11-2008

A neural network model for a hierarchical spatio-temporal memory

Kiruthika RAMANATHAN
Singapore Management University, kiruthikar@smu.edu.sg

Luping SHI
Agency for Science, Technology and Research

Jianming LI
Agency for Science, Technology and Research

Kian Guan LIM
National University of Singapore

Zhi Ping ANG
National University of Singapore

See next page for additional authors

Citation

RAMANATHAN, Kiruthika; SHI, Luping; LI, Jianming; LIM, Kian Guan; ANG, Zhi Ping; and TOW, Chong Chong. A neural network model for a hierarchical spatio-temporal memory. (2008). *Advances in Neuro-Information Processing: 15th International Conference, ICONIP 2008, Auckland, New Zealand, November 25-28: Proceedings*. 5506, 428-435.

Available at: https://ink.library.smu.edu.sg/sis_research/7392

Author
Kiruthika RAMANATHAN, Luping SHI, Jianming LI, Kian Guan LIM, Zhi Ping ANG, and Chong Chong TOW

A Neural Network Model for a Hierarchical Spatio-temporal Memory

Kiruthika Ramanathan[1,*], Luping Shi[1], Jianming Li[1], Kian Guan Lim[1], Ming Hui Li[2], Zhi Ping Ang[2], and Tow Chong Chong[1,2]

[1] Data Storage Institute, (A*STAR) Agency for Science, Technology and Research, DSI Building, 5 Engineering Drive 1, Singapore 117608
[2] Dept of Electrical and Computer Engineering, National University of Singapore
kiruthika_r@dsi.a-star.edu.sg

Abstract. The architecture of the human cortex is uniform and hierarchical in nature. In this paper, we build upon works on hierarchical classification systems that model the cortex to develop a neural network representation for a hierarchical spatio-temporal memory (HST-M) system. The system implements spatial and temporal processing using neural network architectures. We have tested the algorithms developed against both the MLP and the Hierarchical Temporal Memory algorithms. Our results show definite improvement over MLP and are comparable to the performance of HTM.

1 Introduction

The HST-M is developed based on two principles of cognitive modeling. The first is the neural aspect, where, development of recognition is intertwined with that of the brain, both at the cellular and at the modular levels, and where learning is an incremental, connectionist process. The second is the computational modeling, which forces theories of brain structure to be explicit, resulting in a more detailed specification that that which is available in works of Psychology and Neuroscience.

Recent books such as Rethinking Innateness [1] and The Algebraic Mind [2] argue that representation of information in the brain can be achieved by implementing a set of connectionist networks. In the field of connectionism, much work has been done that attempt to model cognition based on neural networks. Of these, several models such as Neocognitron [4, 5], HMAX [6, 7] and HTMs [8] use the principles of hierarchy and the uniformity of the neocortex to design algorithms to improve pattern recognition. While Neocognitron and HMAX are spatial algorithms, HTM proposes a Bayesian model that makes use of the additional temporal information in the data to solve the pattern recognition problem, thereby leading to better results.

In this paper, we propose a connectionist model to perform pattern recognition based on the spatial and temporal properties of the data. The model proposed is hierarchical, uniform and learns in an unsupervised manner. While the principle off the model is similar to that of the HTM, the implementation is based on neural networks, as opposed to a Bayesian model that is proposed by Numenta [8]. The model has been tested on three datasets and the results compared both with traditional MLPs and an implementation of HTM [17].

[*] Corresponding author.

M. Köppen et al. (Eds.): ICONIP 2008, Part I, LNCS 5506, pp. 428–435, 2009.

The work of Vernon Mountcastle cited in Hawkins [9] pioneered a set of research experiments on the human neocortex, which showed that the cortical structure is fairly uniform, with the same general computation performed in each unit of the cortex, as shown in Fig. 1. Each of what we consider specialized sections (visual cortex, auditory cortex etc) had, in fact, learnt its task based on the properties of the training patterns that had been presented to it. A section of the brain could, as has been shown in several research works [10, 11], learn over time to solve a different task.

Fig. 1a and b show the general replicating structure of the cortical column. Fig. 1c shows the hierarchical connections between the various layers of the visual cortex, discussed by Cadieu et al [6] and George et al [8].

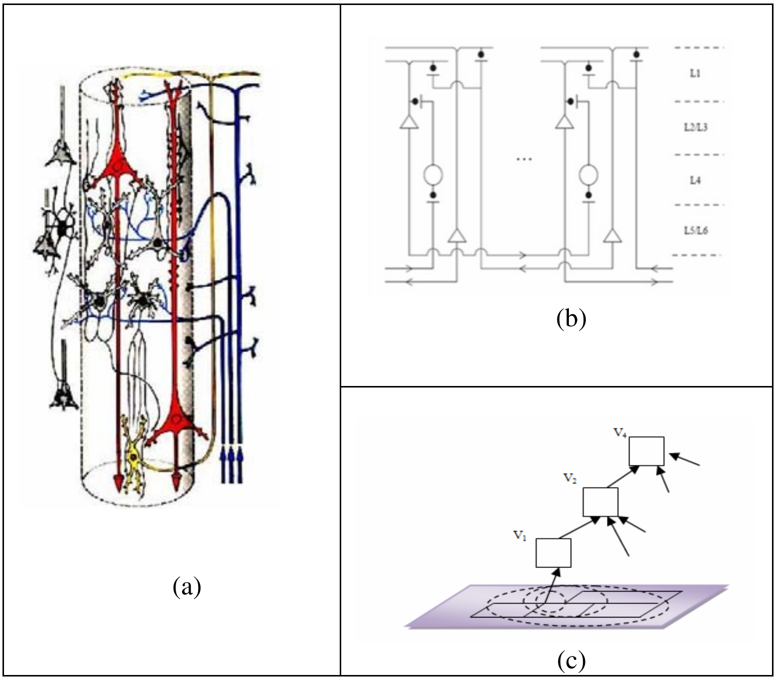

Fig. 1. (a) A depiction off the cortical column as identified by Mountcastle showing the different types of replicating neurons (b) The replicating structure of the cortical column [18] (c). Hierarchy in the layers of the visual cortex when corresponding to an input image.

New evidence comes forth every day that learning is a hierarchical process [12]. Research has long shown that architectures of perception, such as vision and audition are hierarchical in nature. Aspects of memory, as such chunking, also make use of hierarchical architectures. This hierarchical process is not learnt, but is inherent and displayed even in young babies [13].

The HST-M uses principles of uniformity and hierarchy as its roots. A spatio-temporal neural network is the fundamental building block of the HST-M, and is replicated throughout a hierarchical structure.

2 The HST-M Structure and Training Algorithm

Fig. 2b to d shows the structure of the HST-M and that of the component neural networks. We work with a dataset $\mathbf{D} = \{\mathbf{P_1}, \mathbf{P_2}, ...\mathbf{P_n}\}$, composed of the patterns $\mathbf{P_1}, \mathbf{P_2}, ...\mathbf{P_n}$, where each pattern is represented by a m dimensional feature vector such that $\mathbf{P}_i = \{p_{i,1}, p_{i,2}, ..., p_{i,m}\}$.

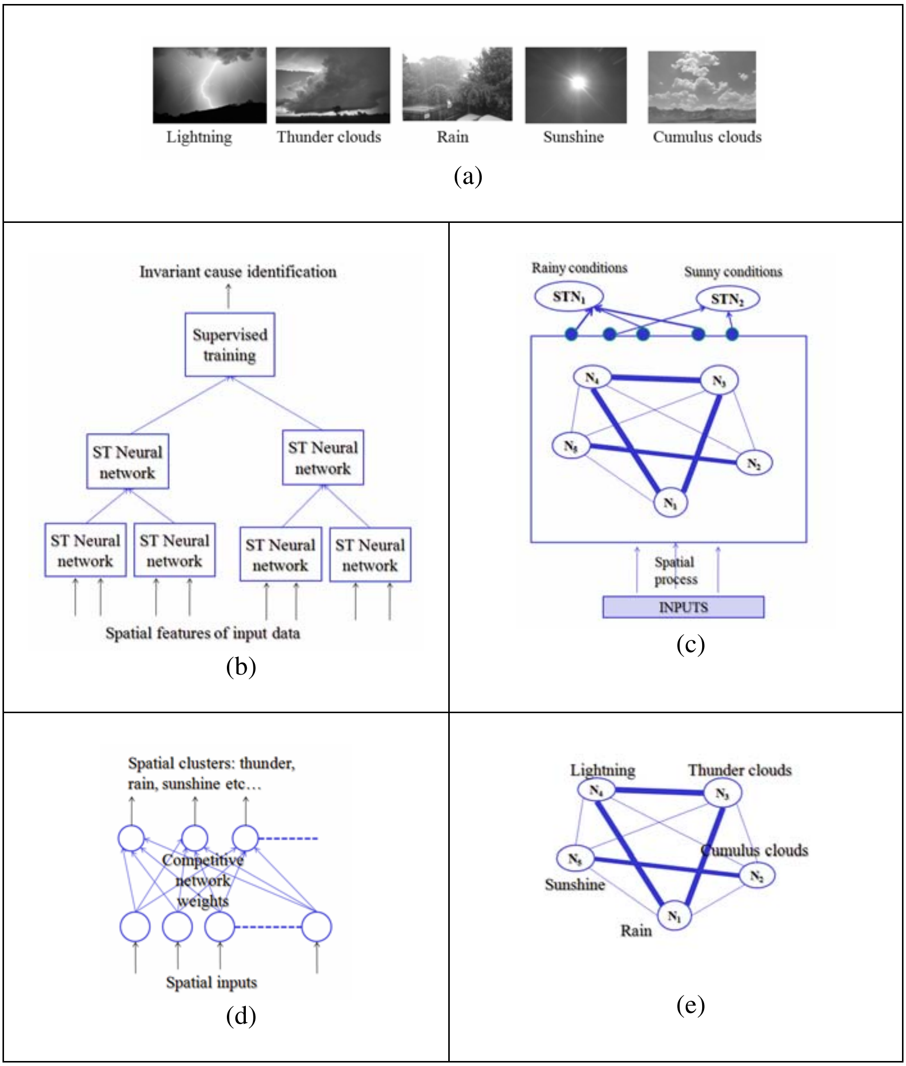

Fig. 2. The connectionist structure of a HST-M showing (a) A sample set of data, (b) The system architecture, (c) the structure of a spatio-temporal node, (d) the spatial processing unit and (e) the temporal processing unit

Consider now, that the dataset $\mathbf{D}$ consists of *32x32* images representing weather conditions, as indicated in Fig. 2a. The images are of a continuing time sequence, and consist of scenes from five categories: {*Sunshine, Rain, Cumulus Clouds, Thunder Clouds, Lightning*}. Intuitively, we know that, based on the co-occurrence correlation of the information, the data can be divided into two associated groups consisting of *{Sunshine, Cumulus Clouds}* and *{Rain, Thunder Clouds, Lightning}*. In other words, the patterns in the associated groups have a higher probability of appearing together in time.

The HST-M processes the patterns by first splitting the m features into k sub vectors of arbitrary size, such that $\mathbf{P_i} = \{\mathbf{p_{i,1}}, \mathbf{p_{i,2}}, \ldots, \mathbf{p_{i,k}}\}, k < m$. Each sub vector $\mathbf{p_{i,j}}$ is fed into one of the lower Spatio-Temporal (ST) Neural Networks of the HST-M network in Fig. 2b. The input to a node in the upper layer is the concatenation of the outputs of the lower layer nodes linked to it.

With reference to the example in Fig. 2, each image is truncated into a smaller segment, say *8x8*. The lower layer if the HST-M is therefore made up of 4 ST Neural Networks, each processing a segment of the image.

Each ST Neural network Fig. 2c is made up of the following architecture: (a) An incremental competitive network (Fig. 2d) that processes the input data according to its spatial properties and (b) A Temporal neural network (Fig. 2e) that connects spatial groups according to their temporal co-occurrence correlation.

Spatial information processing using a competitive neural network: For the problem in Fig. 2, each competitive neural network [3] illustrated in Fig. 2d takes as its input a 64 dimensional vector, representing a segment of the image. The competitive neural network determines the number of spatial clusters in the image segments, thereby dividing the data into 5 spatial clusters, as indicated in Fig. 2d.

Our implementation of competitive networks differs from the traditional Kohonen implementation in that it uses a user defined threshold value τ_1 to dynamically change the number of neurons. The algorithm begins by setting the weights of the first neuron to correspond to the first pattern. Using a defined threshold, the weight and number of neurons is incremented until the system stagnates. The algorithm is therefore suited for incremental adaptive learning.

Temporal information processing: Based on the non-directional co-occurrence correlation of the input data, the weights are adjusted to values as indicated in Fig. 2e, using the update rule in (1). In general, where $S_{(t=i)}$ represents the competitive neuron activated at time *t=i,* and where α and β are two outputs of the spatial processor, and $\mathbf{S}$ represents the state space of the spatial network outputs, a discrete update rule can be expressed as

$$\forall \alpha, \beta \in \mathbf{S}, if \left\lfloor S_{(t=i)} = \alpha \right\rfloor \& \left\lfloor S_{(t=i+1)} = \beta \right\rfloor, W_{\alpha,\beta} ++ \tag{1}$$

Where W refers to the weights of the temporal network.

An ST Neural Network (Fig. 2) therefore consists of a temporal network, whose inputs are the outputs of the spatial network. A maximum group size is imposed to form groups of temporal patterns based on the weights. Based on the processing of the spatio-temporal neurons, the images in Fig. 2a will be finally grouped into two clusters, indicated in Fig. 2b, representing “rainy conditions” and “sunny conditions” respectively.

Hierarchical learning: The implementation of hierarchical learning ensures that each node performs a decision at its own level. For instance, the node which processes information represented in the lower left hand corner of the images in Fig. 2a is likely to form only one temporal group as all the images being given to the system show a black patch. The ability of each ST Neural Network to make inferences at its own level makes the system robust to scaling and skewing, as can be seen from the experimental results.

3 Experimental Results

3.1 Details of Experimental Data

Experiments were run on three datasets: Bitworm, LMH letter recognition, and Pictures. Of these, the Bitworm and the Pictures datasets are benchmark data obtained from Numenta [16]. The LMH letter recognition dataset was generated by us. The details of the datasets are given in Table 1.

Table 1. Details of experimental datasets

Name	Number of inputs	Number of Classes	Number of patterns for training	Testing data
Bitworm	16	2	400	Scaled worms of size 3 to 13 bits
LMH letter recognition	256 (16x16)	3	300	Scaled and Skewed images
Pictures	1024 (32x32)	41	398	Distorted images and Gaussian nose

The bitworm data [16] represents a one dimensional 8 bit worm moving left and right across the 16 bit screen. The worm belongs to one of two classes, solid and textured. Samples of the position of the worm at different time intervals were taken to train the HST-M. Generalization on the bitworm HST-M network was performed using discrete, patterns of solid and textured bitworms. The worms presented to the system range in length between 3 and 13 bits.

The LMH training data consists of moving 16x16 images of the three letters L,M and H, where the patterns undergo training on scaling and translation. Testing is performed using a set of distorted patterns. Fig. 3 shows samples of training and testing data. In the pictures dataset [16] we used consists of 398 black and white images which represent 41 kind of moving objects. The HST-M is tested using noisy data, where the training patterns are distorted using a Gaussian noise of varying standard deviations.

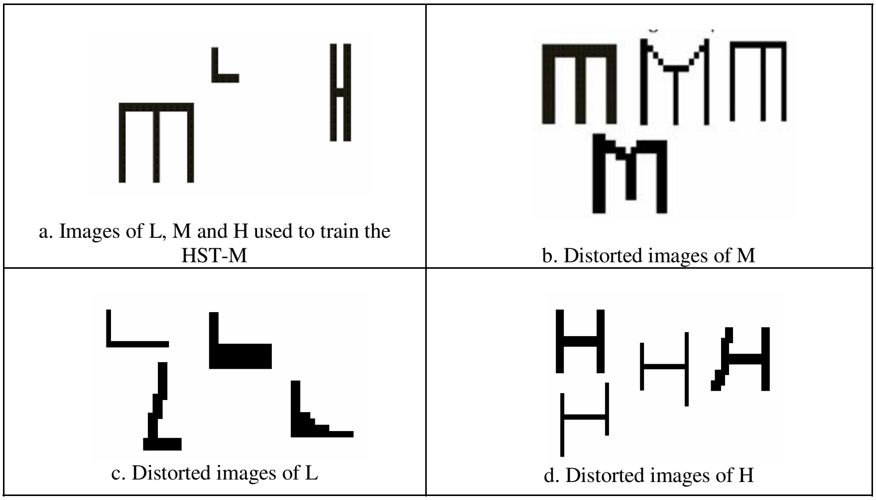
a. Images of L, M and H used to train the HST-M
b. Distorted images of M
c. Distorted images of L
d. Distorted images of H

Fig. 3. Examples of training and testing images of the LMH data for the HST-M memory. The images in b, c and d were used in testing to determine the generalization accuracy of the system.

3.2 Experimental Results

Table 2 shows the experimental results obtained using the various datasets. The results of the HST-M was compared with two networks, traditional MLP implemented using Cascade Correlation [15] with the Quickprop [14] algorithm and an implementation of the HTM architecture by Numenta which proposes the Spatio Temporal architecture using Bayesian probabilistic models [8].

Table 2. Details of experimental results showing the mean and standard deviation of HST-M accuracy compared with the accuracy off MLP and HTM

Name	HSTM architecture	Size of input vector to each bottom node	Mean HST-M accuracy (Std dev) (%)	MLP accuracy (%)	Accuracy of HTM implementation (%)
Bitworm	Multiple	4/8	96.41 (2.46)	60.24	97.86
LMH letter recognition	16,4,1	16	94.43 (5.85)	33.33	92.31
Pictures	32,16,4,1	16	83.8 (9.57)	10.52	80.62

The HST-M algorithm shows a definite improvement of traditional MLP methods. The average accuracy obtained is also comparable to the performance of HTMs. The maximum accuracy obtained is 99.3%, 96.0% and 91.2% for the bitworm, LMH and pictures data respectively, which is higher than the results obtained by HTM.

4 Conclusions and Future Work

The HST-M builds up upon existing literature, making use of the concepts of hierarchy, spatial, and temporal processing to classify patterns. This method of training is considered effective as it models the hierarchical structure of the neo-cortex, thereby achieving time invariant pattern recognition. In this paper, we have studied the method of developing and training a hierarchical spatio-temporal memory structure using neural network algorithms and representations and have tested the system developed on several datasets. The initial results obtained are encouraging, especially in the field of image processing and recognition, where the training algorithms show capability in recognizing noisy and distorted images and is also capable of dealing with scaling and skewing.

Avenues for further work in this area are many. In the algorithm end, they include optimizing the individual neural network algorithms used in training, reducing the number of user-input parameters by automatically determining them in relation to the training set, and determining automatically the optimal hierarchical structure for a problem by using a growing algorithm. In the application end, potential future work includes experimenting with real world, grayscale data and experimenting in other sensory domains.

References

1. Elman, J.L., Bates, E.A., Johnson, M.H., Smith, A.K., Parisi, D., Plunkett, K.: Rethinking Innateness: A Connectionist Perspective on Development (Neural Networks and Connectionist Modeling). MIT Press, Cambridge (1996)
2. Marcus, G.F.: The Algebraic Mind: Integrating Connectionism and Cognitive Science. MIT Press, Cambridge (2001)
3. Kohonen, T.: Self Organizing Maps. Springer, Berlin (1997)
4. Fukushima, K.: Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics 36(4), 93–202 (1980)
5. Fukushima, K., Miyake, S., Ito, T.: Neocognitron: a neural network model for a mechanism of visual pattern recognition. IEEE Transactions on Systems, Man, and Cybernetics 13(3), 826–834 (1983)
6. Cadieu, C., Kouh, M., Pasupathy, A., Conner, C., Riesenhuber, M., Poggio, T.A.: A Model of V4 Shape Selectivity and Invariance. J. Neurophysiol. 98, 1733–1750 (2007)
7. Riesenhuber, M.: Appearance Isn't Everything: News on Object Representation in Cortex. Neuron 55, 341–344 (2007)
8. George, D., Hawkins, J.: A hierarchical Bayesian Model of Invariant Pattern Recognition in the Visual Cortex. International Joint Conference on Neural Networks 3, 1812–1817 (2005)
9. Hawkins, J., Blakeslee, S.: On Intelligence. Owl Books, New York (2004)
10. Rakic, P., Bourgeois, J.P., Eckendorf, M.F., Zecevic, N., Goldman-Rakic, P.S.: Concurrent Overproduction of synapses in diverse regions of the primate cerebral cortex. Science 232, 232–235 (1986)
11. Stelzner, D.J.: Short-circuit recovery from spinal injury. Nature Medicine 14(1), 19–20 (2008)
12. Van Hessen, D.C., Anderson, C.H., Felleman, D.J.: Information processing in the primate visual system: an integrated systems perspective. Science 2255(5043), 419–423 (1992)

13. Babies use grown up memory tricks. New Scientist Magazine 17 (July 19, 2008)
14. Fahlman, S.E.: Faster learning variations on back-propagation: An empirical study. In: Proceedings of the 1988 Connectionist models Summer School (1988)
15. Fahlman, S.E., Lebiere, C.: The cascade correlation learning architecture. In: Advances in neural information processing systems 2, pp. 524–532. Morgan Kaufmann, San Francisco (1990)
16. Numenta, `http://www.numenta.com`
17. George, D., Jaros, B.: The HTM learning algorithms, `http://www.numenta.com/for-developers/education/Numenta_HTM_Learning_Algos.pdf`
18. Thomas, D.: A computational model of the cerebral cortex. In: The Proceedings of Twentieth National Conference on Artificial Intelligence (AAAI 2005), pp. 938–943. MIT Press, Cambridge (2005)