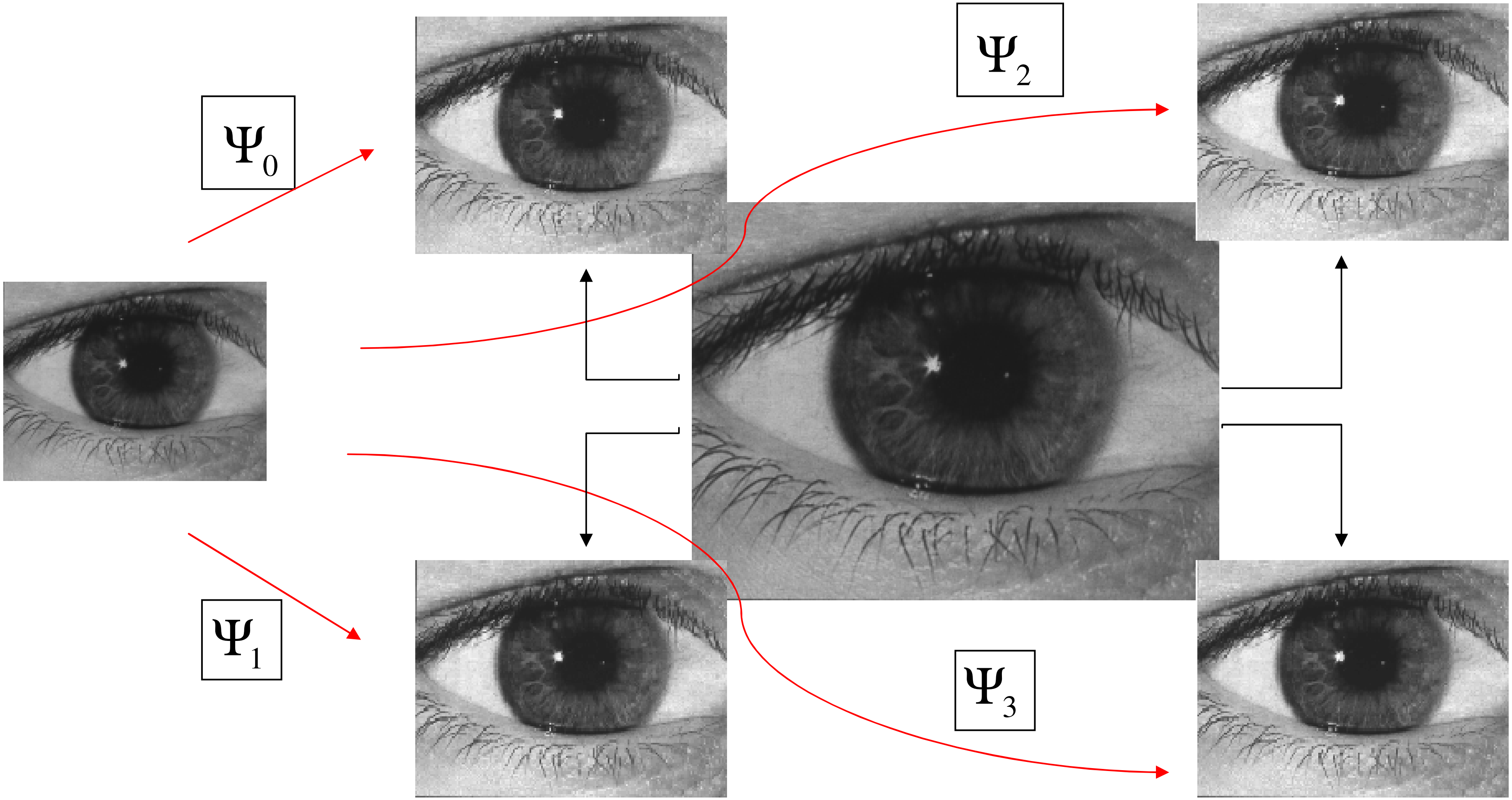
training images



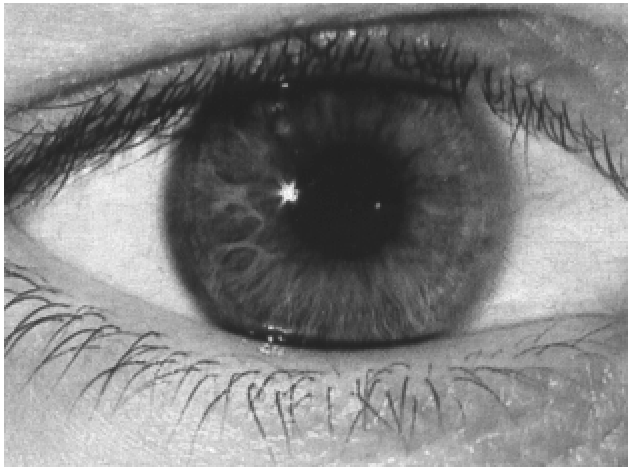
initial error
7 X 7
3 X 3 X 3
3 X 3 X 5
17p X 3
5 X 5 X 3
5 X 5 X 5
17p X 5
MAE Error (%)
Number of examples
0.4
0.35
0.3
0.25
0.2
0
100000
200000
300000
400000
500000
600000
700000

Resolution Enhancement

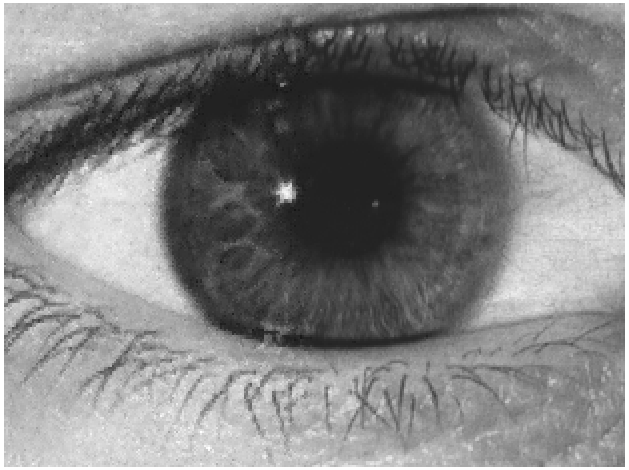




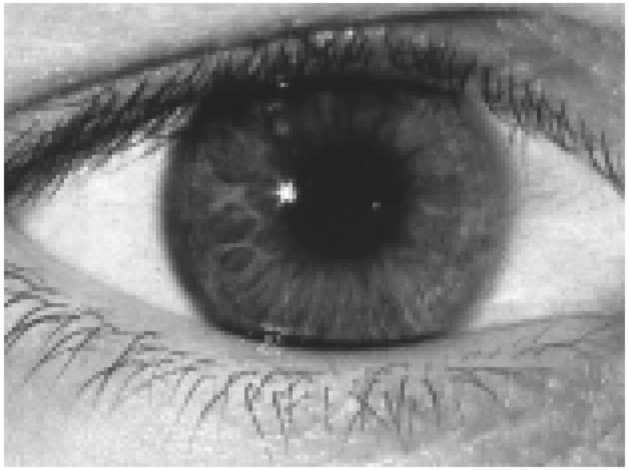
Ψ_0
Ψ_1
Ψ_2
Ψ_3



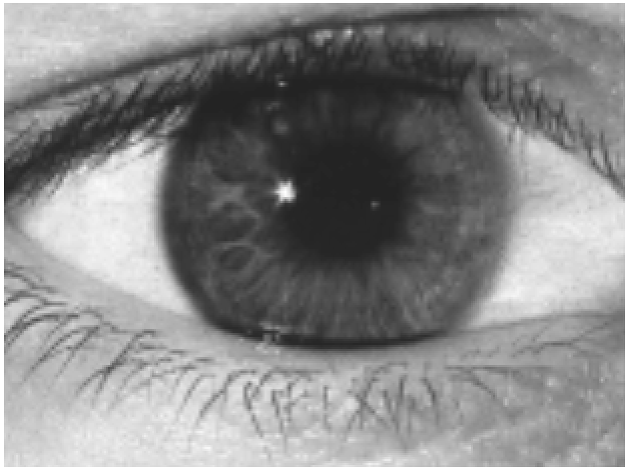
Original



Aperture: 3x3x21x51

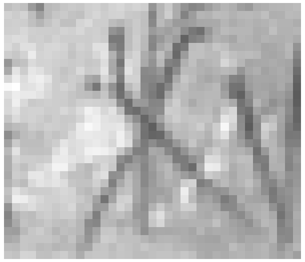


Linear

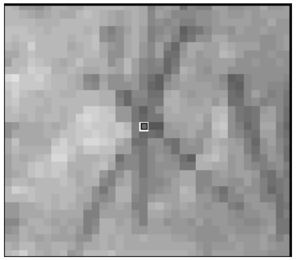


Bilinear

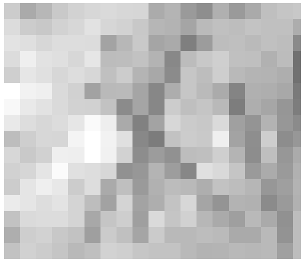
Zoom



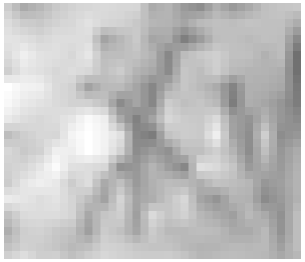
Original



Aperture: 3x3x21x51



Linear



Bilinear

Independent Constraints

Constraints

Restriction of the operators space

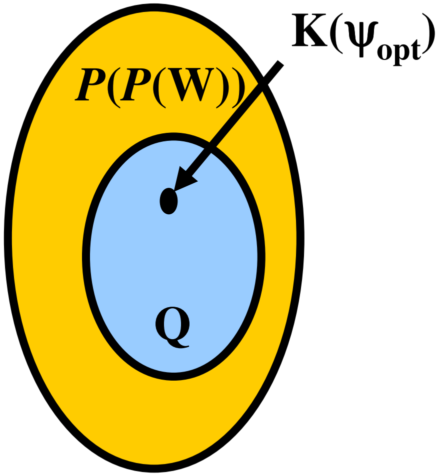
$\mathbf{K}(\psi_{opt}) \in \mathbf{Q} \subseteq P(P(\mathrm{W}))$

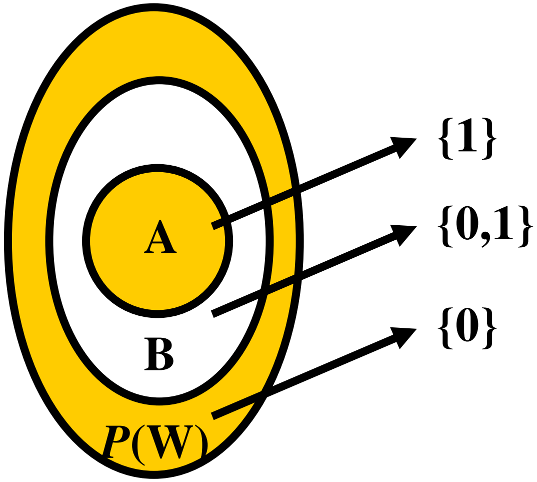
Independent Constraint

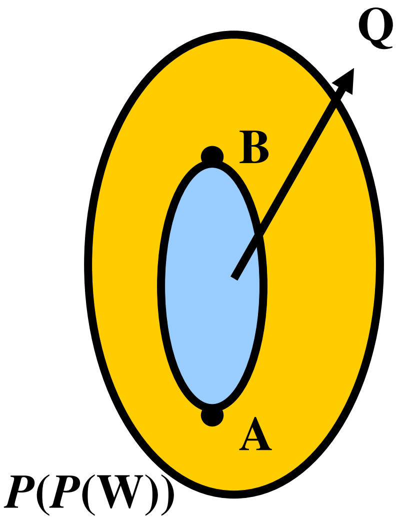
Let be $A,B \subseteq P(\mathrm{W})$ **with** $A \subseteq B$:

$h_\psi(x)=1\ \forall\ x \in A\quad \&\quad h_\psi(x)=0\ \forall x \notin B,$

$\forall \psi : \mathbf{K}(\psi) \in \mathbf{Q}$







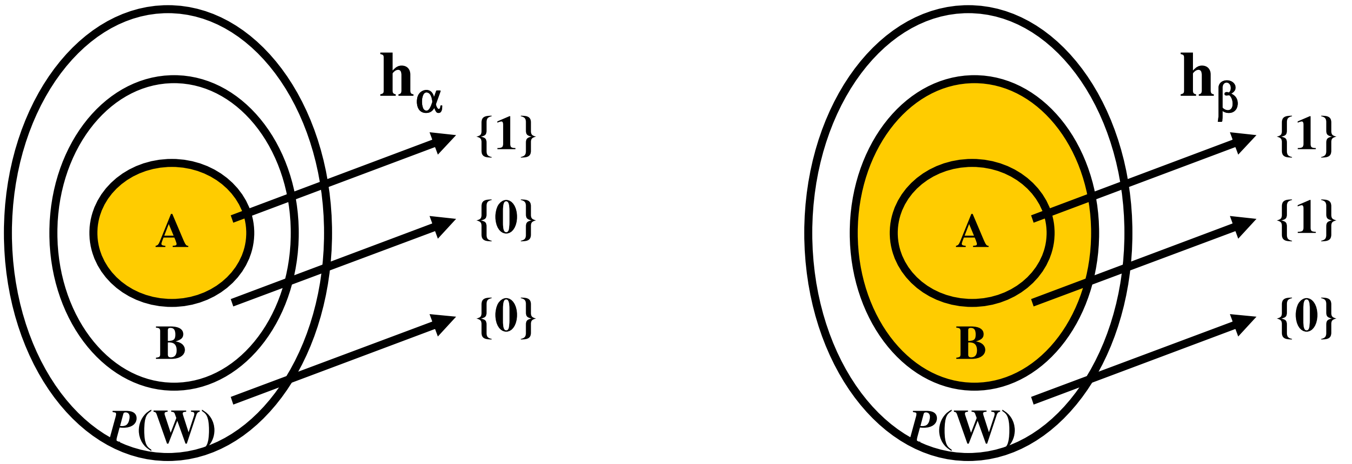
Independent Constraints

<u>Proposition</u>: if Q is an independent restriction then exist a par of operators (α,β) such that, for any $\psi \in \Psi_W$

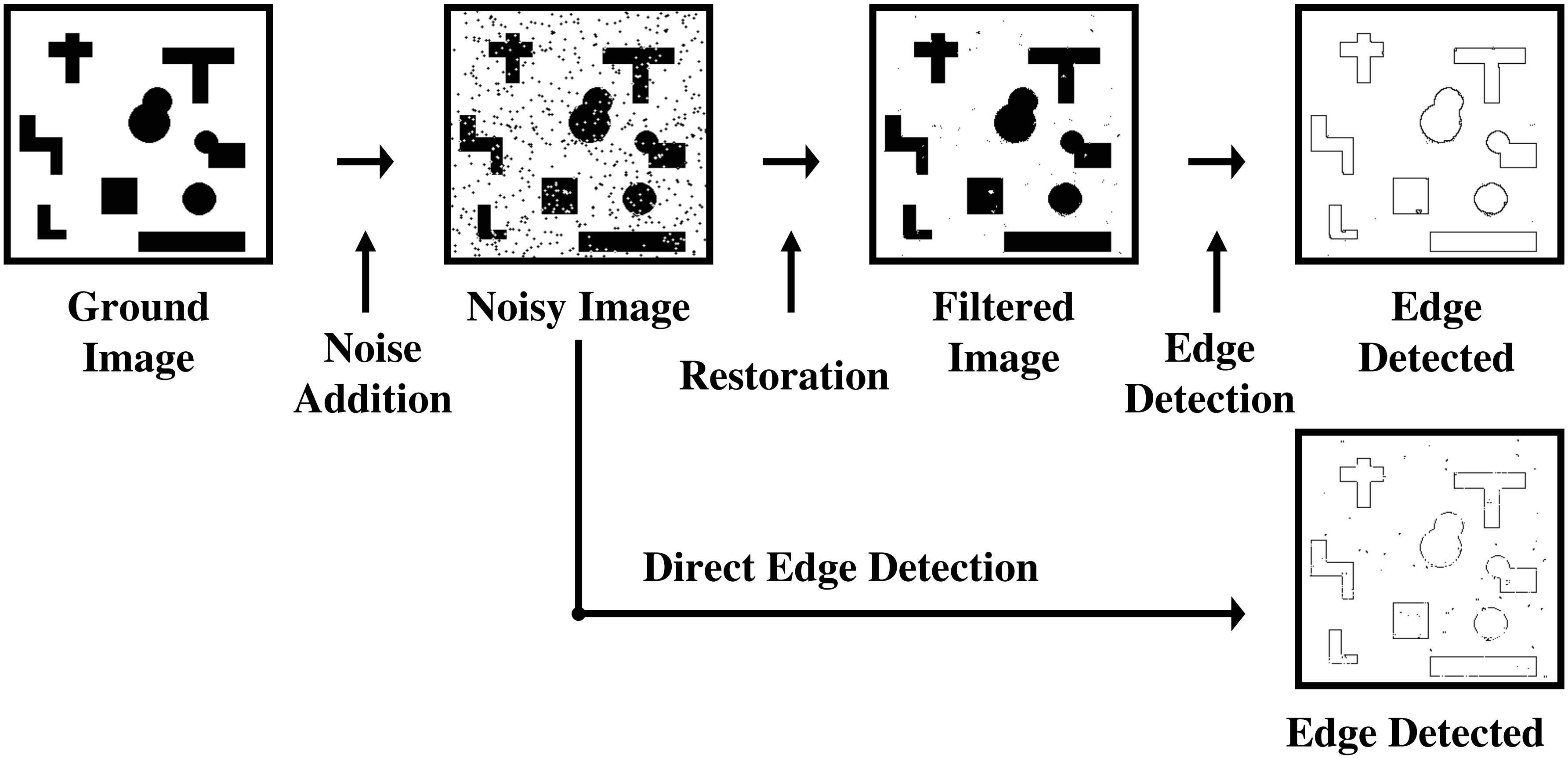
$$K(\psi) \in Q \Leftrightarrow \alpha \leq \psi \leq \beta$$

where $K(\alpha) = A$ and $K(\beta) = B$

- **All independent constraint is characterized by two operators α and β**
- **The pair (α,β) is called "Envelope"**



Noise Edge Detection



Restoration → a) **Machine design** of the restoration

ψ_{pac} designed by examples

→ b) **Human-machine design** of the restoration

$\psi_{con} = (\psi_{pac} \cap \beta) \cup \alpha$

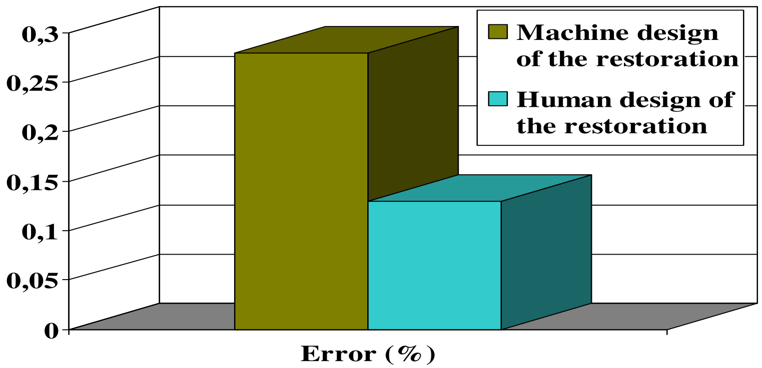
$\alpha = \delta_{B\oplus B}\varepsilon_{B\oplus B}\delta_{B}\varepsilon_{B}$ and $\beta = \varepsilon_{B\oplus B}\delta_{B\oplus B}\varepsilon_{B}\delta_{B}$

α and β are alternating sequential filters with

$P[\ \alpha(S) \leq I \leq \beta(S)\ ] \approx 1$

B is the 3x3 square

Machine design of the restoration	Human-Machine design of the restoration
0.28 %	0.13 %



Noise Edge Detection

Edge Detection

a) **Machine design over noisy images**

ζ_{pac} designed by examples from noisy images

b) **Human design after restoration**

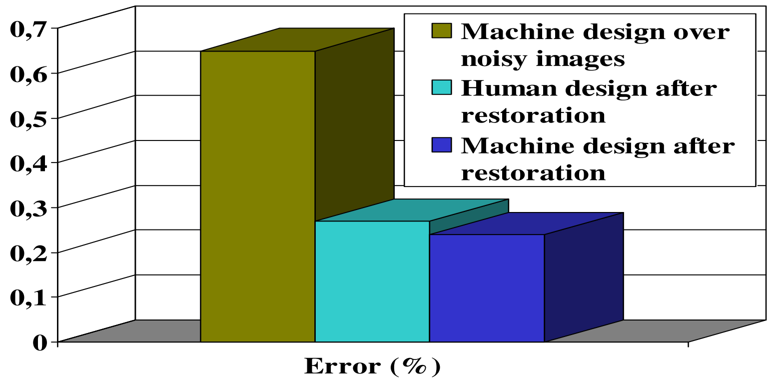
$\zeta = I_d - \varepsilon_B$

B is the 3x3 square

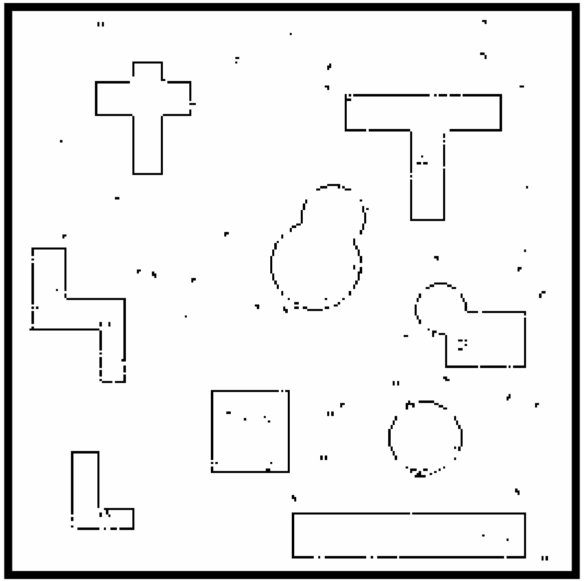
c) **Machine design after restoration**

ζ_{pac} designed by examples from restored images

Machine design over noisy images	Human design after restoration	Machine design after restoration
0.65 %	0.27 %	0.24 %

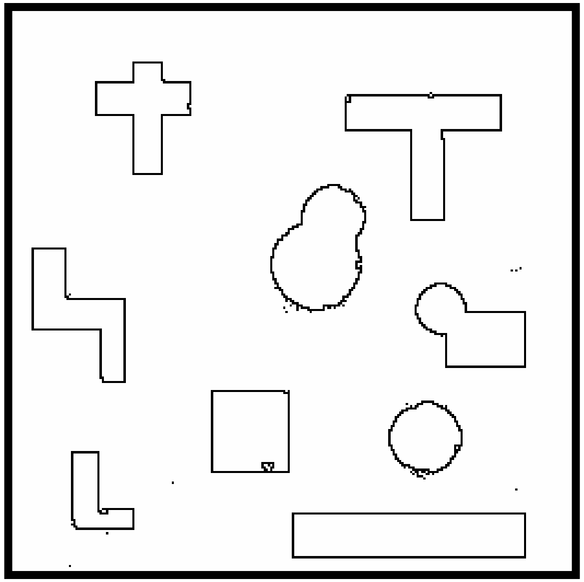


Noise Edge Detection



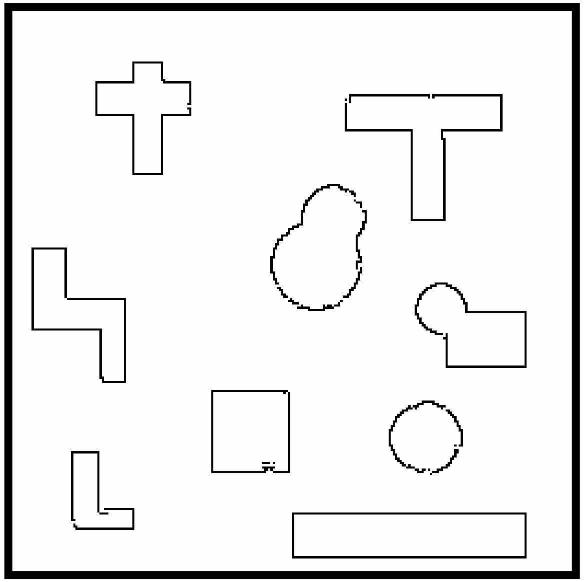
Machine design over noisy images

Error = 0.65%



Human design after restoration

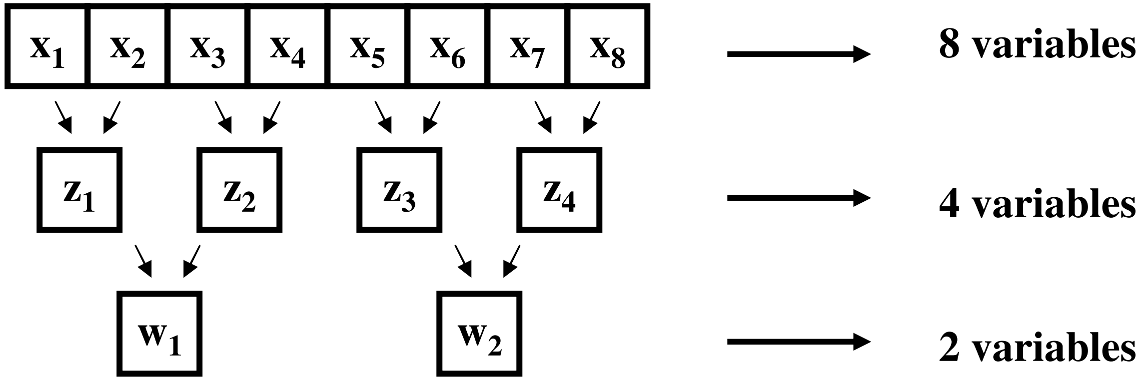
Error = 0.27%

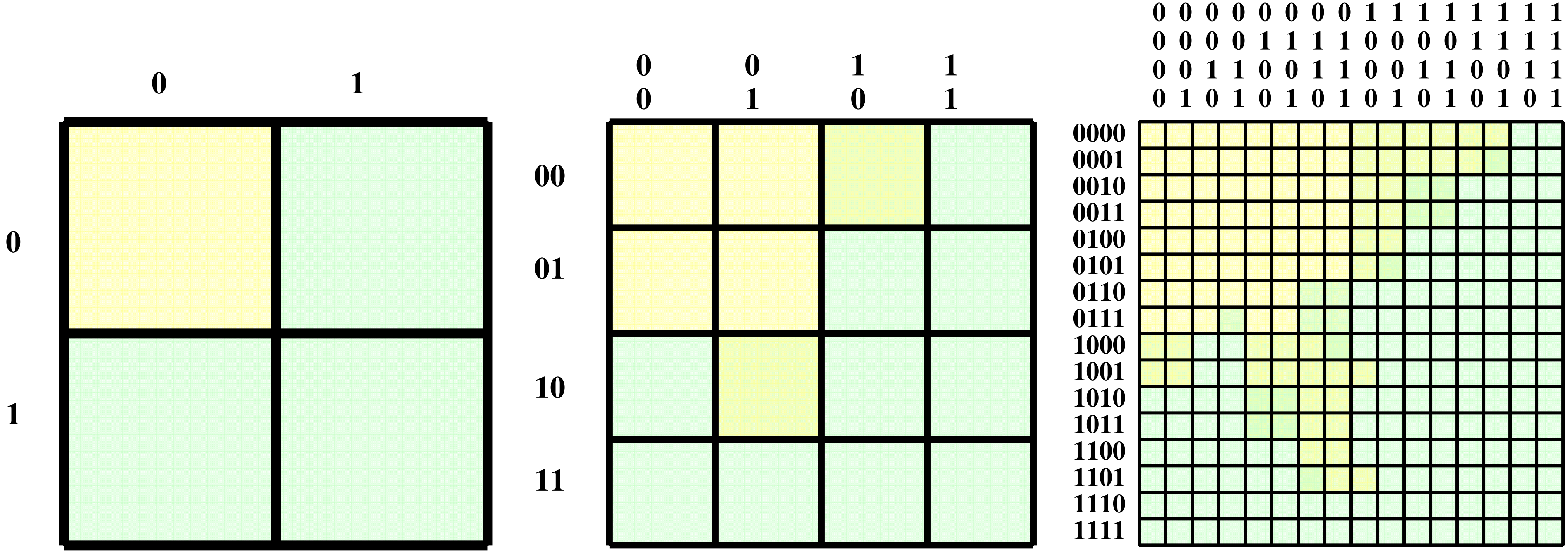


Machine design after restoration

Error = 0.24%

Multiresolution Constraint



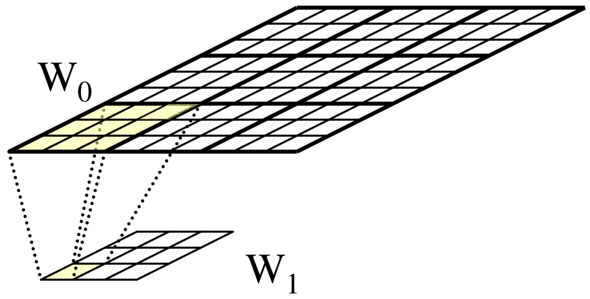


2 variables: $2^2=4$

4 variables: $2^4=16$

8 variables: $2^8=256$

Multiresolution Constraint



$D_1 = P(W_1)$

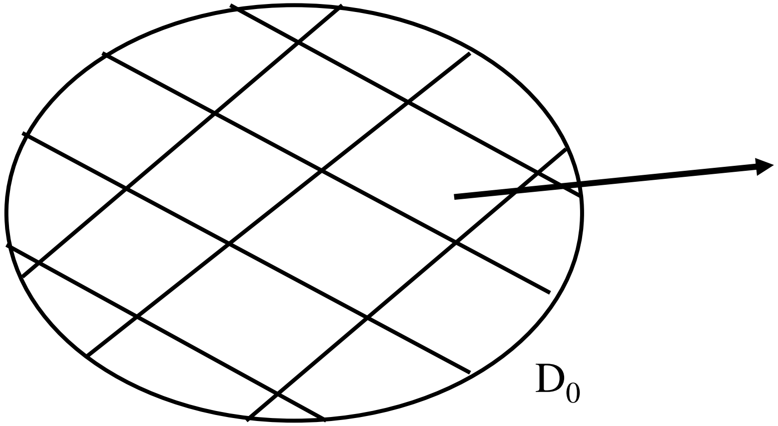
$D_0 = P(W_0)$

$\mathbf{z}_i = p_i(\mathbf{x}_{i1}, \ldots, \mathbf{x}_{i9})$, $\mathbf{z} = p(\mathbf{x})$, $p=(p_1, .. p_9)$

Let ϕ:$D_1 \rightarrow \{0,1\}$, it defines the operator Ψ_ϕ on D_o by

$\Psi_\phi (\mathbf{x}) = \phi(p(\mathbf{x}))$

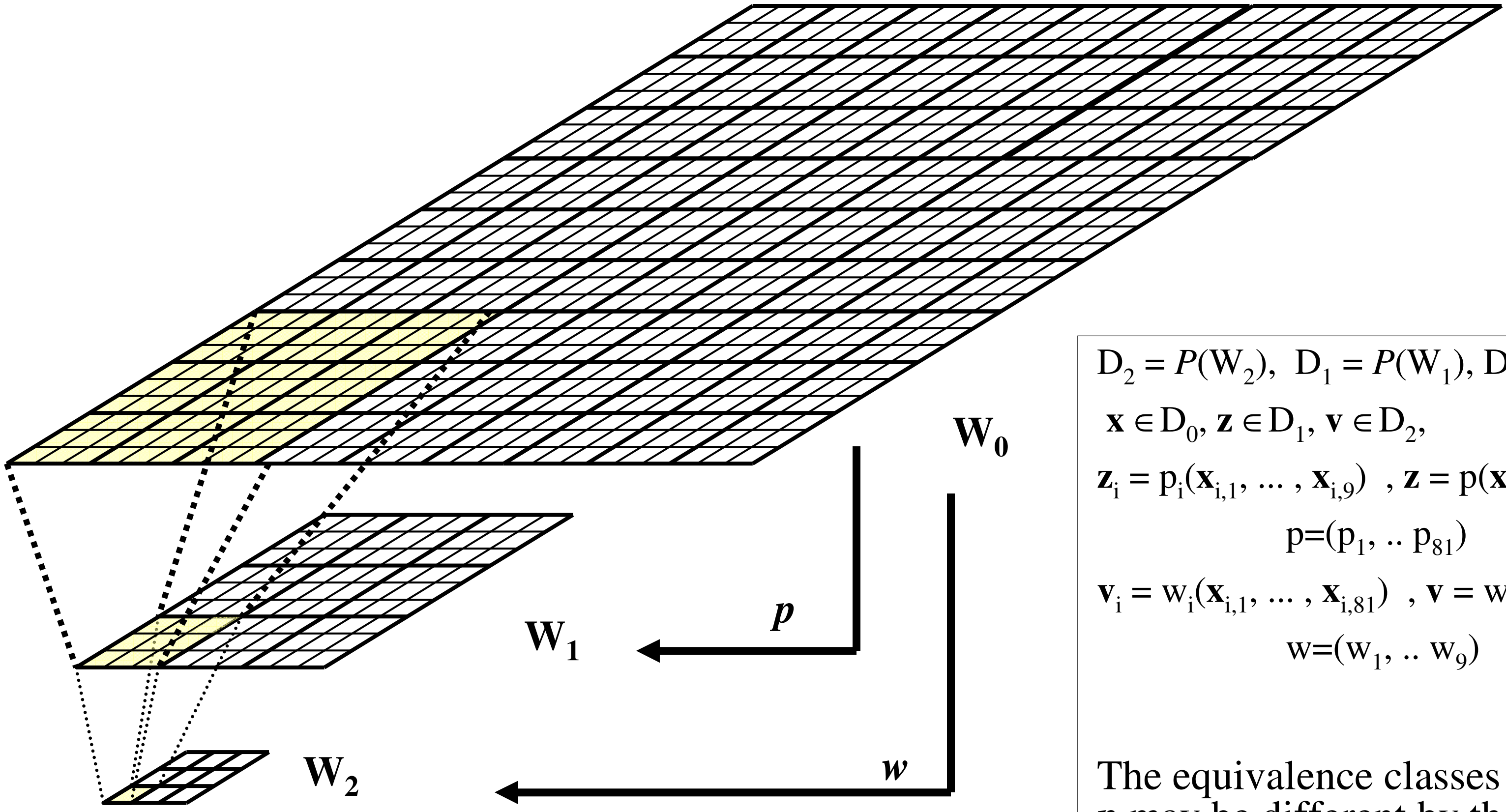
The operador Ψ_ϕ is constrained by resolution to D_1



Equivalence classes defined by

$p(\mathbf{x}) = p(\mathbf{y})$

Multiresolution Constraint



$D_2 = P(W_2)$, $D_1 = P(W_1)$, $D_0 = P(W_0)$

$\mathbf{x} \in D_0$, $\mathbf{z} \in D_1$, $\mathbf{v} \in D_2$,

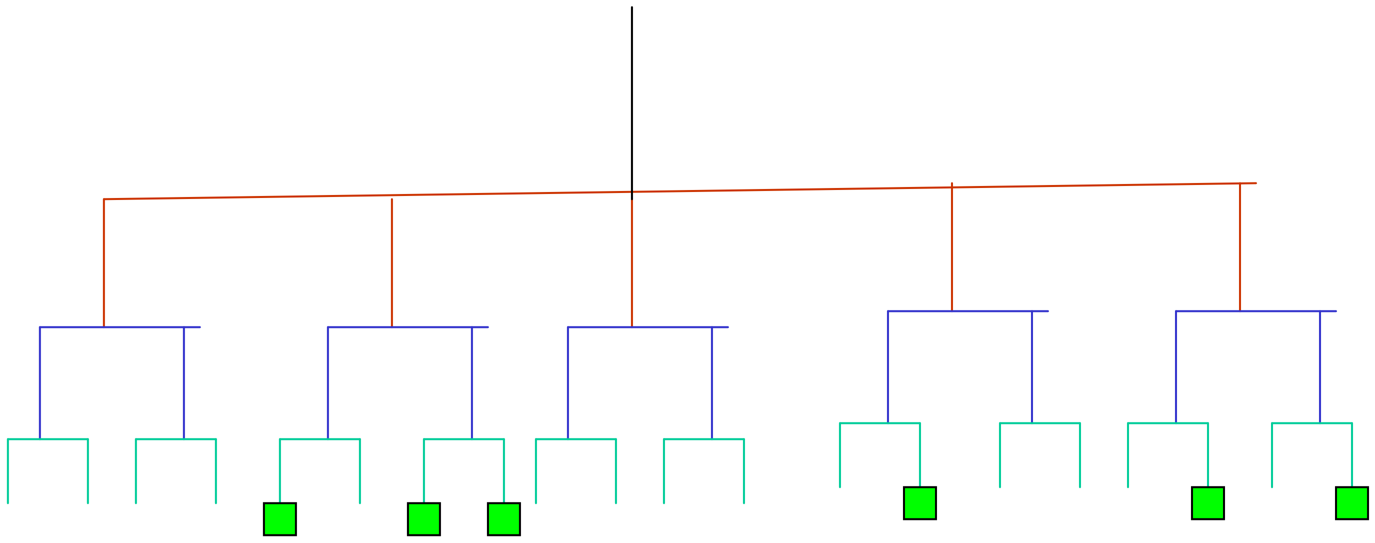
$\mathbf{z}_i = p_i(\mathbf{x}_{i,1}, \dots, \mathbf{x}_{i,9})$, $\mathbf{z} = p(\mathbf{x})$,

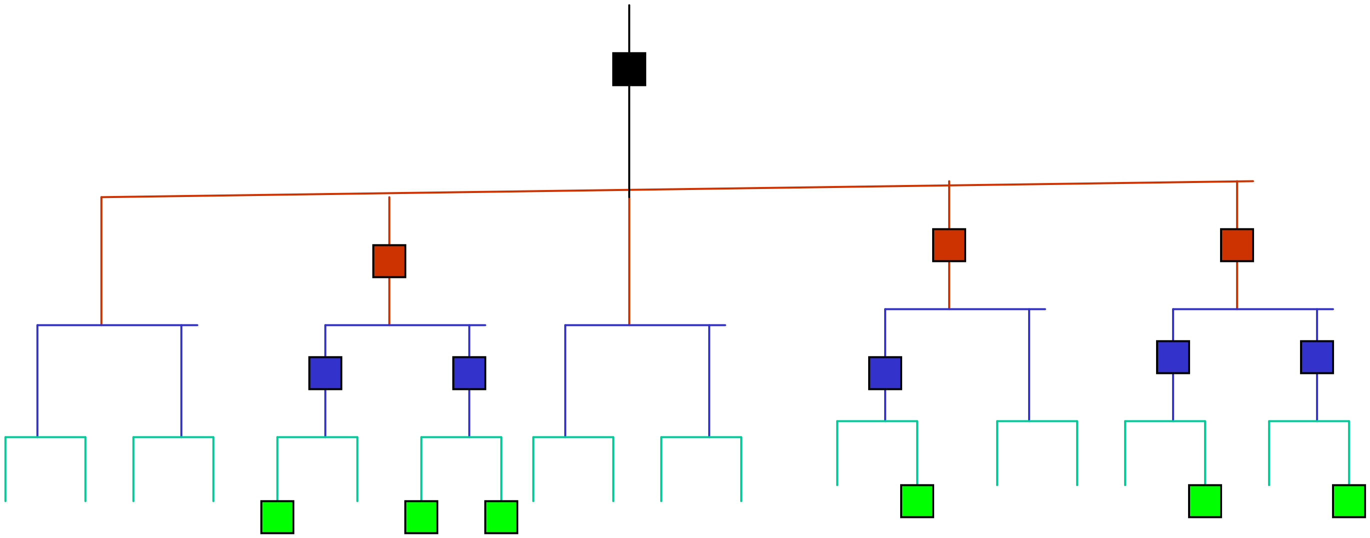
$p=(p_1, .. p_{81})$

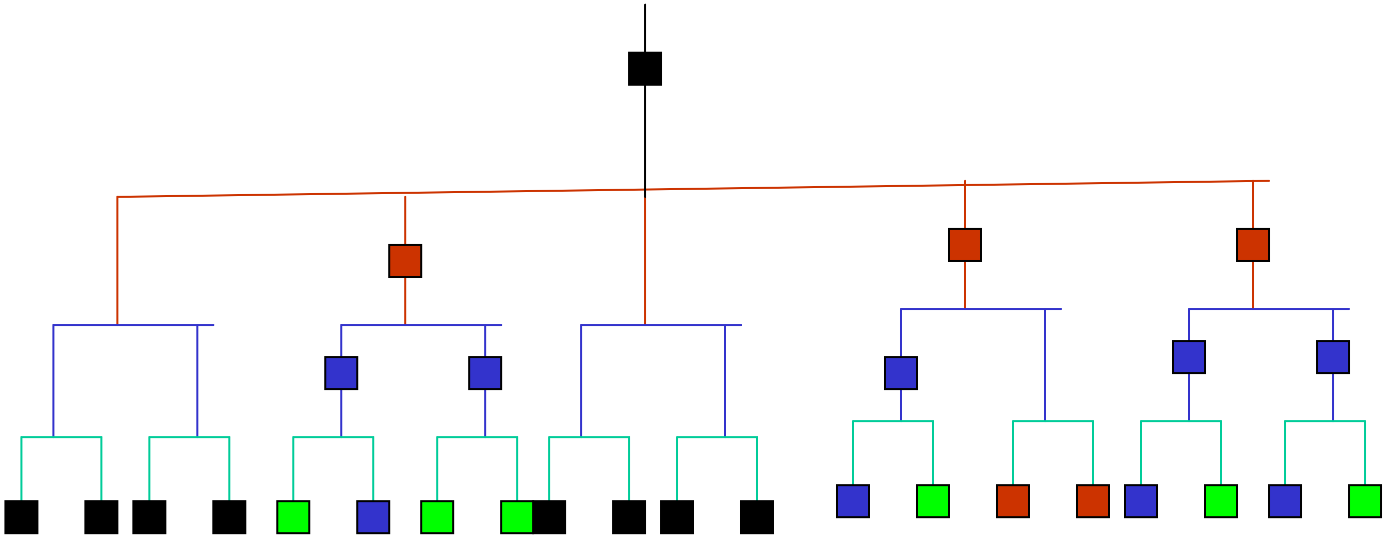
$\mathbf{v}_i = w_i(\mathbf{x}_{i,1}, \dots, \mathbf{x}_{i,81})$, $\mathbf{v} = w(\mathbf{x})$,

$w=(w_1, .. w_9)$

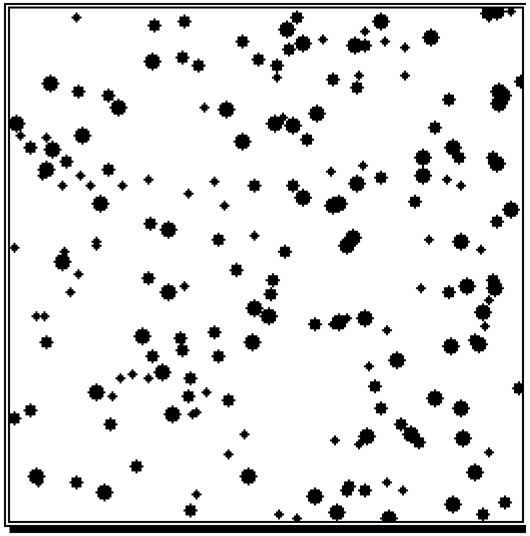
The equivalence classes defined by ***p*** may be different by the ones defined by ***w***.



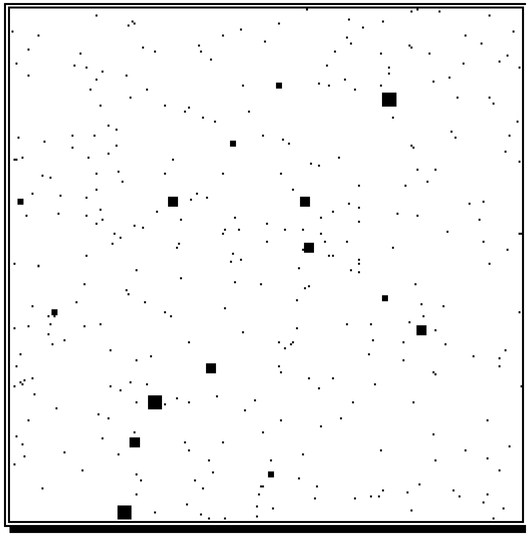




Multiresolution Noise



image



noise

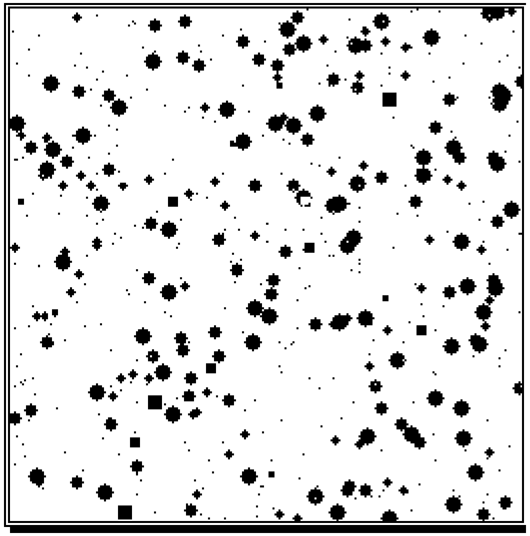
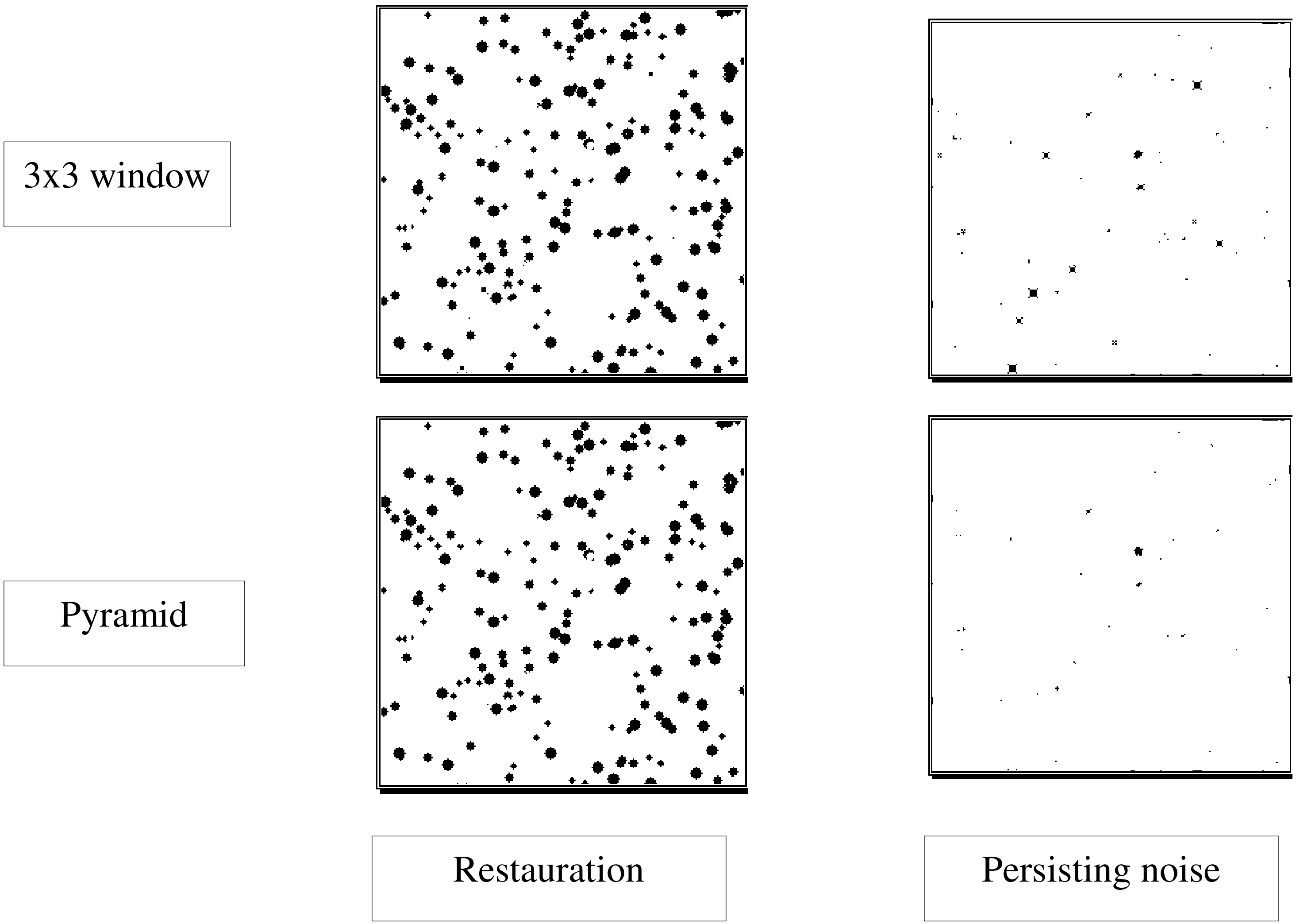
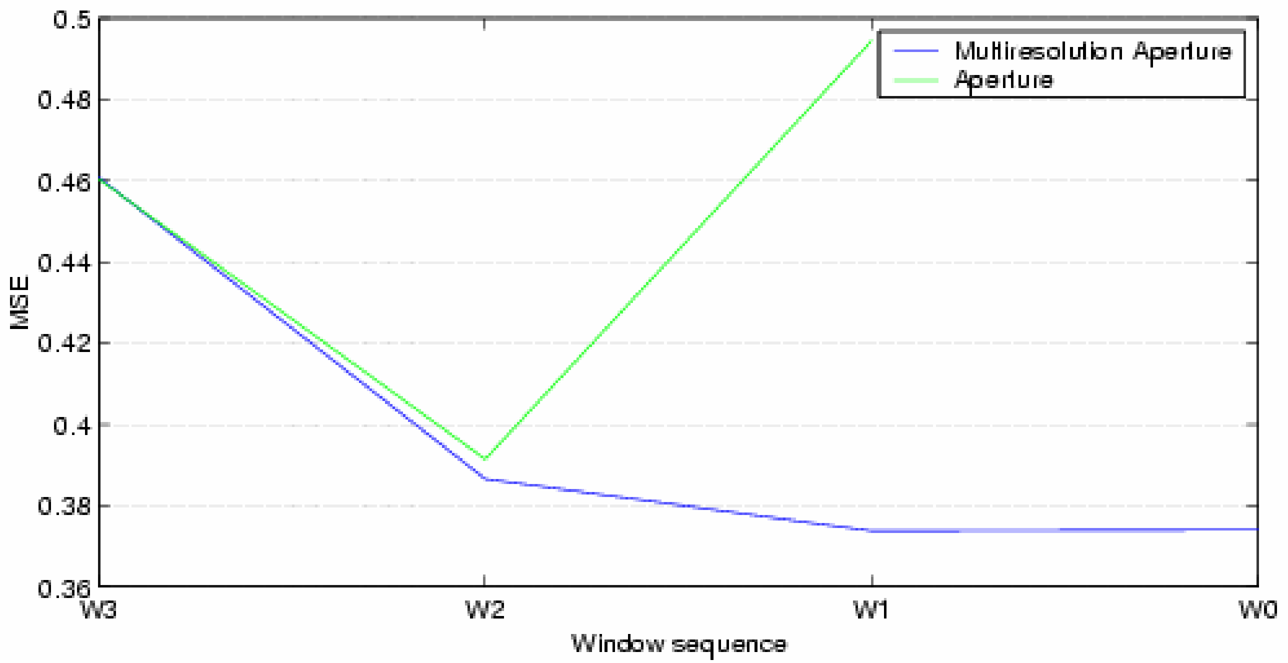


image + noise


3x3 window
Pyramid
Restauration
Persisting noise



Multiresolution Aperture
Aperture
MSE
0.5
0.48
0.46
0.44
0.42
0.4
0.38
0.36
W3
W2
W1
W0
Window sequence

Dynamical Systems

Finite Lattice Dinamical System

$\mathbf{x} : T \to \mathcal{L}^n$ $\mathbf{y} : T \to \mathcal{L}^m$ $\mathbf{x}[t] \in \mathcal{L}^n$

$\mathbf{u} : T \to \mathcal{L}^n$

$$\mathbf{x}[t+1] = \Phi_t(\mathbf{x}[t-N], ..., \mathbf{x}[t], ..., \mathbf{x}[t+N], \mathbf{u}[t-N], ..., \mathbf{u}[t], ..., \mathbf{u}[t+N])$$

$$\mathbf{y}[t] = \Psi_t(\mathbf{x}[t-N], ..., \mathbf{x}[t], ..., \mathbf{x}[t+N], \mathbf{u}[t-N], ..., \mathbf{u}[t], ..., \mathbf{u}[t+N])$$

$S(\Phi_t, \Psi_t)$

$\Phi_t : \mathcal{L}^{2(2N+1)n} \to \mathcal{L}^n$

$\Psi_t : \mathcal{L}^{2(2N+1)n} \to \mathcal{L}^m$

Representation

$$\mathbf{x}_j[t+1] = \phi_{t,j}(\mathbf{x}[t-N], \ldots, \mathbf{x}[t], \ldots, \mathbf{x}[t+N], \mathbf{u}[t-N], \ldots, \mathbf{u}[t], \ldots, \mathbf{u}[t+N])$$

$$\mathbf{y}_k[t] = \psi_{t,k}(\mathbf{x}[t-N], \ldots, \mathbf{x}[t], \ldots, \mathbf{x}[t+N], \mathbf{u}[t-N], \ldots, \mathbf{u}[t], \ldots, \mathbf{u}[t+N])$$

$$\mathbf{x}_j[t] \in \mathcal{L}$$

$$\phi_{t,j} : \mathcal{L}^{2(2N+1)n} \to \mathcal{L}$$

$$\psi_{t,k} : \mathcal{L}^{2(2N+1)n} \to \mathcal{L}$$

The component functions have canonical morphological representations

System: input-output

$$\mathbf{y}[t] = S_{(\mathbf{x}[0],\ldots,\mathbf{x}[N],\ldots,\mathbf{x}[2N])}(\Phi_t, \Psi_t)(\mathbf{u}[t-N], \ldots, \mathbf{u}[t], \ldots, \mathbf{u}[t+N])$$

$$\mathbf{y} = S_{(\mathbf{x}[0],\ldots,\mathbf{x}[N],\ldots,\mathbf{x}[2N])}(\Phi_t, \Psi_t)(\mathbf{u})$$

Filter

$$\mathbf{u} : T \rightarrow \mathcal{L}^n \qquad \mathbf{y} : T \rightarrow \mathcal{L}^m$$

$$\Gamma : \mathcal{L}^{(2N+1)n} \rightarrow \mathcal{L}^m$$

$$\mathbf{y}[t] = \Gamma_t(\mathbf{u}[t-N], \ldots, \mathbf{u}[t], \ldots, \mathbf{u}[t+N])$$

For example, processing of motion images.

Input-free systems

$$\mathbf{x}[t+1] = \Phi_t(\mathbf{x}[t-N], ..., \mathbf{x}[t], ..., \mathbf{x}[t+N])$$

$$\mathbf{y}[t] = \Psi_t(\mathbf{x}[t-N], ..., \mathbf{x}[t], ..., \mathbf{x}[t+N])$$

Causal systems

$$\mathbf{x}[t+1] = \Phi_t(\mathbf{x}[t-N], ..., \mathbf{x}[t], \mathbf{u}[t-N], ..., \mathbf{u}[t])$$

$$\mathbf{y}[t] = \Psi_t(\mathbf{x}[t-N], ..., \mathbf{x}[t], \mathbf{u}[t-N], ..., \mathbf{u}[t])$$

Time translation invariant systems

$$\Phi : \mathcal{L}^{2(2N+1)n} \to \mathcal{L}^n$$

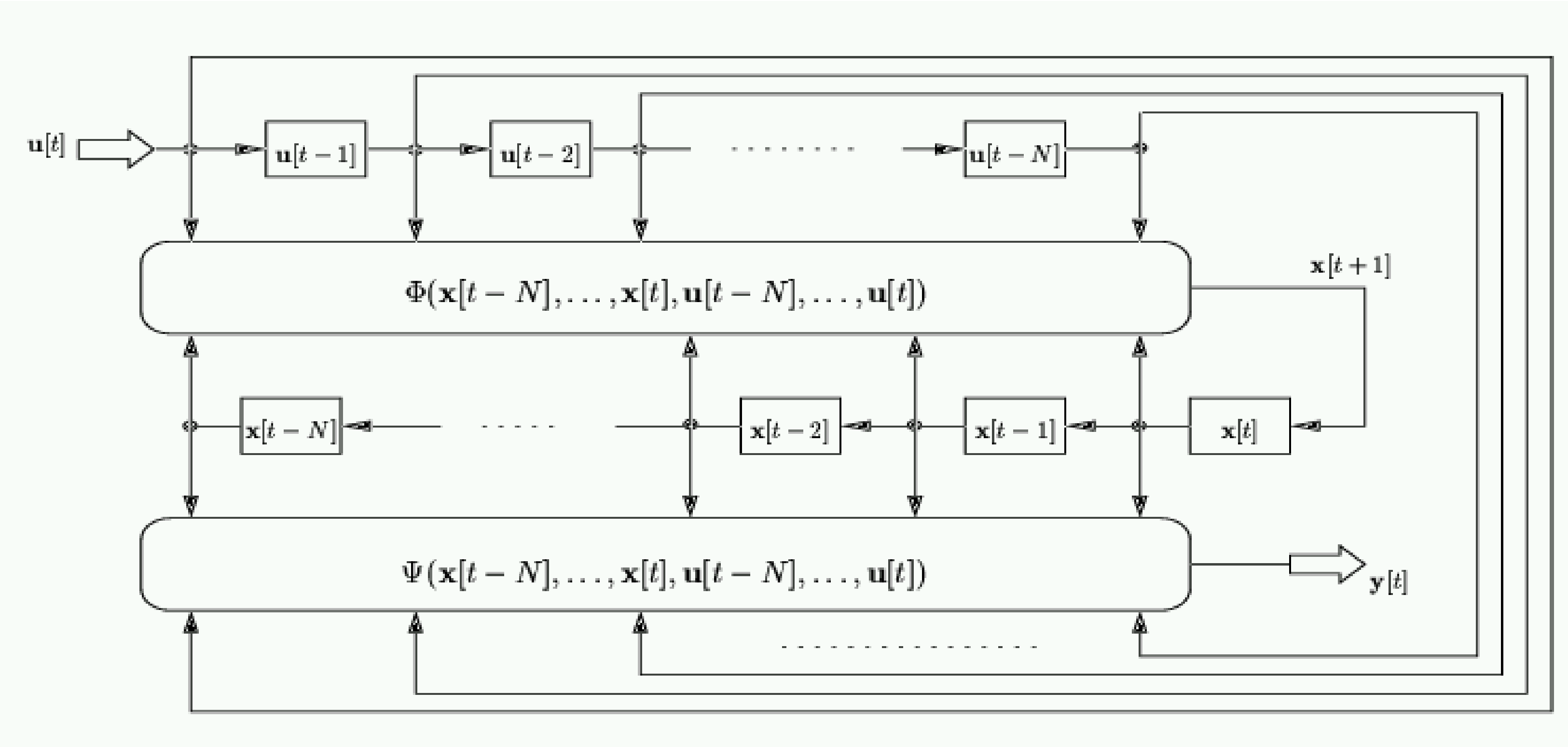
$$\Psi : \mathcal{L}^{2(2N+1)n} \to \mathcal{L}^m$$

$$S(\Phi, \Psi)$$

$$\Phi_t = \Phi$$

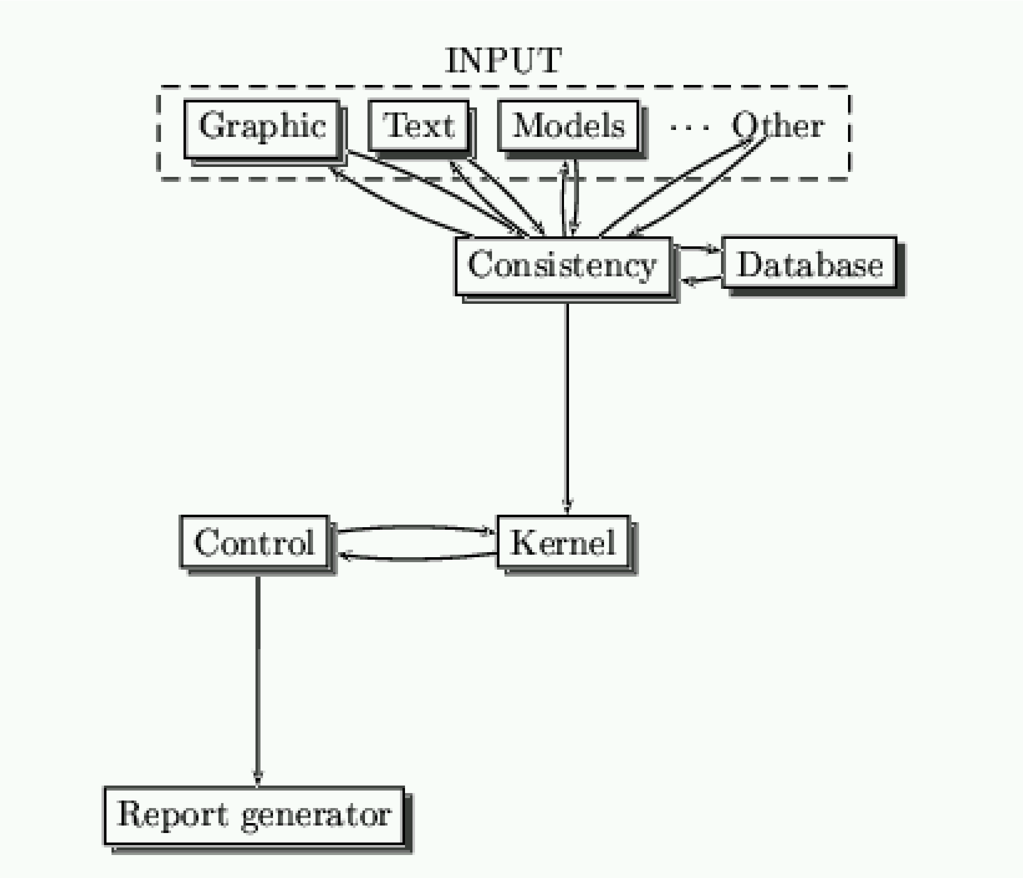
$$\Psi_t = \Psi$$

Causal, time translation invariant

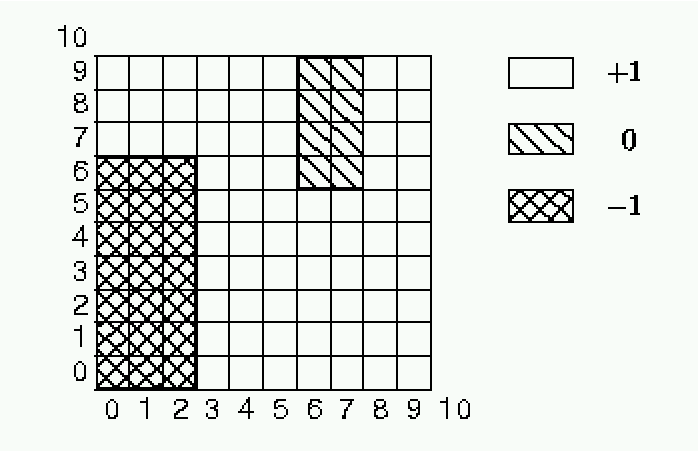


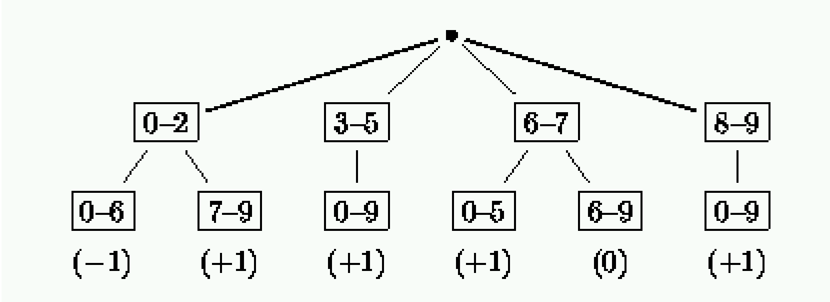
A simulator for chain dynamical systems

Simulator Architecture



Functions Representation

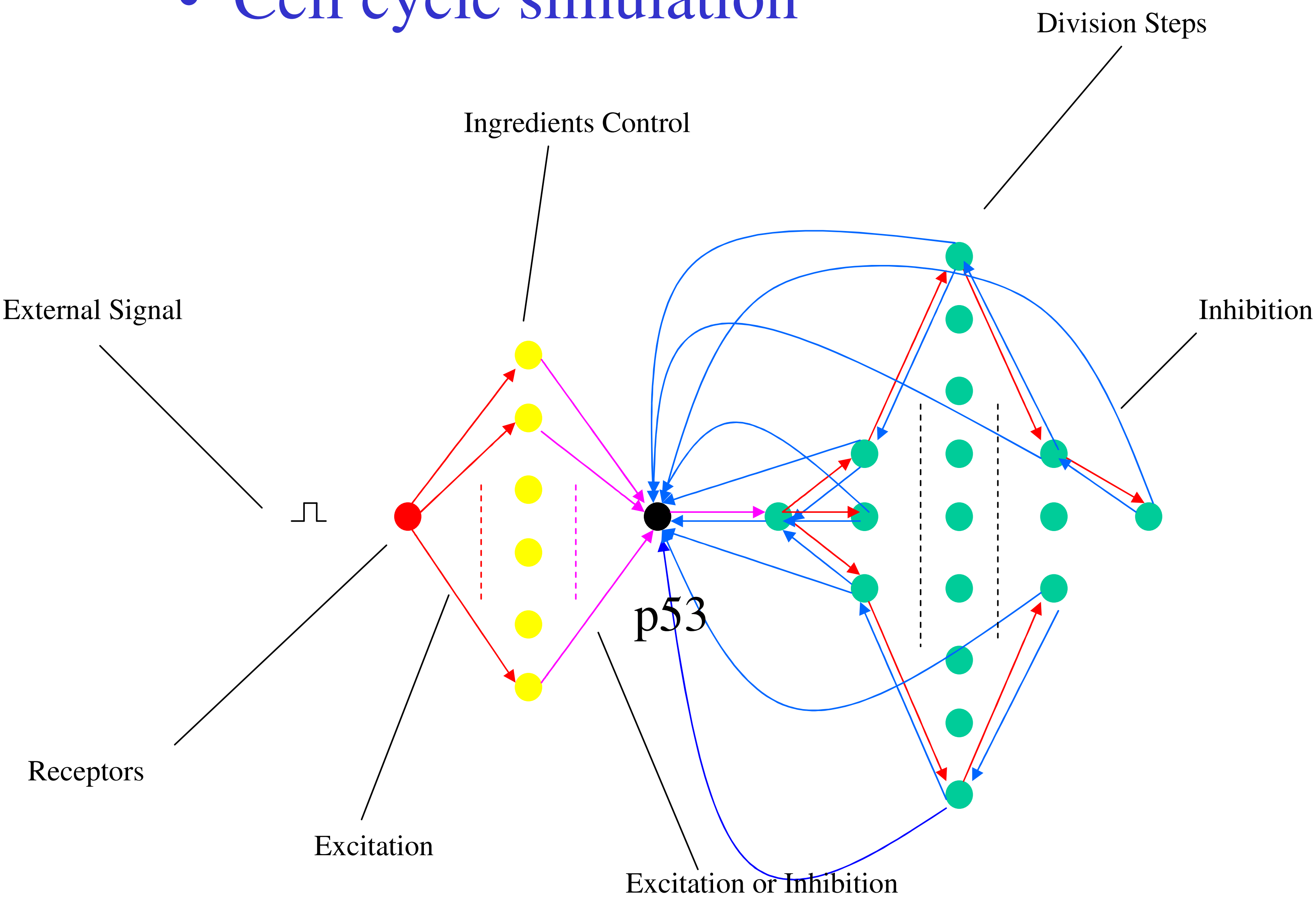


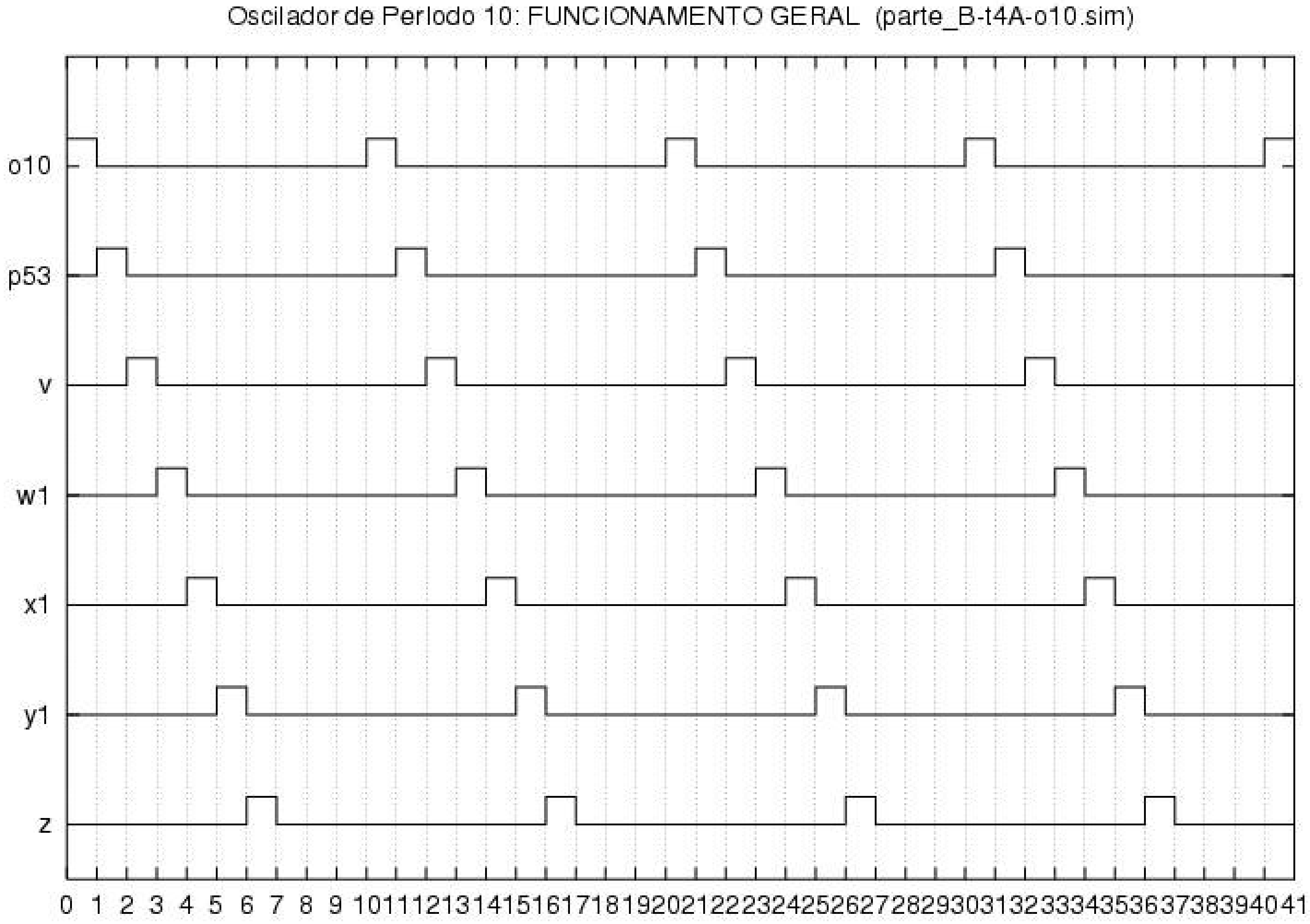


System Description

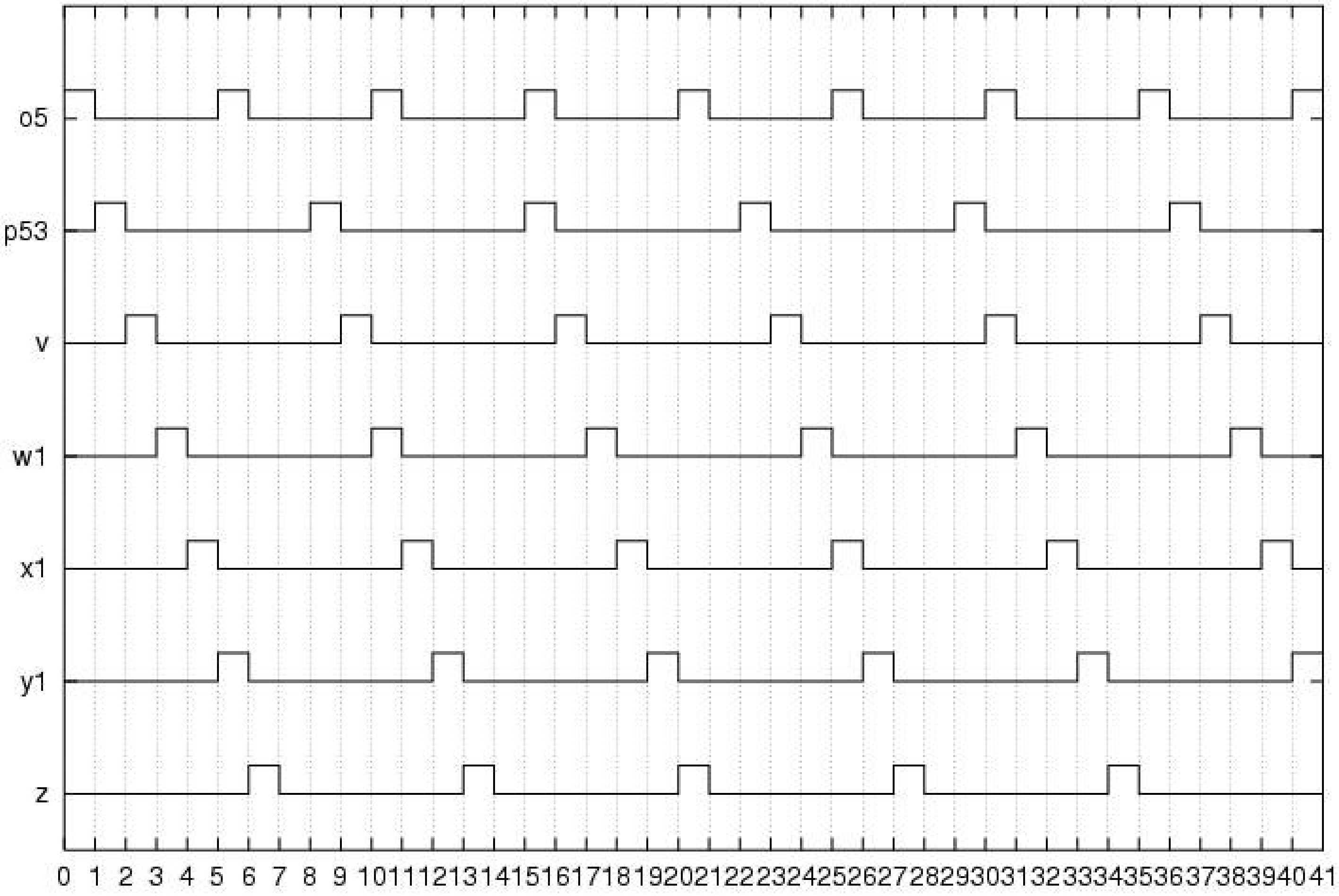
```
Gene g1:{1.00 f1 g1[4] g1[3] g1[2] g1[1] g2[6] g2[5] g2[4] g2[3] g2[2] g2[1];};
Gene g2:{1.00 f2 g1[4] g1[3] g1[2] g1[1] g2[6] g2[5] g2[4] g2[3] g2[2] g2[1];};

# Function definitions
# ----------------------


def f1: [0000000000..1000101010]: 1;
def f2: [0000000000..0111100000]: 1;

# History
# -------------------------
hist g1: [0 0 0 0 0 0 0 0 0 0 0 0];
hist g2: [0 0 0 0 0 0 0 0 0 0 0 0];

end
```

- Cell cycle simulation

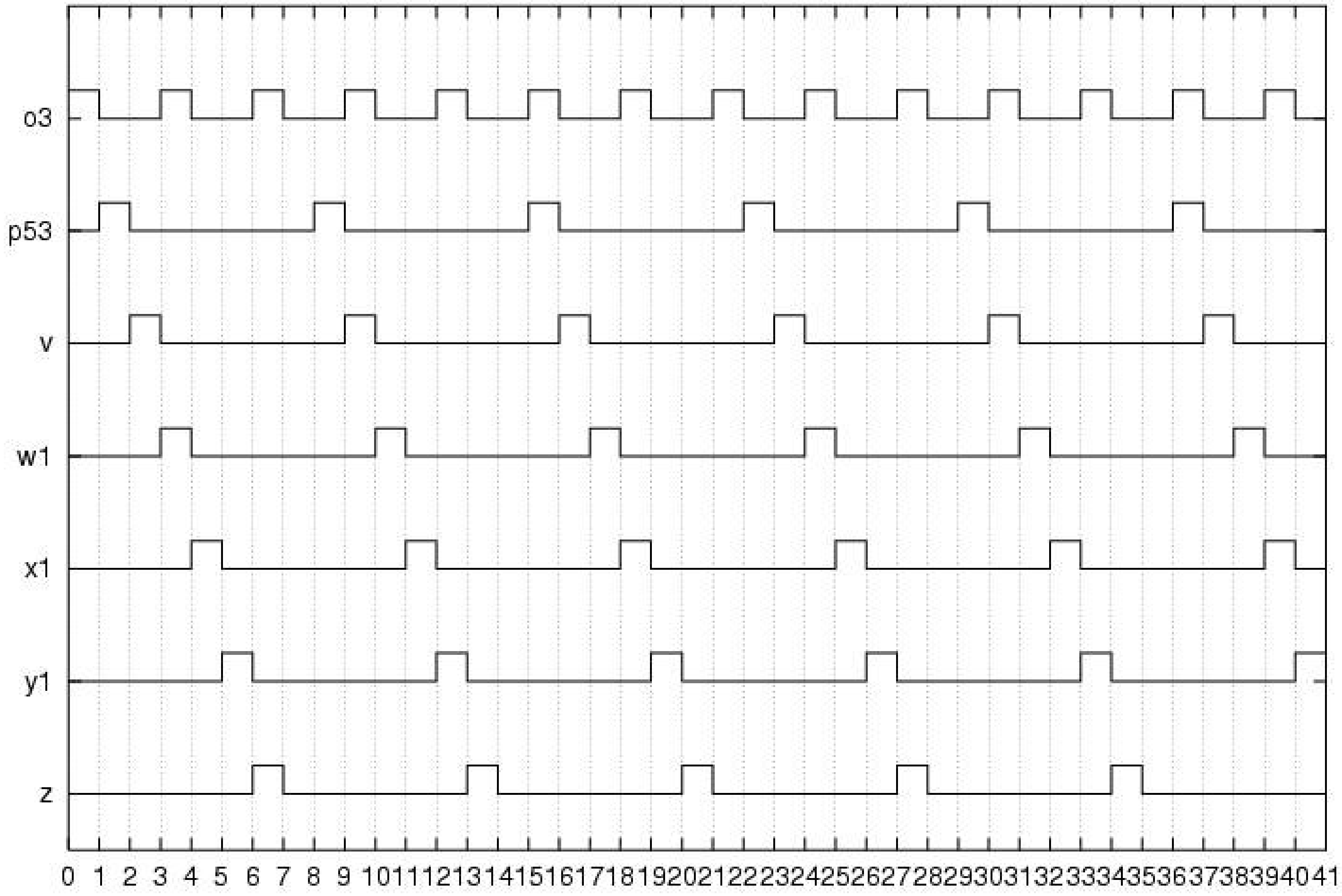


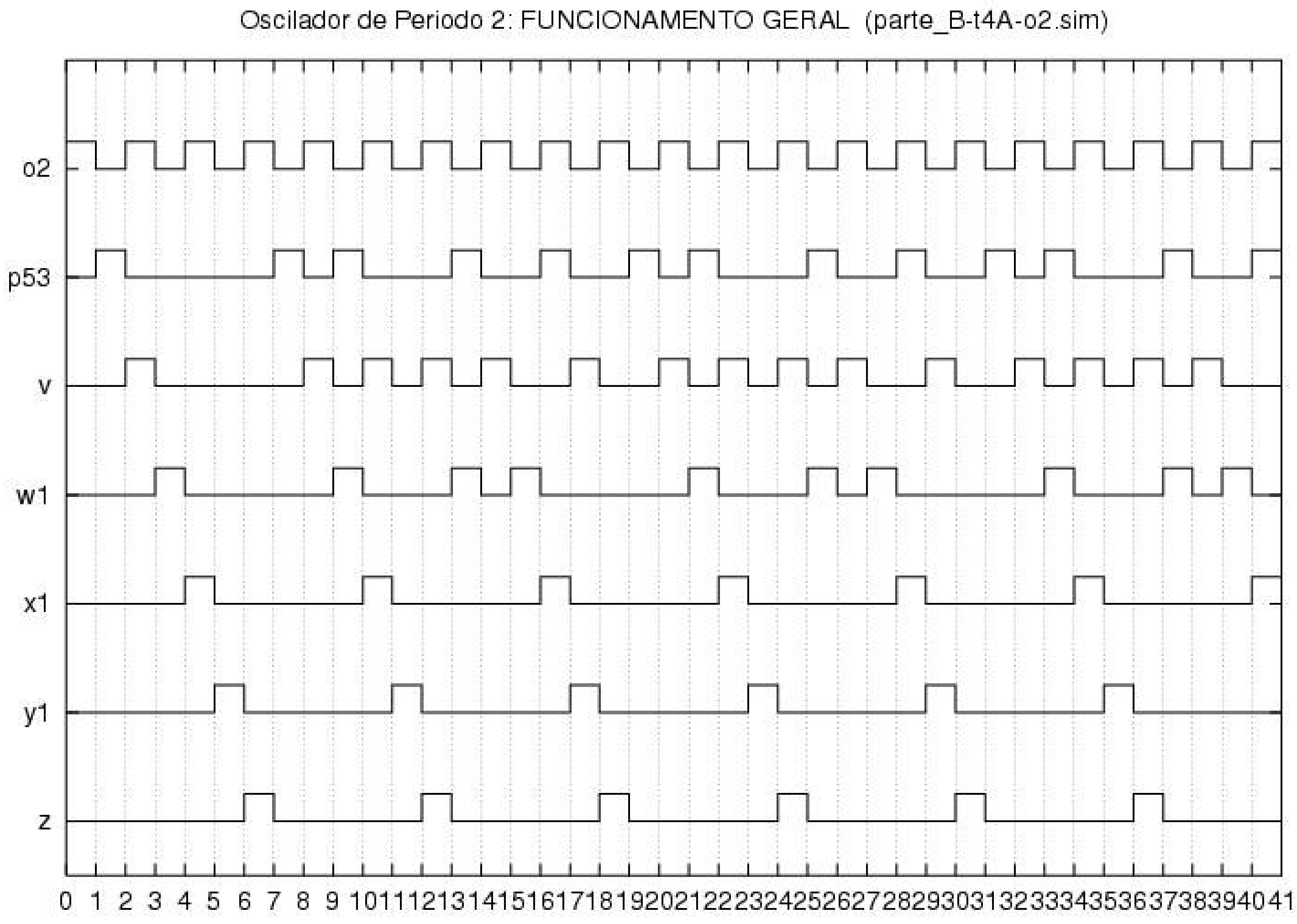

Oscilador de Período 10: FUNCIONAMENTO GERAL (parte_B-t4A-o10.sim)
o10
p53
v
w1
x1
y1
z
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41

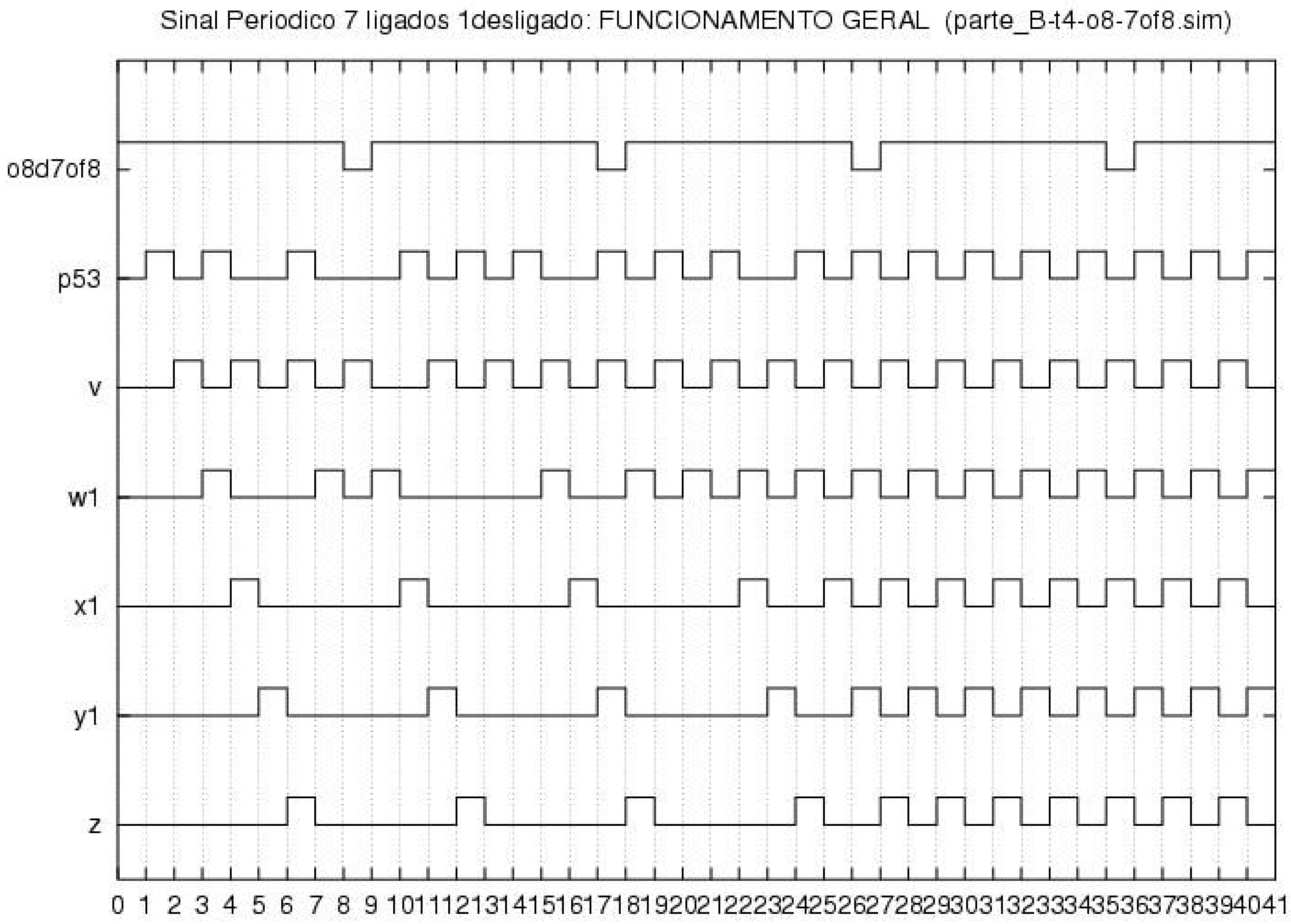
Oscilador de Periodo 5: FUNCIONAMENTO GERAL (parte_B-t4A-o5.sim)



Oscilador de Periodo 3: FUNCIONAMENTO GERAL (parte_B-t4A-o3.sim)



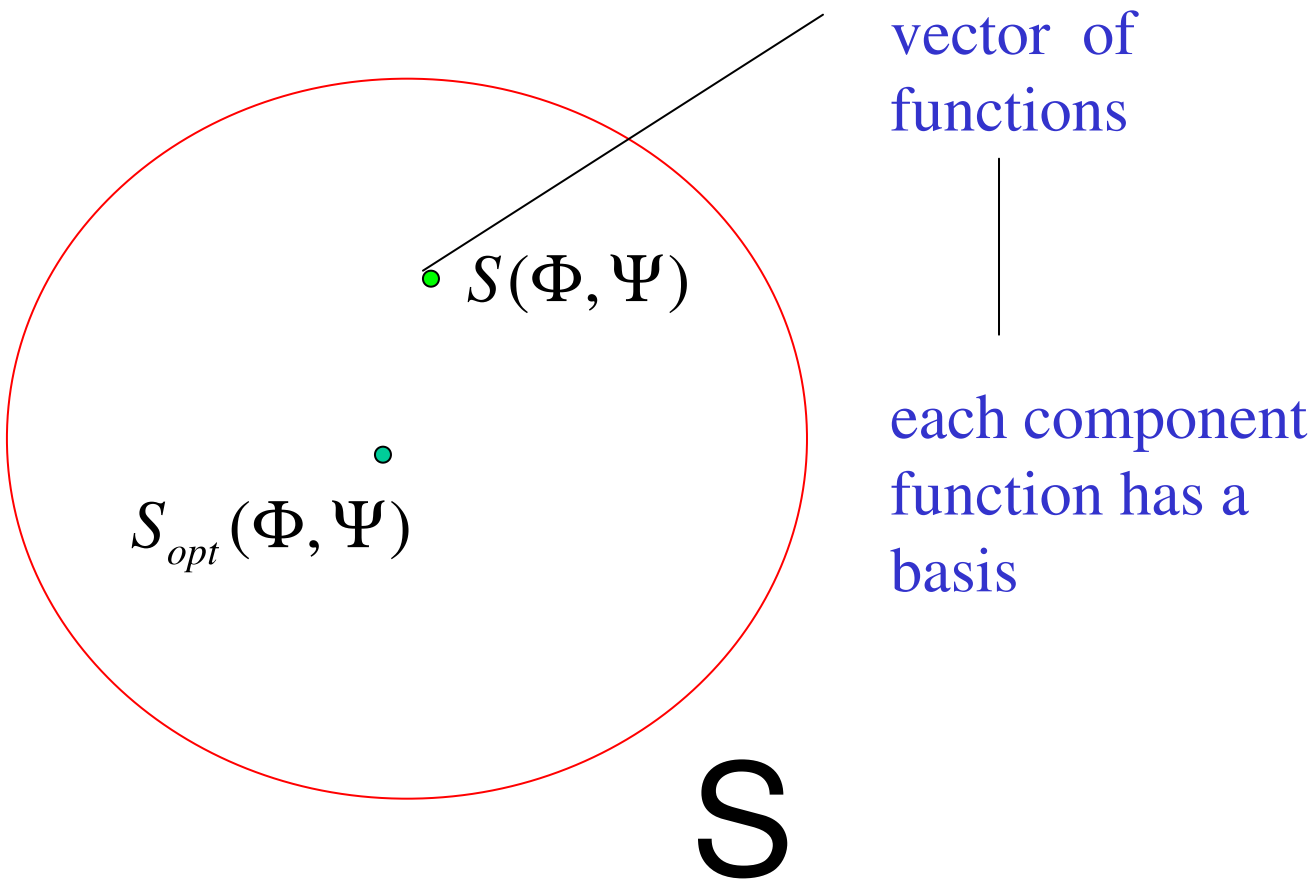

Oscilador de Periodo 2: FUNCIONAMENTO GERAL (parte_B-t4A-o2.sim)
o2
p53
v
w1
x1
y1
z
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41



Sinal Periodico 7 ligados 1desligado: FUNCIONAMENTO GERAL (parte_B-t4-o8-7of8.sim)
o8d7of8
p53
v
w1
x1
y1
z
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41

A model for system identification

Model


vector of functions
$S(\Phi, \Psi)$
each component function has a basis
$S_{opt}(\Phi, \Psi)$
S

System Error

$$Er\big(S(\Phi,\Psi)\big) \quad = \quad E\Big[\, l\big(\, S_{(\mathbf{X}[0],\ldots,\mathbf{X}[N])}(\Phi,\Psi)(\mathbf{U}[t-N],\ldots,\mathbf{U}[t-1],\mathbf{U}[t]),\mathbf{I}[t]\,\big)\Big]$$

$$l : \mathcal{L}^m \times \mathcal{L}^m \rightarrow \Re^+$$

Stationary Conditions

$$P(\mathbf{x}[t-N],...,\mathbf{x}[t-1],\mathbf{x}[t],\mathbf{u}[t-N],...,\mathbf{u}[t-1],\mathbf{u}[t],\mathbf{i}[t]) = p$$

$$\begin{aligned} Er\left(S(\Phi,\Psi)\right) &= E\left[\, l\left(S_{(\mathbf{X}[0],...,\mathbf{X}[N])}(\Phi,\Psi)(\mathbf{U}[t-N],...,\mathbf{U}[t-1],\mathbf{U}[t]),\mathbf{I}[t]\right)\right] \\ &= \sum_{(\mathbf{x}[t-N],...,\mathbf{x}[t],\mathbf{u}[t-N],...,\mathbf{u}[t],\mathbf{i}[t])\in\mathcal{L}^{2(N+1)n}\times\mathcal{L}^m} l\left(S_{(\mathbf{x}[t-N],...,\mathbf{x}[t])}(\Phi,\Psi)(\mathbf{u}[t-N],...,\mathbf{u}[t]),\mathbf{i}[t]\right)\times \\ &\qquad p(\mathbf{x}[t-N],...,\mathbf{x}[t],\mathbf{u}[t-N],...,\mathbf{u}[t],\mathbf{i}[t]) \end{aligned}$$

Component Error

$$Er_k\left[\,S(\Phi,\Psi)\,\right] = E\Big[\,l_k\big(\,S_{(\mathbf{X}[0],\ldots,\mathbf{X}[N])}(\Phi,\Psi)(\mathbf{U}[t-N],\ldots,\mathbf{U}[t-1],\mathbf{U}[t])_k,\mathbf{I}_k[t]\,\big)\,\Big].$$

$$l_k : \mathcal{L} \times \mathcal{L} \rightarrow \Re^{+}$$

Additive Loss Function

$$l = \sum_{k=1}^{m} c_k l_k \qquad c_k \in \Re^+$$

$$Er(S(\Phi, \Psi)) = \sum_{k=1}^{m} Er_k\left[\, S(\Phi, \Psi) \,\right]$$

$$e_{MAE}(\mathbf{a}, \mathbf{b}) = \sum_{k=1}^{m} |\, \mathbf{a}_k - \mathbf{b}_k \,|$$

$$e_{MAE} = \sum_{k=1}^{m} e_{kMAE}$$

$$\mathbf{a}, \mathbf{b} \in \{0, 1\}^m$$

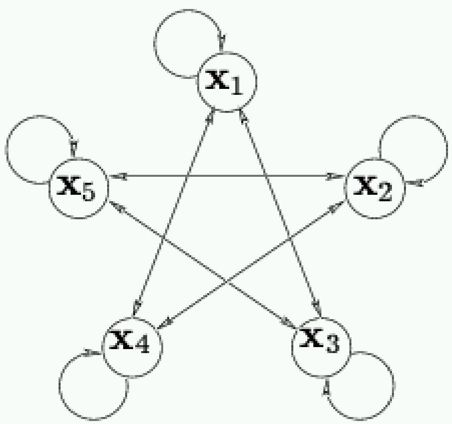
$$e_{kMAE}(a, b) = |a - b|$$

Independence Condition

- Under additive loss function optimize the system error is equivalent to optimize the system components error
- The problem of system identification is reduced to a family of problems of lattice operator design.

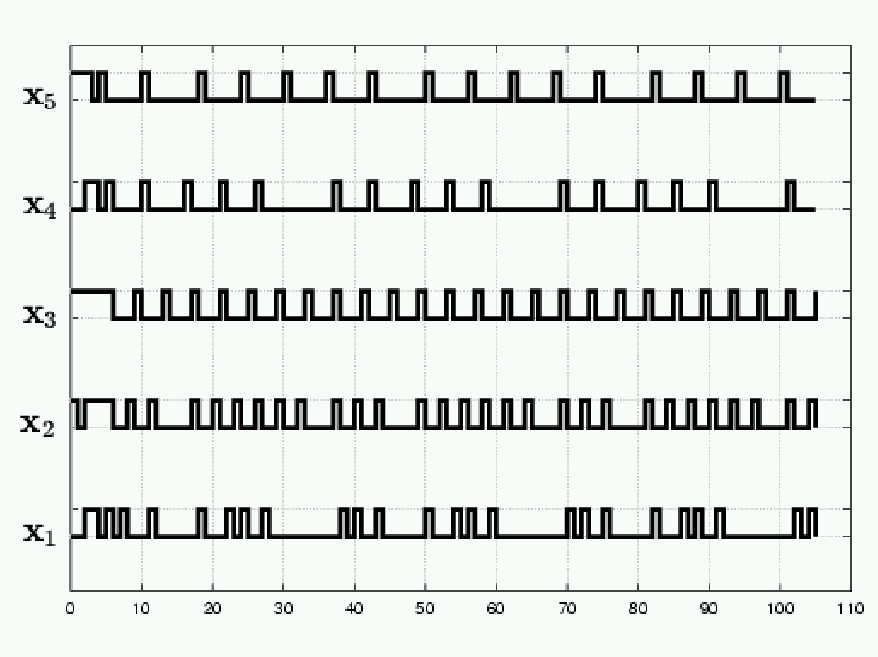
Identification of Dynamical Systems

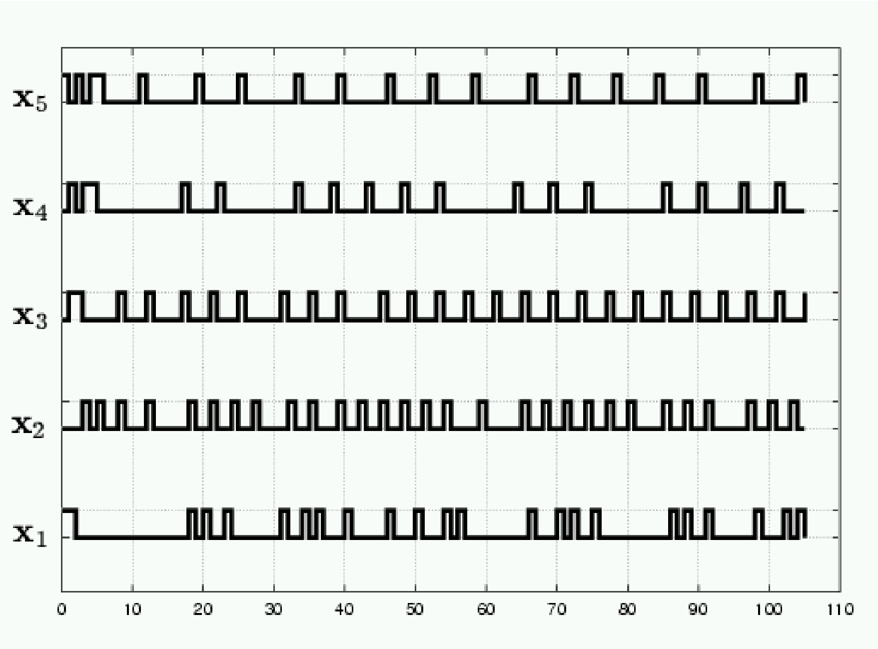
A Boolean System



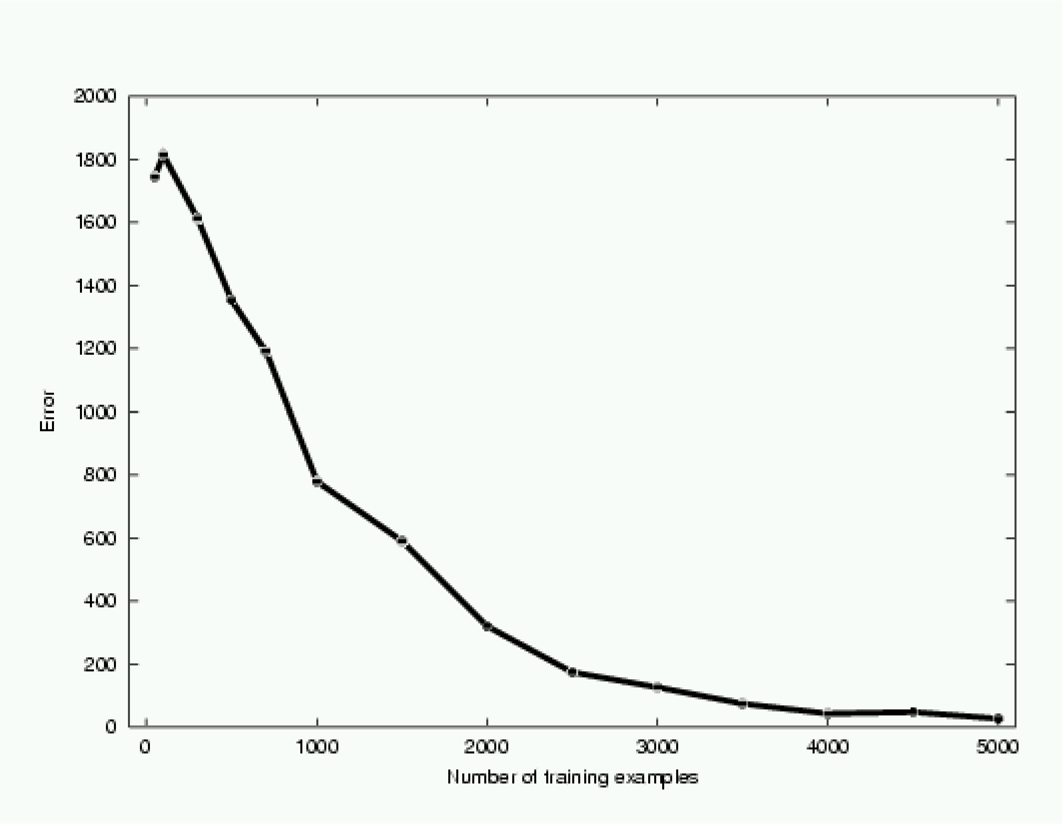
$$\mathbf{x}_1[t+1]=1 \iff \begin{cases} \mathbf{x}_1[t]=0 \\ \text{and} \\ \Big[\Big((\mathbf{x}_3[t]=1 \text{ or } \mathbf{x}_3[t-1]=1 \text{ or } \mathbf{x}_3[t-2]=1) \text{ and} \\ \quad (\mathbf{x}_4[t]=1 \text{ or } \mathbf{x}_4[t-1]=1 \text{ or } \mathbf{x}_4[t-2]=1) \Big) \\ \quad \text{or} \\ \quad \Big(\mathbf{x}_3[t]=\mathbf{x}_3[t-1]=\mathbf{x}_3[t-2]=\mathbf{x}_3[t-3]=\mathbf{x}_3[t-4]=0 \text{ and} \\ \quad \mathbf{x}_4[t]=\mathbf{x}_4[t-1]=\mathbf{x}_4[t-2]=\mathbf{x}_4[t-3]=\mathbf{x}_4[t-4]=0 \Big) \Big] \end{cases}$$

System simulation

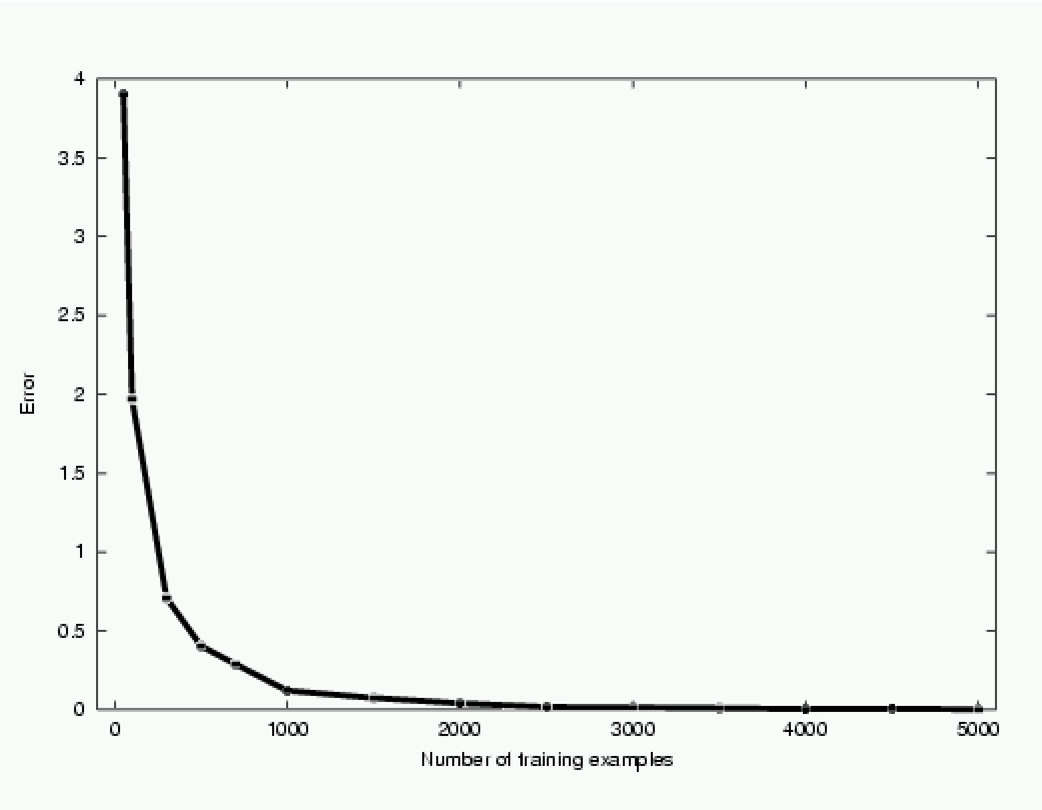


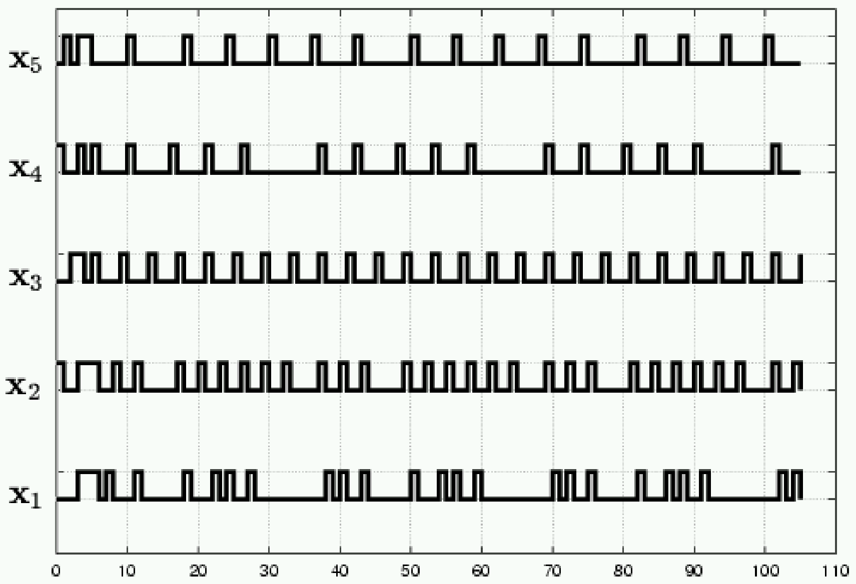


System identification: system error

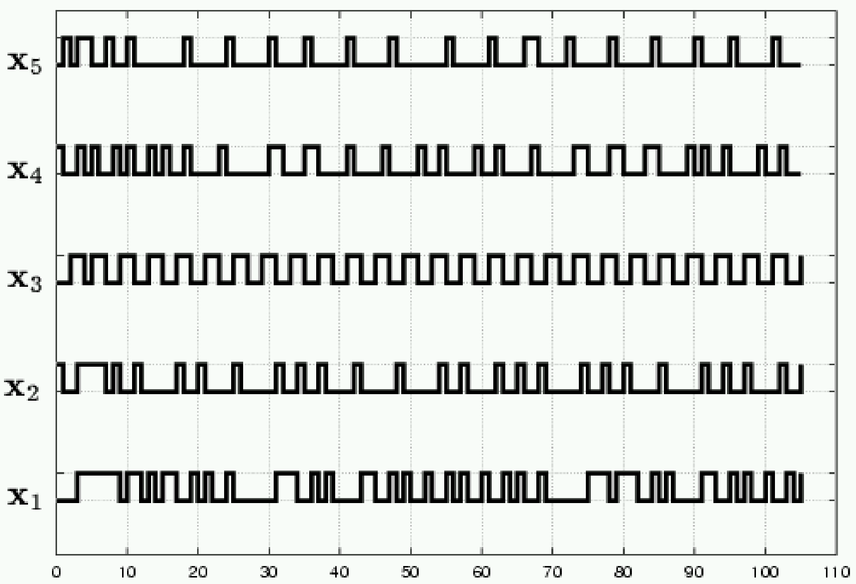


System identification: transition error

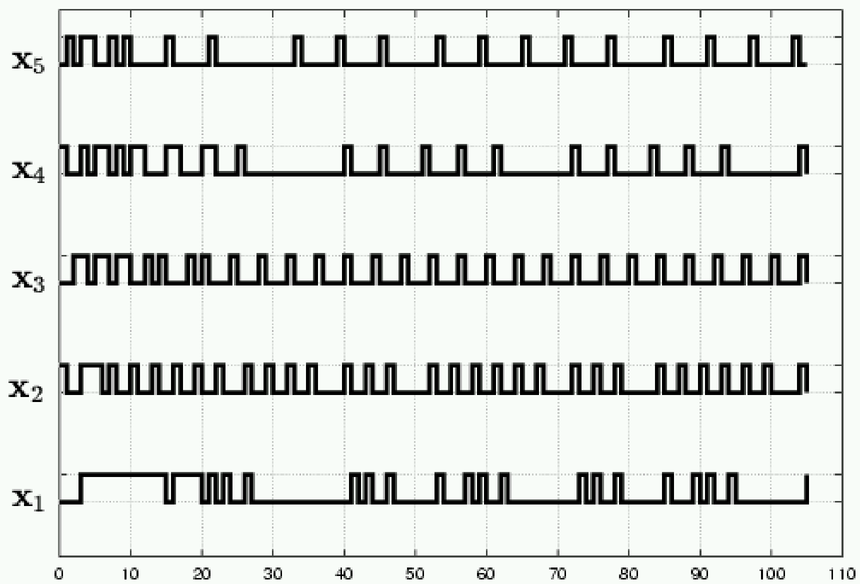




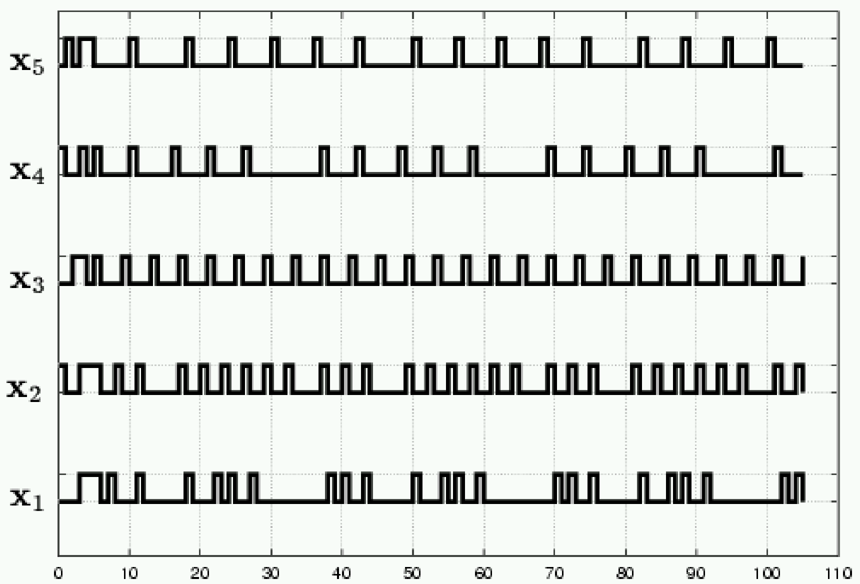
Ideal



Est.: 100 training examples

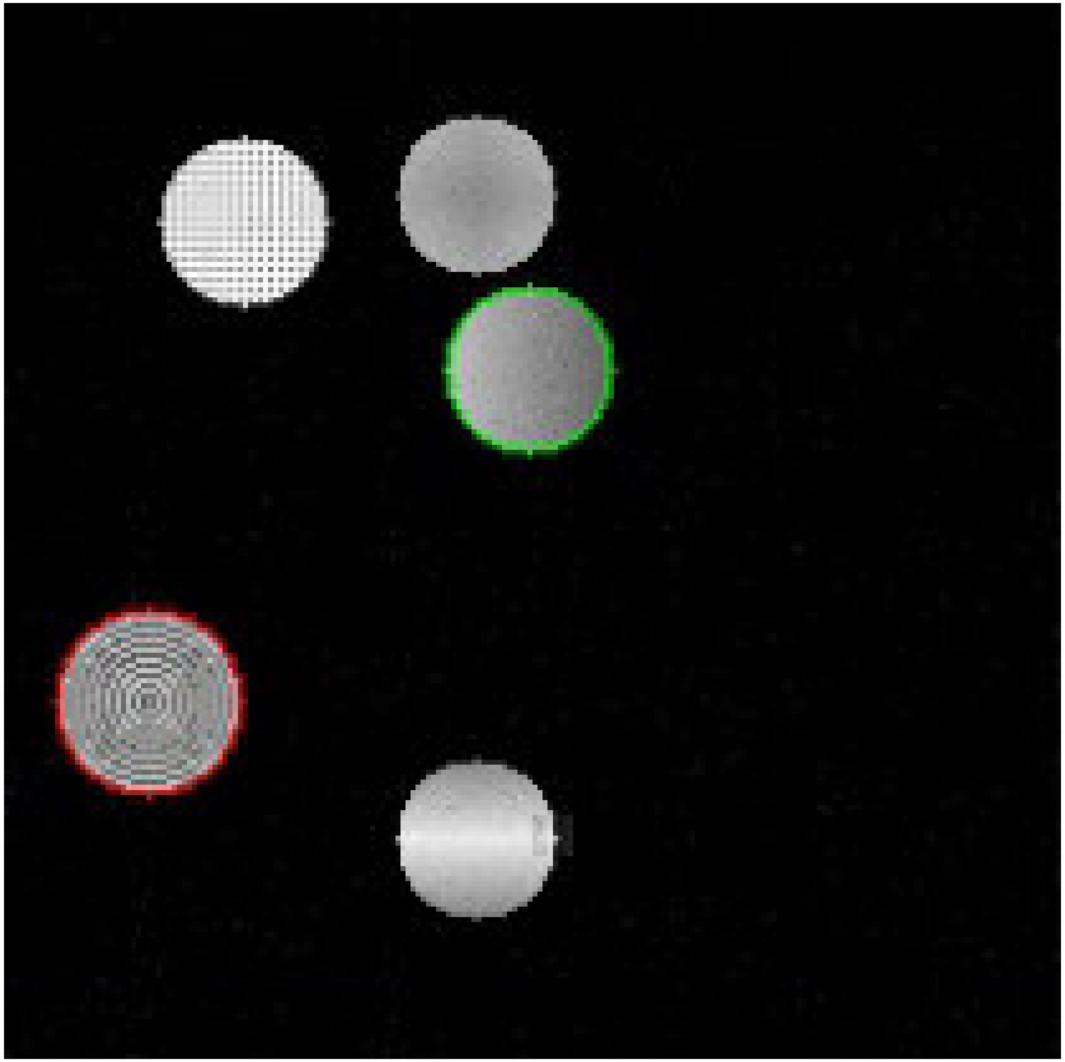


Est.: 500 training examples

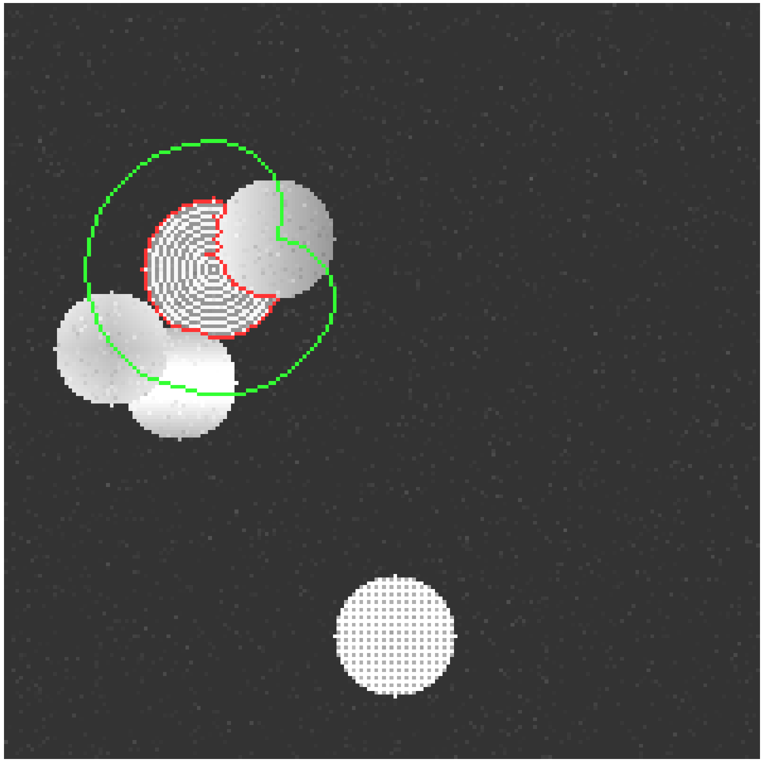


Est.: 1500 training examples

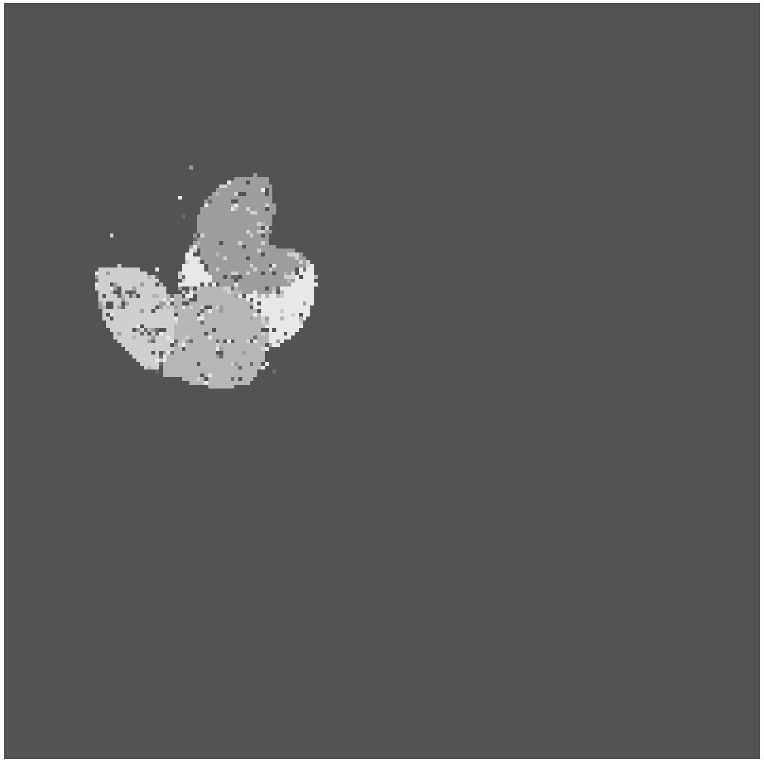
Motion Segmentation



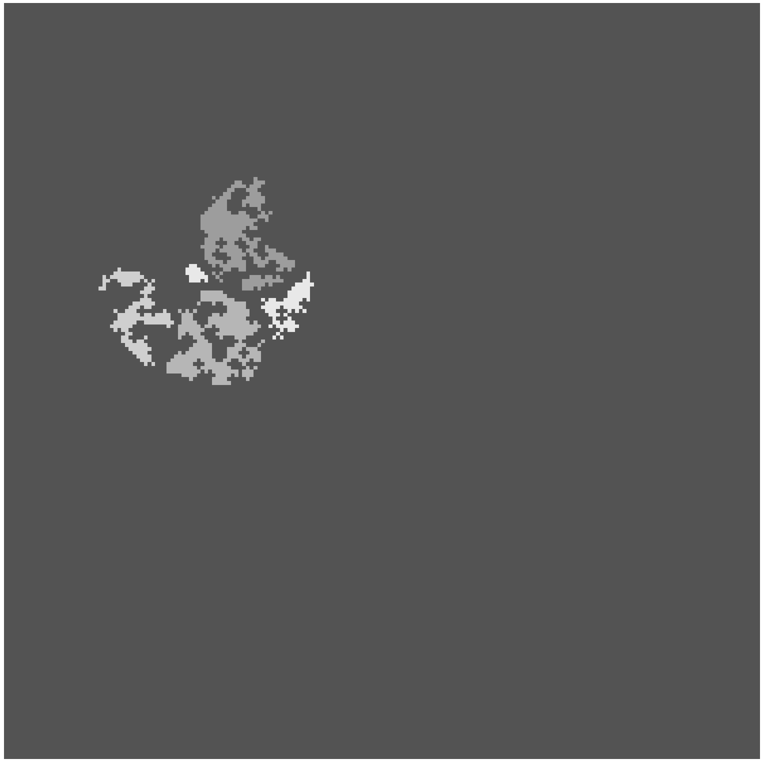
Mask



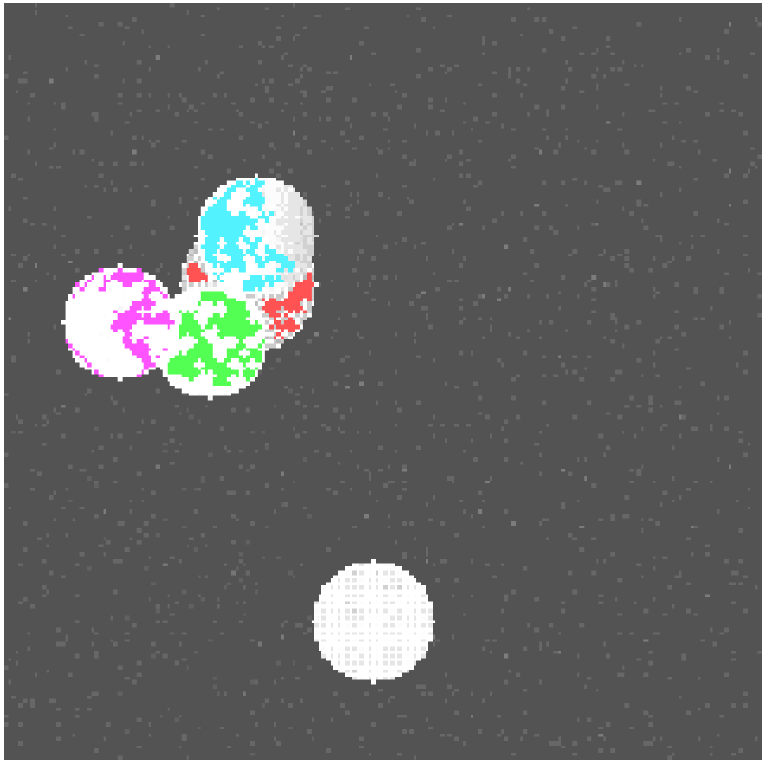
Predictor result



Filtering

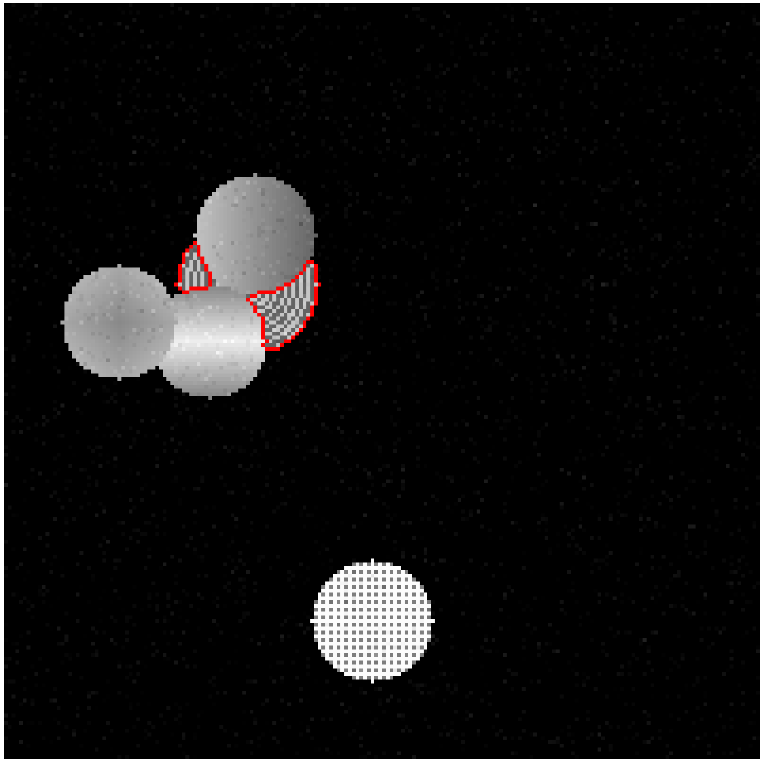


Color Composition

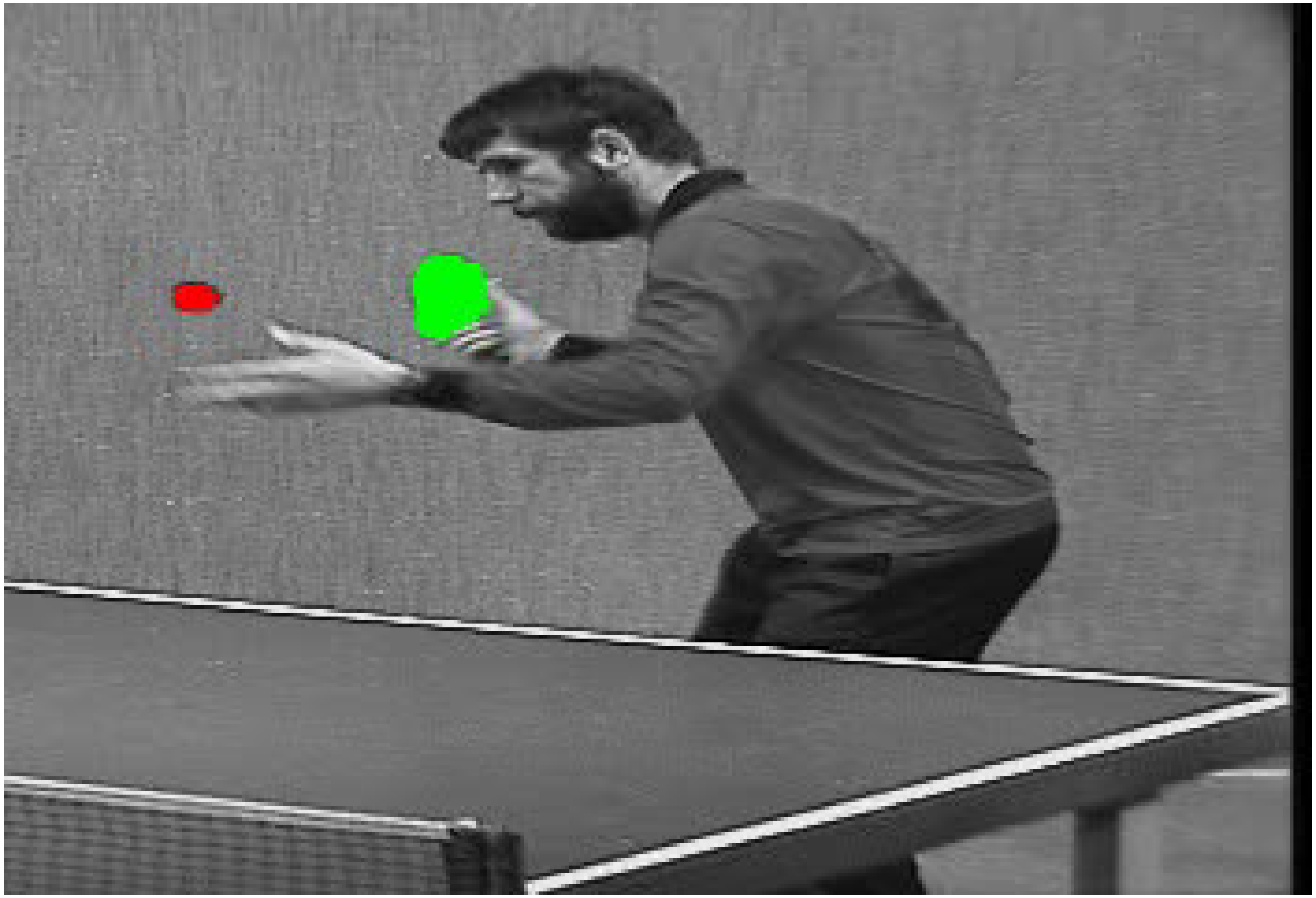


Watershed

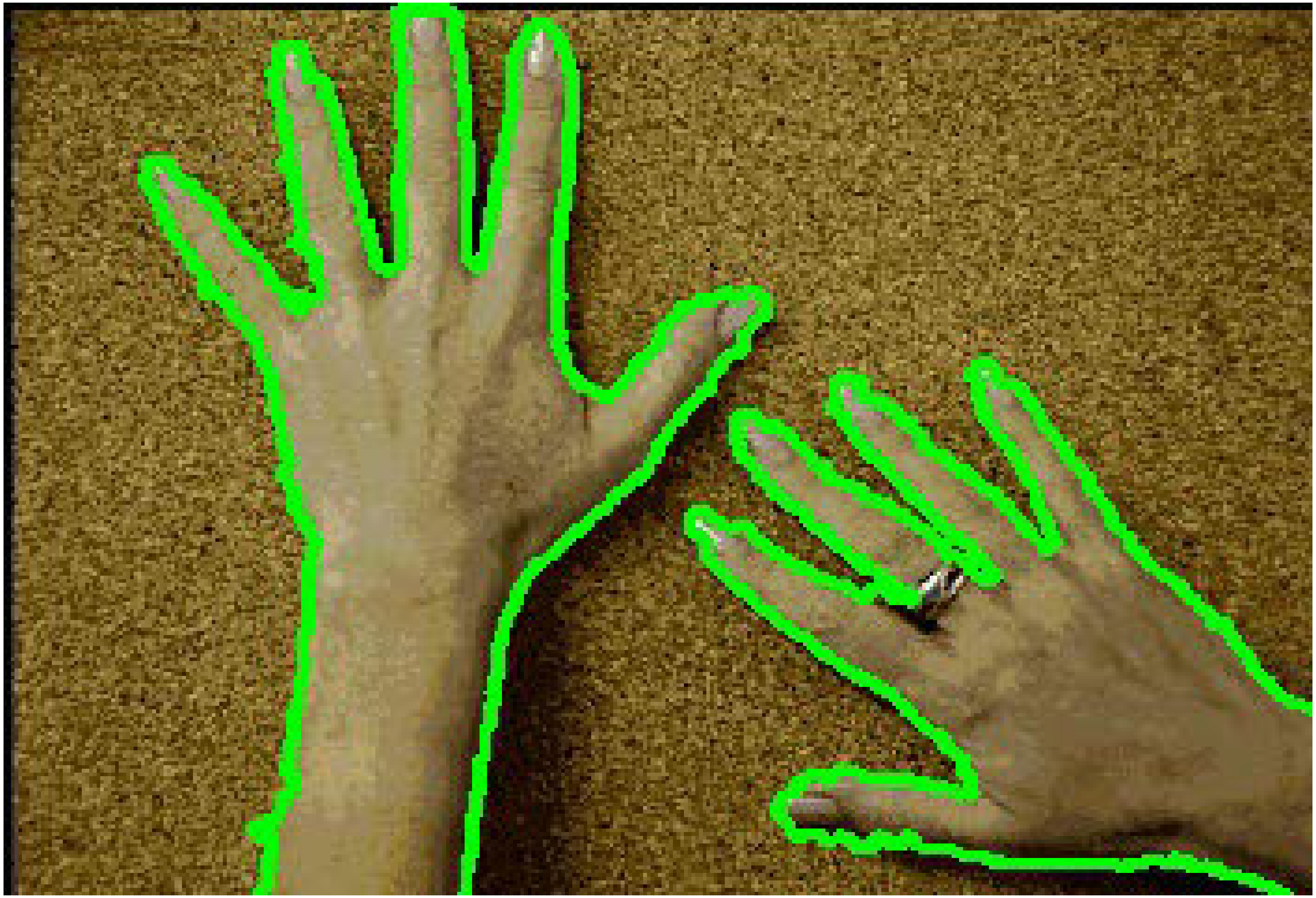
Color Composition



Motion Segmentation



Motion Segmentation



Conclusion

- Presented the notion of Lattice Dynamical System
- Proposed a model for LDS identification
- Under additive condition, system identification reduces to a family of problems of lattice operator design
- Some examples were presented
- This perspective unifies theories such as switching theory, discrete automatic control and reinforcement learning.