KNOWLEDGE ENGINEERING & DISCOVERY RESEARCH INSTITUTE
AN INSTITUTE OF AUT UNIVERSITY

AUCKLAND UNIVERSITY OF TECHNOLOGY
TE WĀNANGA ARONUI O TAMAKI MAKAU RAU

The International Neural Network Society (INNS)

6th Balkan Conference in Informatics
17th Panhellenic Conference in Informatics
Thessalonica, Greece, 19-21.09.2013
Keynote Speech

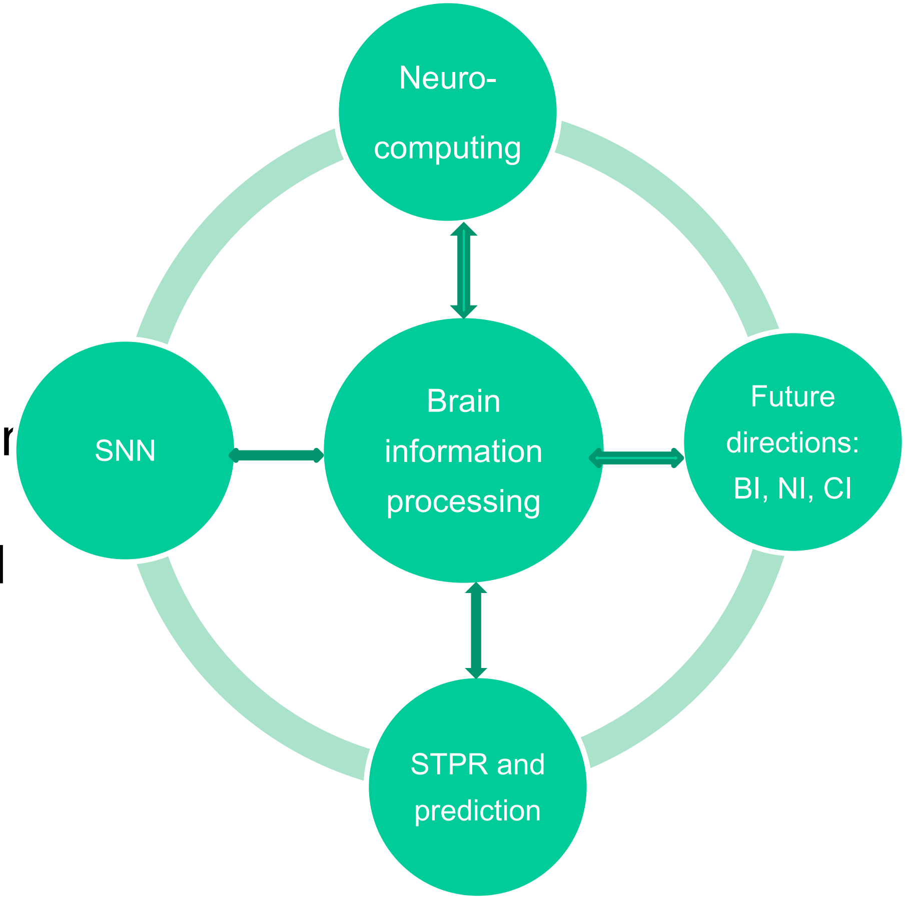
Neurocomputation as Brain Inspired Informatics: Methods, Systems, Applications

Nikola Kasabov, FIEEE, FRSNZ,
2013 Royal Academy of Engineering Distinguished Visiting Fellow

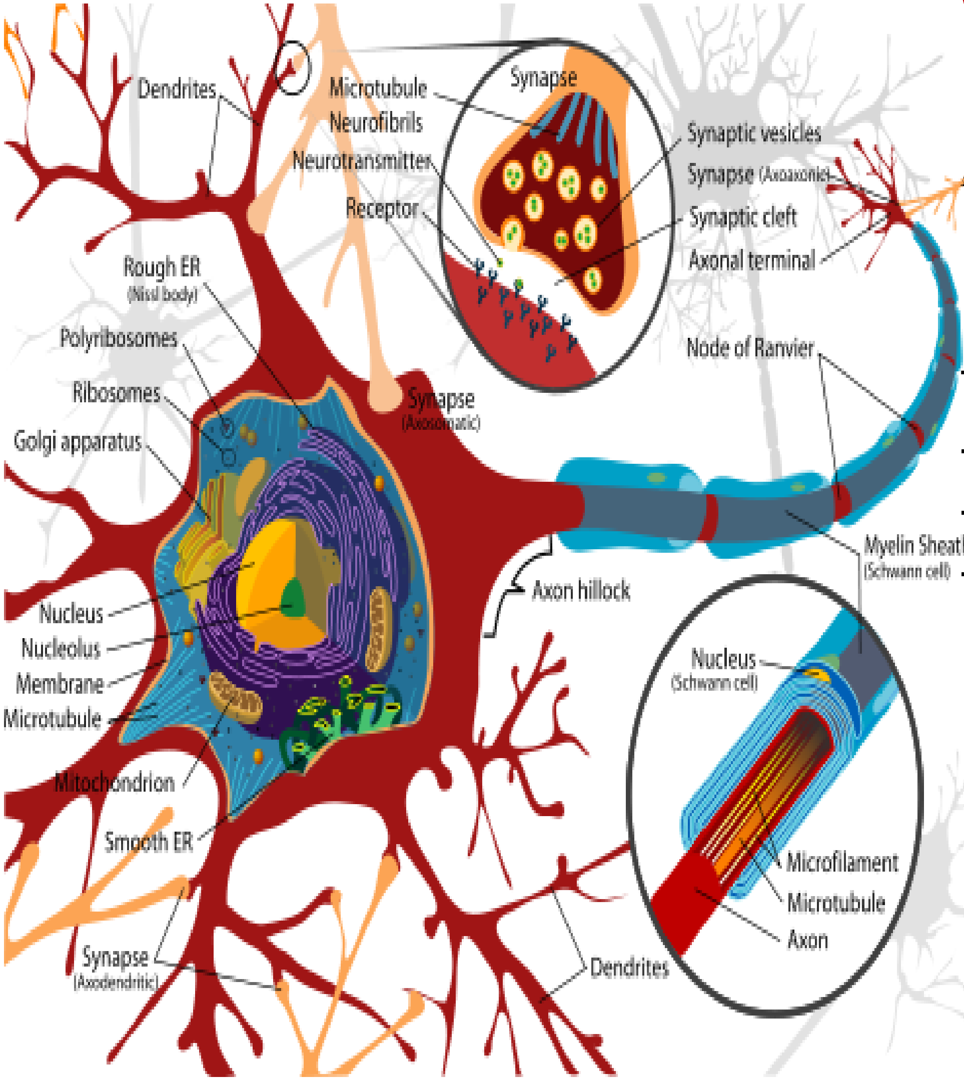
Professor and Director,
Knowledge Engineering and Discovery Research Institute (KEDRI),
Auckland University of Technology, New Zealand

Content

1. Brain information processing
2. Neurocomputing
3. Spiking Neural Networks (SNN)
4. SNN for Spatio/Spectro-Temporal Pattern Recognition and Predictior
5. Advantages and limitations of SNN
6. Future Directions



1. Brain Information Processing



Why do we need to look for inspiration from the brain for our informatics methods and systems?

A single neuron is very rich of information processes: time; frequency; phase; field potentials; molecular (genetic) information; space.

Three, mutually interacting memory processes:

- short term (membrane potential);
- long term (synaptic weights)
- genetic (gene and protein information)

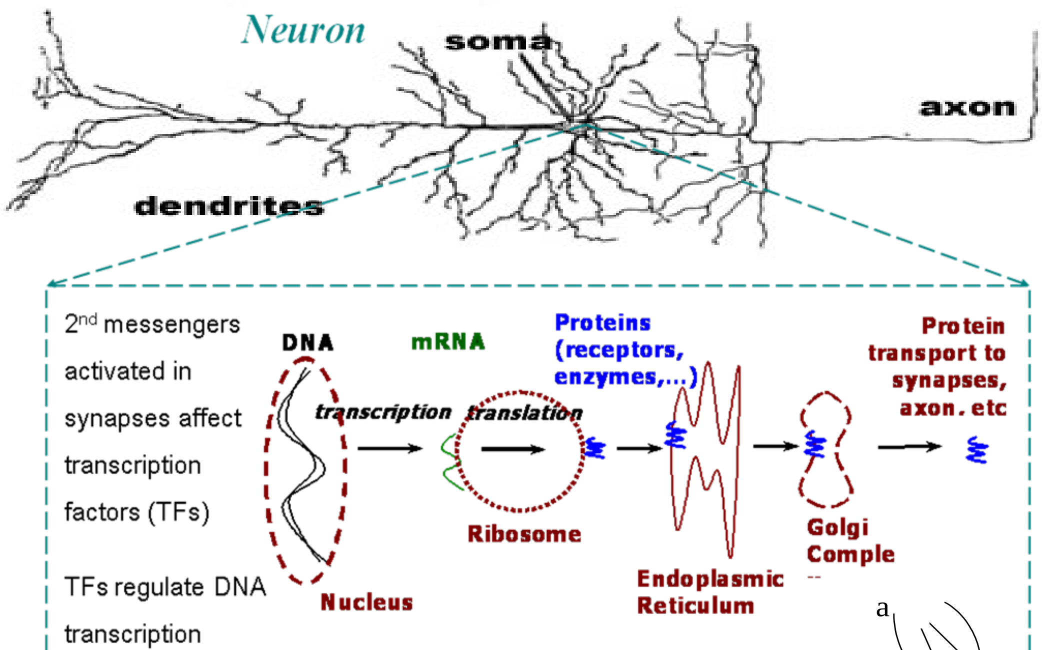
The brain is always evolving (developing, learning, changing) in a spatio-temporal way

The brain is an excellent spatio-temporal information processing machine

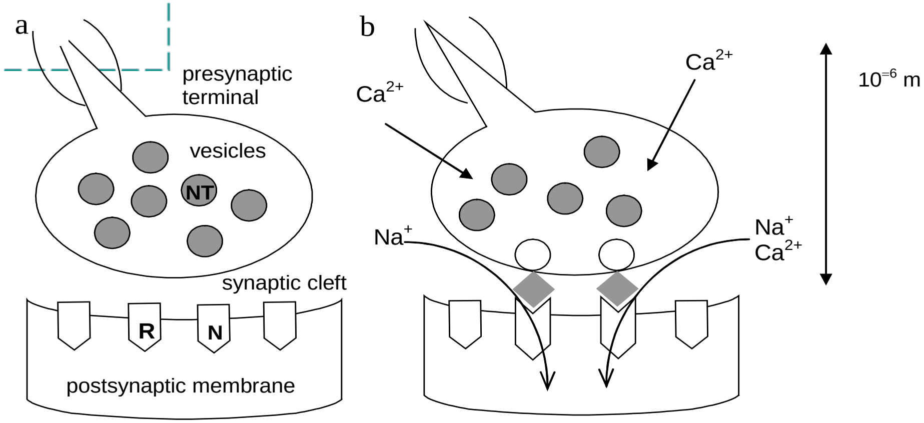
A vast amount of brain data has been collected (e.g. the EU Human Brain project, 1bln Euros, starting in 2013)

The challenge: To create brain-inspired informatics methods for efficient computation

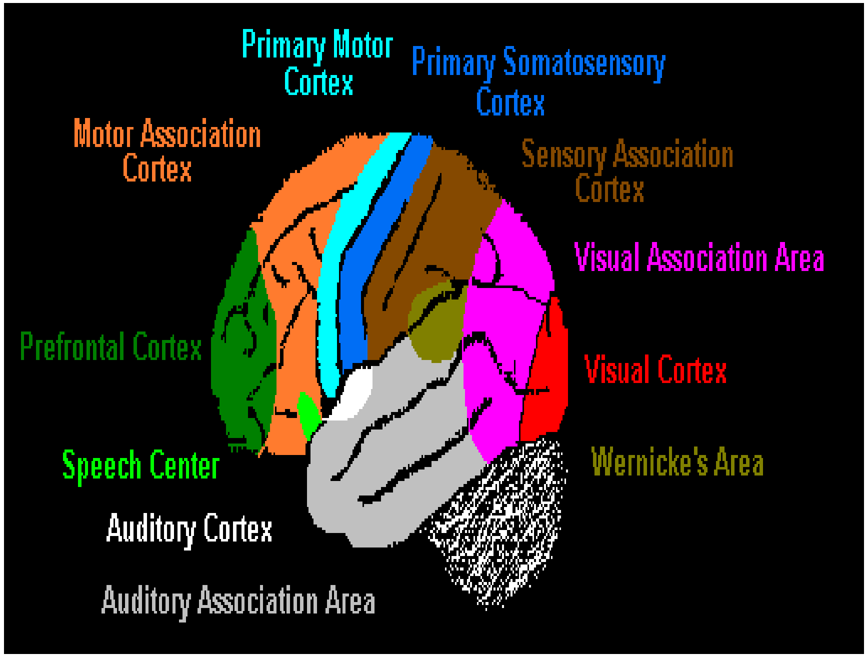
Neurogenetic information processes in neurons and synapses

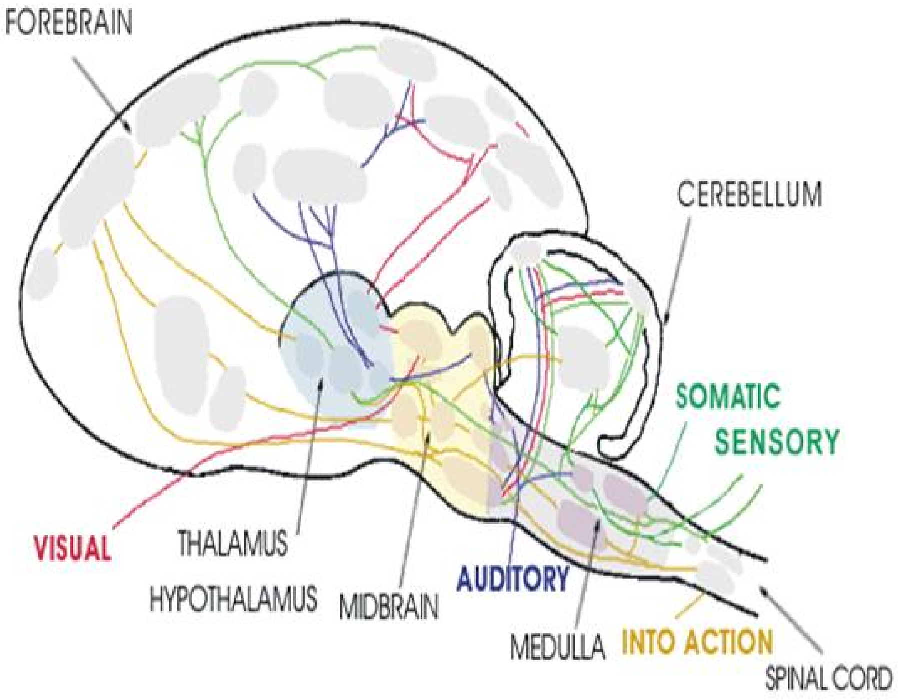


- Nature via Nurture
- Complex interactions between thousands of genes (appr. 6000 expressed in the brain) and proteins (more than 100,000)
- Different time-scales
- Stochastic processes
- Integration of Bio- and Neuroinformatics



Brain information processes are spatio/spectro -temporal



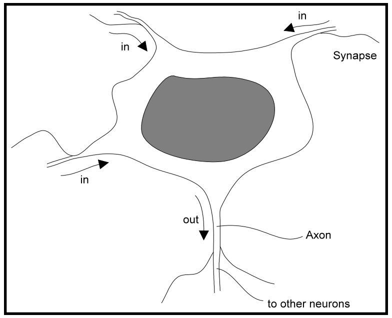


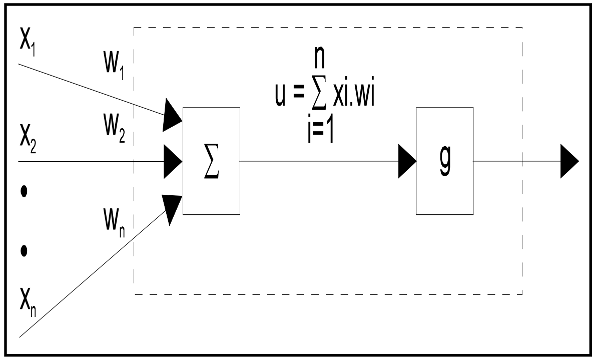
2. Neurocomputation

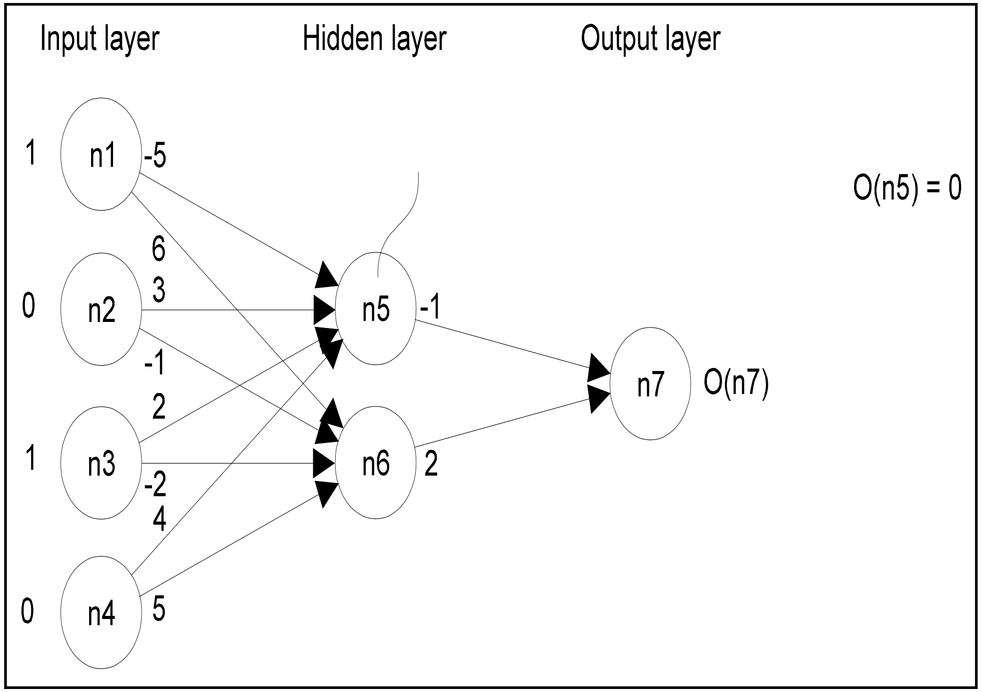
Artificial Neural Networks:

- NN are computational models that mimic the nervous system in its main function of adaptive learning.
- ANN can *learn* from data and make *generalisations*
- ANN are *universal computational models*
- Stable information is represented in the connections
- Dynamic information is represented as activity of neurons at a certain time

- ANN are non-von Neumann machines!!!







First Generation of ANN

- 1943, McCulloch and Pitts - a model of a neuron,
- 1960, Widrow and Hoff- Adelaine,
- 1962, Rosenblatt - Perceptron,
- 1971- 1986, Amari, Rumelhart and others, Multilayer perceptron

Societies and conferences:

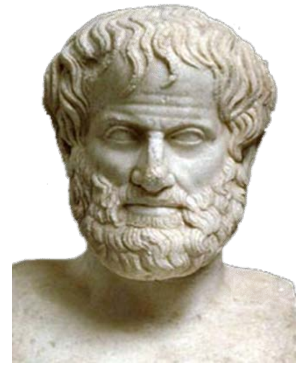
- 1980, International Neural Network Society (INNS) www.inns.org, Grossberg, IJCNN -> IJCNN 2015!!!

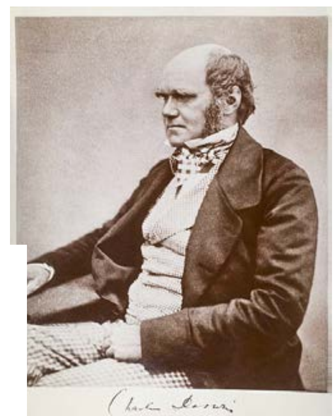
- 1992, European Neural Network Society, (ENNS), J.Taylor (1936-2012), ICANN

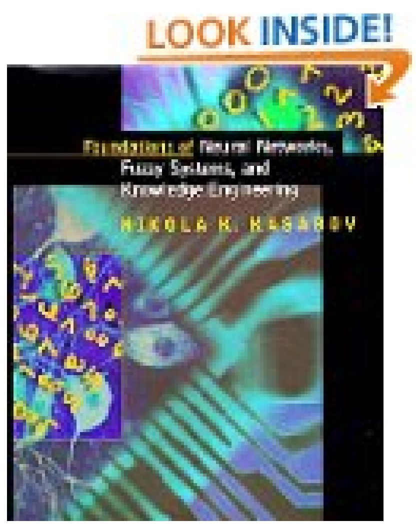
- 1993, Asia Pacific Neural Network Assembly (APNNA), www.apnna.net, Shun-ichi Amari, ICONIP

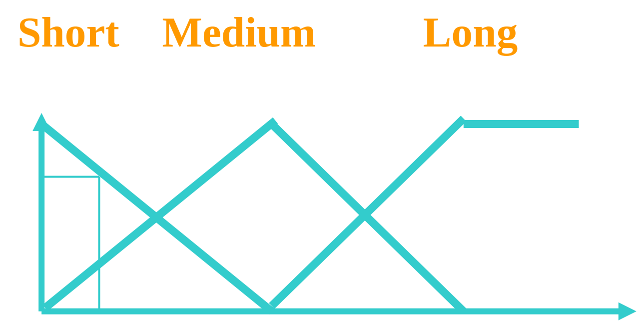
Second Generation of ANN: Hybrid neuro-symbolic, neuro-evolutionary and neuro-fuzzy information processing

- Connectionist -symbolic systems (Propositional logic, Aristotle, 4c BC)
- Fuzzy and neuro-fuzzy systems (Zadeh, 1965; Yamakawa 1989; Kosko 1992; Kasabov 1992)
- Neuro-evolutionary computation
- Kasabov, Foundations of neural networks, fuzzy systems and knowledge engineering, MIT Press 1996
- IEEE Comp. Intelligent Society



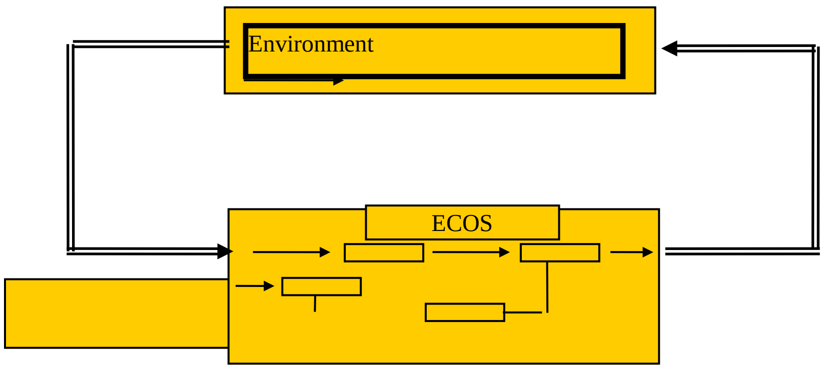


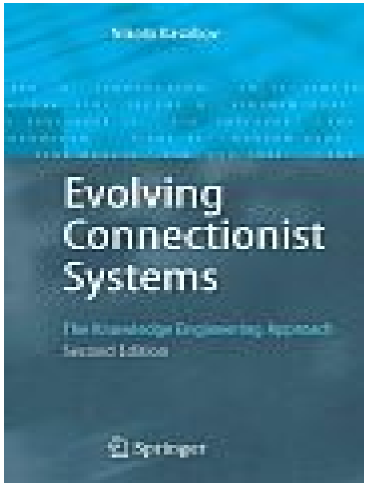




Evolving Connectionist Systems (ECOS)

- ECOS are modular connectionist-based systems that evolve their structure and functionality in a continuous, self-organised, in on-line, adaptive, interactive way from incoming information facilitating knowledge discovery (Kasabov, 1998, 2002, 2007).





- Main features of ECOS:
  - Evolving (developing) both the structure and the functionality
  - Incrementally learning, adaptive;
  - Knowledge –based (extracting and inserting rules)
  - Local learning (no catastrophic forgetting)
  - Memory based: leave a track of the learning process

M.Watts, *Ten years of Kasabov's evolving connectionist systems, IEEE Tr SMC- part B, 2008.*

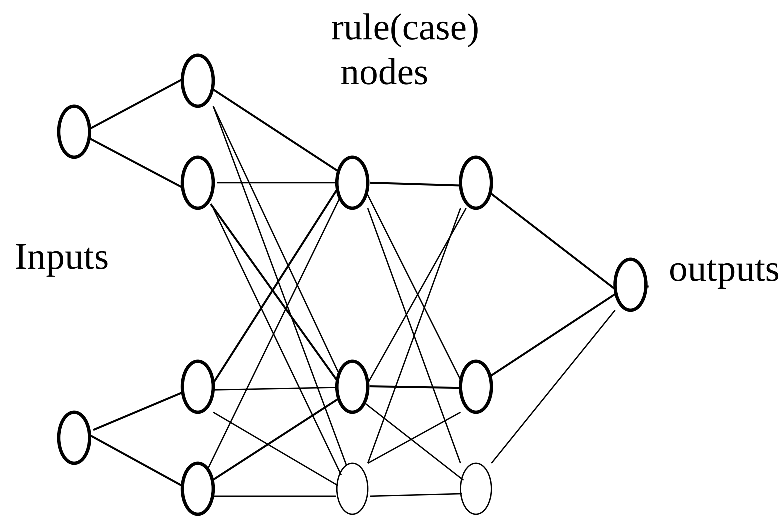
Evolving Fuzzy Neural Network (EFuNN)

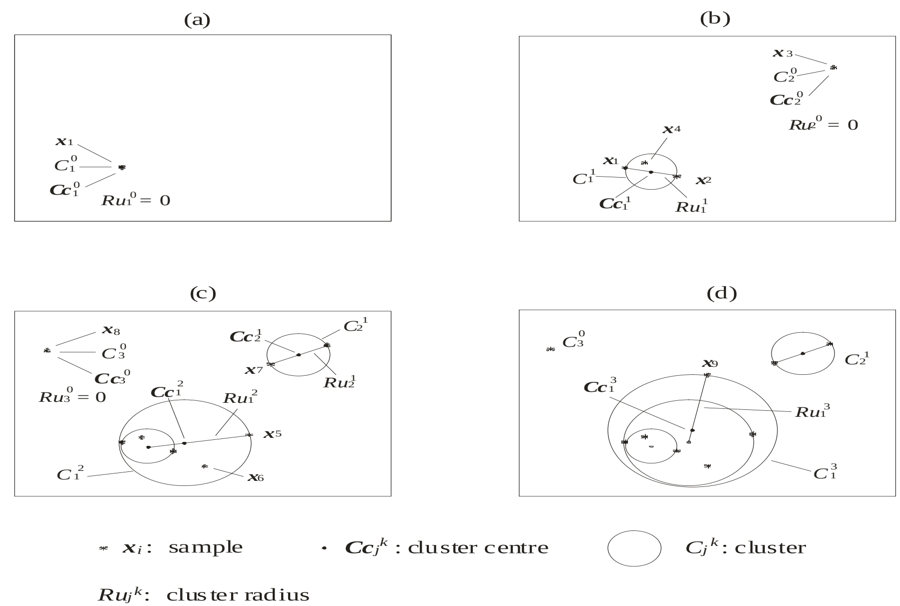
- Incremental, supervised clustering
- Input and/or output variables can be non-fuzzy (crisp) or fuzzy
- Hidden nodes evolve to capture clusters (prototypes) of input vectors
- Input weights change based on *Euclidean distance* between input vectors and prototype nodes (evolving clustering):

$$\Delta \mathbf{w} = \mathrm{lrate} \; * \; E(\mathbf{x}, Rn)$$

- Output weights evolve to capture local output function and change based on output error.

- EFuNN, N. Kasabov, IEEE Tr SMC, 2001
- DENFIS, N.Kasabov , Q.Song, IEEE Tr FS, 2002
- ECOS Toolbox available in MATLAB
- NeuCom Software available: www.kedri.info

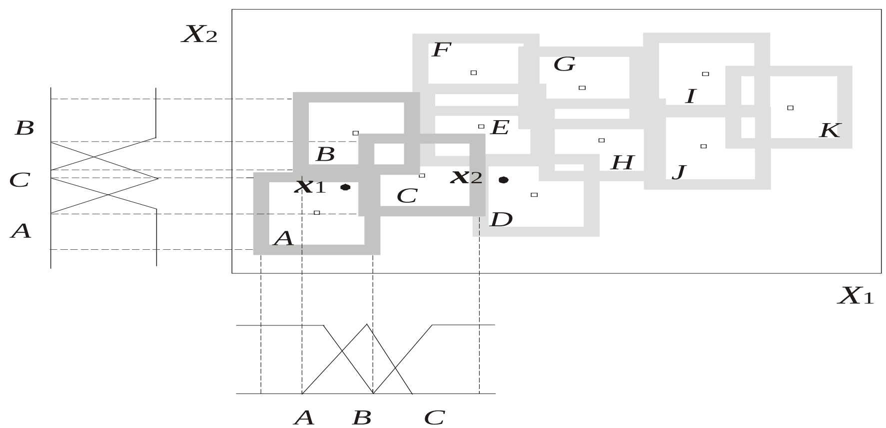




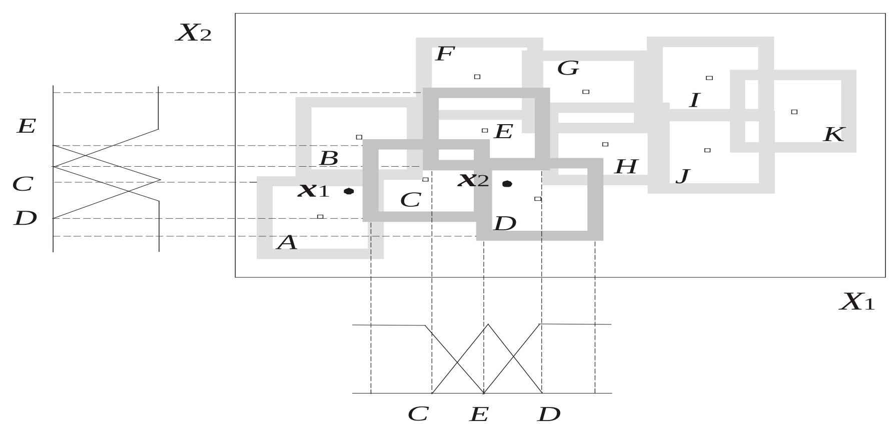
DENFIS: Evolving Neuro-Fuzzy Inference System

(DENFIS, Kasabov and Song, 2002, IEEE Tr Fuzzy Systems, 600 citations

(a) Fuzzy rule group 1 for a DENFIS



(b) Fuzzy rule group 2 for a DENFIS



DENFIS algorithm:

(1) Learning:

- Unsupervised, incremental clustering.
- For each cluster there is a Takagi-Sugeno fuzzy rule created: IF x is in cluster Cj THEN yj = fj (x),

where: yi = β0 + β1 x1 + β2 x2 + … + βq

- Incremental learning of the function coefficients and weights of the functions through least square error

(2) Fuzzy inference over fuzzy rules:

- For a new input vector x = [x1,x2, … , xq] DENFIS chooses m fuzzy rules from the whole fuzzy rule set for forming a current inference system.
- The inference result is:

$$y = \frac{\Sigma_{i=1,m} [\, \omega i \; fi \, (x1, x2, \ldots, xq)]}{\Sigma_{i=1,m} \, \omega i}$$

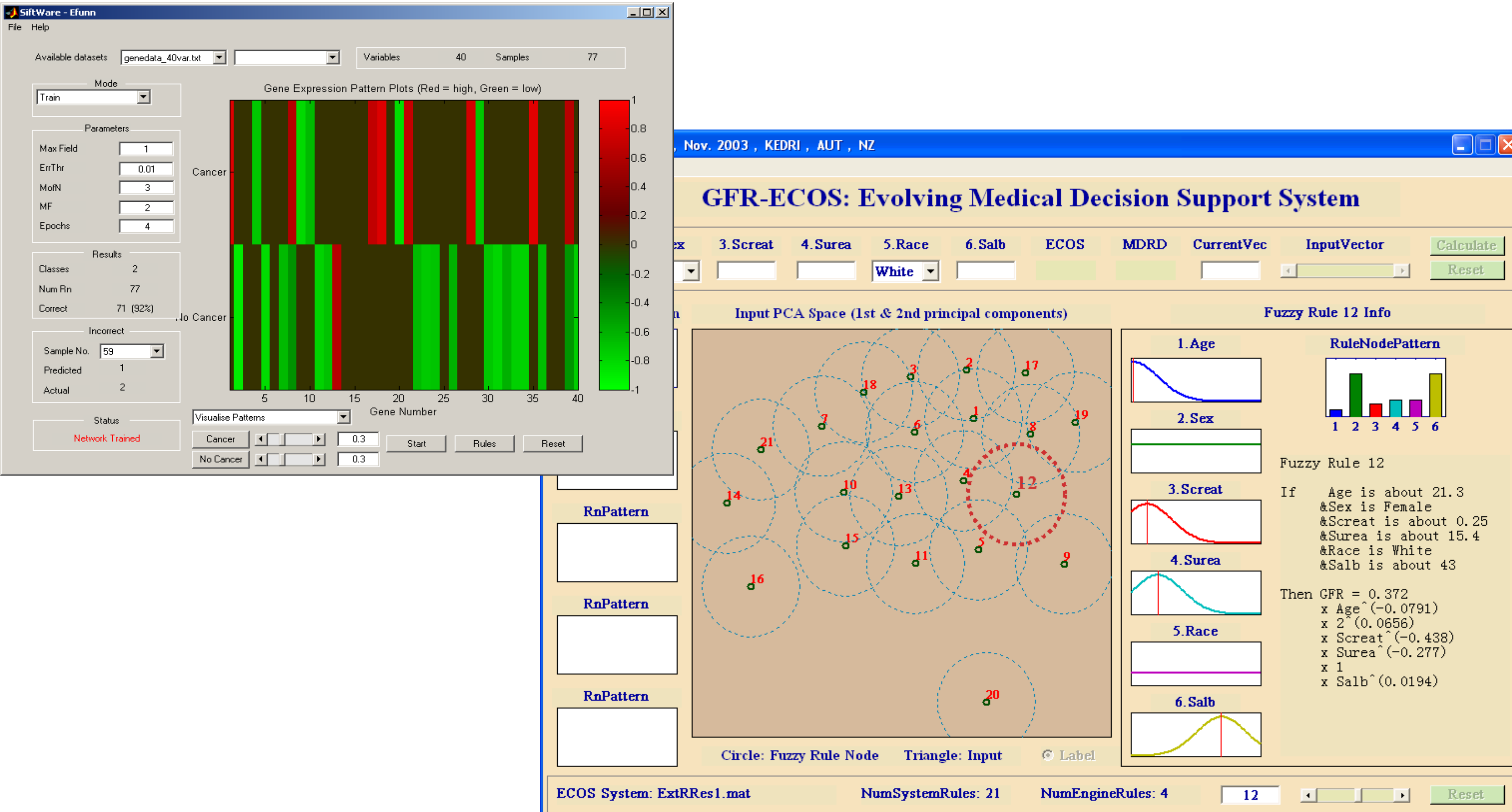
nkasabov@aut.ac.nz www.kedri.info

Applications of ECOS

- Bioinformatics -

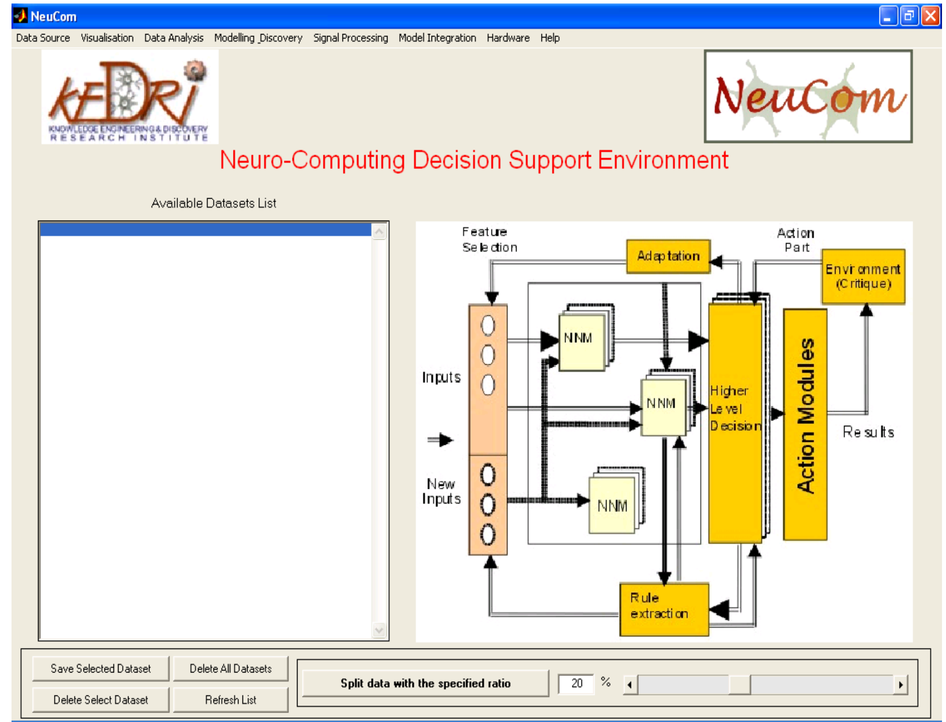
- Neuroinformatics

- Decision support systems



NeuCom: A Software Environment for NeuroComputing, Data Mining and Intelligent System Design (www.theneucom.com)

- A generic environment, that incorporates 60 traditional and new techniques for intelligent data analysis and the creation of intelligent systems, including:
  - Statistical methods
  - Neural networks

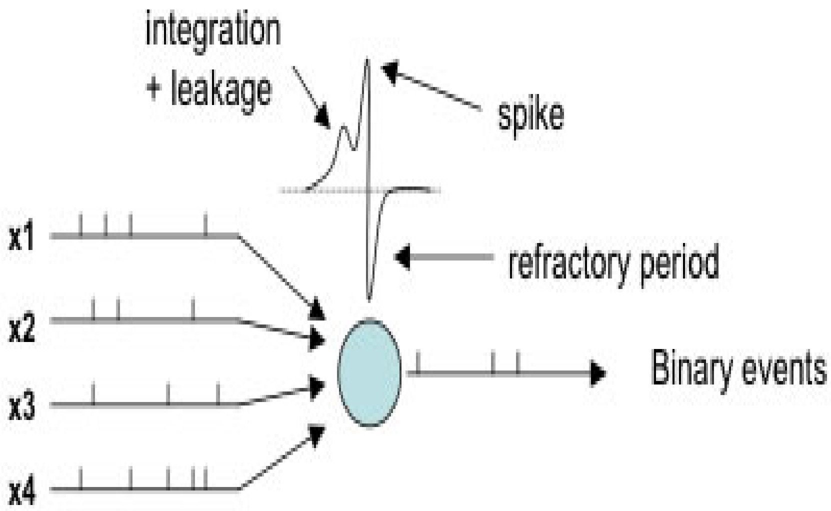


- Methods for feature selection
- Methods for classification
- Methods for prediction
- Methods for knowledge extraction
- Fast data analysis and visualisation
- Fast model prototyping
- A free copy available for education and research from: www.theneucom.com
- DENFIS for prediction
- ECF for classification

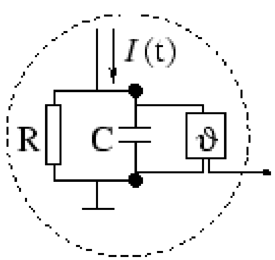
3. Third Generation of ANN: Spiking Neural Networks:

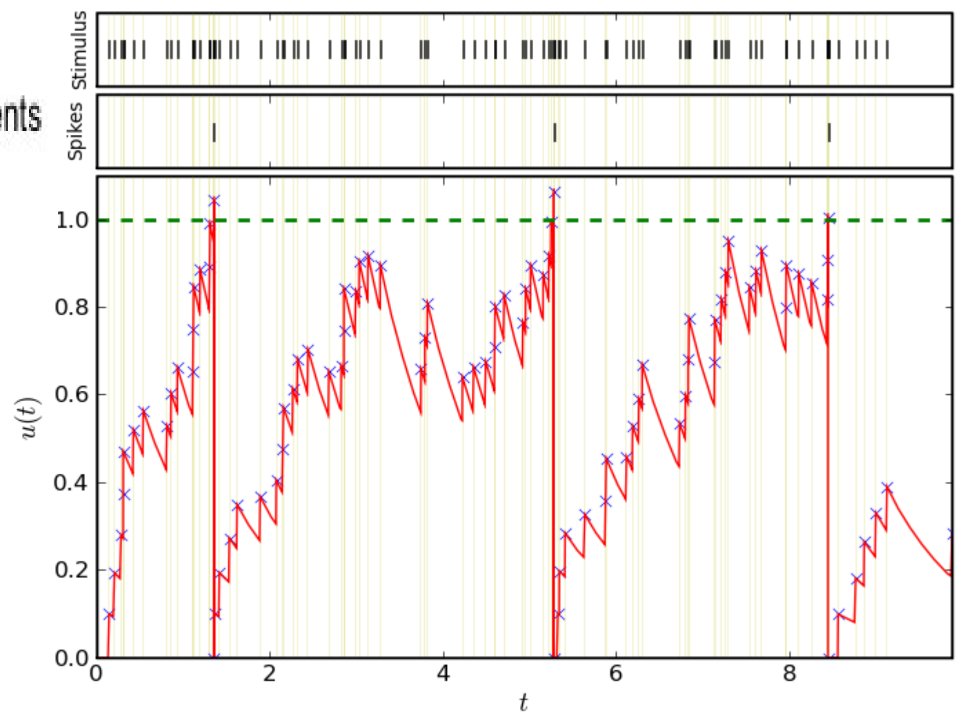
Models of spiking neurons: (Hodgkin-Huxley 1952; Abbott, 2000; Maas, Izhikevich; other)

Most popular is the Leaky Integrate and Fire Model (LIF) .



$$\tau_m \frac{du}{dt} = -u(t) + RI(t)$$

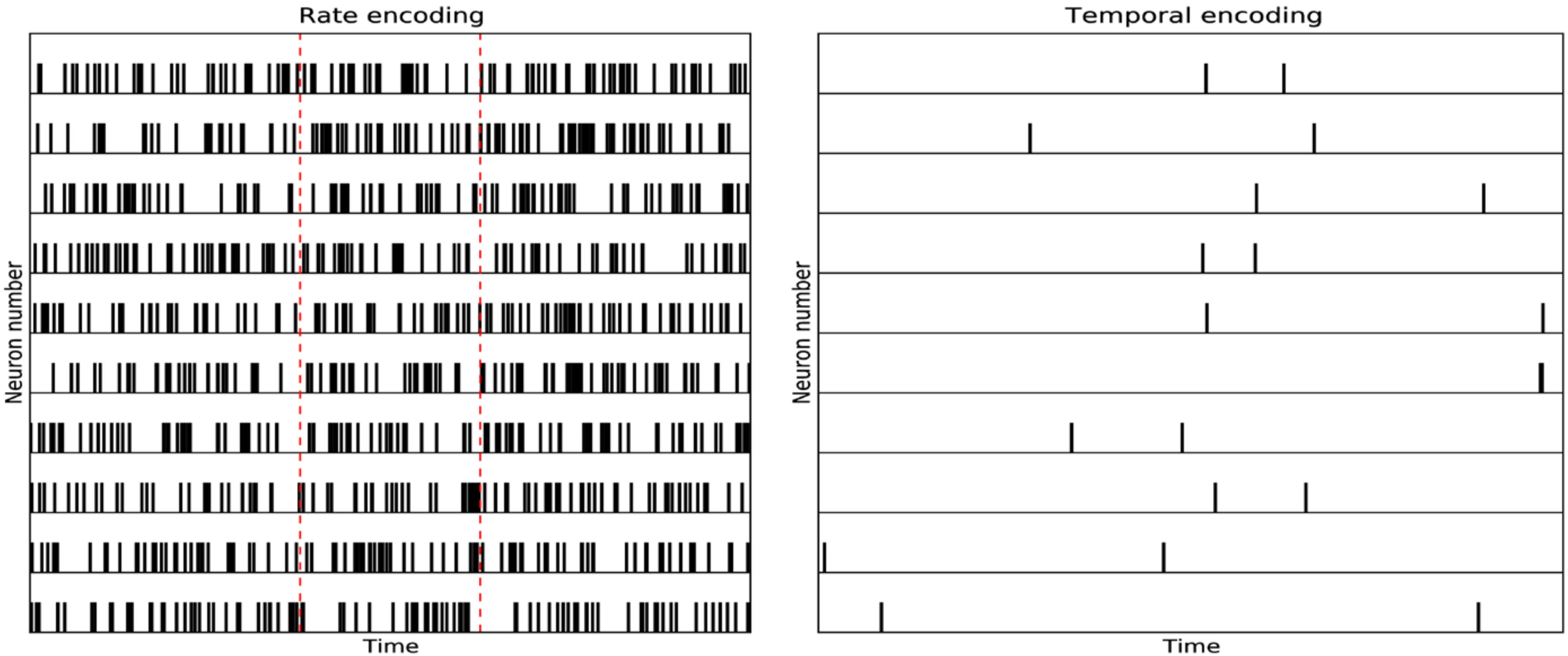




Representing information as spikes: Rate vs time-based

- Rate-based coding: A spiking characteristic within a time interval, e.g. frequency.

- Time-based (temporal) coding: Information is encoded in the time of spikes. Every spike matters! For example: class A is a spike at time 10 ms, class B is a spike at time 20 ms.

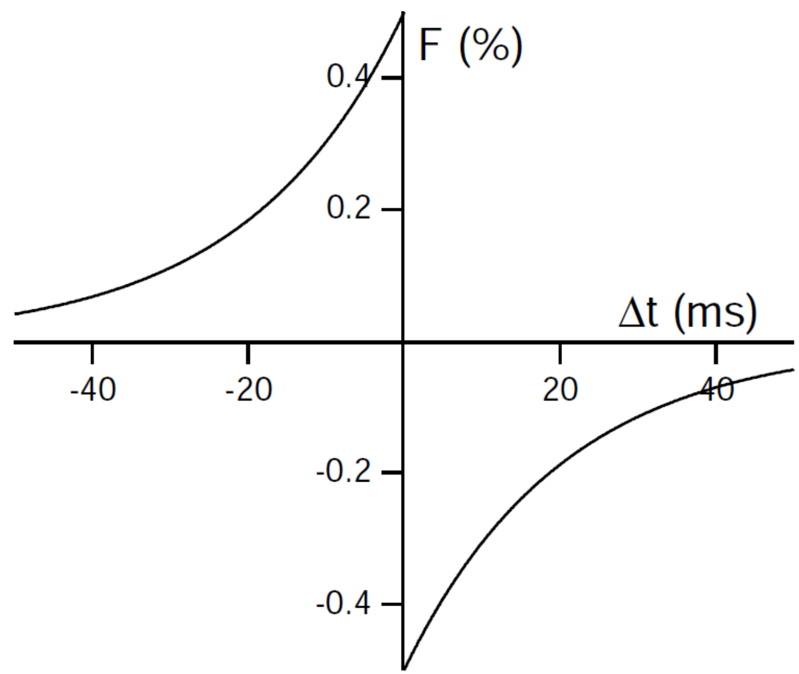


Methods for learning in SNN:
Spike-Time Dependent Plasticity (STDP)
(Abbott and Nelson, 2000).

- Hebbian form of plasticity in the form of long-term potentiation (LTP) and depression (LTD)
- Effect of synapses are strengthened or weakened based on the timing of pre-synaptic spikes and post-synaptic action potential.
- Through STDP connected neurons learn consecutive temporal associations from data.

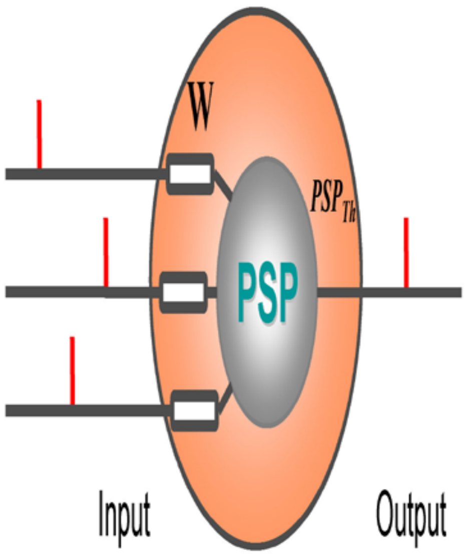
Pre-synaptic activity that precedes post-synaptic firing can induce LTP, reversing this temporal order causes LTD

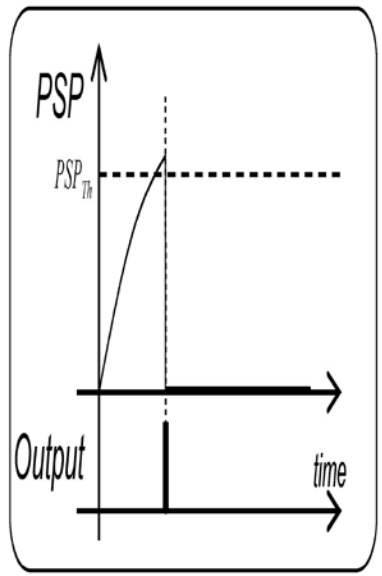
Δt=tpre -tpost



The rank order (RO) learning rule

(Thorpe et al, 1998)





$$\Delta w_{ji} = m^{\mathrm{order}(j)}$$

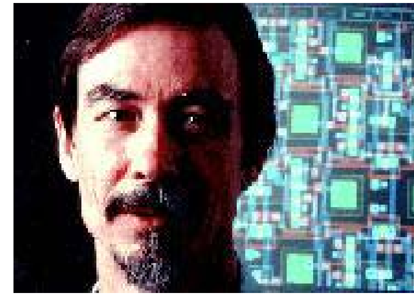
$$u_i(t) = \begin{cases} 0 & \text{if fired} \\ \sum\limits_{j|f(j)<t} w_{ji} m_i^{\mathrm{order}(j)} & \text{else} \end{cases}$$

PSP max = SUM ($m^{\mathrm{order}\,(j,i(t))}$ $w_{j,i}$(t)), for j=1,2.., N; t=1,2,...,T

$0 < m < 1$;

PSP_{Th}=C. PSPmax ($0 < C < 1$) – this is the parameter that allows the neuron to learn to spike before the whole pattern is presented)

Progress in neuromorphic computation



Hodgin- Huxley model (1952)

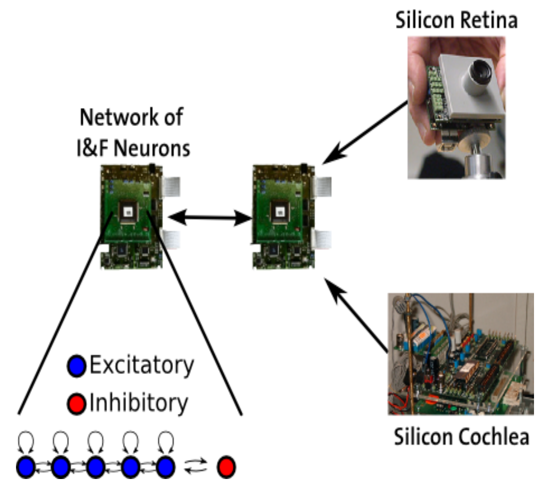
Carver Mead (1989): A hardware model of an IF neuron:
The Axon-Hillock circuit;

INI Zurich SNN chips (Giacomo Indivery, 2008 and 2012)

FPGA SNN realisations (McGinnity, Ulster, 2010);

The IBM chip (D.Modha, 2012): 256 LIF neurons and 64k synapses in a chip.

U. Manchester SpiNNaker (2^{16} computer chips, 2011; 1 mln neurons 2013) and the Stanford U., NeuroGrid (Kwabena Boahen et al), 1mln neurons on a board, 63 bln connections ; hybrid - analogue /digital)



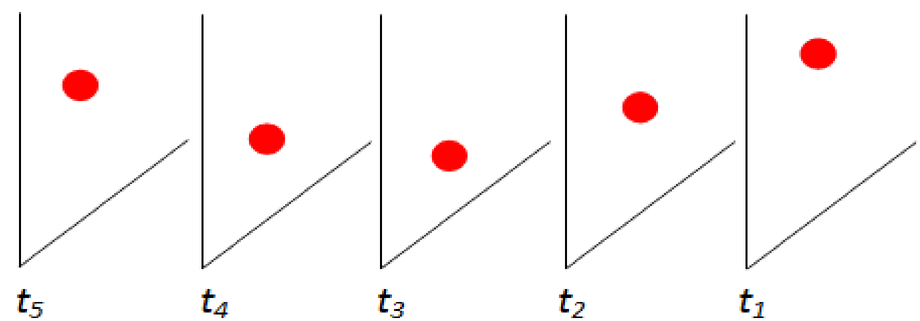
The challenge: Technology is available, but how do we use it for **integrated spatio-temporal data** modelling and STPR?

4. SNN for Spatio/Spectro-Temporal Pattern Recognition

- Most real world data is spatio- or spectro- temporal.
- In STPR problems spatial and temporal components of the information are interrelated.
- Examples of spatio-temporal data and related problems are:

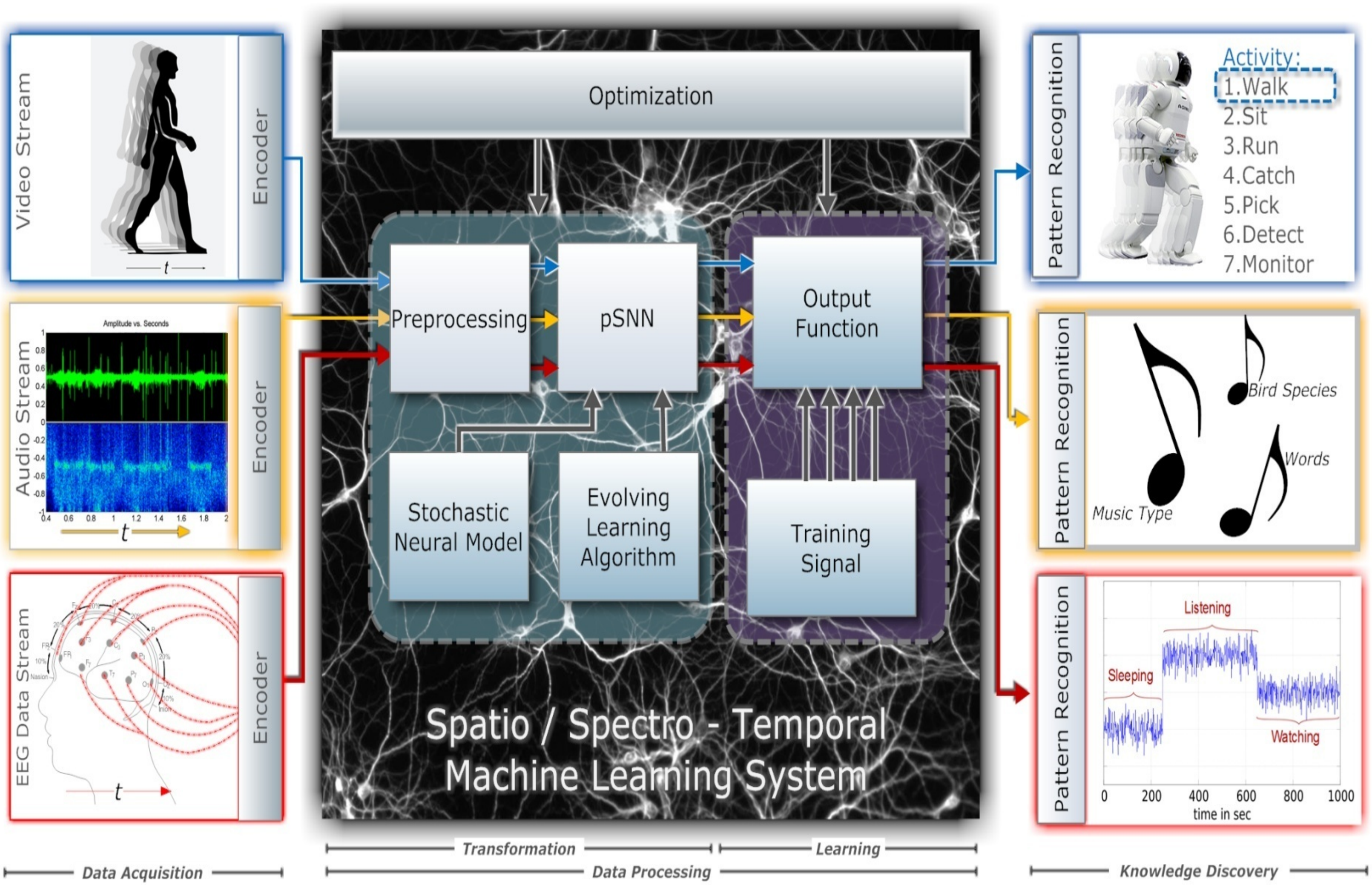
a) Object movement recognition from video data
b) Audio/video data modelling
c) Multisensor temporal data integration
d) Brain signals (EEG, MEG, fMRI)
e) Brain- computer interfaces
f) Motor control for prosthetics
g) Ecological and environmental data, e.g. earthquake prediction
h) Robot control
i) Cyber-security data

- Goal: Developing new methods

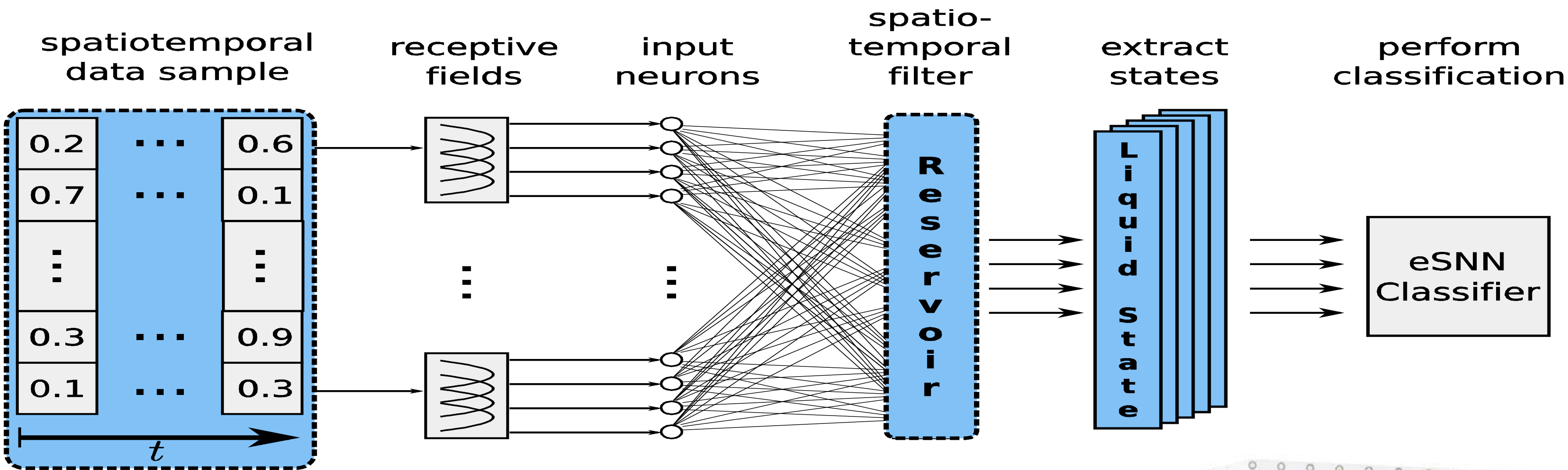


Spatio-temporal input data

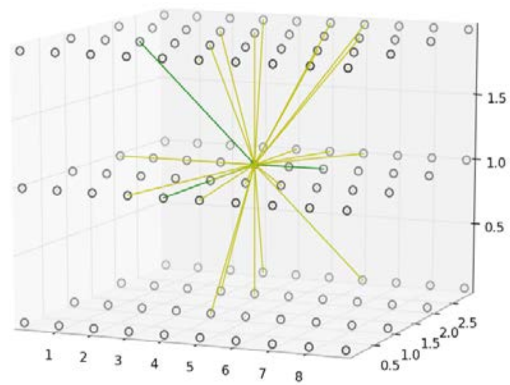
The EvoSpike Project: EU FP7Marie Curie (http://ncs.ethz.ch/projects/evospike)



Reservoir-based eSNN for STPR



- *Maass, W., Natschl ̈ager, T., Markram, H.: Real-time computing without stable states, Neur. Comp. 14(11),2002;*
- Input (feature) neurons connected to part of the LSM
- Output neurons connected to part of the LSM
- LSM recurrent connections, e.g. small world connections
- Excitatory 80%, Inhibitory 20%
- Learning in LSM: STDP; spike time delay ....???
- Polychronization (Izhikevich): 'opening the box'?

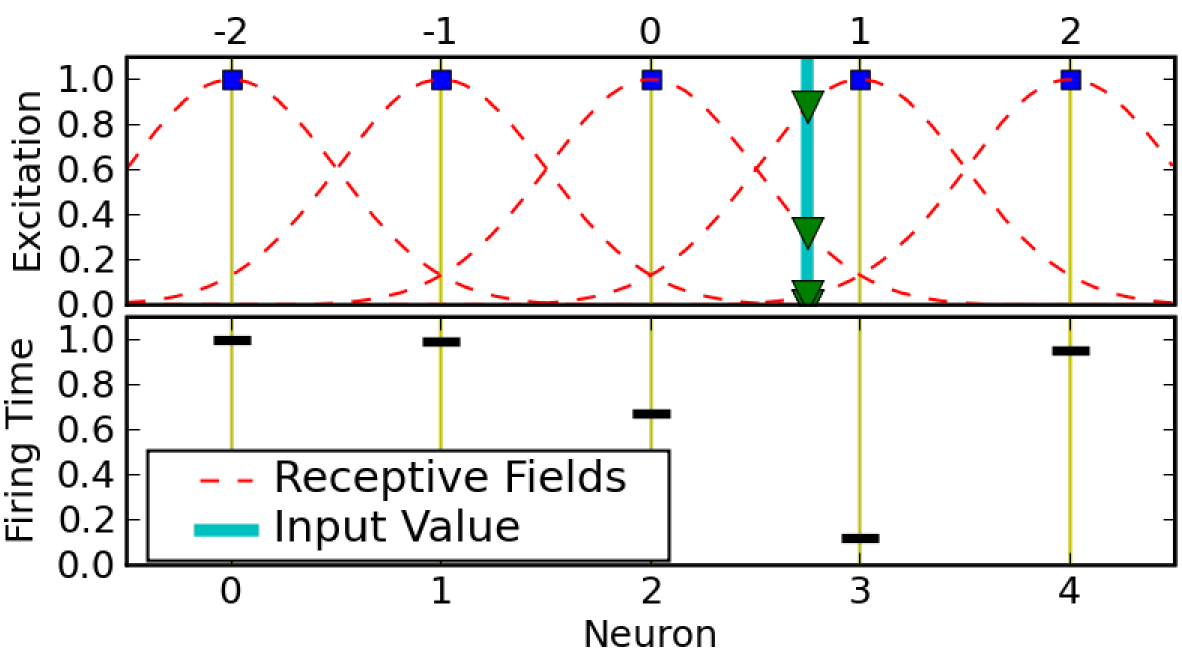


$$p_{a,b} = C \times e^{-D^2_{a,b}/\lambda^2}$$

Encoding input data into spikes

Rank Order Population Encoding

- Distributes a single real input value to multiple neurons and may cause the excitation and firing of several responding neurons
- Implementation based on Gaussian receptive fields introduced by Bothe *et al* . 2002

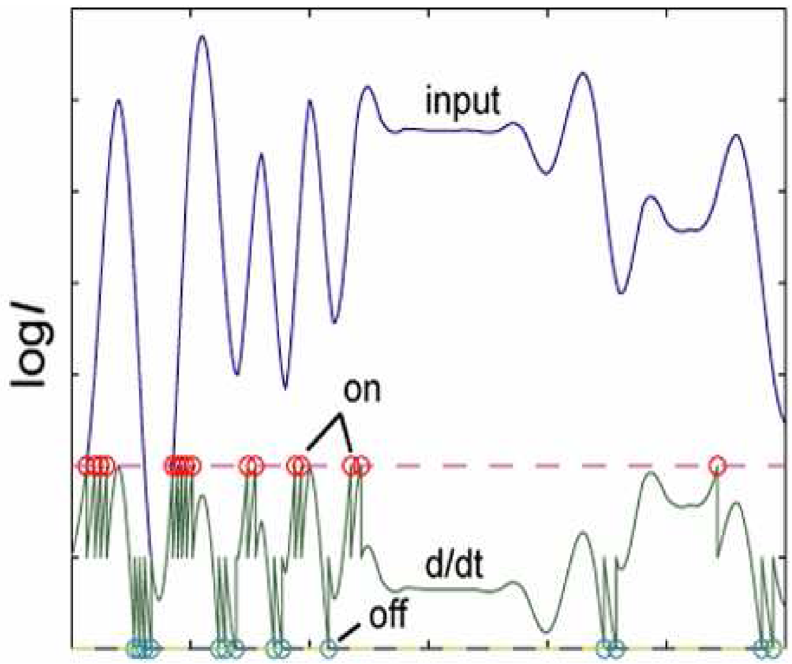


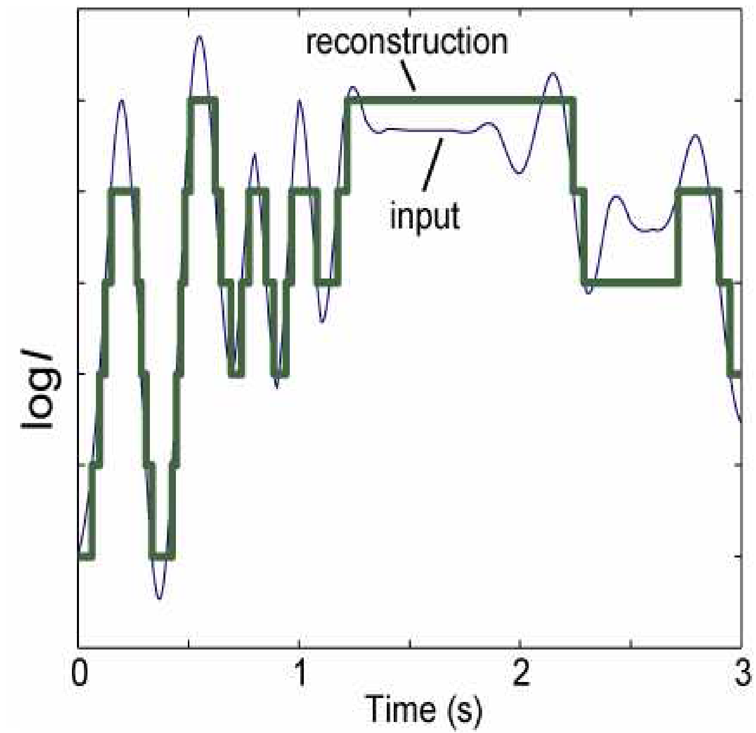
Address Event Representation (AER) Encoding

A spike is generated only if a change in the input data occurs

Silicon Retina (Tobi Delbruck, INI, ETH/UZH, Zurich), DVS128

Silicon Cochlea (Shih-Chii Liu, INI, ETH/UZH, Zurich)





Evolving SNN (eSNN) as a classifier

- eSNN: Creating and merging neurons based on localised information (Kasabov, 2007; Wysoski, Benuskova and Kasabov, 2006-2009)
- Uses the first spike principle (Thorpe et al.) for fast on-line training
- For each input vector

a) Create (evolve) a new output spiking neuron and its connections

b) Propagate the input vector into the network and train the newly created neuron

$$u_i(t) = \begin{cases} 0 & \text{if fired} \\ \sum_{j|f(j)<t} w_{ji} m_i^{\mathrm{order}(j)} & \text{else} \end{cases}$$

$$\Delta w_{ji} = m^{\mathrm{order}(j)}$$

Weights change based on the spike time arrival

c) Calculate the similarity between weight vectors of newly created neuron and existing neurons: IF similarity > Threshold THEN Merge newly created neuron with the most similar neuron

$$W \Leftarrow \frac{W_{new} + NW}{1+N}$$

where N is the number of samples previously used to update the respective neuron.

d) Update the corresponding threshold ϑ:

$$\vartheta \Leftarrow \frac{\vartheta_{new} + N\vartheta}{1+N}$$

- Schliebs, S. and N.Kasabov, Evolving spiking neural networks: A Survey, *Evolving Systems*, Springer, 2013.

nkasabov@aut.ac.nz,

Dynamic Evolving SNN (deSNN)

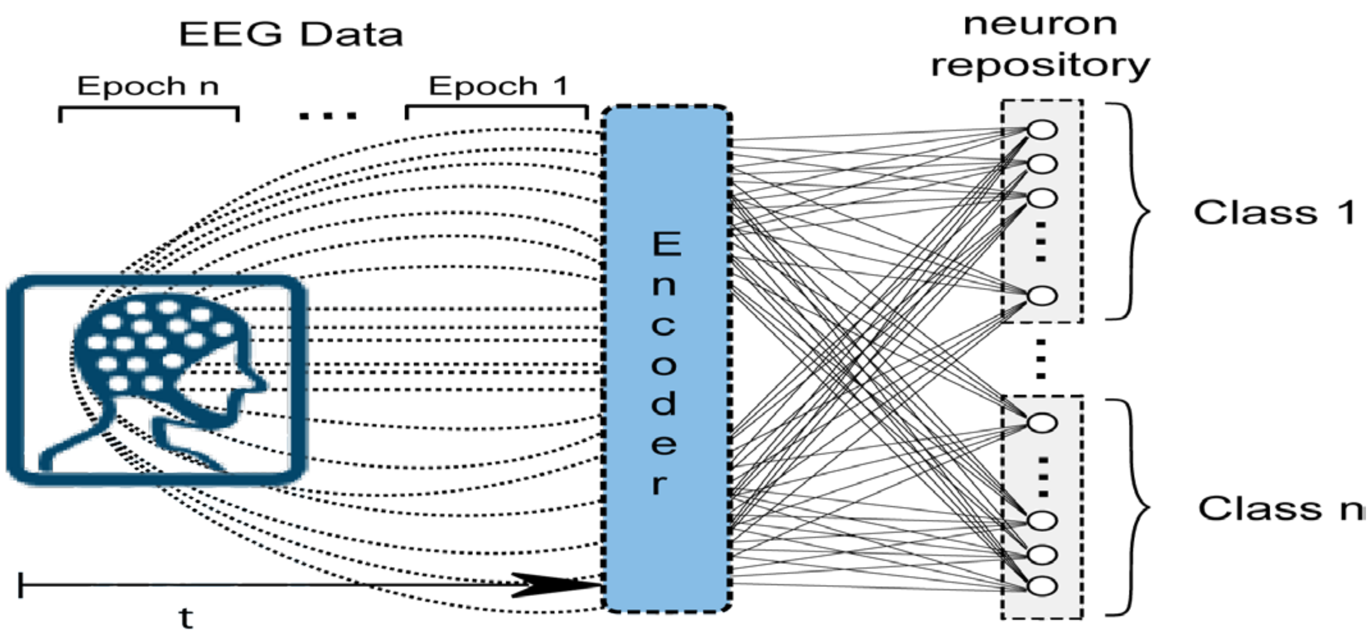
(Kasabov, Dhoble, Nuntalid, Indivery, Neural Networks, 2013)

- Combine: (a) RO learning for weight initialisation based on the first spikes:

$$\Delta w_{ji} = m^{\mathrm{order}(j)}$$

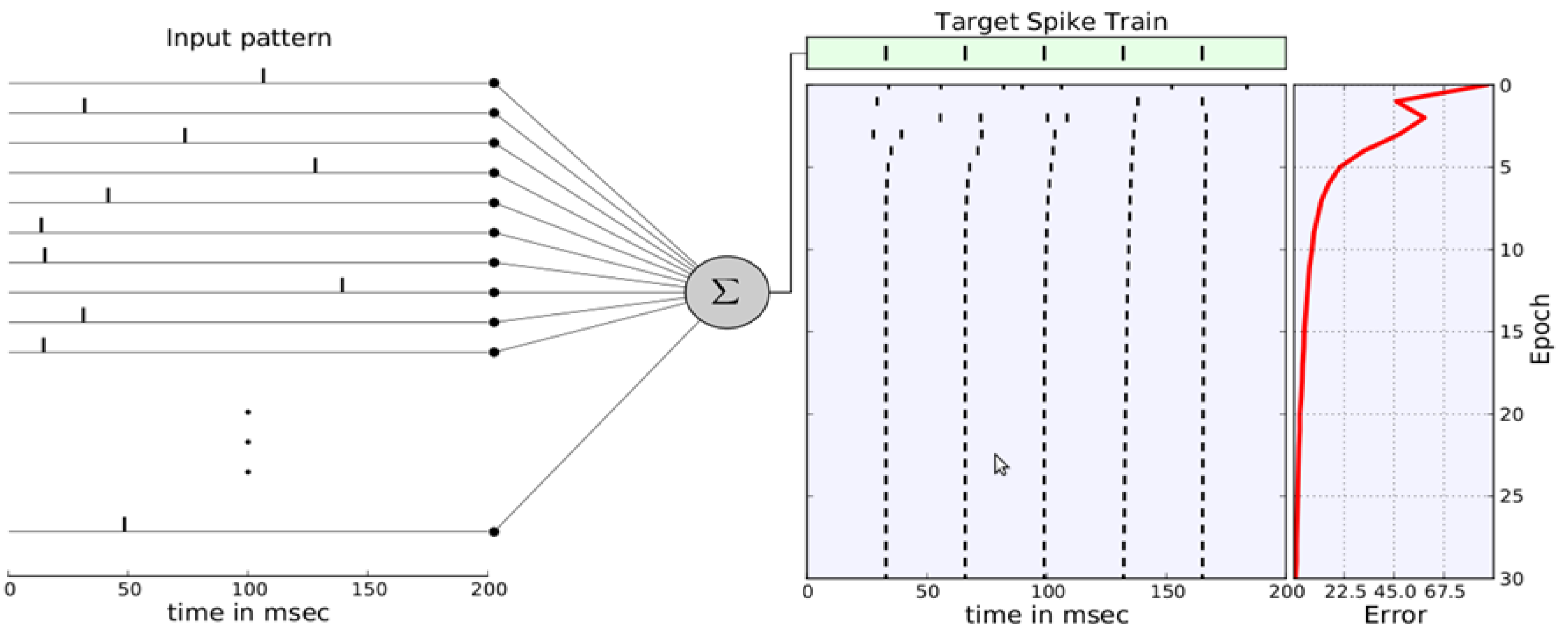
(b) SDSP for learning further input spikes at a synapse.

- A new output neuron is added to a respective output repository for every new input pattern learned. Neurons may merge.
- The figure below shows the deSNN architecture on a case study for EEG STPR.



SPAN: Spike Pattern Association Neuron and the Delta Rule

(A.Mohhemed et al, EANN 2011, ICONIP2011, IJNS, 2012; Neurocomputing, 2012))

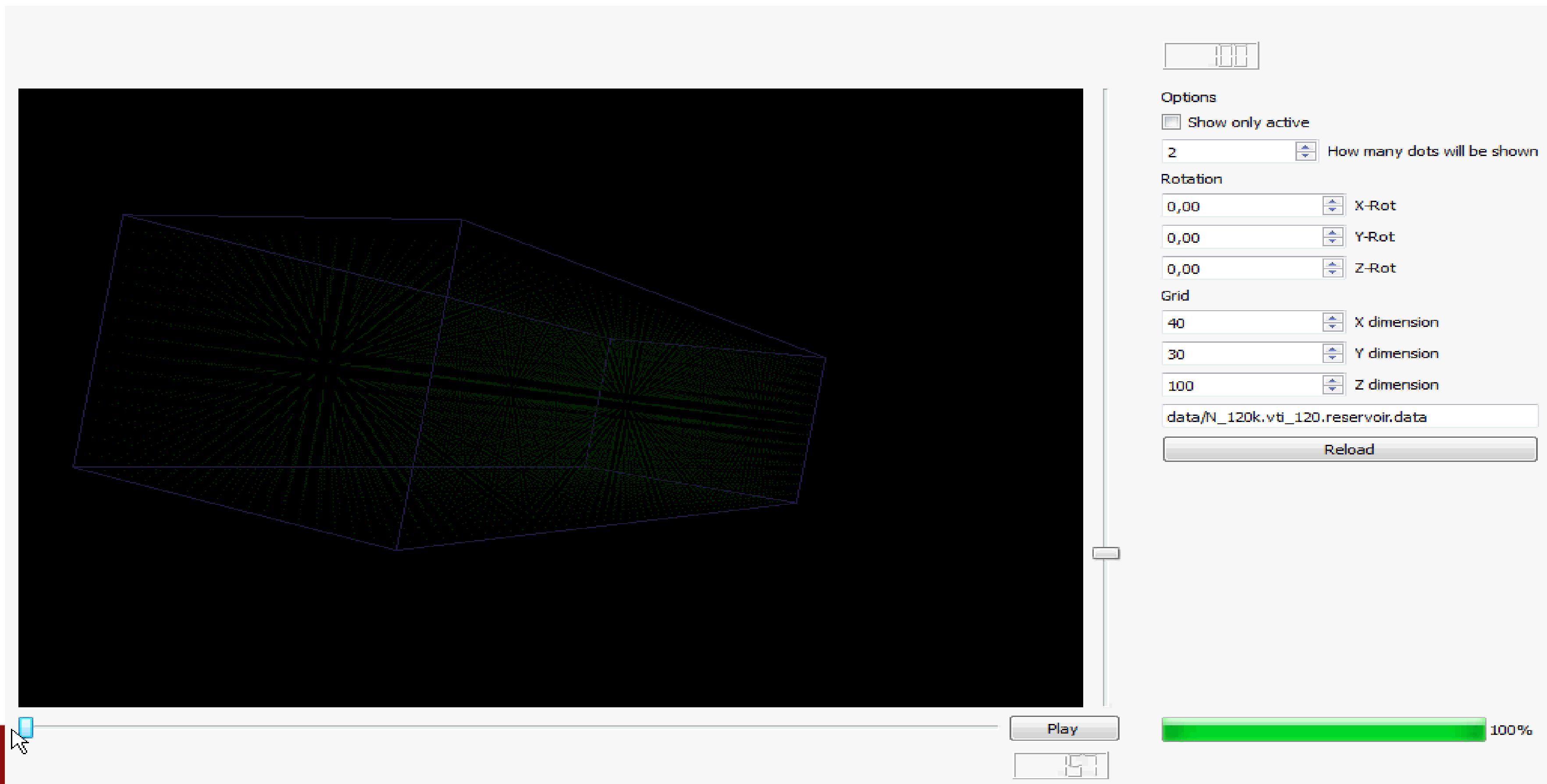


A single output neuron is trained to respond with a temporally precise output spike train to a specific spatio-temporal input.

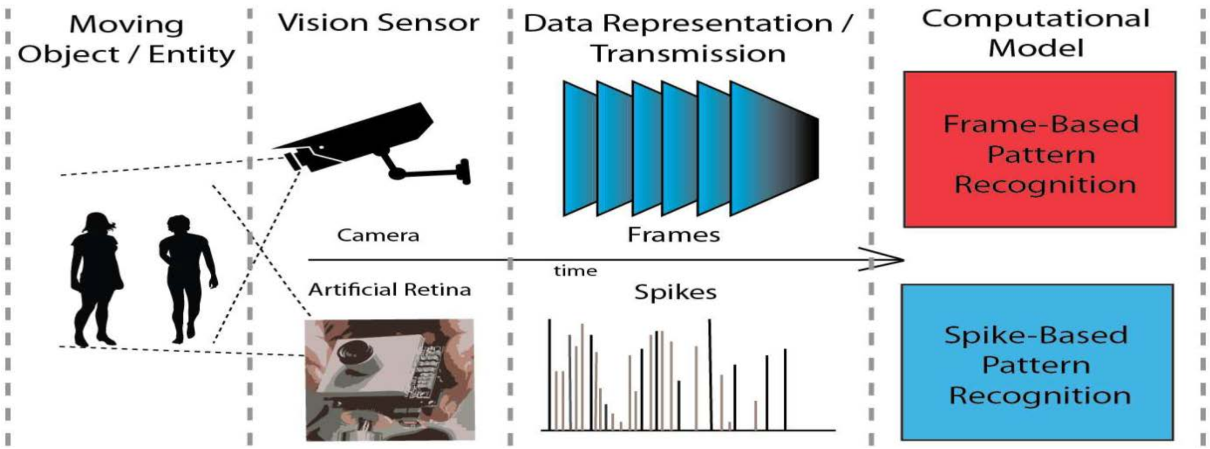
Spike pattern association neuronal models: SpikeProp; ReSuMe; Tempotron; Chronotron.

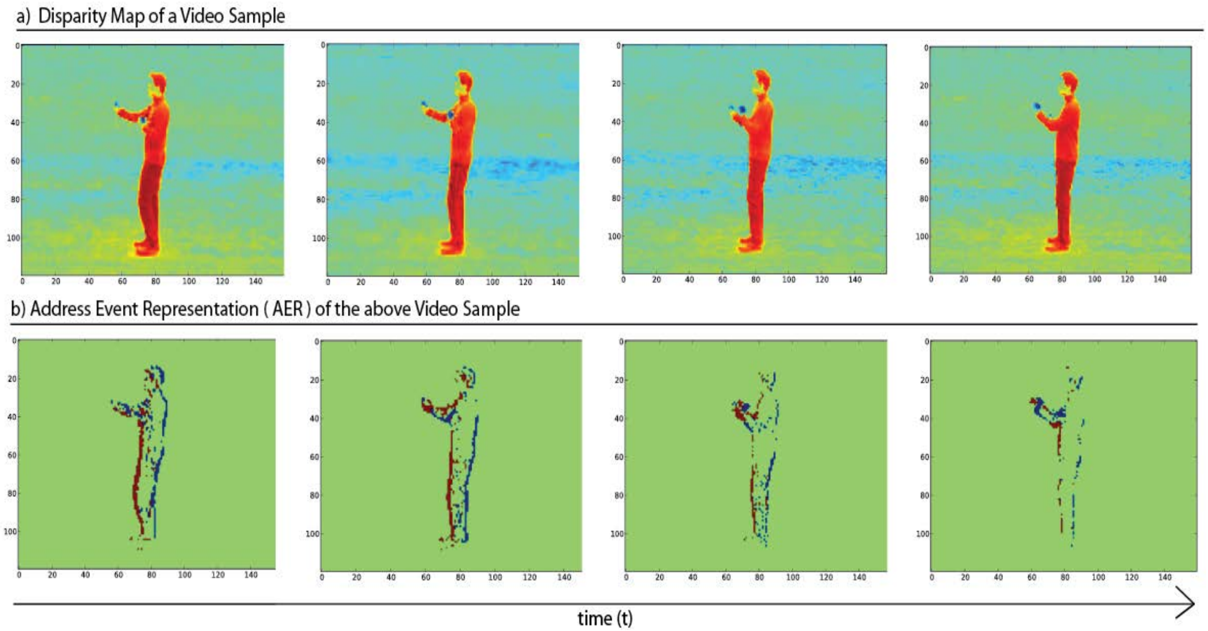
The EvoSpike Simulator

A collection of modules and functions written in Python using functions from Brian library:
- Converting continuous-value input data into spike trains;
- SNN for spatio-temporal pattern recognition (SPAN, deSNN, LSM deSNN, …);
- Knowledge extraction from trained eSNN;
- Presenting results and visualisation of learning processes ;
- Connecting software modules with neuromorphic hardware.



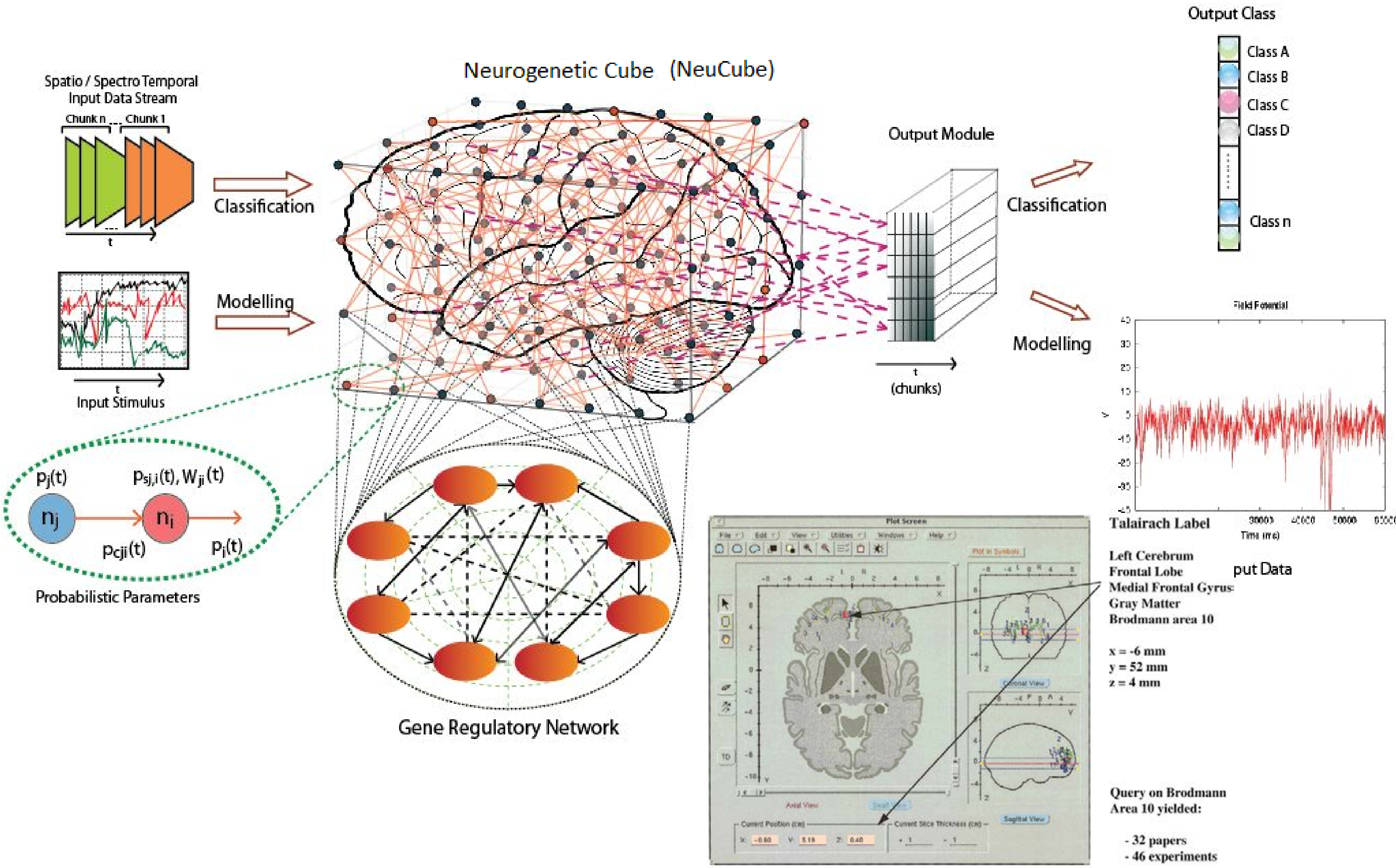
Moving object recognition – frame-based vs AER





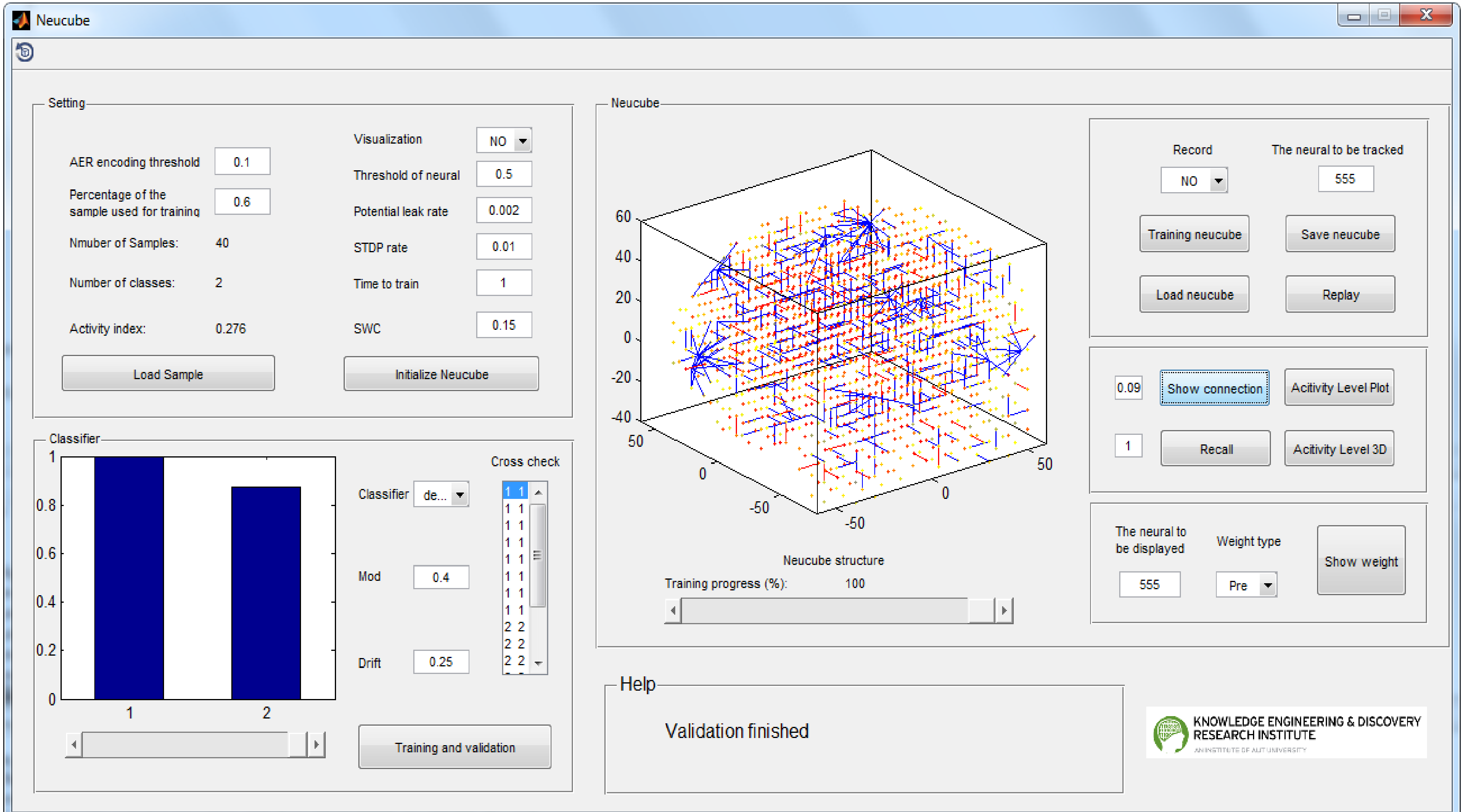
The NeuCube Architecture for Integrated Brain Data Modelling and brain STPR

(Kasabov, Springer LNAI 7477, 2012; Kasabov, NN 2013)

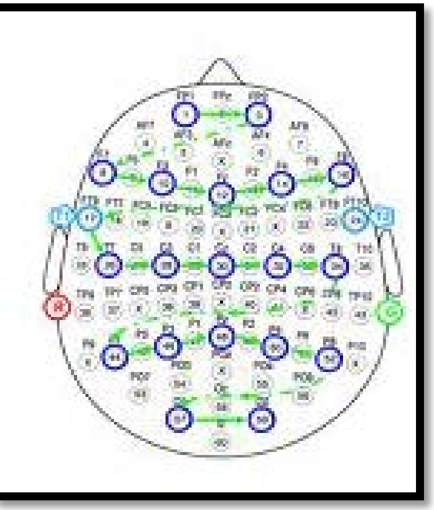


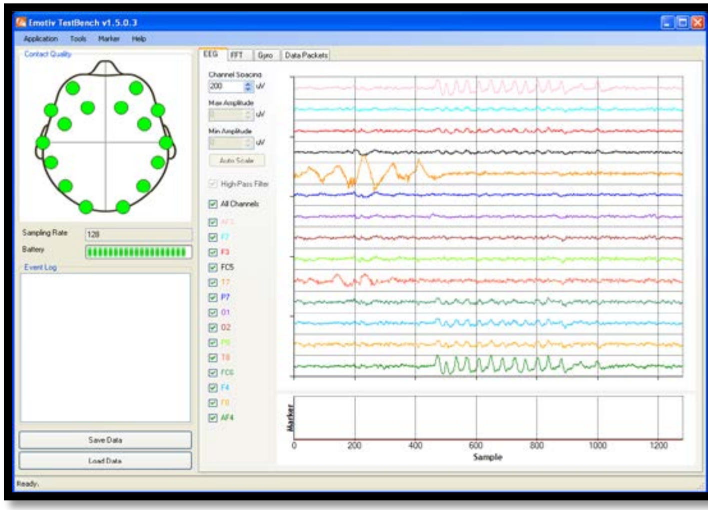
NeuCube Implementation

- Module input data encoding into spikes (e.g. AER)
- Module 3D reservoir (e.g. 1471 neurons)
- Module classifier (e.g. deSNN)

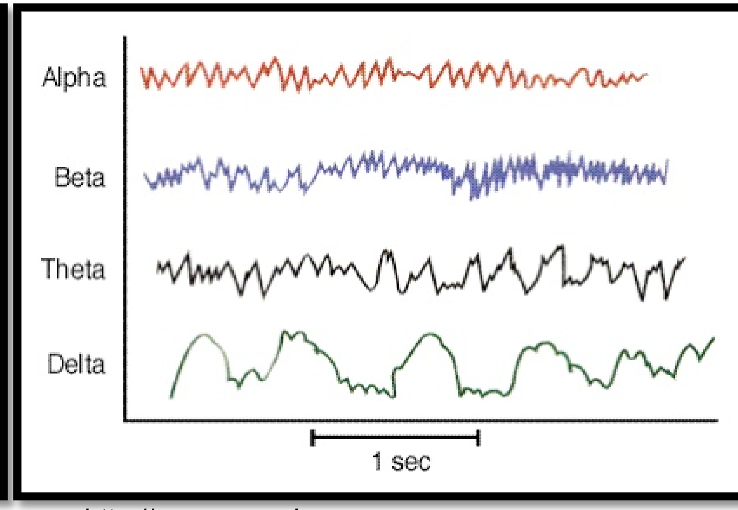


EEG STPR in NeuCube

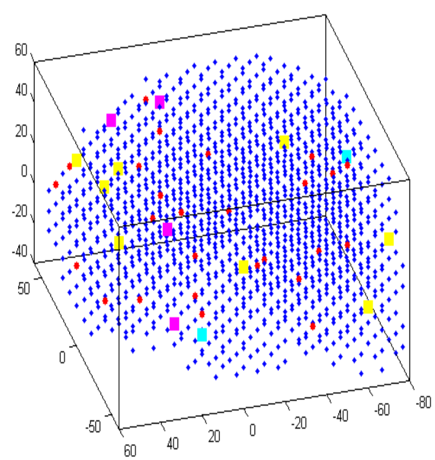


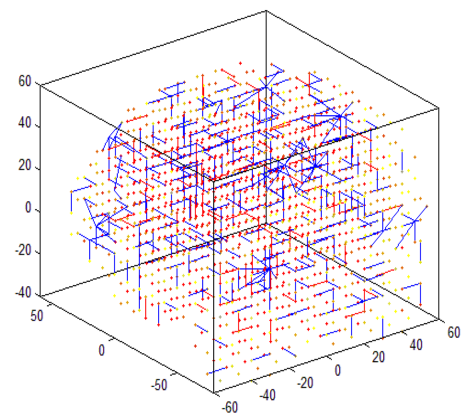


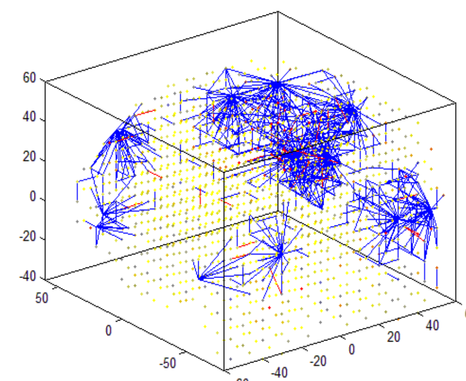
(McFarland, Anderson, MÜller, Schlögl, Krusienski , 2006)



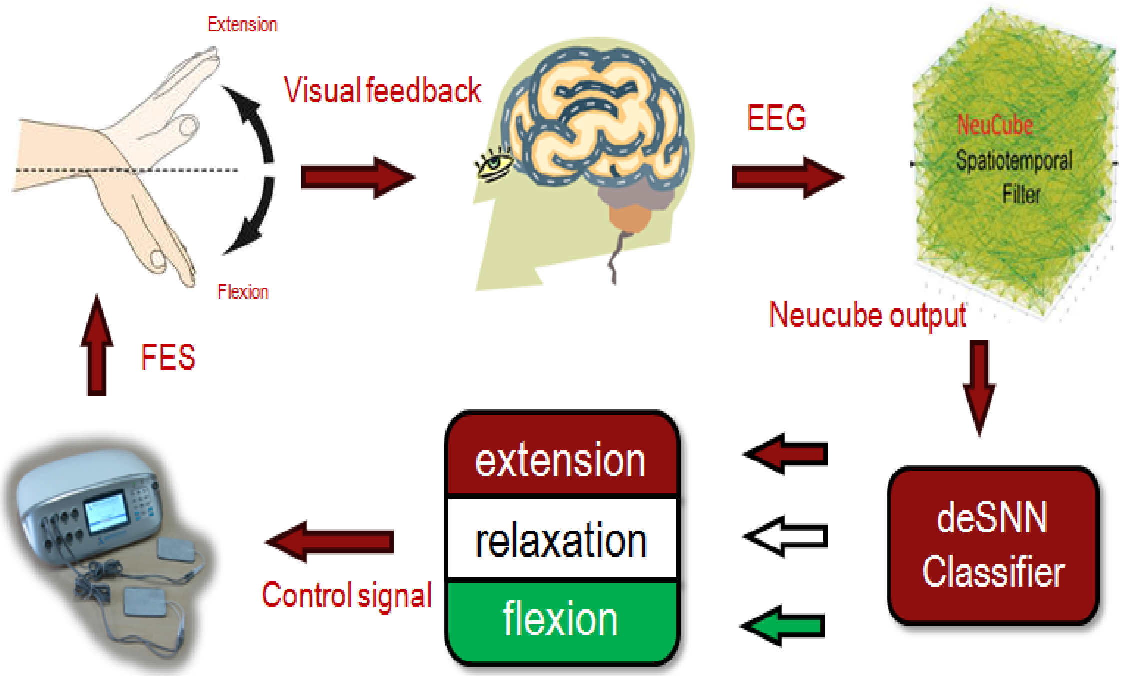
http://www.nuroshop.com

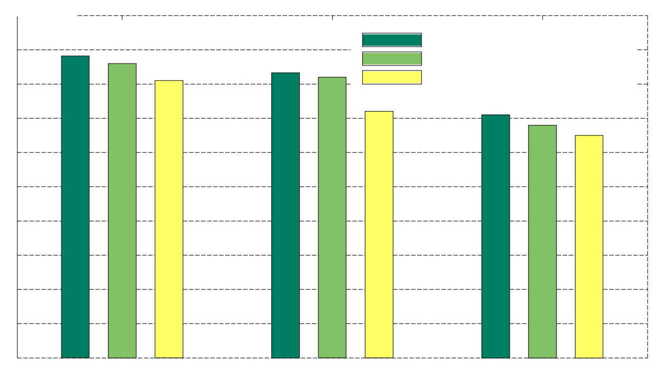






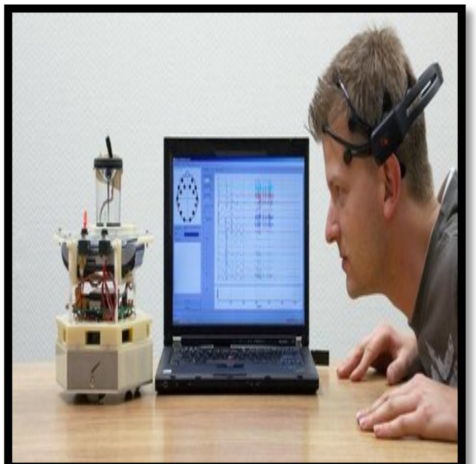
NeuCube for Neurorehabilitation



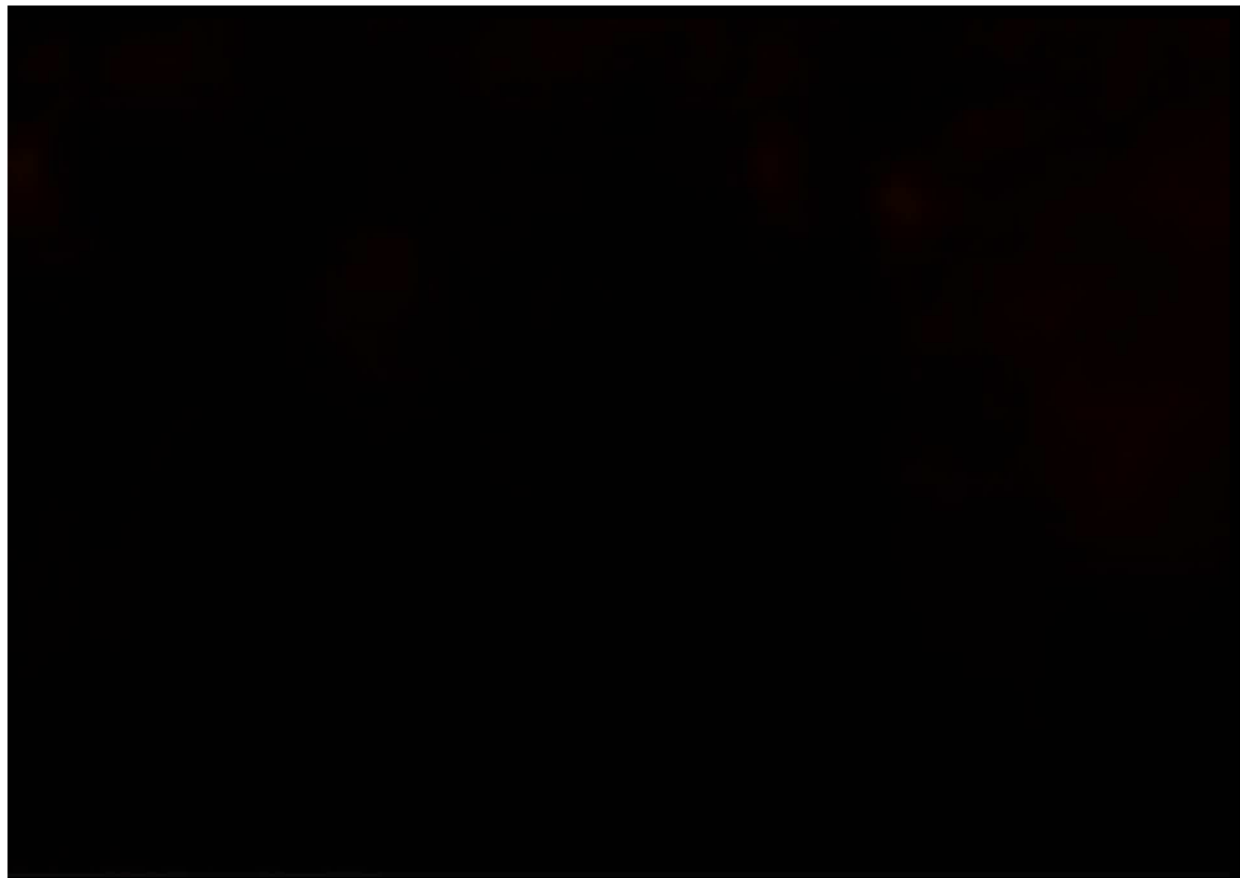


NeuCube for BCI

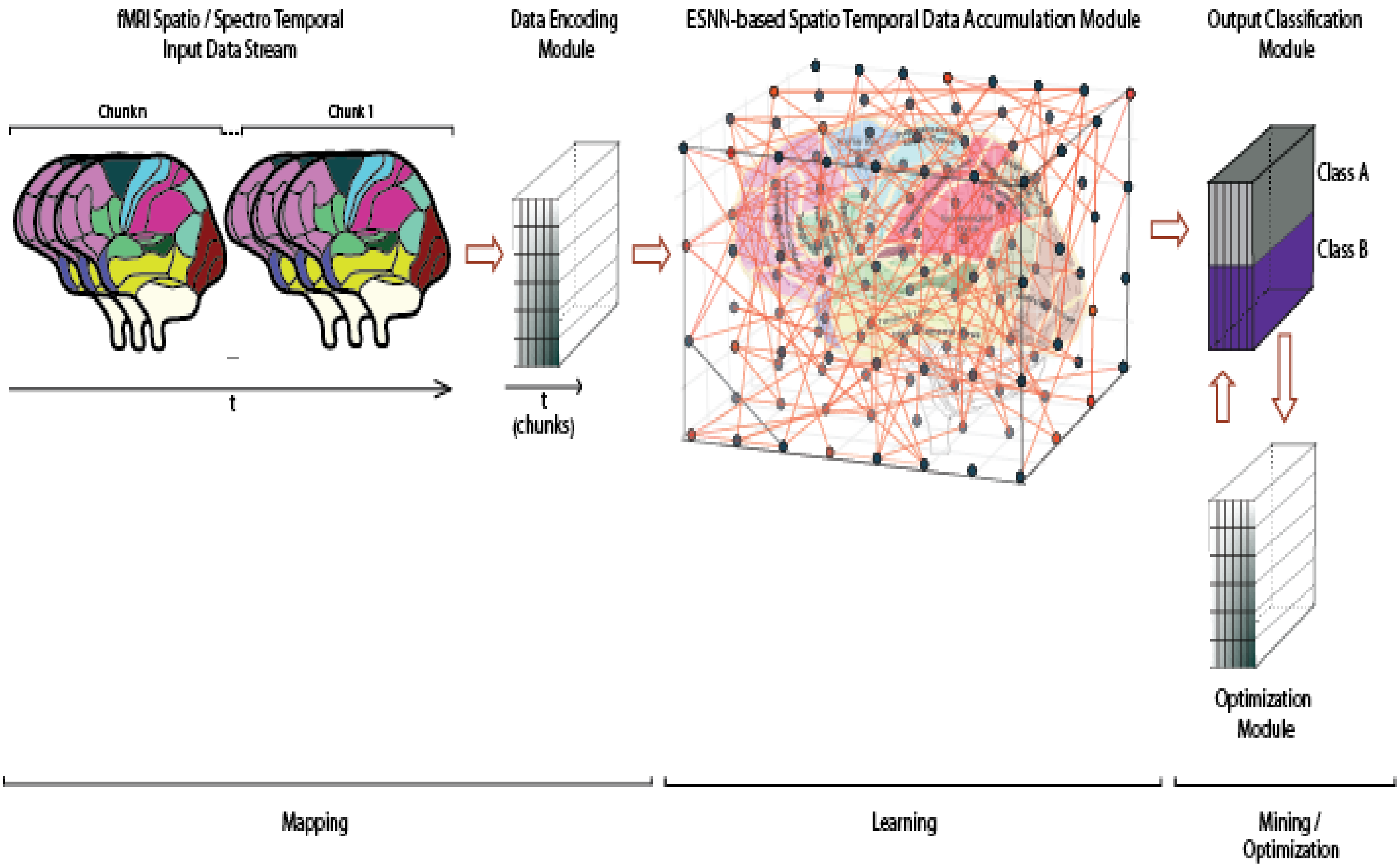
- Brain-Computer Interfaces (BCIs) are interfaces that allow humans to communicate directly with computers or external devices through their brains (e.g. EEG signals)
- Experiments with the WITH robot from KIT, prof. Yamakawa (S.Schliebs)



http://www.nzherald.co.nz

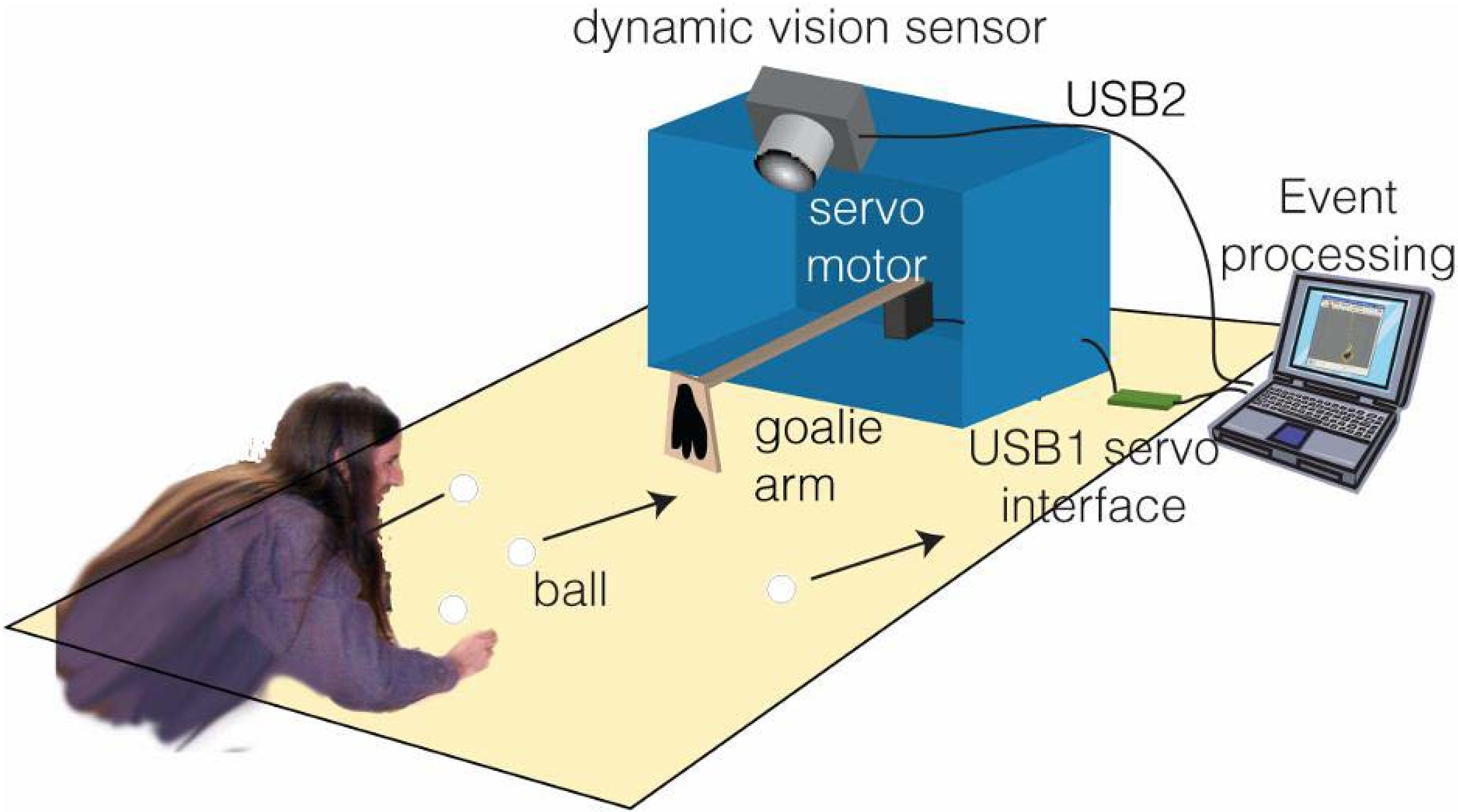


NeuCube for fMRI STBD



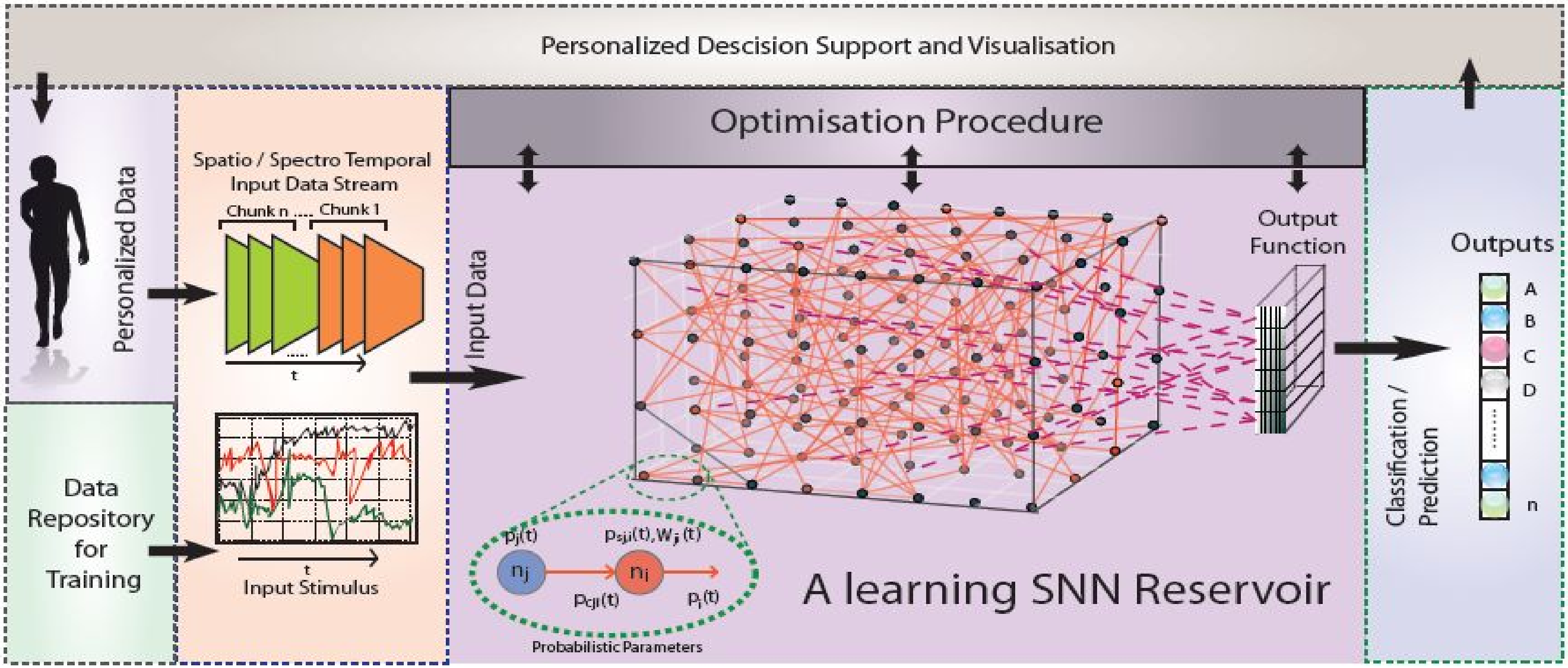
Early detection of a moving object with DVS and SNN

(T.Delbruck, INI, ETH. Zurich)



Personalised Predictive Systems

N. Kasabov and R.Hu, Integrated Optimisation Method for Personalised Modelling and Case Study Applications for Medical Decision Support, *J. Functional Informatics and Personalised Medicine*, 2011

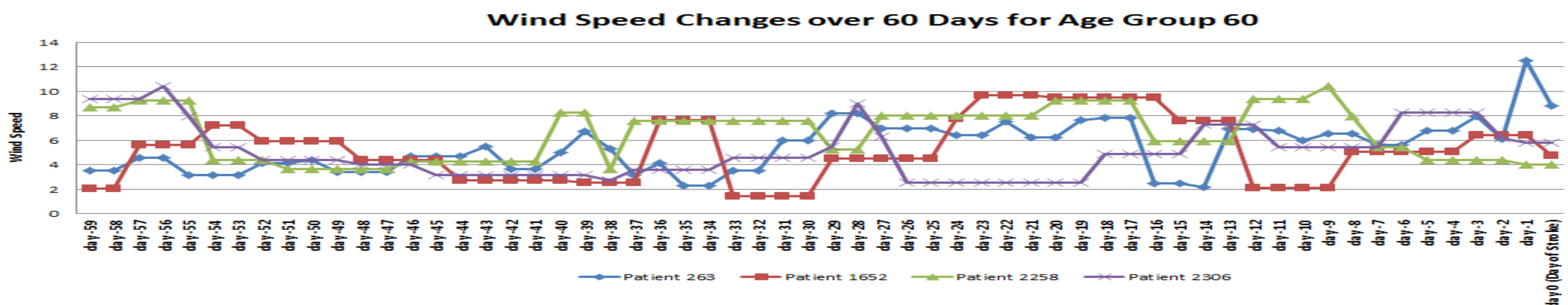


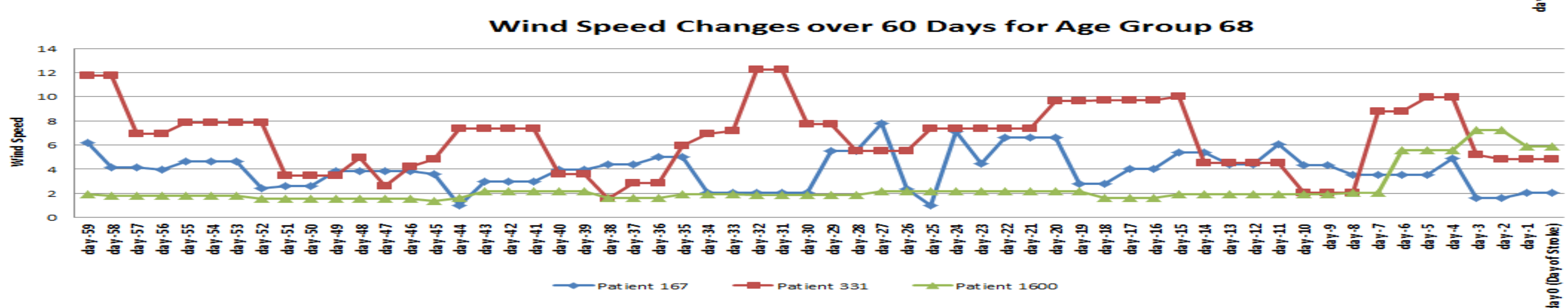
Personalised stroke occurrence prediction

The dataset consists of 11,453 samples (all with first occurrence of stroke).

Each sample is described by 9 features/variables:

- 4 static patient clinical features (age, gender, history of hypertension, smoking status, geographical region)
- 5 temporal weather variables: Temperature; Humidity; Atmospheric Pressure (kPA); Wind Speed (Knots) and Wind Chill (Degrees Celsius).
- All of these weather parameters were measured over a 60-day period preceding data of stroke (including the day of stroke occurrence as the last day).

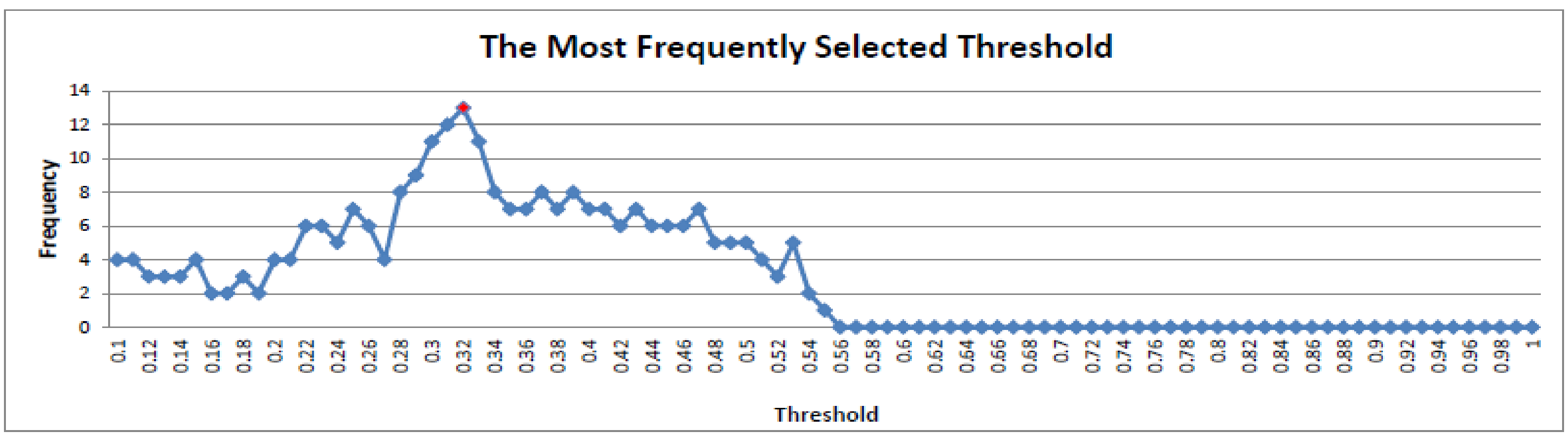




Results for personalised early stroke prediction

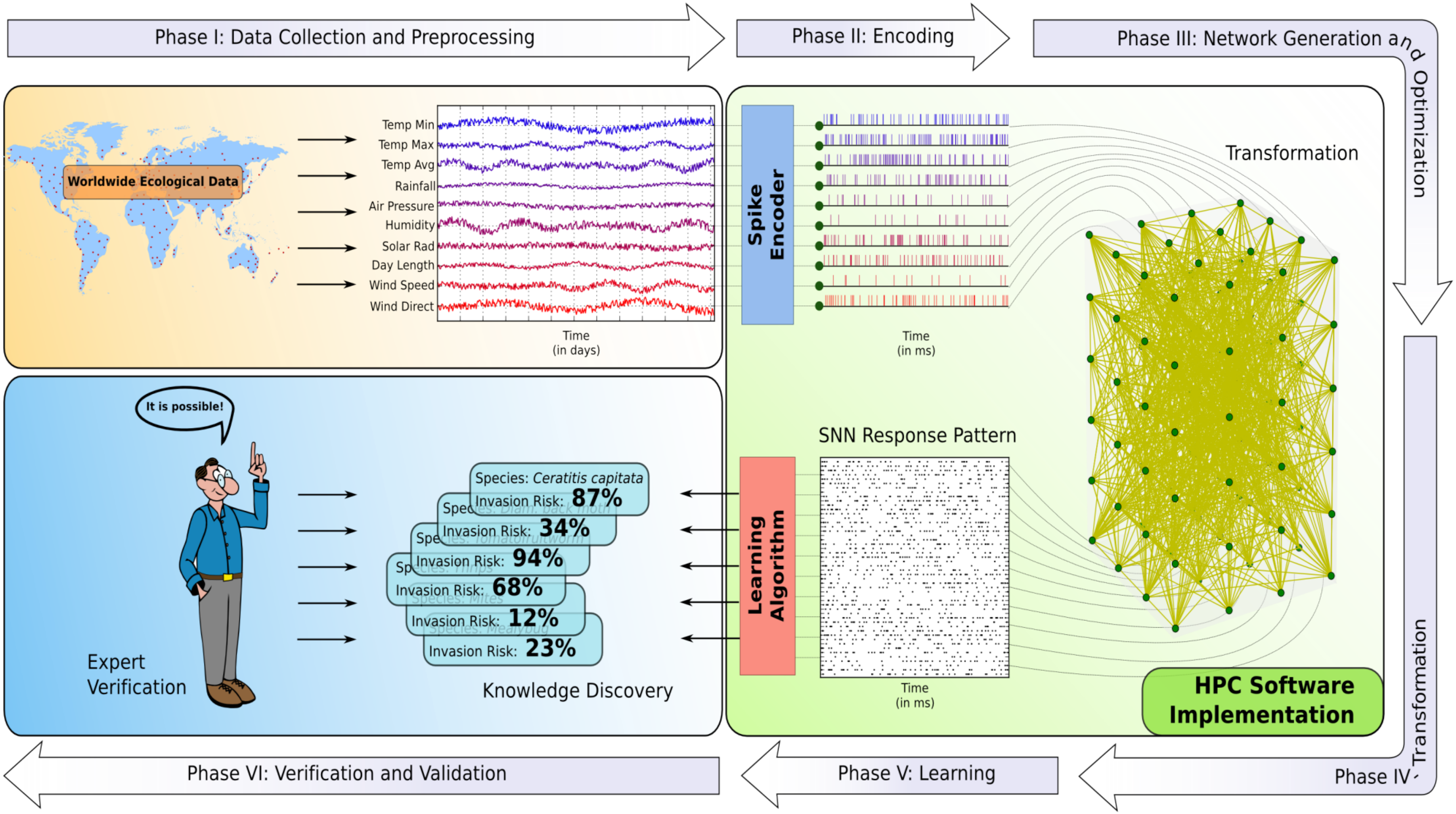
- SNN achieve better accuracy
- SNN predict stroke much earlier than other methods

Method	Overall accuracy (%)	TP – stroke prediction (%)	TN – no stroke (%)
Multiple Linear regression (MLR)	67.50	65	70
SVM	72.5	65	80
MLP	87.5	85	90
PMeSNNr	94	88	100

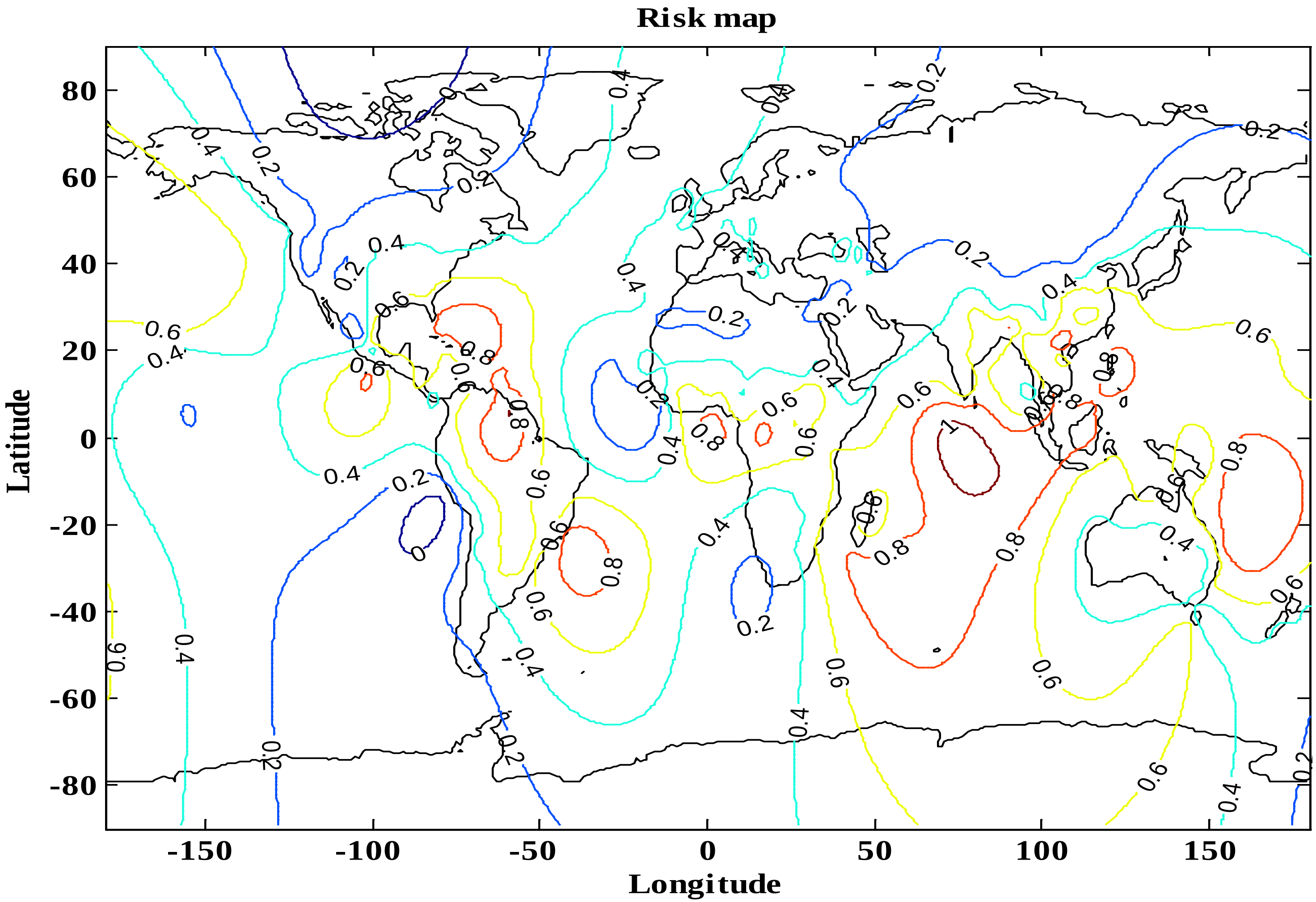


Early estimation of risk of establishment of invasive species on a certain *location* at a certain *time*

(S.Schliebs, Defoin-Platel, N.Kasabov, S.Worner et al, Neural Networks, No.22, 2009)



Example: Through modelling a world map was created for the estimation of the probability of *p.citri* insect establishment



Other applications of SNN for STPR

- Multisensor systems
- Wearable technologies in sport (e.g. wearable coach), in medicine (implants)
- On-line stream of data processing
- Autonomous mobile robot control
- Predictive systems across domain areas
-

5. Advantages and limitations of SNN

Advantages:

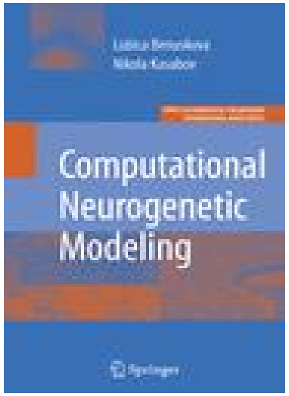
- Universal computational mechanism
- Extendable models, with more biologically related knowledge as it become available (e.g. genes, quantum information)
- Can learn spatio-temporal relationships from spatio-temporal data
- Fast and less computationally demanding (spikes are easy to compute)
- Adaptive to new data and non stationary inputs
- Robust to changing dynamics of the data

Problems and limitations

- Sensitive to parameter values
- Large number of parameters that need to be optimised
- Unknown properties in terms of dealing with different types of spatio-temporal data
- No rigid information theory yet!

6. Future Directions

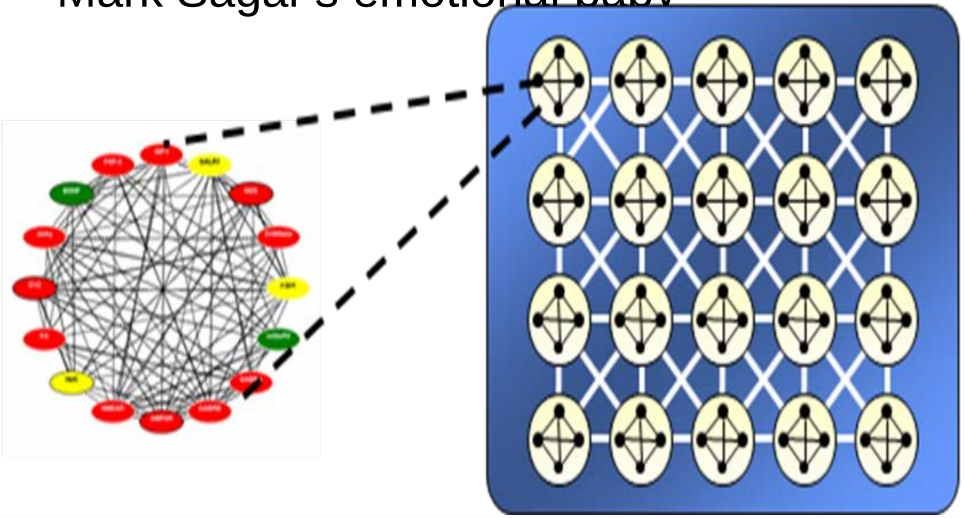
Fourth generation of ANN: Computational Neuro-Genetic Modelling (CNGM)



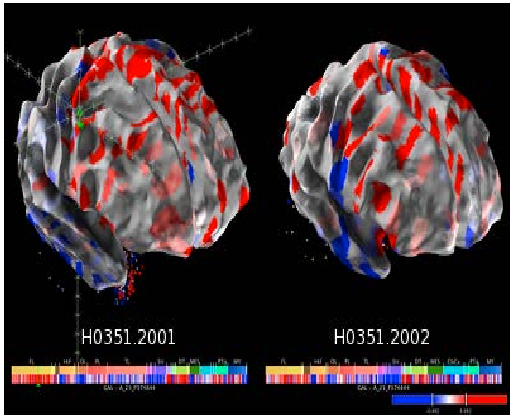
- Benuskova and Kasabov (2007)

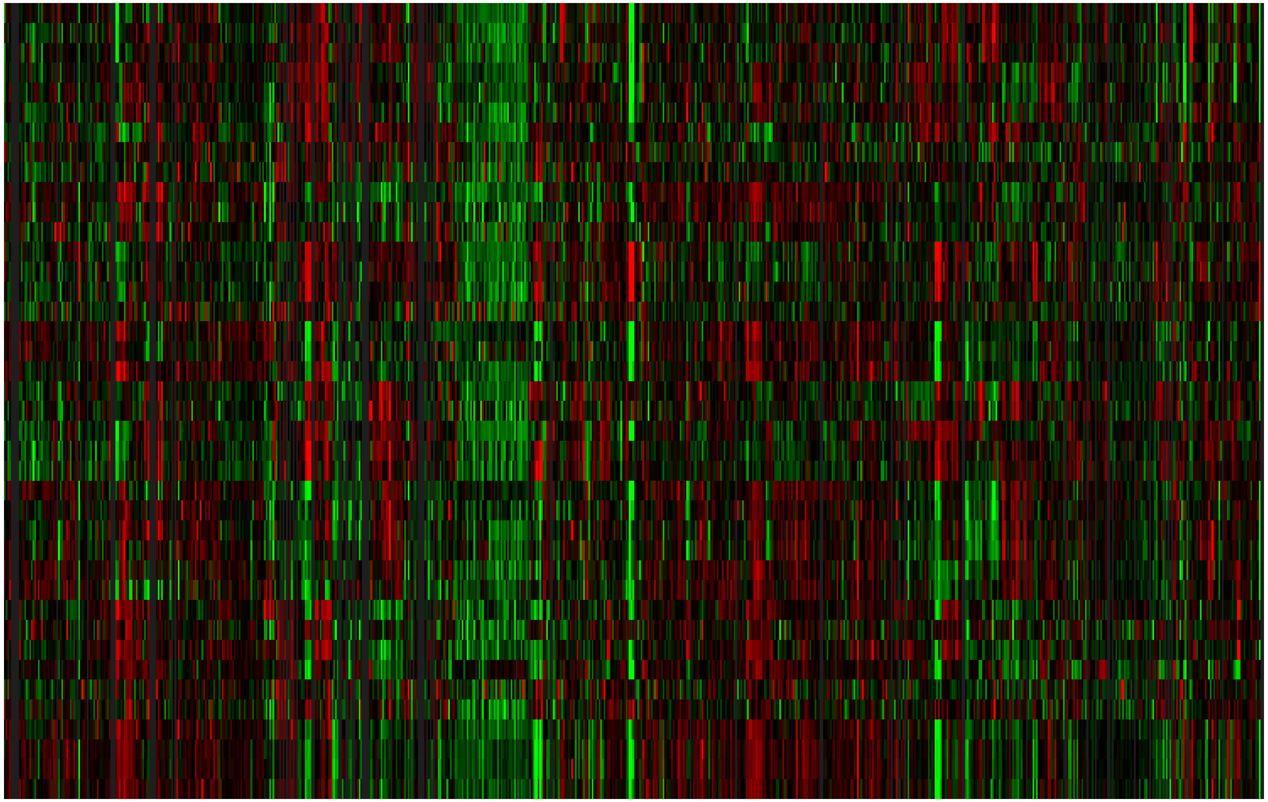
SNN that incorporate a gene regulatory network (GRN) as a dynamic paramter systems to capture dynamic interaction of genes (parameters) related to neuronal activities of the SNN.

- Functions of neurons and neural networks are influenced by internal networks of interacting genes and proteins forming an abstract GRN model.
- The GRN and the SNN function at different time scales.
- Mark Sagar's emotional baby



Neurogenetic STBD: The Allen Brain Institute Map (http://www.brain-map.org)

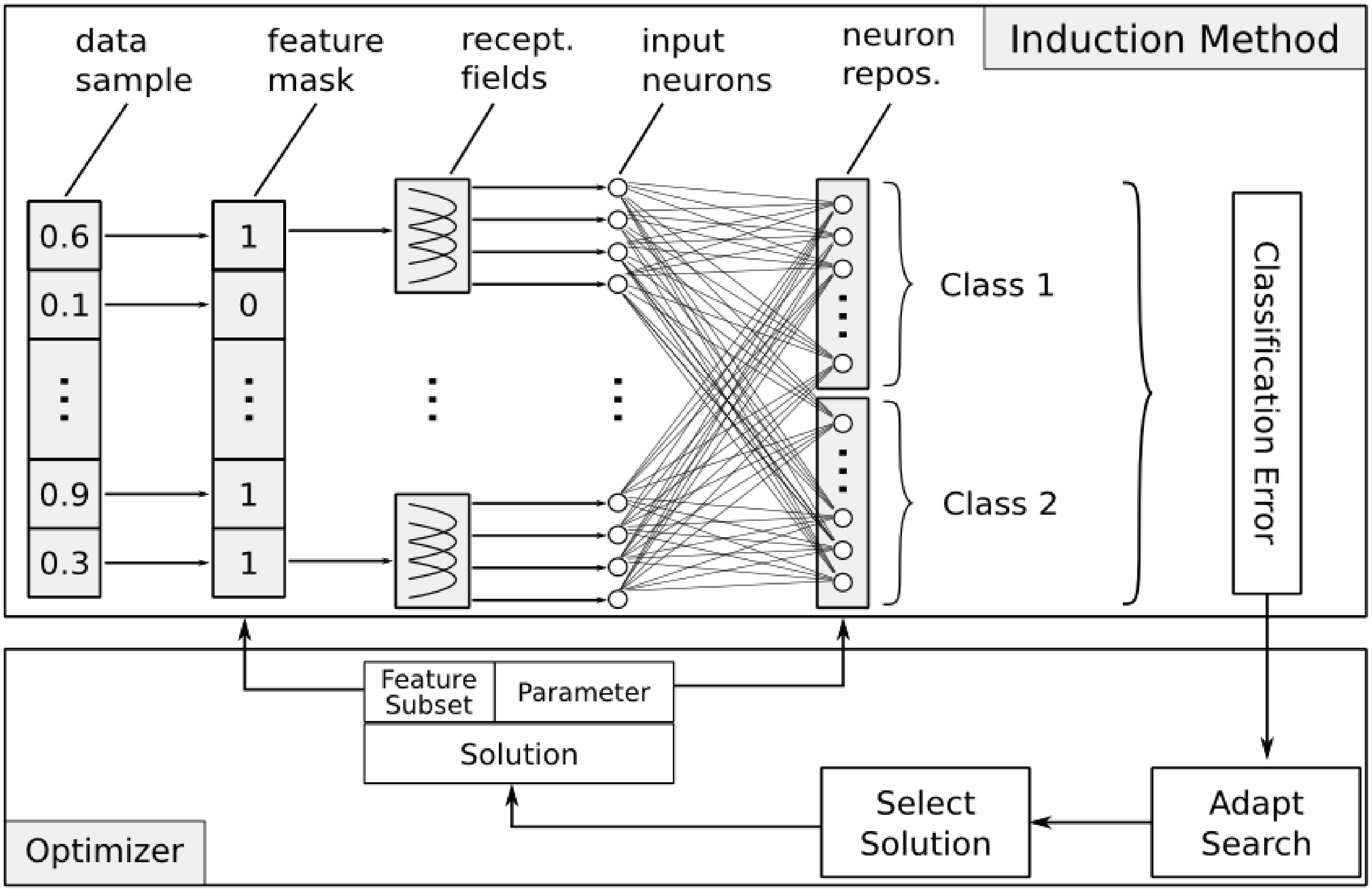




From the Brain Explorer: The Expression level of the genes (on the y-axis): ABAT A_23_P152505, ABAT A_24_P330684, ABAT CUST_52_PI416408490, ALDH5A1 A_24_P115007, ALDH5A1 A_24_P923353, ALDH5A1 A_24_P3761, AR A_23_P113111, AR CUST_16755_PI416261804, AR CUST_85_PI416408490, ARC A_23_P365738, ARC CUST_11672_PI416261804, ARC CUST_86_PI416408490, ARHGEF10 A_23_P216282, ARHGEF10 A_24_P283535, ARHGEF10 CUST_) at different slices of the brain (on the x-axis) (from www.brain-map.org) (http://www.alleninstitute.org)

Quantum-inspired EC for the optimisation of eSNN

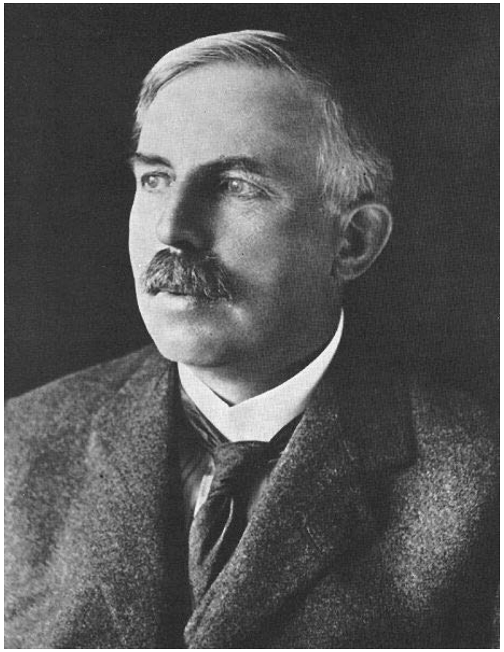
(Kasabov, 2007-2008; S.Schliebs, M.Defoin-Platel and N.Kasabov, 2008)



Fifth generation of ANN: Quantum Neurocomputation

- Quantum principles: superposition; entanglement, interference, parallelism
  - Quantum bits (qu-bits)

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \qquad |\alpha|^2 + |\beta|^2 = 1$$



- - Quantum vectors (qu-vectors)

$$\left[\begin{array}{c|c|c|c} \alpha_1 & \alpha_2 & \dots & \alpha_m \\ \beta_1 & \beta_2 & \dots & \beta_m \end{array}\right]$$

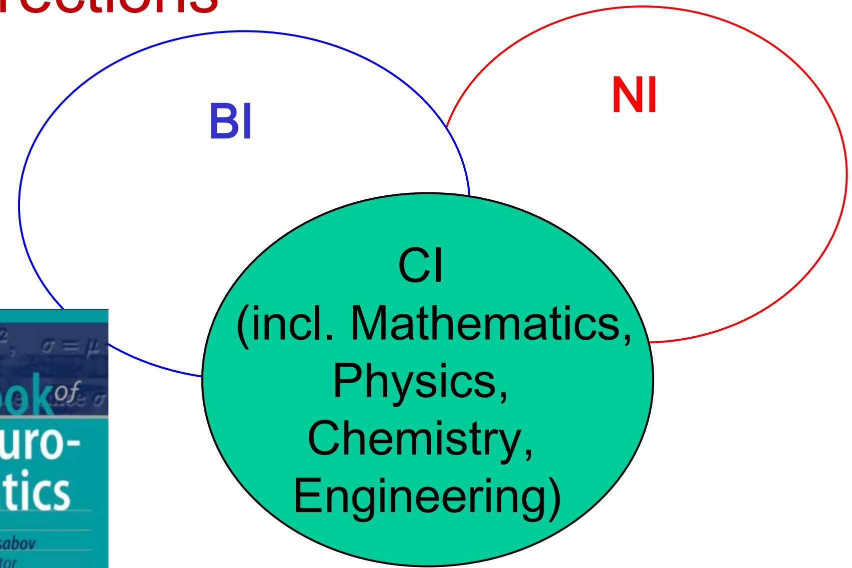
- Quantum gates

$$\begin{bmatrix} \alpha_i^j(t+1) \\ \beta_i^j(t+1) \end{bmatrix} = \begin{bmatrix} \cos(\Delta\theta) & -\sin(\Delta\theta) \\ \sin(\Delta\theta) & \cos(\Delta\theta) \end{bmatrix} \begin{bmatrix} \alpha_i^j(t) \\ \beta_i^j(t) \end{bmatrix}$$

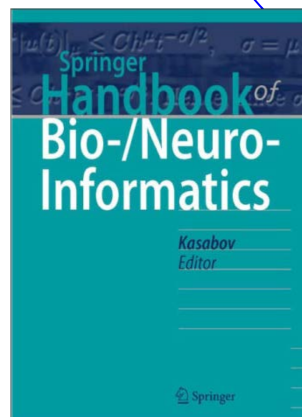
- Applications:
  - Specific algorithms with polynomial time complexity for NP-complete problems (e.g. factorising large numbers, Shor, 1997; cryptography)
  - Search algorithms (Grover, 1996), O($N^{1/2}$) vs O(N) complexity)
  - Quantum associative memories
  - Quantum inspired evolutionary algorithms and neural networks

…Future Directions

- Further interdisciplinary research in the three areas of CI, BI and NI

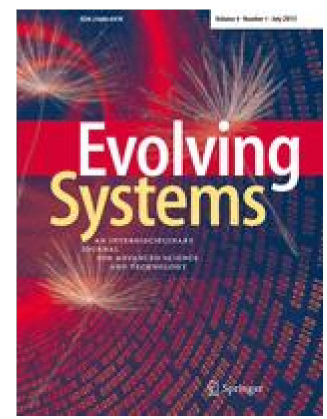


- The Springer Handbook of Bio-Neuroinformatics, 2013 (N.Kasabov, ed)



- The Springer Series in Bio-Neuroinformatics (N.Kasabov, ed)

- Springer journal *Evolving Systems*
  (ed. Angelov, Filev, Kasabov)



KEDRI: The Knowledge Engineering and Discovery Research Institute at AUT (www.kedri.aut.ac.nz)

References

[1] N.Kasabov (2007) Evolving Connectionist Systems: The Knowledge Engineering Approach, Springer, London (www.springer.de) (first edition published in 2002)

[2] S.Wysoski, L.Benuskova, N.Kasabov, Evolving Spiking Neural Networks for Audio-Visual Information Processing, Neural Networks, vol 23, issue 7, pp 819-835, September 2010.

[3] N.Kasabov, To spike or not to spike: A probabilistic spiking neural model, Neural Networks, 23, 1, 2010, 16-19.

[4] Mohemmed,A., Schliebs,S., Kasabov,N.,SPAN: Spike Pattern Association Neuron for Learning Spatio-Temporal Sequences, Int. J. Neural Systems, 2012.

[5] Kasabov, N., Dhoble, K., Nuntalid, N., G. Indiveri, Dynamic Evolving Spiking Neural Networks for On-line Spatio- and Spectro-Temporal Pattern Recognition, Neural Networks, 41, 2013, 188-201.

[6] Benuskova, L and N.Kasabov (2007) Computaional Neurogenetic Modelling, Springer.

[7] Defoin-Platel, M., S.Schliebs, N.Kasabov, Quantum-inspired Evolutionary Algorithm: A multi-model EDA, IEEE Trans. Evolutionary Computation, vol.13, No.6, Dec.2009, 1218-1232

[8] Nuzly, H., N.Kasabov, S.Shamsuddin (2010) Probabilistic Evolving Spiking Neural Network Optimization Using Dynamic Quantum Inspired Particle Swarm Optimization, Proc. ICONIP 2010, Part I, LNCS, vol.6443.

[9] S.Soltic, N.Kasabov, Knowledge extraction from evolving spiking neural networks with a rank order population coding, Int.J.Neural Systems, Vol. 20, No. 6 (2010) 437-445, World Scientific Publ.

[10] N.Kasabov (ed) (2013) The Springer Handbook of Bio- and Neuroinformatics, Springer, September, 2013..

[11] Schliebs, Michael Defoin Platel, Susan Worner and Nikola Kasabov, Integrated Feature and Parameter Optimization for Evolving Spiking Neural Networks: Exploring Heterogeneous Probabilistic Models, Neural Networks, 22, 623-632, 2009.

[12] S. Pang, Tao Ban, Youki Kadobayashi and Nikola K. Kasabov, LDA Merging and Splitting with Applications to Multi-agent Cooperative Learning and System Alteration, IEEE Transactions On Systems, Man And Cybernetics, -Part B. 42(2): 552-564, 2012.

[13] Kasabov, N., NeuroCube EvoSpike Architecture for Spatio-Temporal Modelling and Pattern Recognition of Brain Signals, in: Mana, Schwenker and Trentin (Eds) ANNPR, *Springer LNAI* 7477, 2012, 225-243

[14] Furber, S., To Build a Brain, IEEE Spectrum, vol.49, Number 8, 39-41, 2012.