Applied Artificial Intelligence in Embedded Systems

Keep It Stupid Simple

Jan Jacobs

Agenda

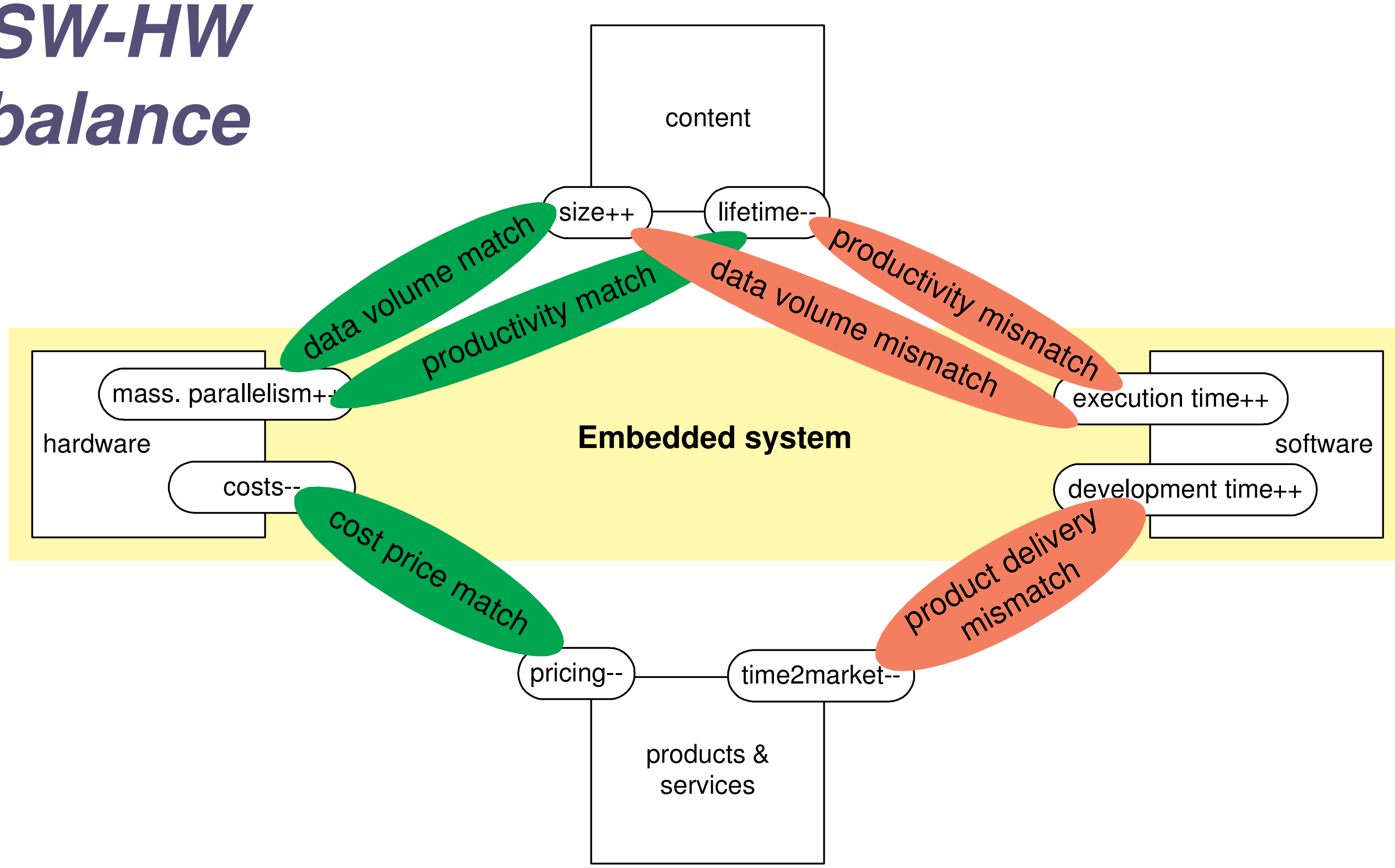
- Current problems
- Vision
- Self Organising Map
- Applications
  - Social network 

  - Diagnostics / testing
  - Robot arm control 

- Subsumption architecture
- Conclusions

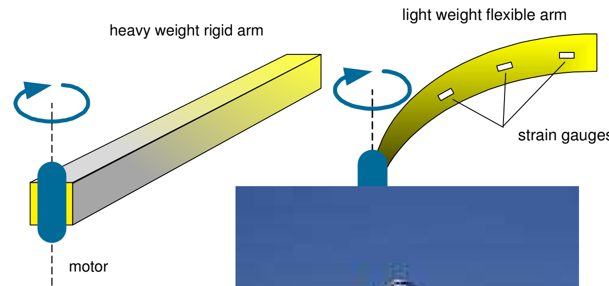
Current problems(1)

SW-HW balance

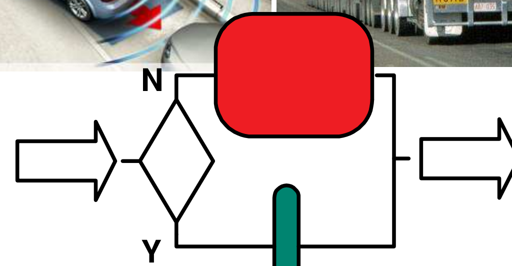


Current problems(2)

Other trends in ES



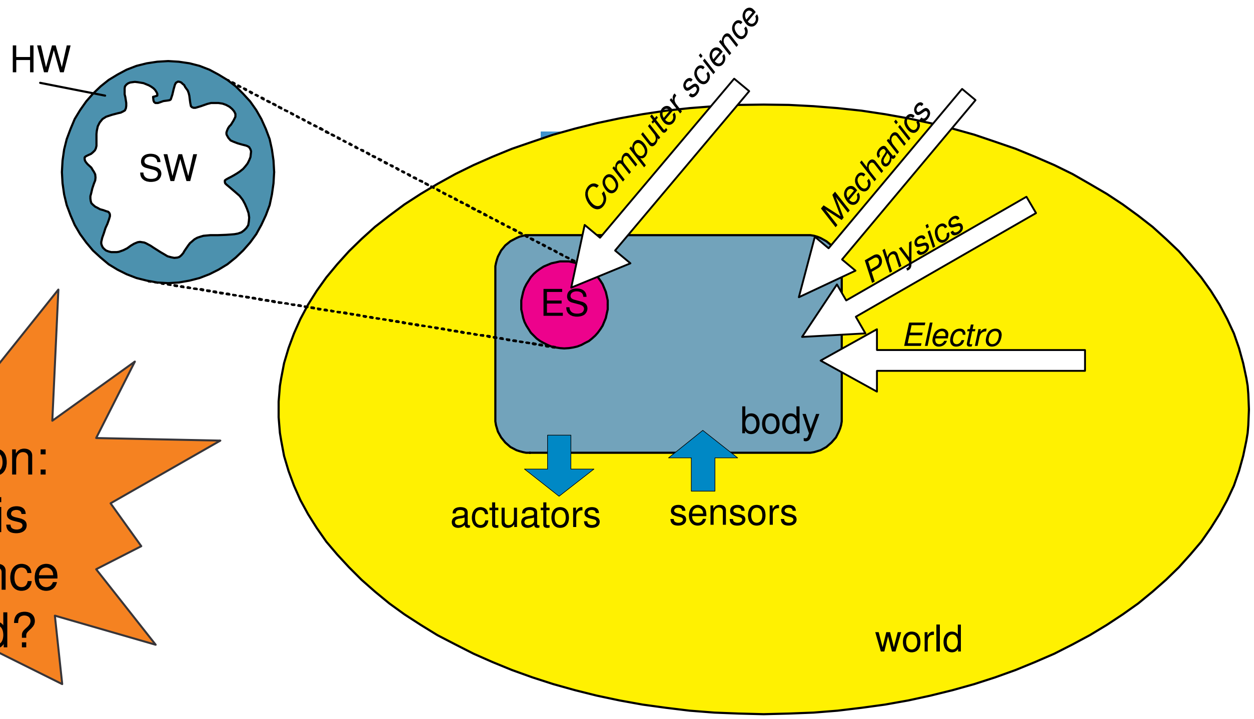
- parsimony (energy, weight, size, ...)
- fault tolerance / robustness
- multiple (concurrent) goals
- deterministic real-time response
- easier design and test
- reduction of prior knowledge, and
- in particular that of world model



Agenda

- Current problems
- **Vision**
- Self Organising Map
- Applications
  - Social network demo
  - Diagnostics / testing
  - Robot arm control demo
- Subsumption architecture
- Conclusions

Vision(1)



Question: where is intelligence located?

Current believe:

Embedded Systems (ES) hosts 100% of all intelligence

Denise

Conclusion:

Intelligent behaviour is not exclusively domain of Computer Science (CS)

Vision(2)

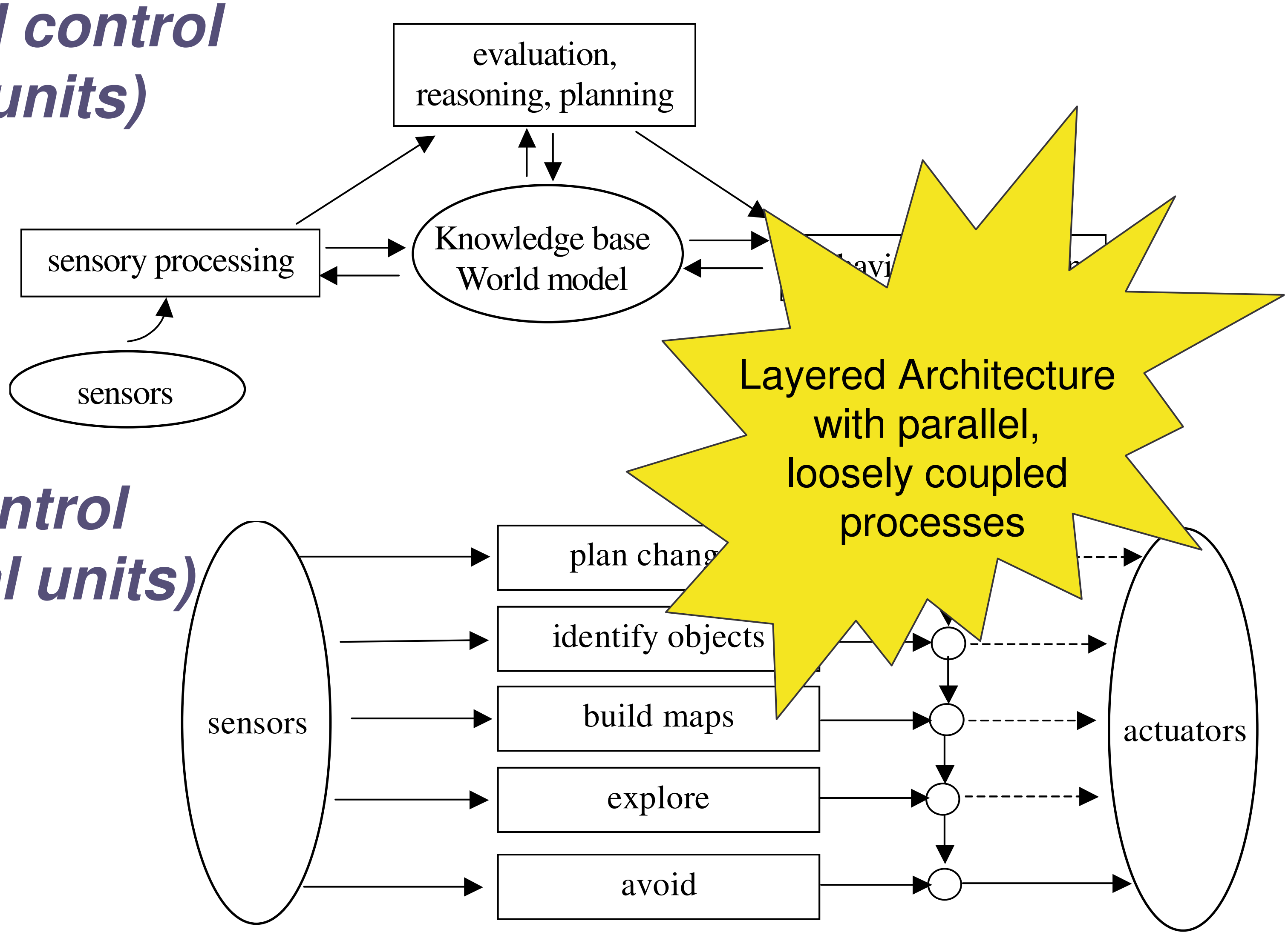
Lego Mindstorms:

"…intelligence is not a "box" residing inside the brain, but is distributed throughout the organism and requires the interaction with the environment as an essential component"

Vision(3)

Hierarchical control (functional units)

Reactive control (behavioural units)



Agenda

- Current problems
- Vision
- **Self Organising Map (SOM)**
- Applications
  - Social network 

  - Diagnostics / testing
  - Robot arm control 

- Subsumption architecture
- Conclusions

Self Organising Map (SOM) background(1)

biological plausibility of topological ordering (Teuvo Kohonen, HUT Finland)

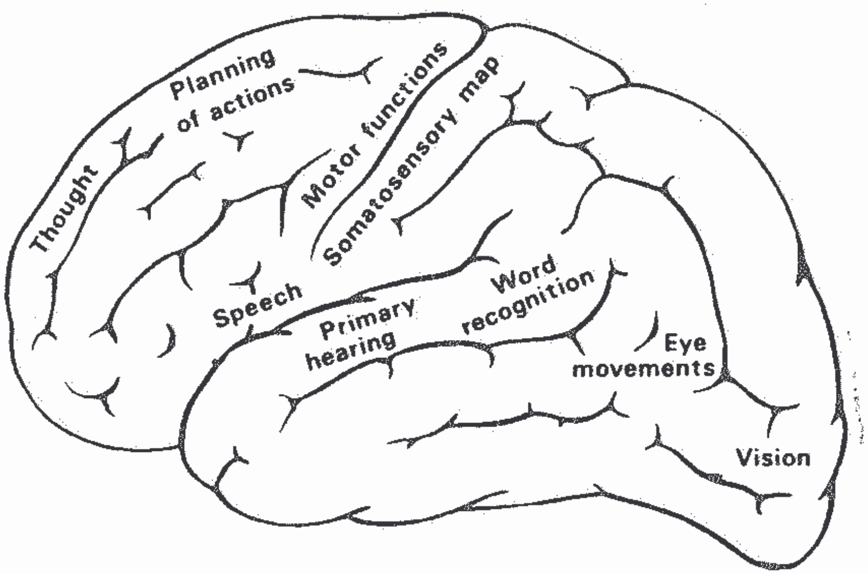


Fig. 2.7. Brain areas

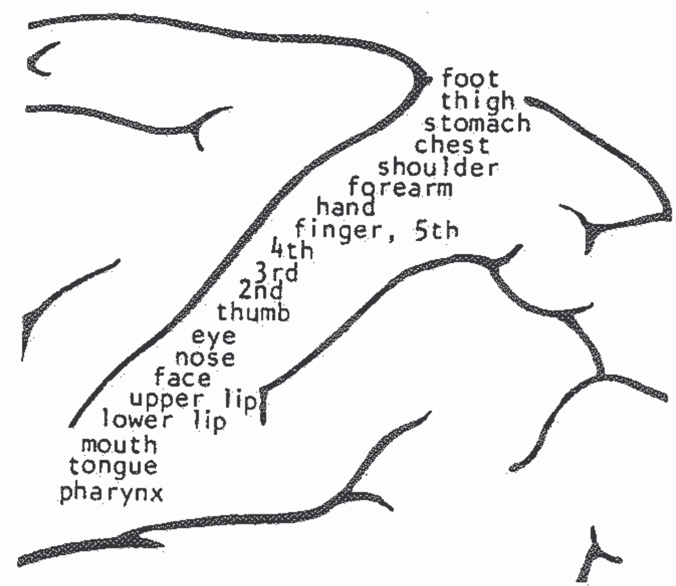
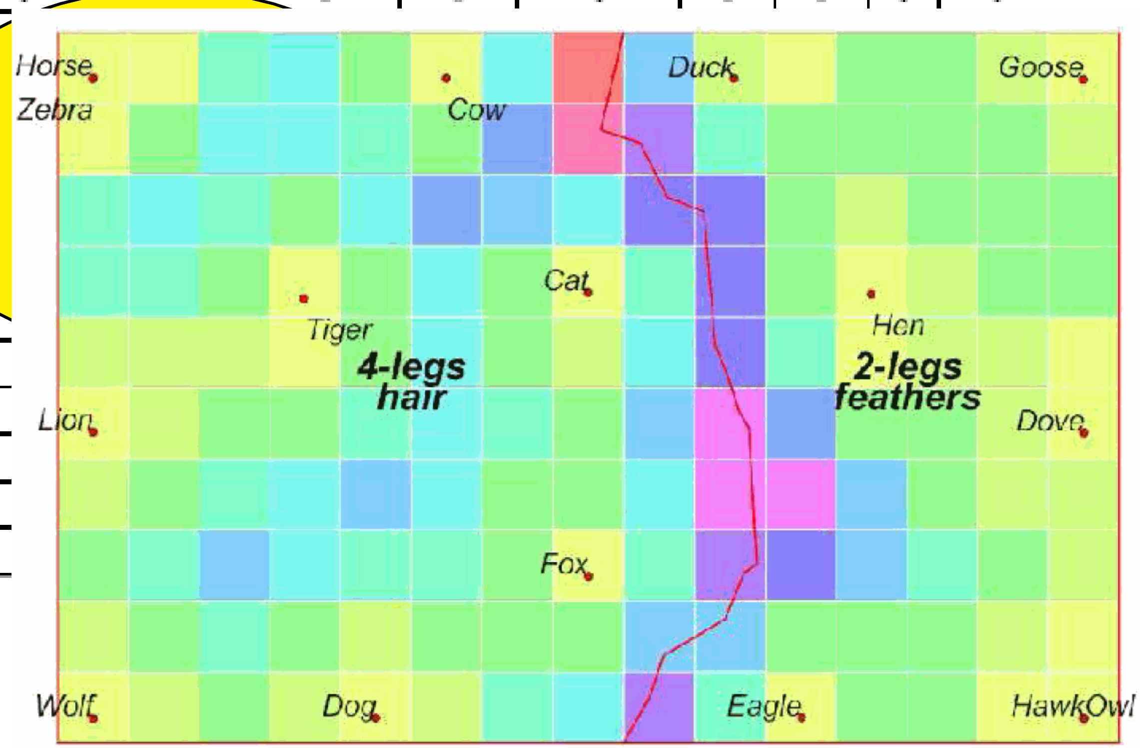


Fig. 2.8. The somatotopic map

SOM background(2)

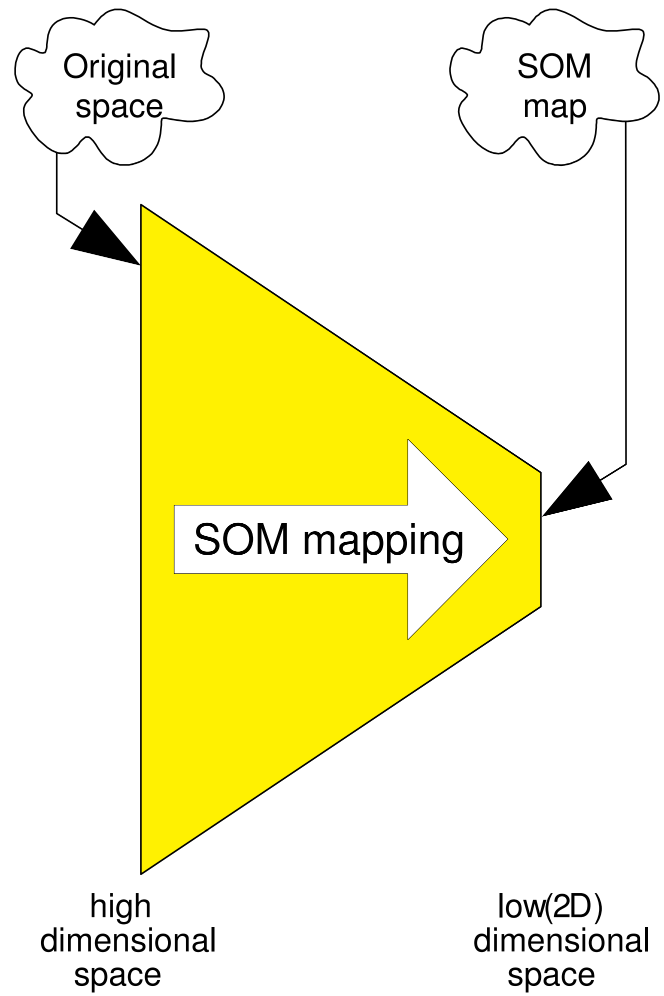
	small	medium	big	Two legs	Four legs	Hair	Hooves	Mane	Feathers	Hunt	Run	Fly	Swim
Dove	1	0	0	1	0	0	0	0	1	0	0	1	0
Hen	1	0	0	1	0	0	0	0	1	0	0	0	0
Duck	1	0	0	1	0	0	0	0	1	0	0	0	1
Goose	1	0	0	1	0	0	0	0	1	0	0	1	1
Owe	1	0	0										
Hawk	1	0											
Eagle	0	1											
Fox	0	1											
Dog	0	1											
Wolf	0	1											
Cat	1	0	0										
Tiger	0	0	1										
Lion	0	0	1										
Horse	0	0	1										
Zebra	0	0	1										
Cow	0	0	1										



SOM background(3)

Conclusions

- Self organisation; no prior model
- Dimension compression
- Mapping preserves structure
- Visualisation with topological ordering

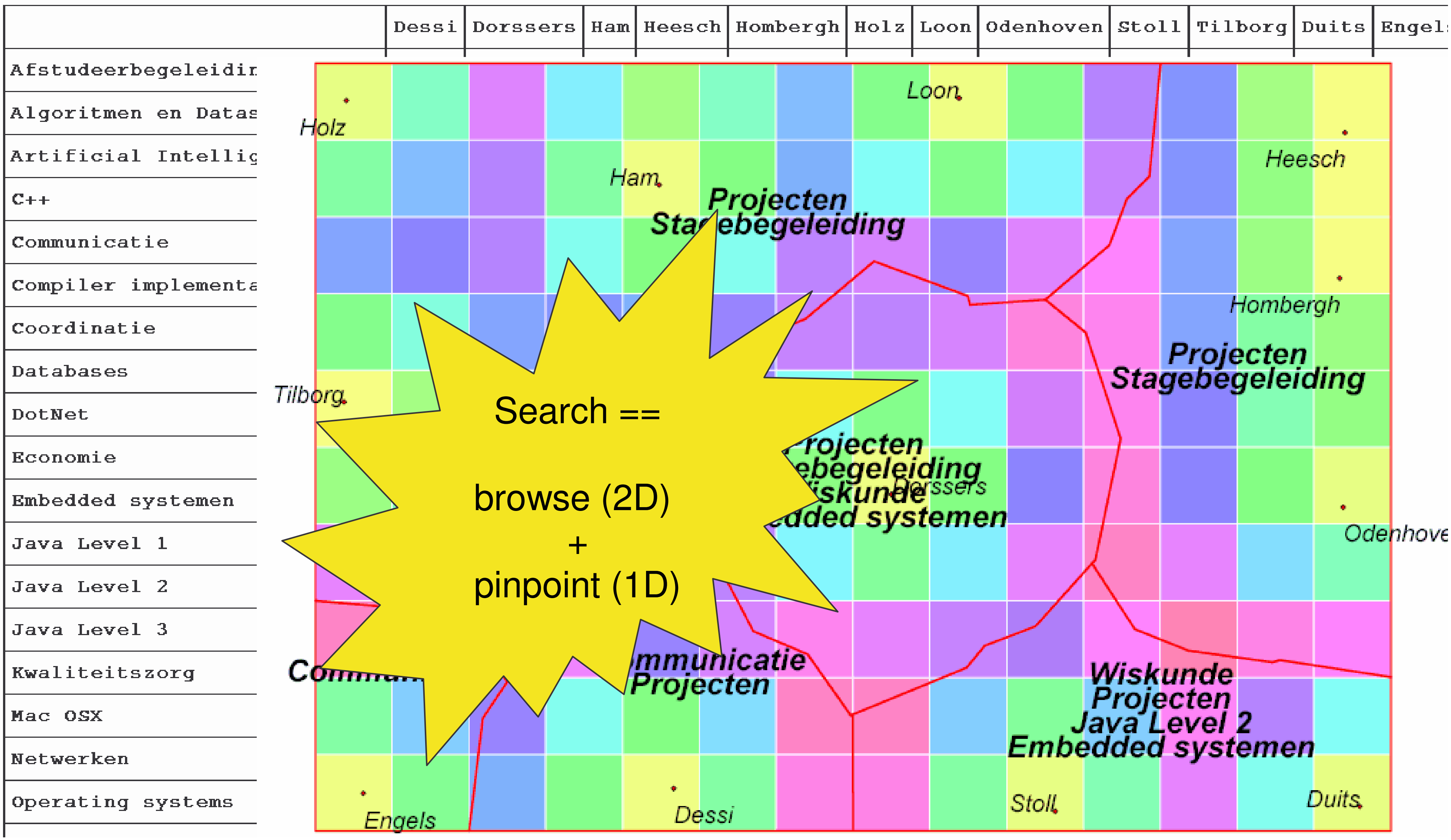


Agenda

- Current problems
- Vision
- Self Organising Map
- **Applications**
  - **Social network** demo
  - Diagnostics / testing
  - Robot arm control demo
- Subsumption architecture
- Conclusions

Social network mapping(1)

Search a person without programming



Social network mapping(2)

Questions:

•How are expertises distributed?

•Who is the expert in this field?

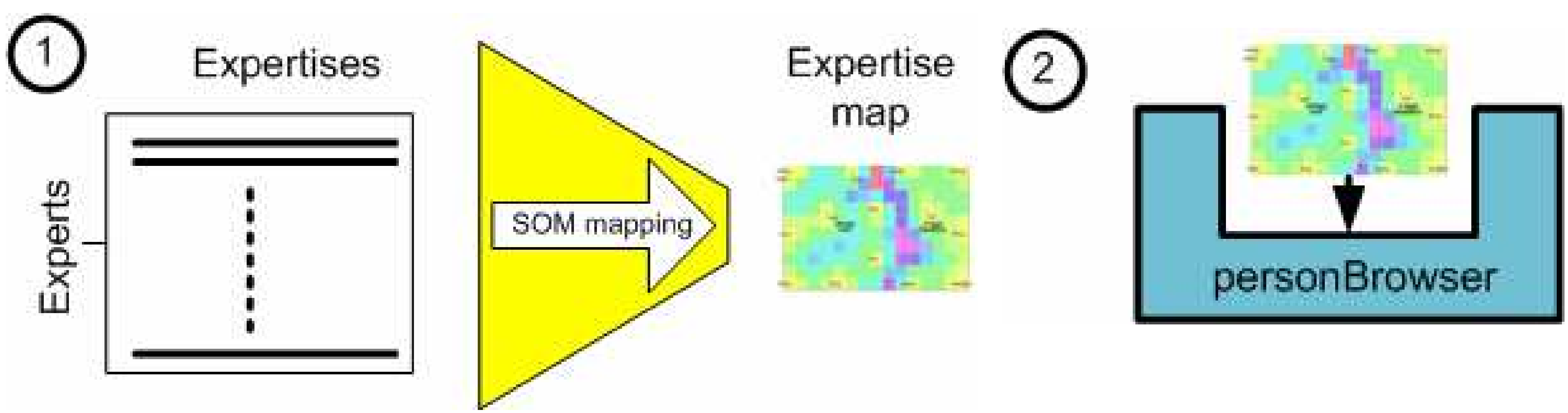
•Who has similar experience?

Steps:

1. train

2. use

demo



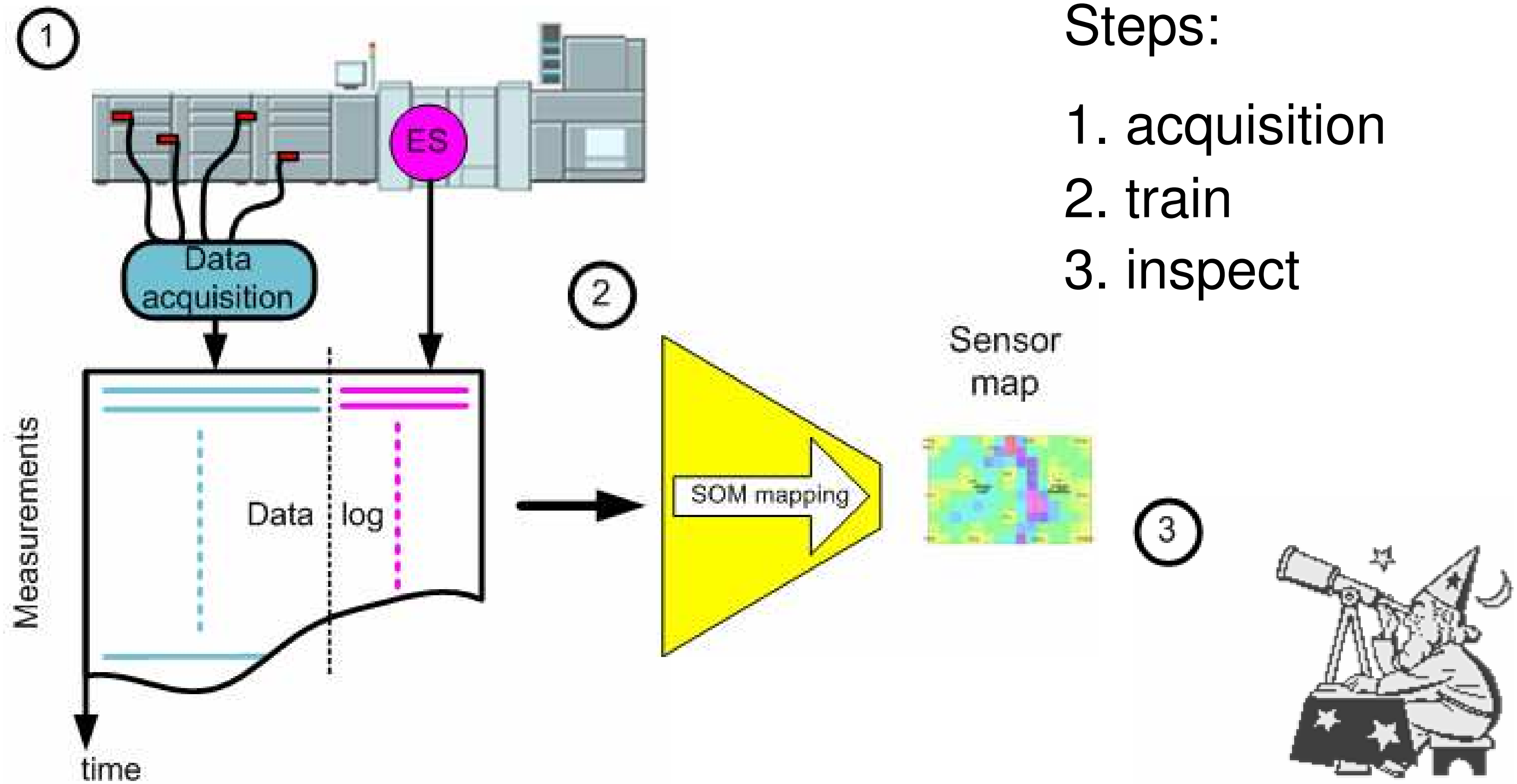
Agenda

- Current problems
- Vision
- Self Organising Map
- Applications
  - Social network
  - Diagnostics / testing
  - Robot arm control
- Subsumption architecture
- Conclusions

COFFEE
BREAK

Anomaly Detection(1)

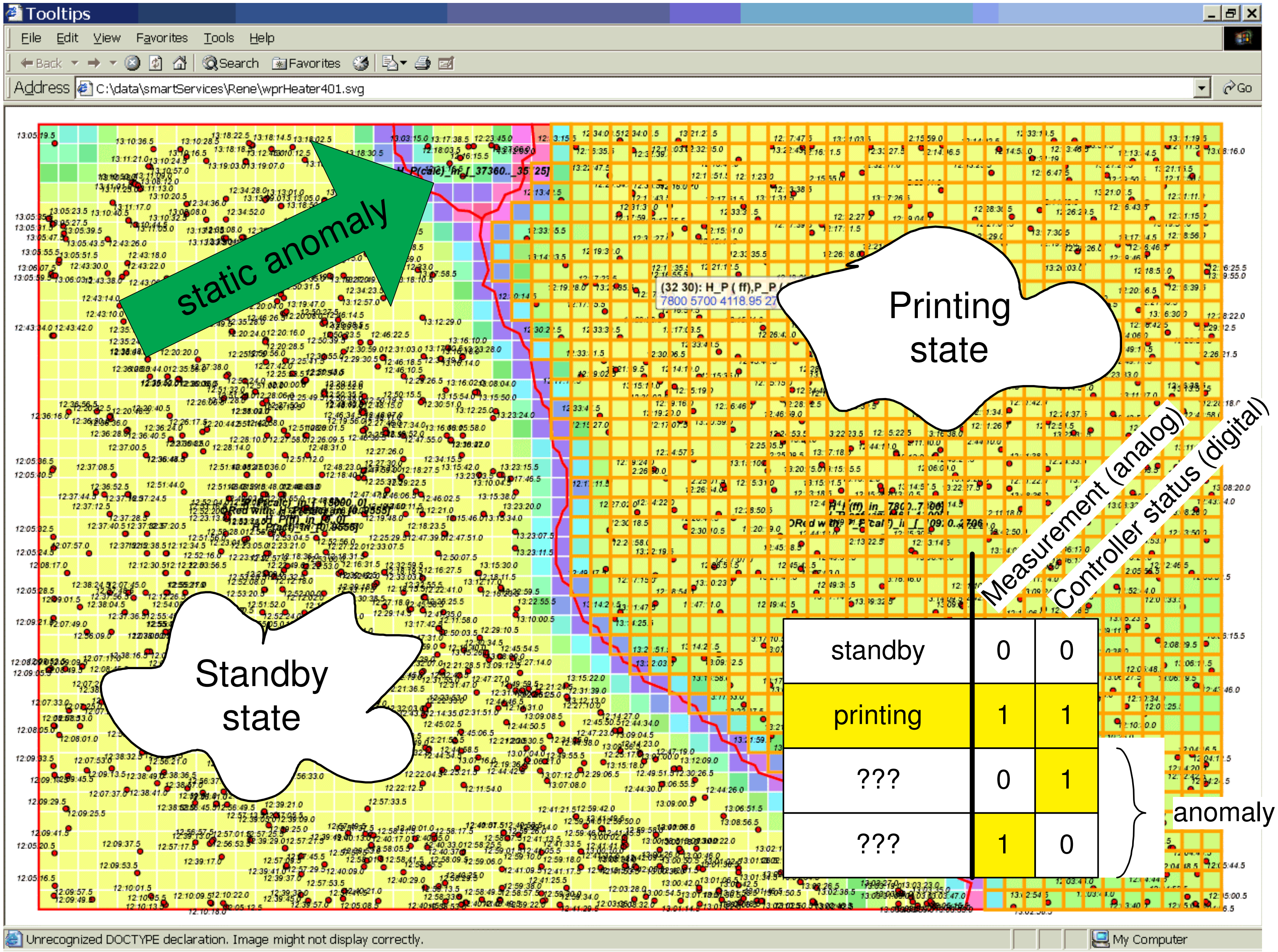
Detect deviating patterns without programming



Steps:

1. acquisition
2. train
3. inspect

Anomaly Detection(2)



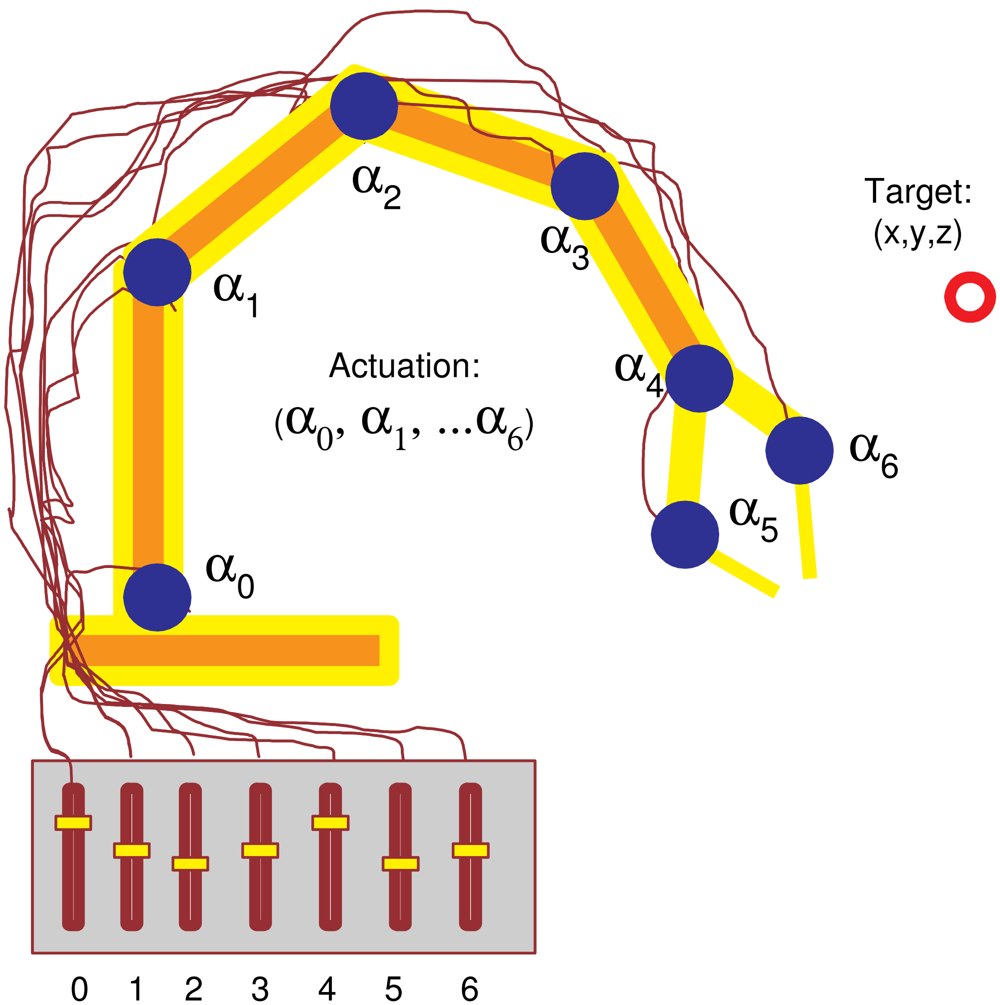
	Measurement (analog)	Controller status (digital)
standby	0	0
printing	1	1
???	0	1
???	1	0

Agenda

- Current problems
- Vision
- Self Organising Map
- Applications
  - Social network
  - Diagnostics / testing
  - **Robot arm control** demo
- Subsumption architecture
- Conclusions

Cognitive Actuation(1)

Inverse Kinematics



Assignment: control the 7 sliders such that the gripper moves smoothly to the target.

Cognitive Actuation(2)

Actuate without modelling

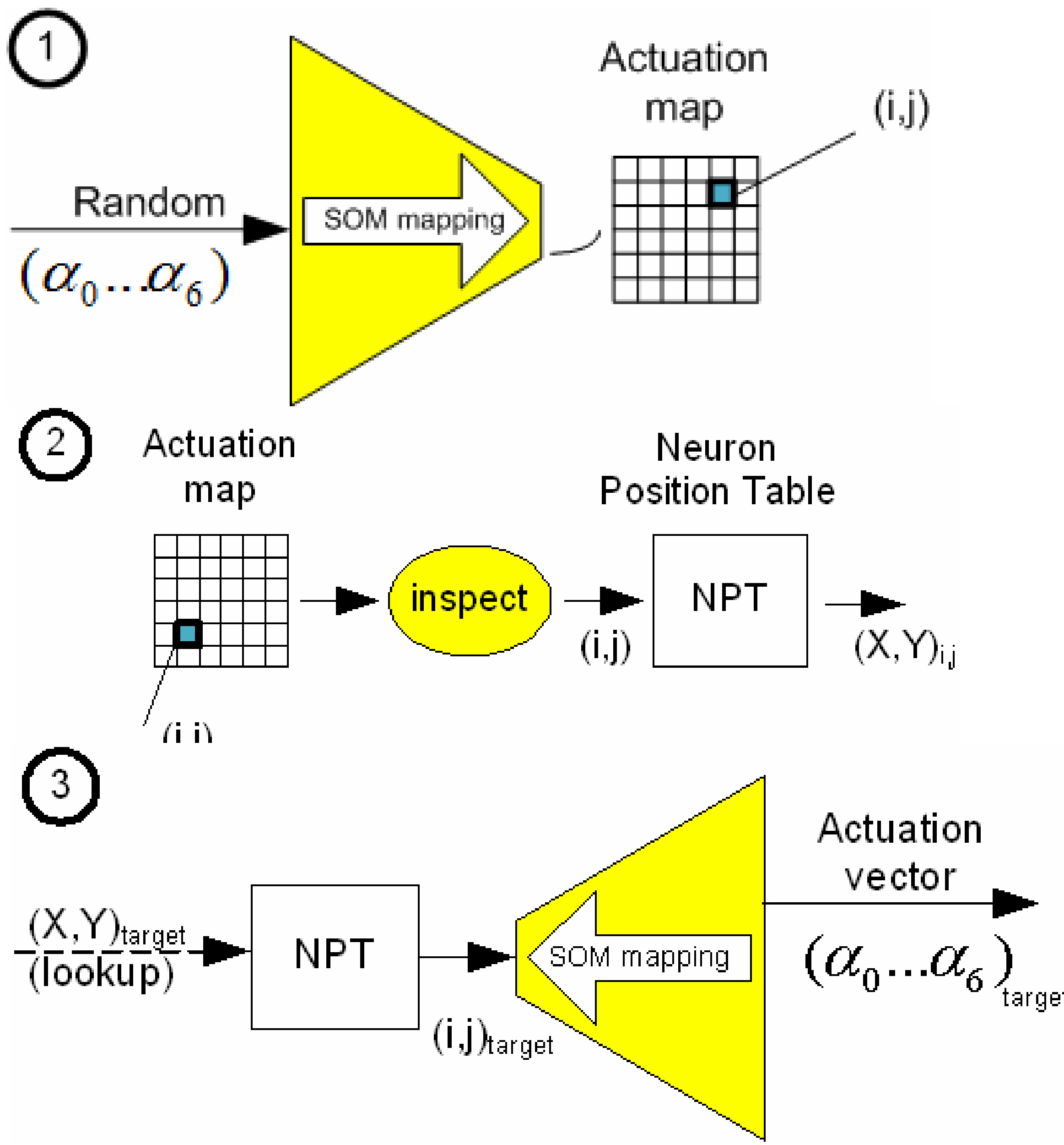
- Baby babbling movie
- Randomly select actuation angles
- Train SOM (highD → 2D)
- Use reversely (2D → highD)
- Solve inverse kinematics problem

Cognitive Actuation(3)

Actuate without modelling

Steps:

1. Train
2. Inspect
3. Use

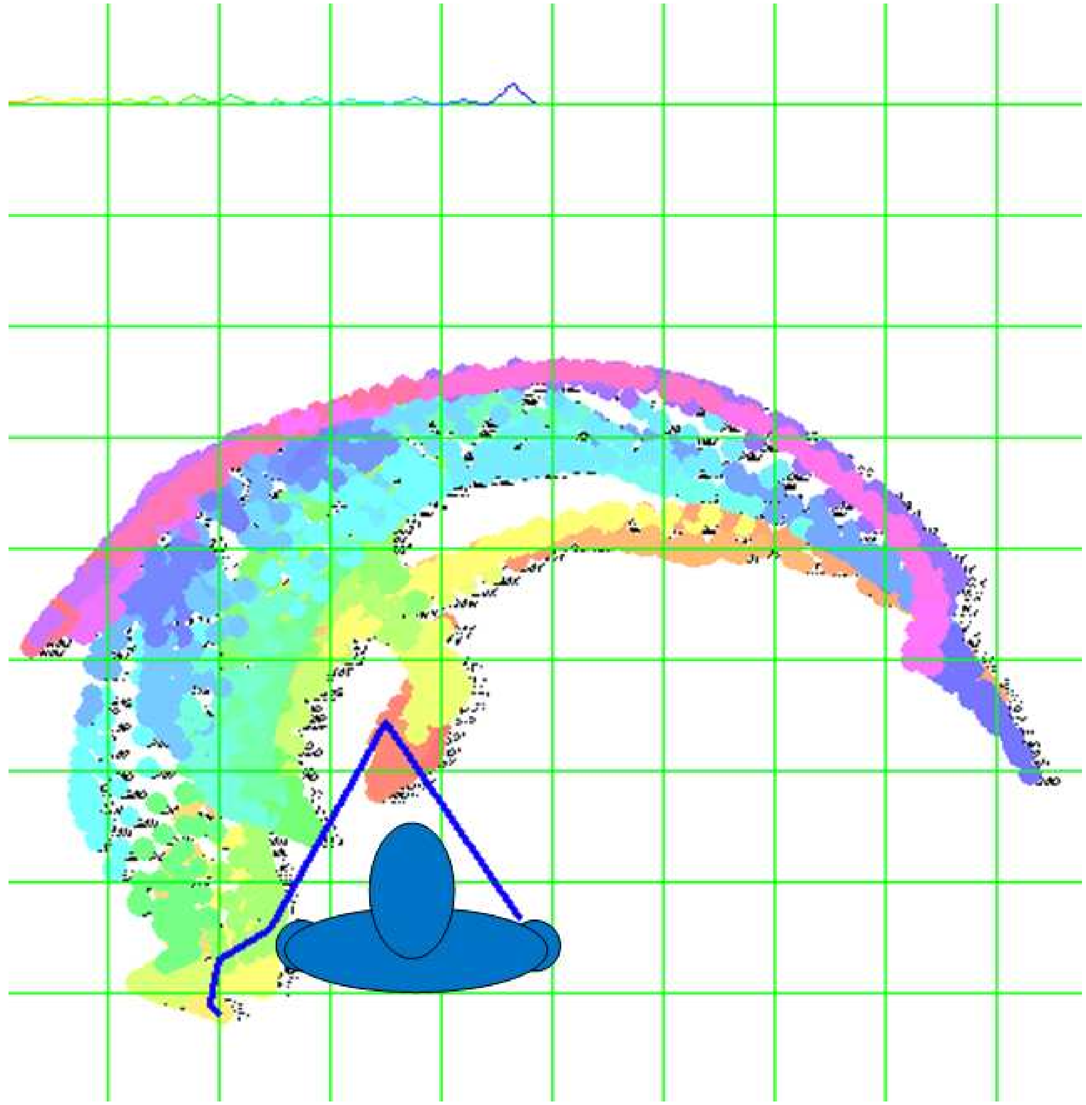


Cognitive Actuation(4)

Human arm joint

6D state space

1. shoulder, 40 cm, 0..135^{o}
2. elbow, 40 cm, 0.. 90^{o}
3. wrist, 10 cm, -45..+45^{o}
4. proximal phalanx, 5 cm, 0..90^{o}
5. middle phalanx, 3 cm, 0..90^{o}
6. distal phalanx, carrying the nail, 2 cm, 0..90^{o}



demo

Cognitive Actuation(5)

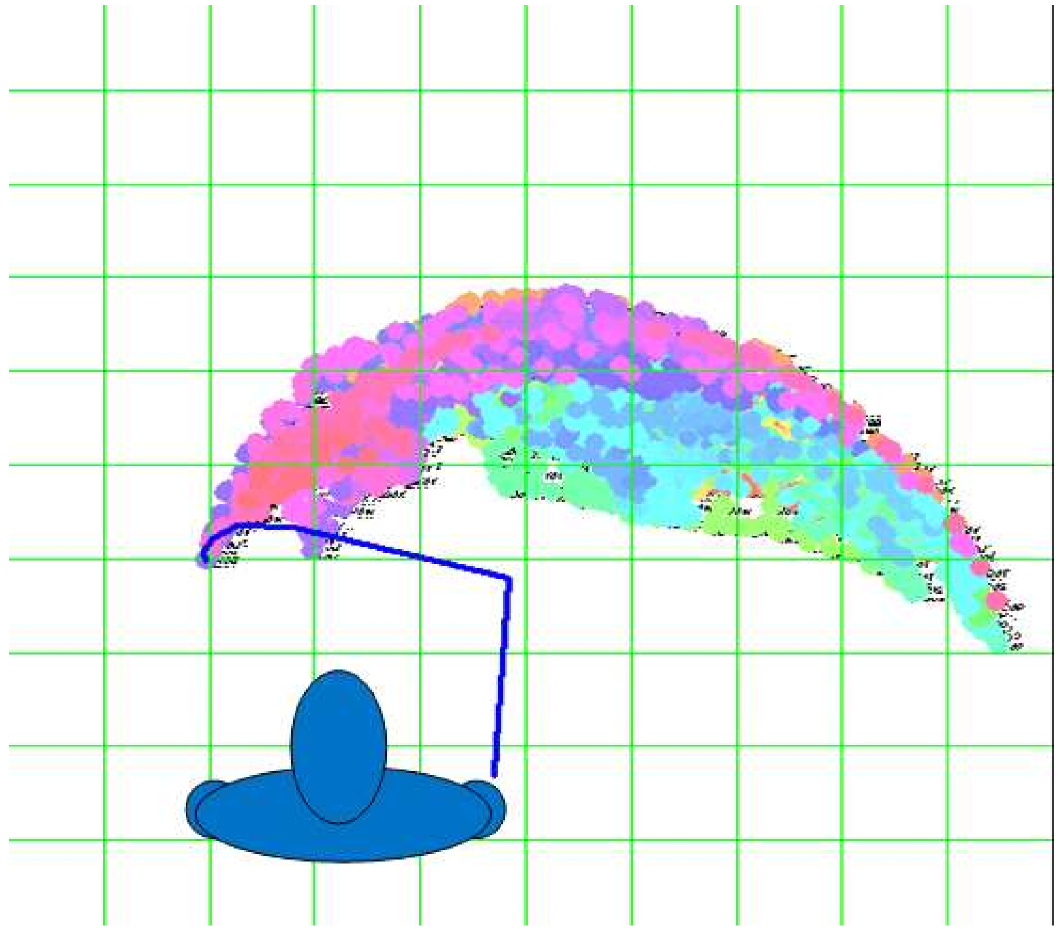
Graceful degradation without modelling

- The arm is “ill”
  - wear (motor brushes)
  - slip (joints)
  - extreme: arm bended, finger missing
- But arm is still able to smoothly avoid “painful” areas

Cognitive Actuation(6)

…and now a shoulder with rheumatism!

1. shoulder, 40 cm, 0..~~135~~ → **90**^{o}
2. elbow, 40 cm, 0.. 90^{o}
3. wrist, 10 cm, -45..+45^{o}
4. proximal phalanx, 5 cm, 0..90^{o}
5. middle phalanx, 3 cm, 0..90^{o}
6. distal phalanx, carrying the nail, 2 cm, 0..90^{o}



demo

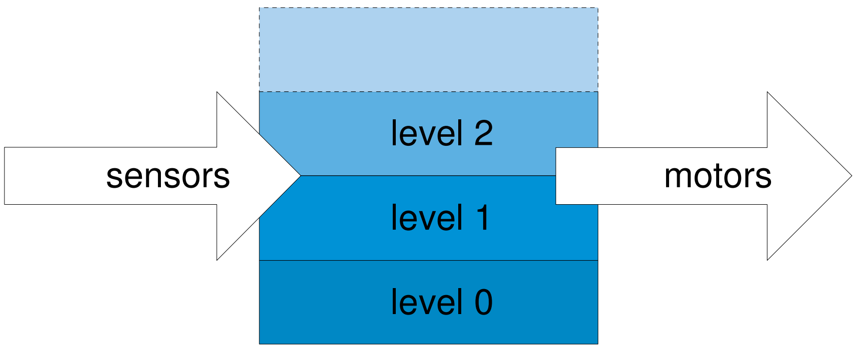
Agenda

- Current problems
- Vision
- Self Organising Map
- Applications
  - Social network
  - Diagnostics / testing
  - Robot arm control
- **Subsumption architecture**
- Conclusions

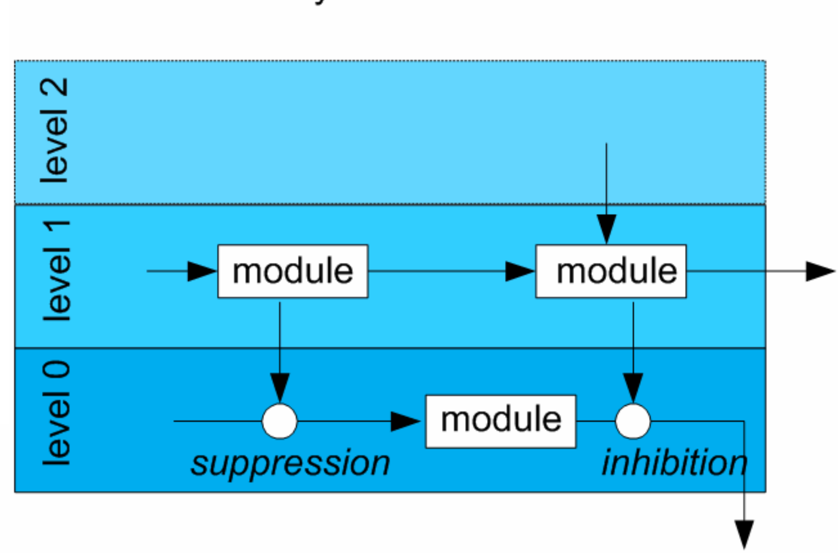
Subsumption architecture(1)

Behavioural layers

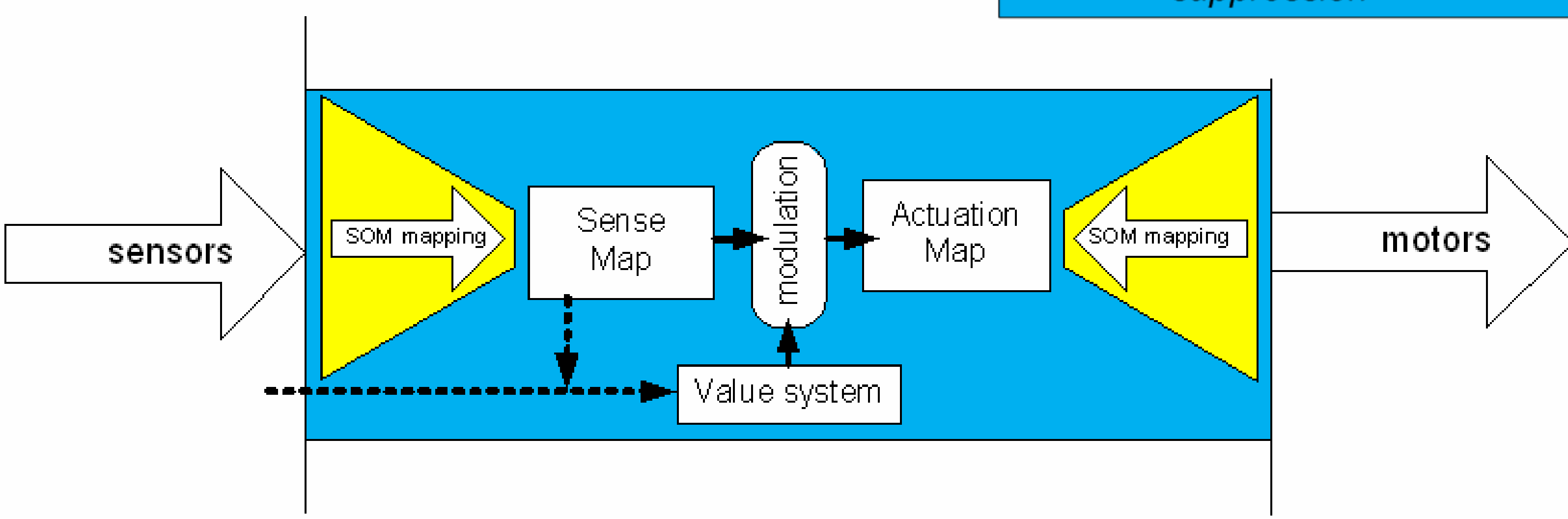
levels of competence



interlayer relations

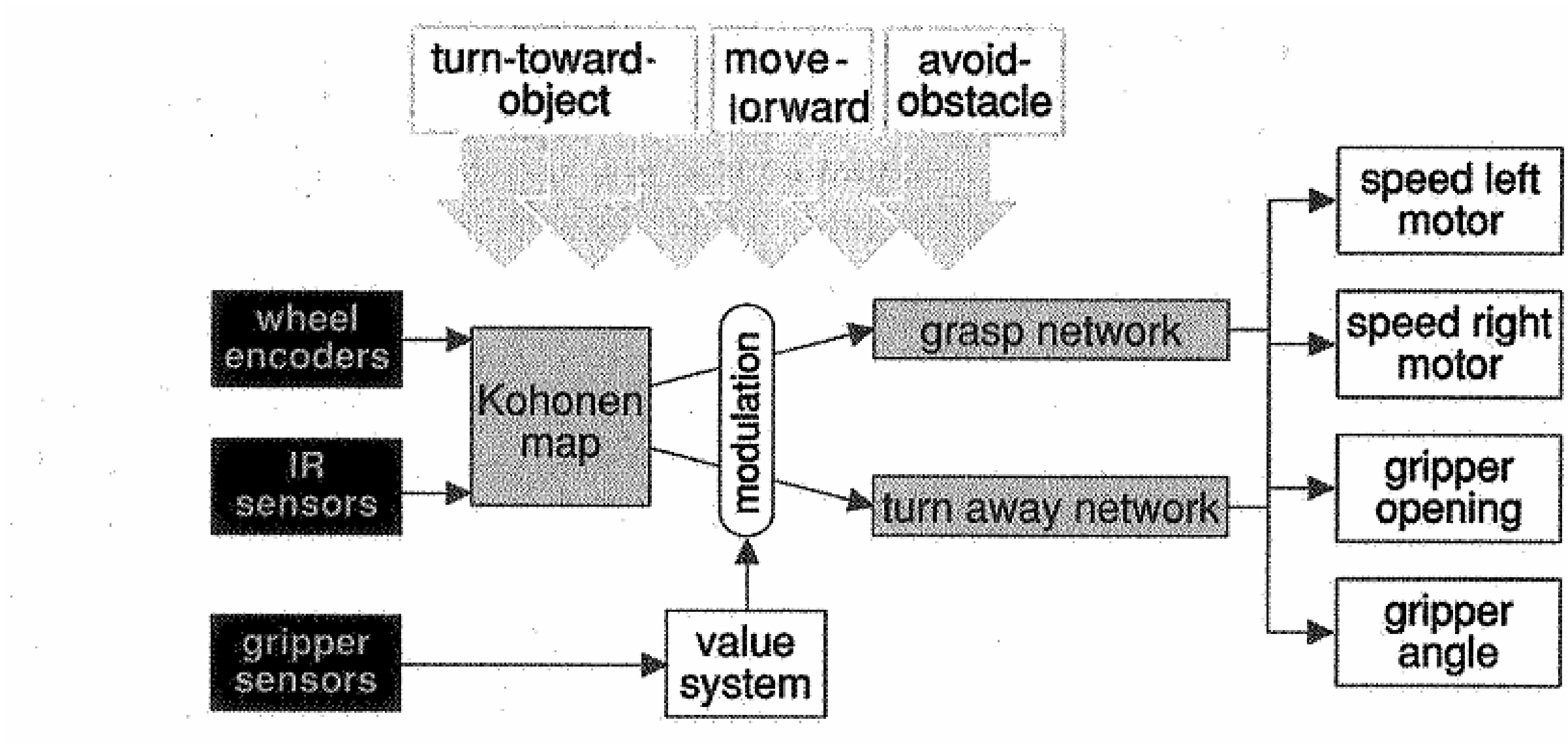


typical layer architecture



Subsumption architecture(2)

value steered sensory-motor



Agenda

- Current problems
- Vision
- Self Organising Map
- Applications
  - Social network
  - Diagnostics / testing
  - Robot arm control
- Subsumption architecture
- **Conclusions**

Conclusions

- "Intelligence" should be distributed through whole machine (not exclusively by Embedded Systems)
- The subsumption architecture provides a framework that may result in less complex ES
  - when? many uncertainties, many dimensions
  - why? less complex, safety for self / others
  - where? hard real-time actions (reflexes, no logic)
- Promising component: Self Organising Map
- SOM acts as a (self-learned) universal 2D format

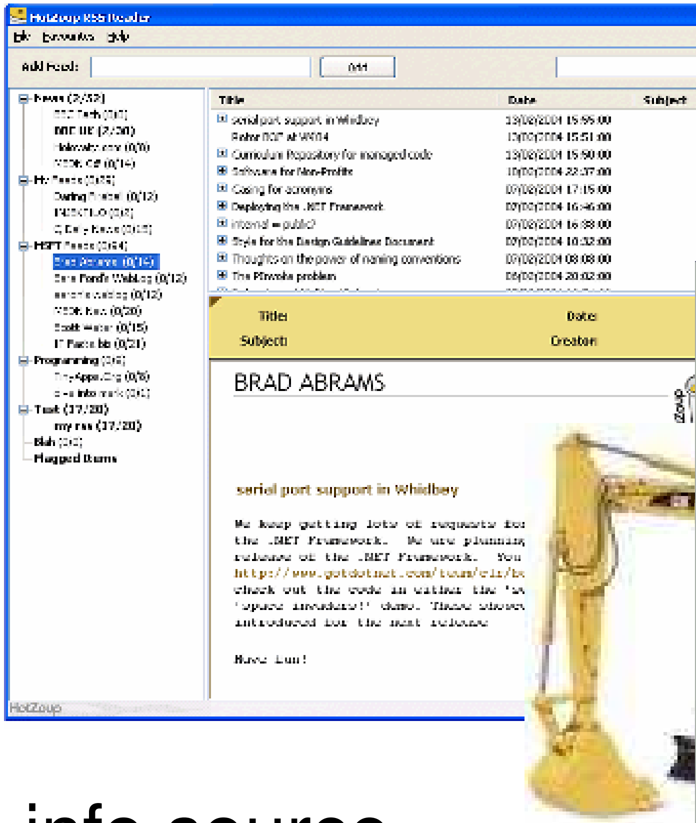
Keep IT *Stupid Simple*

Questions…

Agenda

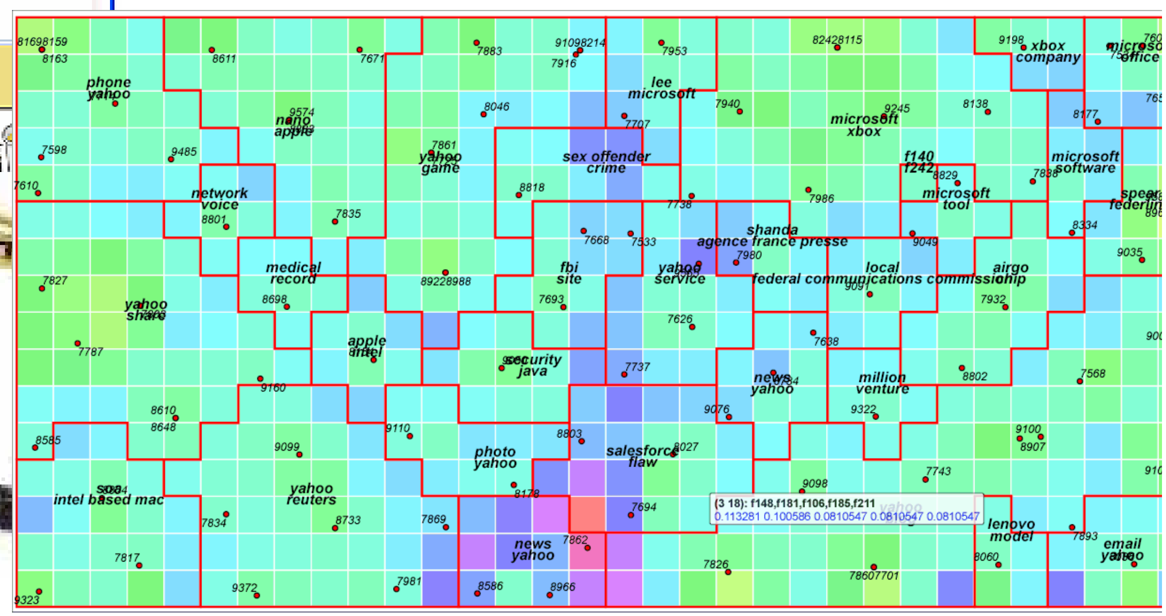
- Current problems
- Vision
- Self Organising Map
- Applications
  - Social network
  - **Document space** demo
  - Diagnostics / testing
  - Robot arm control demo
- Subsumption architecture
- Conclusions

Document mapping(1)



info source

Self Organising map (SOM)

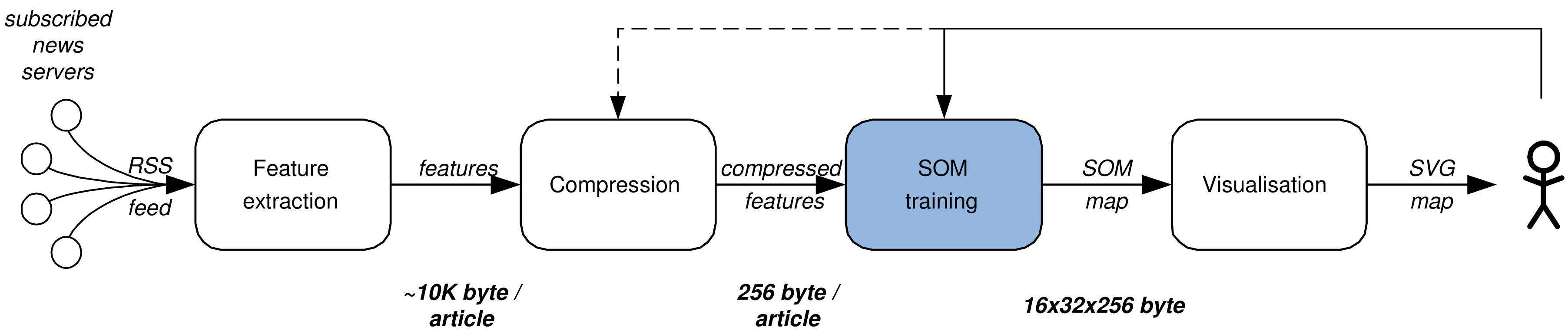


data mining

visualisation

Document mapping(2)

Specification Data mining system

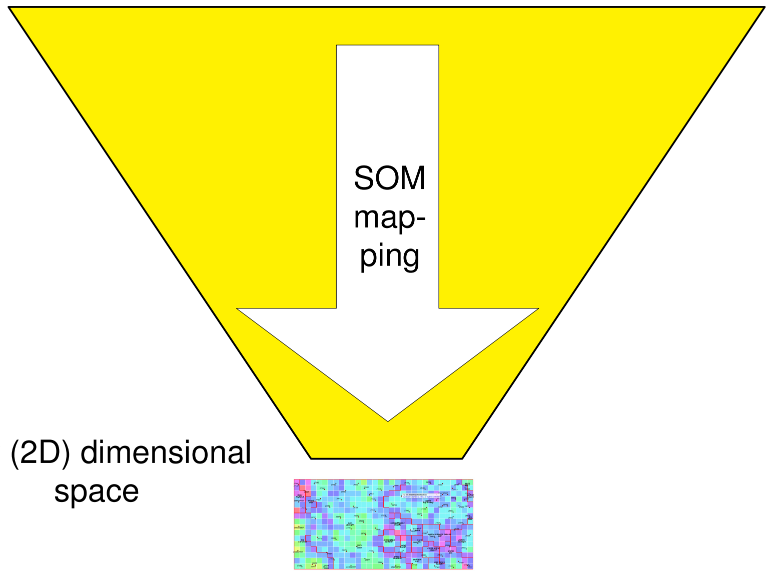


- #articles: 100-1000 (typical 500)
- data mining: few minutes
- SOM training: responsive (few seconds)
- interactive visualisation

Document mapping(3)

Compressed list of important words

High dimensional space



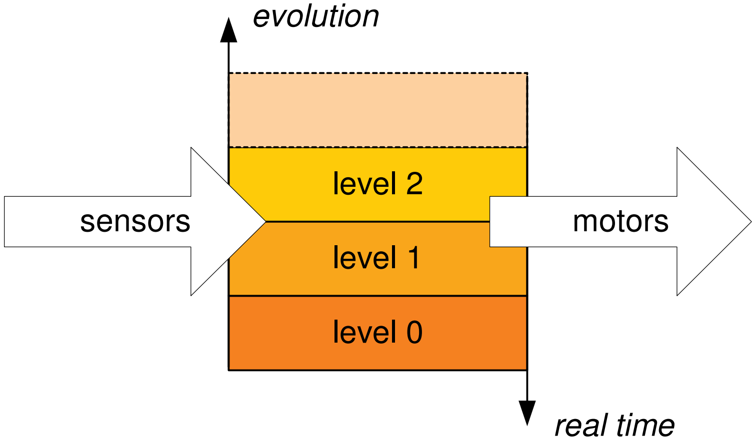
Low (2D) dimensional space

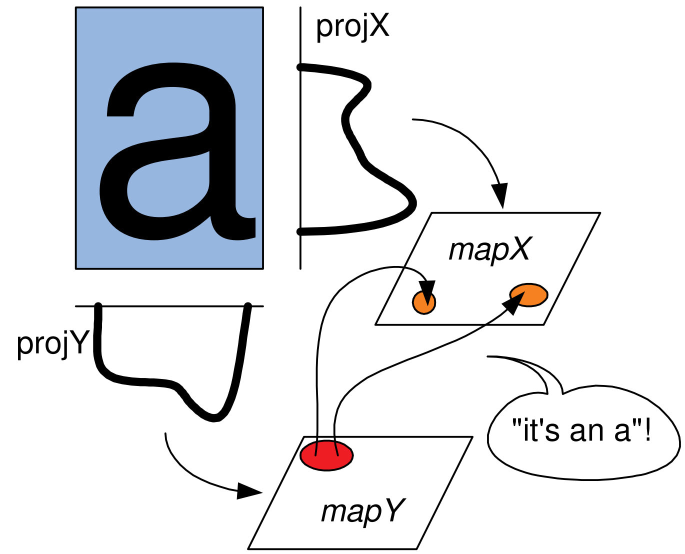
Document map

demo

Future challenges (1)

exciting new theory: embodied intelligence

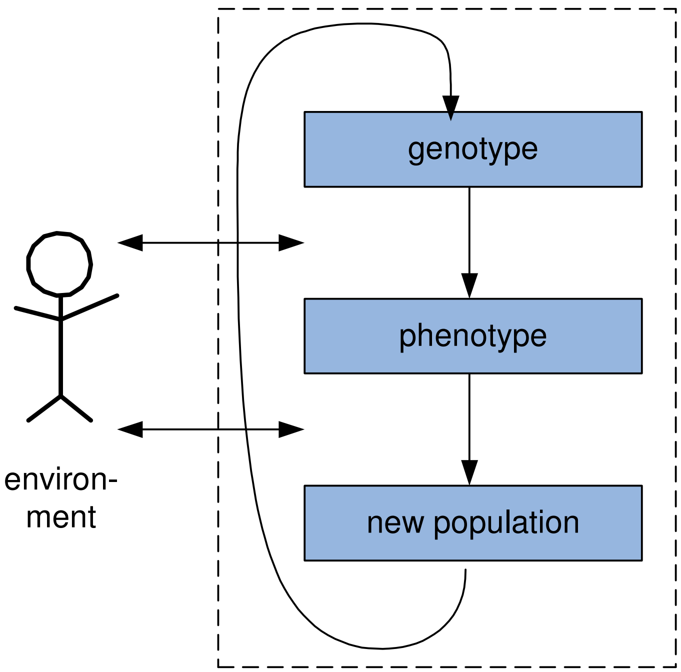


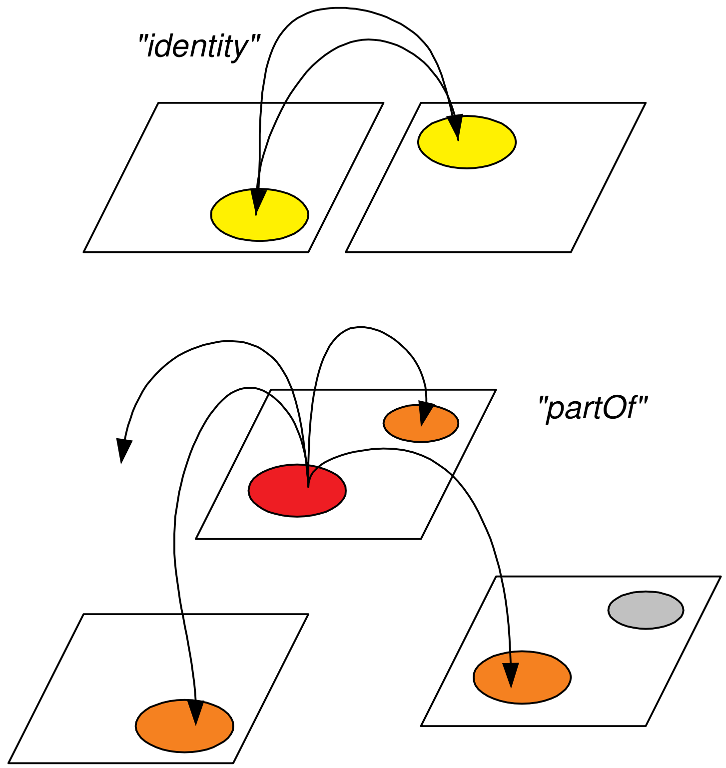


Embedded Engine Control new interdisciplinary methodology? $--, T--?

Document Analysis and Recognition, learning versus programming (complexity), Q++?, robustness?

Future challenges (2)





Document Management Systems
document archiving revisited: use “user” as “environment” in more pro-active DMS (human in loop, office lab)

Embodied ontology extraction
morphological analyses of neurostructures in global maps, grounding text based ontologies

SOM training algorithm & example

for *all epochs* ***do***

decrease α ; decrease σ ;

for *each sample* ***do***

compute high dimensional distance(sample, all_neurons); (1)

determine winning neuron; (2)

compute low dimensional distance(winning_neuron, all_neurons); (3)

compute neighbourhood (low dimensional distance, σ); (4)

correct all_neurons = all_neurons + α . neighbourhood . (sample – all_neurons); (5)

end

end

next sample

sample==198

123	46	84	19
0	19	200	96
250	23	190	55
97	60	85	143

original neuron values

1 2

winner

75	152	114	179
198	179	2	102
52	175	8	143
101	138	113	55

(1) high dimensional distance
(2) winner selection

3

3	2	1	2
2	1	0	1
3	2	1	2
4	3	2	3

(3) two dimensional distance

4

0	0.25	0.5	0.25
0.25	0.5	1	0.5
0	0.25	0.5	0.25
0	0	0.25	0

(4) neighbourhood = Λ_{ij}

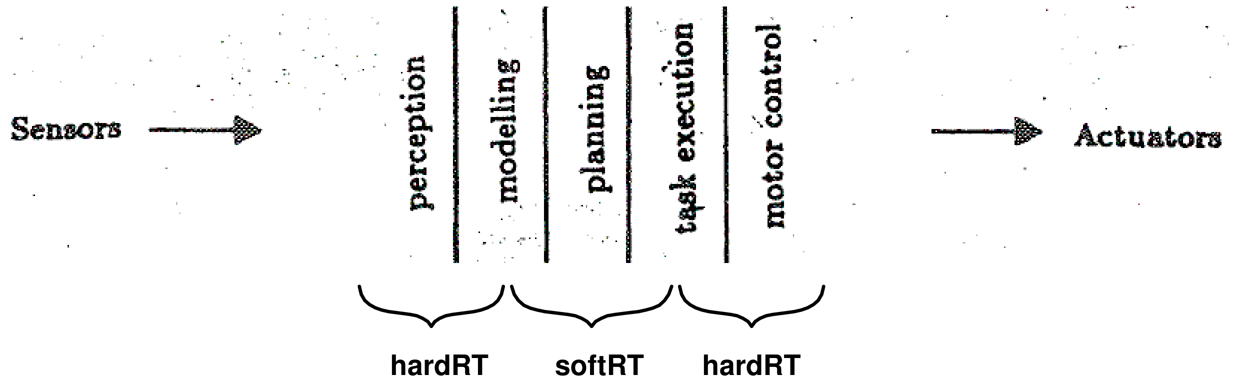
5

123	84	141	63.75
49.5	108.5	198	147
250	66.75	194	90.75
97	60	113.25	143

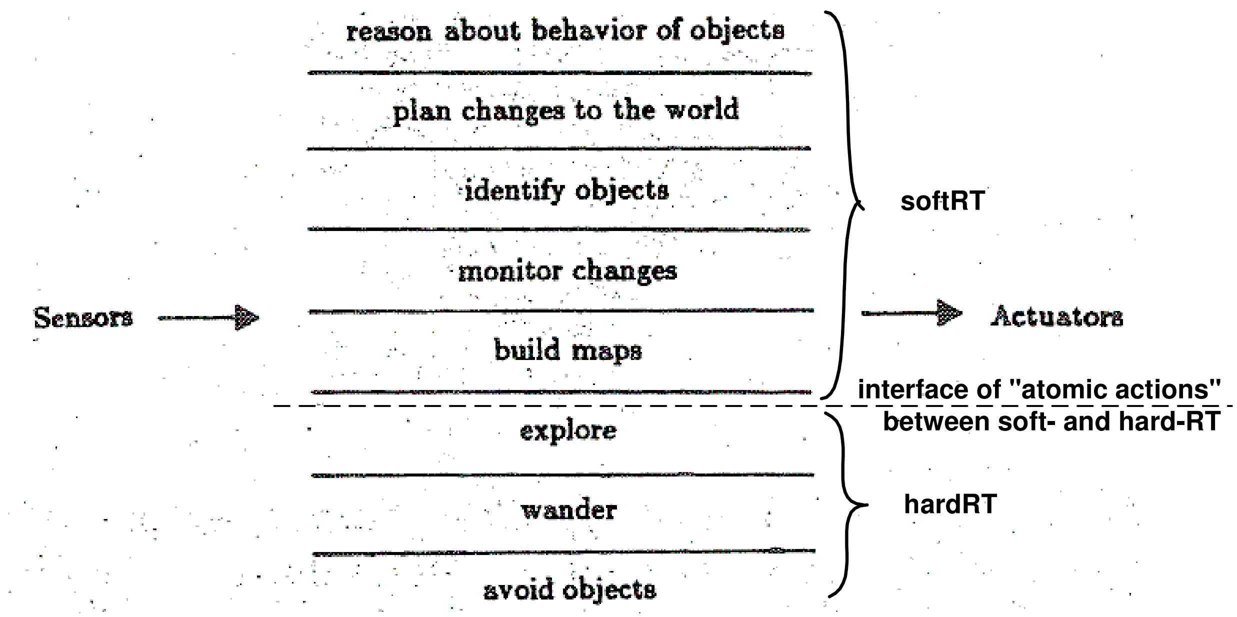
(5) updated neuron values

decrease α
decrease σ
next epoch

Vision(3)



- Hard RT responsiveness deteriorated by soft RT behaviour
- Difficult to test



- Lower: hard RT
- Higher: focus on the longer term
- Easy testing

Early warning system (printer)

What

Optimise uptime by “sensing” a printer problem before the machine fails the customer.

How

•Add cheap sensors and interprete the vibration signals continuously.

• make a fingerprint of an individual engine’s frame vibration

• monitor for deviations in this fingerprint

• inform OBS to schedule preventive maintenance to optimise uptime

Smart spectrogram analysis

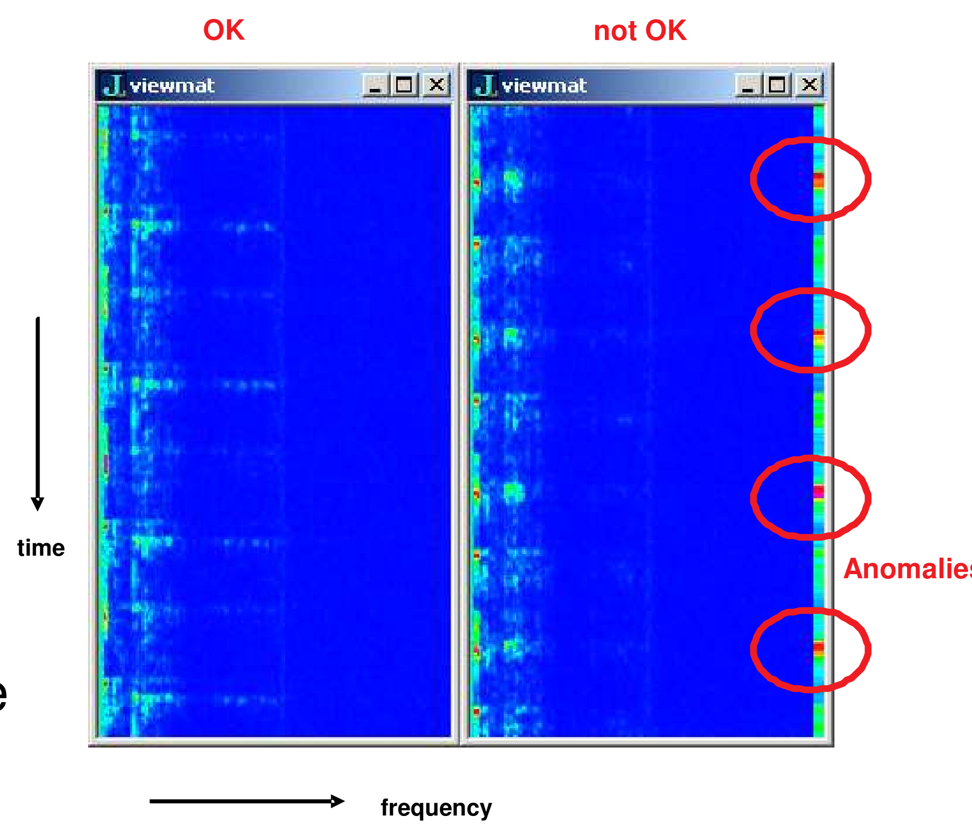


Image processing
with less programming

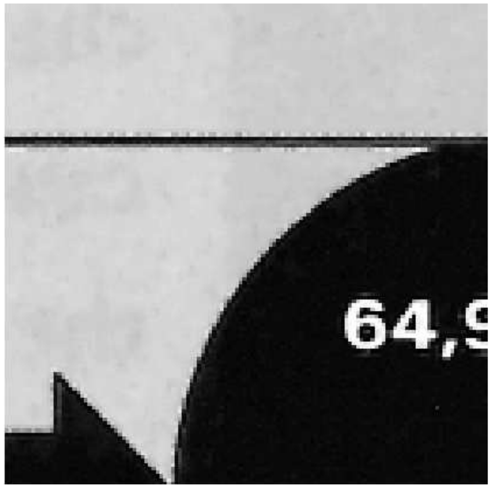
- Image processing by throwing dices
- Simulated Annealing

Introduction

Scanned Business Graphics

Grey-value original scan

[0..255]

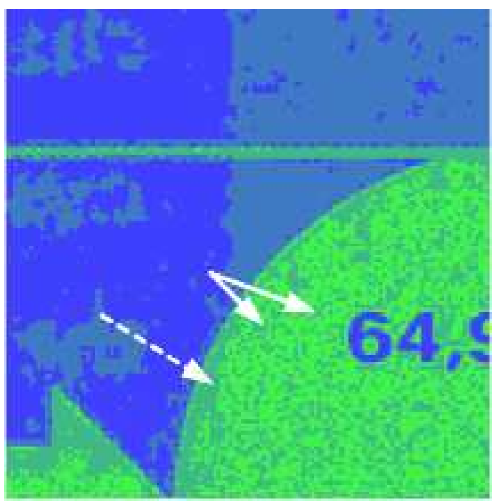


Traditional quantisa-tion

[0..L-1]

(false coloured,

e.g. L=4)



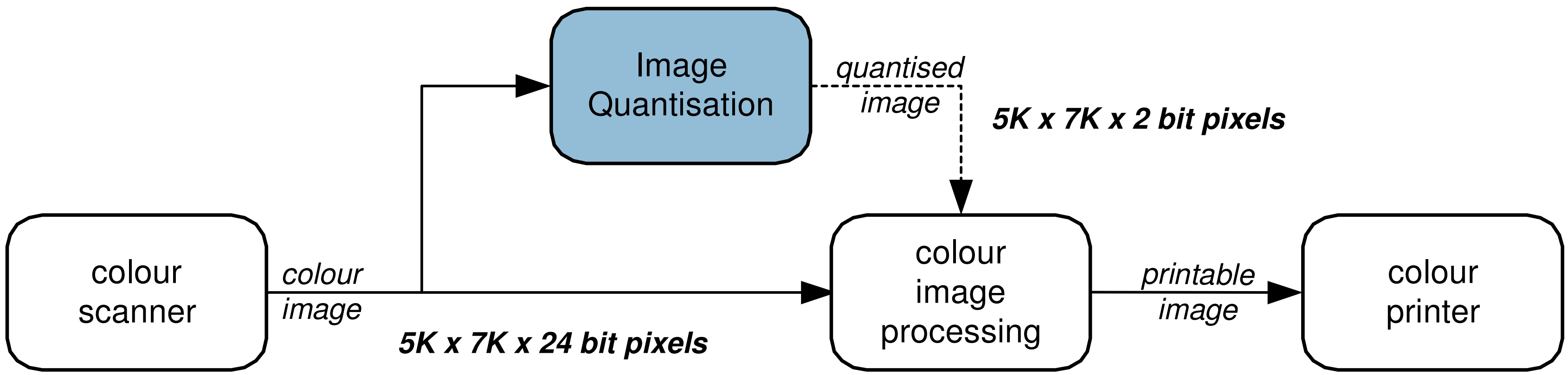
Problems quantisers

- quality (difficult to compress)
- computation time

Question

Can massively // computing improve on quality(compression) and performance?

Specification Quantisation module

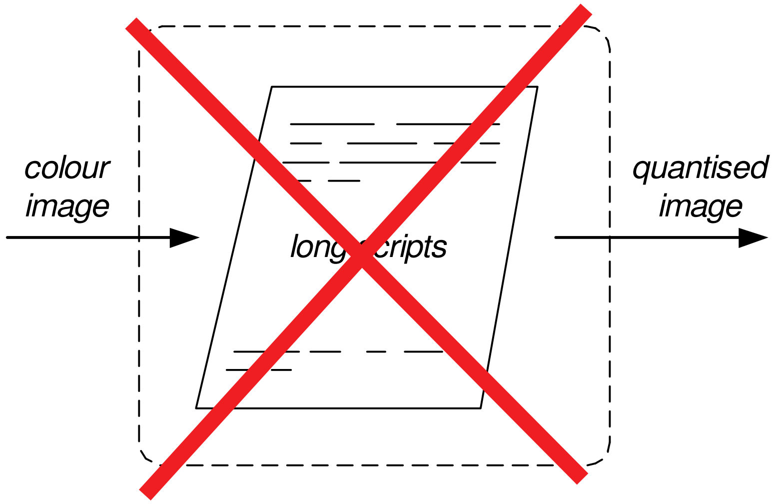


- sizes bitmap: 5,000x7,000
- **improved** quantisation takes 1 minute (Pentium)

Goals *(w.r.t. massively parallel implementation)* :

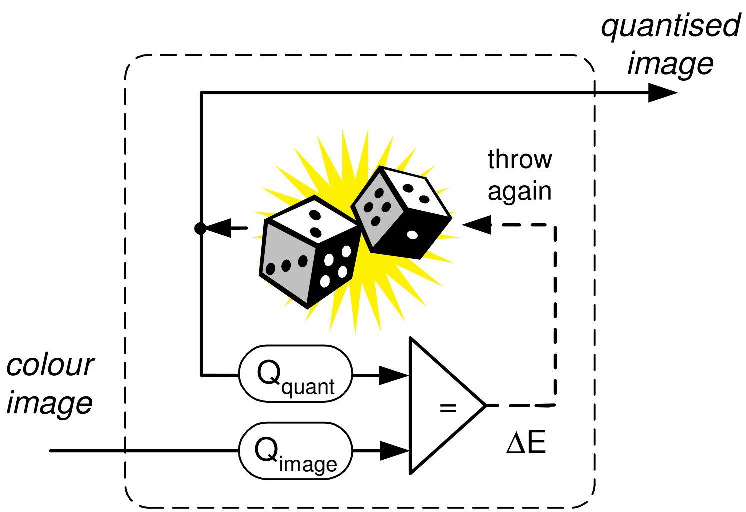
- improve business graphics quality
- faster (few seconds)

Analysis & Design



Complexity $O(L^{W.H})$

Based on 2 steps:
1) perceptual optimisation crit.
2) iterate until convergence (by e.g. Simulated Annealing)

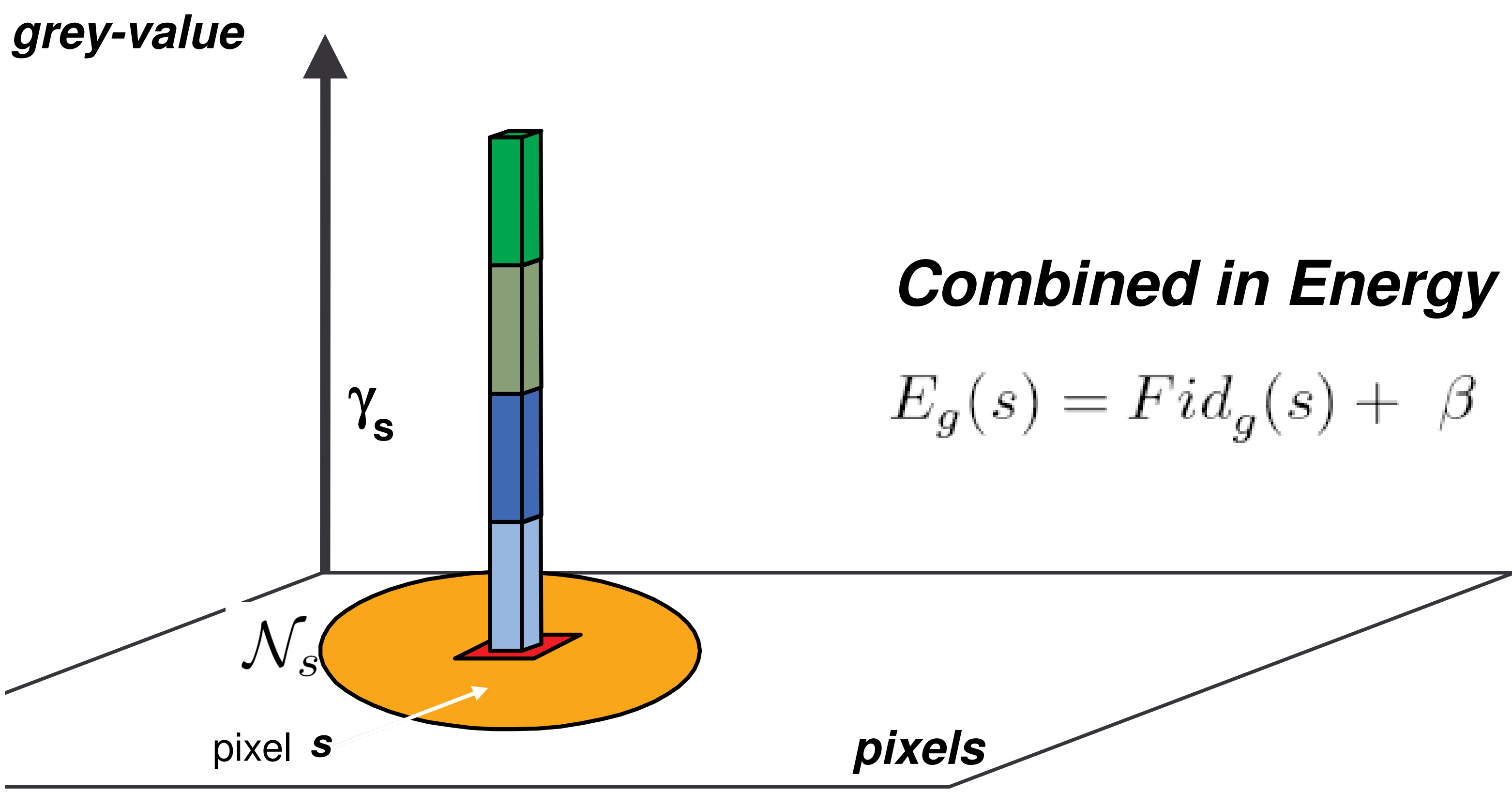


Markov Random Field

How to control loss of information?

- *fidelity* $Fid_g(s) = (\gamma_s - \mu_{g_s})^2$
- *regularity* $Reg_g(s) = \sum_{r \in \mathcal{N}_s} \delta(g_s, g_r)$

γ_s	grey-value of pixel s
μ_g	mean of class g
g_s	quantisation state
$\mathcal{N}_s$	neighbourhood of s
β	weight factor



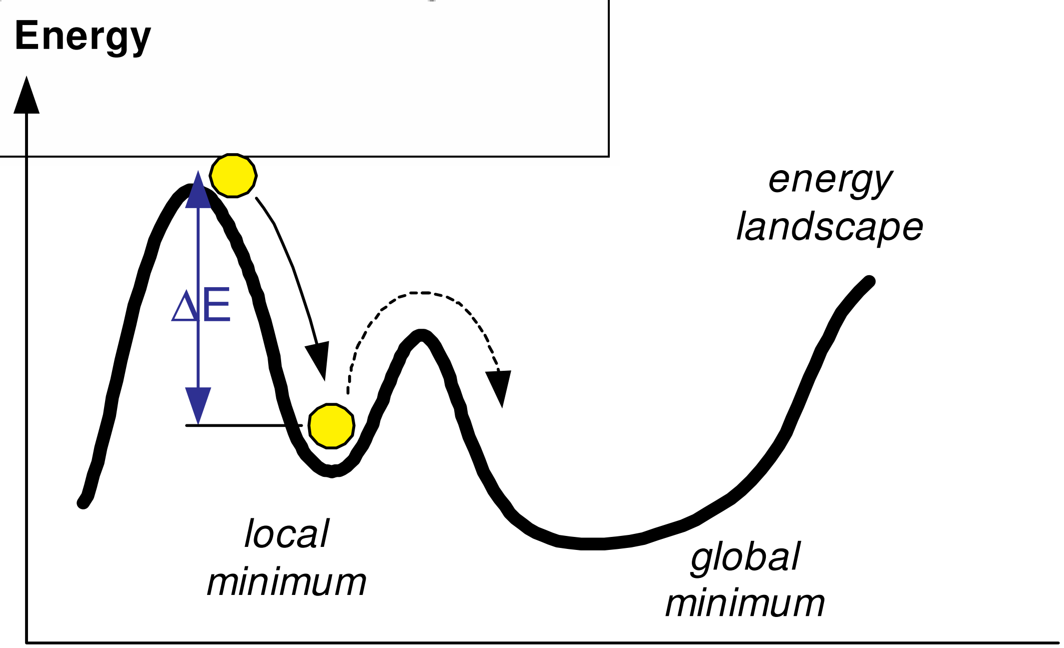
Combined in Energy (local):

$$E_g(s) = Fid_g(s) + \beta \cdot Reg_g(s)$$

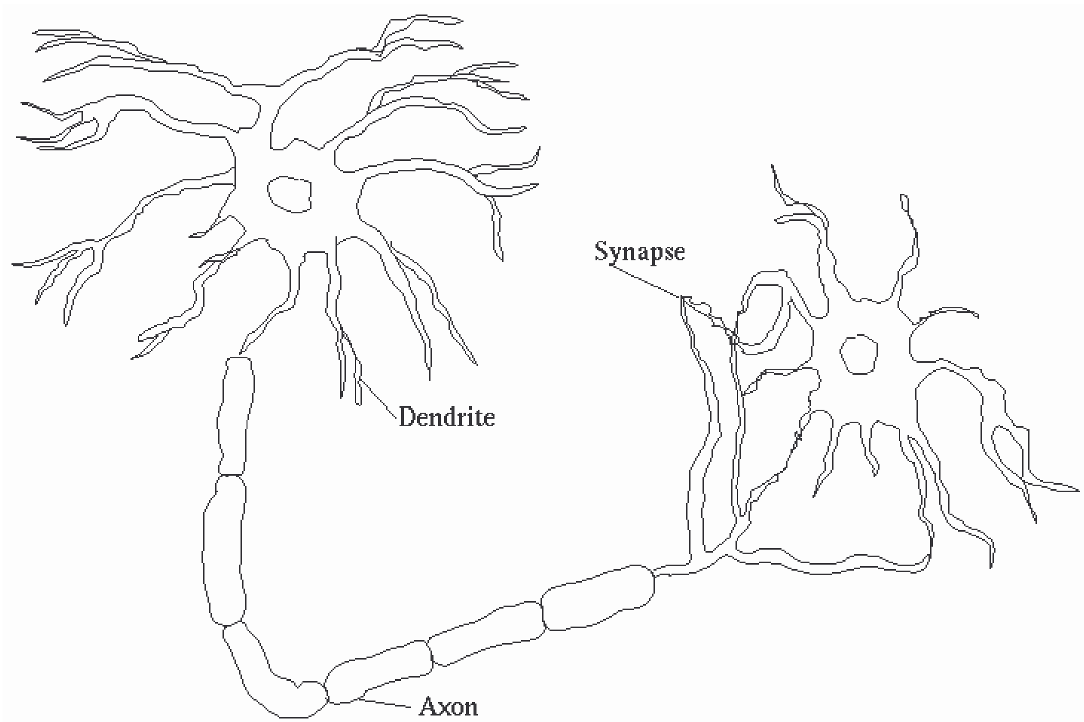
Simulated Annealing

$x \leftarrow$ Random state
for $T \leftarrow T_0, T_0 \cdot C, \ldots, T_0 \cdot C^{A-1}$ **do**
  $\hat{x} \leftarrow$ Slightly changed state x
  $\Delta E \leftarrow E(\hat{x}) - E(x)$
  if $\exp\left\{-\frac{\Delta E}{T}\right\} \geq 1$ **then** {Deterministic acceptance}
    $x \leftarrow \hat{x}$
  else if $\exp\left\{-\frac{\Delta E}{T}\right\} \geq \mathrm{Rand}[0,1)$ **then** {Probabilistic acceptance}
    $x \leftarrow \hat{x}$
  end if
end for

movie



bio neural networks



Facts:

- > 10^{11} neurons
- 10^3 synapses/neuron
- msec performance/ neuron

Source: "Tutorial de Redes Neuronales", la universidad Politecnica de Madrid