Neurocomputing 73 (2010) 1057–1064

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Producing pattern examples from "mental" images ☆

Bruno P.A. Grieco [a], Priscila M.V. Lima [b], Massimo De Gregorio [c], Felipe M.G. França [a,b,*]

[a] PESC/COPPE, Universidade Federal do Rio de Janeiro, Caixa Postal 68511, 21941-972 Rio de Janeiro, RJ, Brazil
[b] LAM—Computer Architecture and Microelectronics Laboratory, Universidade Federal do Rio de Janeiro, Caixa Postal 68511, 21941-972 Rio de Janeiro, RJ, Brazil
[c] Istituto di Cibernetica Eduardo Caianiello, CNR, Via Campi Flegrei 34, 80078 Pozzuoli, NA, Italy

ARTICLE INFO

Available online 21 January 2010

Keywords:
WiSARD
DRASiW
Mental images
Pattern recognition
Pattern generation
Weightless neural networks

ABSTRACT

The WiSARD (Wilkie, Stonham and Aleksander's Recognition Device) weightless neural network model has its functionality based on the collective response of RAM-based neurons. WiSARD's learning phase consists on writing at the RAM neurons' positions addressed (typically through a pseudo-random mapping) by binary training patterns. By counting the frequency of writing accesses at RAM neuron positions during the learning phase, it is possible to associate the most accessed addresses with the corresponding input field contents that defined them. The idea of associating this process with the formation of "mental" images is explored in the DRASiW model, a WiSARD extension provided with the ability of producing pattern examples, or prototypes, derived from learnt categories. This work demonstrates the equivalence of two ways of generating such prototypes: (i) via frequency counting and filtering and (ii) via formulating fuzzy rules. Moreover, it is shown, through the exploration of the MNIST database of handwritten digits as benchmark, how the process of mental images formation can improve WiSARD's classification skills.

1. Introduction

Although very important to many applications of intelligent systems, explaining the resulting choice of a category is a process that may prove hard to accomplish. In particular, in intelligent computational models, such as feedforward neural networks, where knowledge is stored in a distributed fashion (i.e., synaptic weights hold the results of the learning/training process) and pattern classification is performed via an input-to-output unidirectional flow, it is hard to obtain explicit information about *which* pattern(s) the classifier internally associates with a target output. In other words: how the classifier could produce a pattern example, a *prototype*, of a particular class? Apart from being a kind of explanation coming from an artificial classifier, another question would be about the usefulness of such information. This paper presents qualitative and quantitative explorations of a weightless neural system capable of producing prototypes, i.e., self-generated pattern examples. Besides, positive results concerning the use of prototypes as a disambiguation tool in the classification process are discussed.

The pioneering use of n-tuple RAM nodes in pattern recognition problems is due to the work of Bledsoe and Browning in the late 1950s [18]. A few years later, Aleksander introduced the stored logic adaptive microcircuit (SLAM), i.e., n-tuple RAM nodes as basic components for an adaptive learning network [19]. Created by Wilkes, Stonham, and Aleksander in 1984, the WiSARD perceptron was the first artificial neural network machine (and the most representative weightless neural network (WNN) model) to be patented and produced commercially [1]. Many other WNN paradigms were proposed and have been surveyed in [20,13]. WiSARD takes a set of bits as input, which is then parsed into a set of uncorrelated n-tuples. Each n-tuple is used as a specific address of a RAM-based neuron, in such a way that the input field is completely covered. A WiSARD *discriminator* is composed by a set of n-tuples covering the whole input field, and is trained with representative data of a specific class/category. A discriminator recognizes a test pattern via summation of all of its associated RAM neurons' output. Therefore, the WiSARD model is a multi-discriminator, unidirectional architecture, in summary, a perceptron.

The DRASiW model introduces a way of providing the WiSARD model with backward-classification capabilities, so that one can ask for prototypes of already learnt categories [4,7], i.e., each discriminator is able to produce representative examples of a class that have been learnt from trained patterns. In order to make this possible, RAM neurons' positions act as access counters, which contents can be reversed to an internal "retina", where a "mental" image is produced, thus yielding a bidirectional structure. The "mental" image metaphor, associated with the

☆ Funded by: BZG.

* Corresponding author at: PESC/COPPE, Universidade Federal do Rio de Janeiro, Caixa Postal 68511, 21941-972 Rio de Janeiro, RJ, Brazil. Tel.: +55 21 2562 8663; fax: +55 21 2562 8676.

E-mail addresses: bgrieco@mac.com (B.P. Grieco), priscilamvl@gmail.com (P.M. Lima), m.degregorio@cib.na.cnr.it (M. De Gregorio), felipe@cos.ufrj.br (F.M. França).

0925-2312/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2009.11.015

internal "retina" metaphor, was originally explored in [17], which also discusses the cognitive plausibilities related to these ideas. Prototypes, or pattern examples, of trained classes, are obtained from a mental image by simply selecting RAM neurons' positions with highest values.

Fuzzy logic has improved weightless neural network performance on categorization and cognition [8,9] and both Boolean [10] and fuzzy rule [11] extraction has been explored in this context. Based on the preliminary ideas presented in [14], this work proposes an equivalent way of expliciting learnt categories, based on the same information considered in the DRASiW model, in the form of fuzzy rules. The MNIST database of handwritten digits [15] was used to demonstrate how both DRASiW and the equivalent fuzzy rule formulation are able to produce quite representative pattern examples. Moreover, as an interesting byproduct, it is also shown how the process of mental images formation provided by DRASiW can improve WiSARD's classification skills.

The next section explains the WiSARD weightless neural network model. Section 3 describes how "mental" images are produced by the DRASiW generalization. Section 4 demonstrates an equivalent fuzzy rule formulation. The generation of pattern examples by both DRASiW and its equivalent fuzzy rule specification is explored and compared, via experimentation, in Section 5. A novel way to improve WiSARD's classification capabilities is introduced and tested in Section 6. Section 7 presents our conclusions.

2. The WiSARD perceptron

A RAM-discriminator consists of a set of X one-bit word RAM nodes, or RAM neurons, with n inputs and a summing device (Σ) [1]. Any such RAM-discriminator can receive a binary pattern of ($X \times n$) bits as input. The RAM input lines are connected to the input pattern by means of a biunivocal pseudo-random mapping (see left part of Fig. 1). The summing device enables this network of RAM nodes to exhibit—just like other ANN models based on synaptic weights—generalization and noise tolerance [2].

In order to train the discriminator, one has to set all RAM memory locations to "0" and choose a training set formed by binary patterns of ($X \times n$) bits. For each training pattern, a "1" is stored in the memory location of each RAM addressed by this input pattern. Once the training of patterns is completed, RAM memory contents will be set to a certain number of "0"s and "1"s.

The information stored by RAM nodes during the training phase is used to deal with previous unseen patterns. When one of these is given as input, RAM memory contents addressed by the input pattern are read and summed by Σ. The number r thus obtained, which is called the *discriminator response*, is equal to the number of RAMs that output "1". It is easy to see that r necessarily reaches the maximum X if the input pattern belongs to the training set. r is equal to "0" if no n-bit component of the input pattern appears in the training set (not a single RAM outputs "1"). Intermediate values of r express a kind of "similarity measure" of the input pattern with respect to the patterns in the training set.

A system formed by various RAM-discriminators is called WiSARD (Wilkie, Stonham and Aleksander's Recognition Device) [1,13]. Each RAM-discriminator is trained upon a particular class of patterns, and classification by the multi-discriminator system is performed in the following way. When a pattern is given as input, each RAM-discriminator gives a response to that input. The various responses are evaluated by an algorithm which compares them and computes the relative confidence c of the highest response (e.g., the difference d between the highest response and the second highest response, divided by the highest response). A schematic representation of a RAM-discriminator and a 10 RAM-discriminator WiSARD are illustrated by Fig. 1.

The performance of the WiSARD strongly depends on n. Other factors, such as the choice of the training set, the way confidence is calculated, etc., also influences WiSARD's performance. Specialized responses from WiSARD grows with n; on the other hand, generalization capabilities of WiSARD grows inversely with n [2].

3. DRASiW: WiSARD's "mental" images

DRASiW is an extension to the WiSARD model provided with the ability of producing pattern examples, or prototypes, derived from learned categories. RAM-discriminators are modified in what their memory locations may hold and, correspondingly, in their training algorithm. These changes, which produce something very similar to PLN nodes, introduced by Aleksander [3], allows one to store q-bit words in memory locations (where q is usually not greater than 8); this information, in turn, can be exploited in the generation of "mental" images of learned pattern categories (further improving in other ways the behaviour of RAM-discriminators).

The training algorithm of RAM-discriminators is changed in one aspect only: instead of storing "1"s, it just increment (+1) memory location contents that are addressed by input patterns. At the end of the training phase, values of the memory contents will vary between 0 and Y (where Y is the number of training patterns). Fig. 2 shows the result of training the same RAM-discriminator of Fig. 1 with examples of stylized vertical lines (Fig. 2 left), by means of the new algorithm.

The various memory content values can now be associated with subpattern frequency in the training set. For instance, the memory content of the address 010 associated with the $\diamond$−RAM is 5. This value indicates that the subpattern 010 is present 5 times in the training set of Fig. 2. One should notice that the new domain of memory content values (non-negative integers) do not

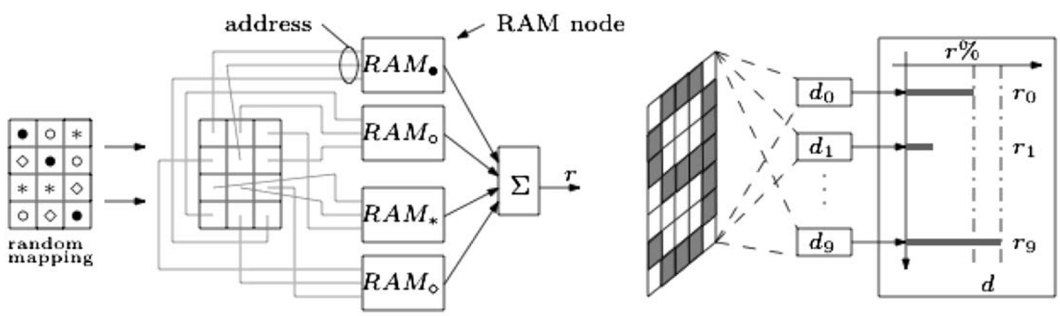

Fig. 1. Example of a RAM-discriminator (left) and of a 10 RAM discriminator WiSARD (right).

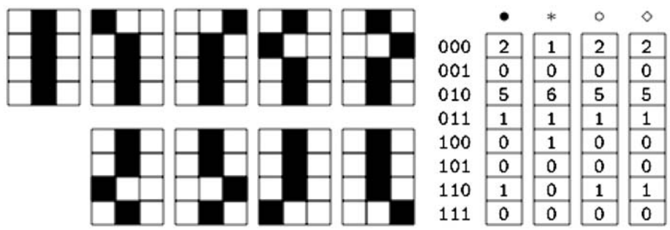

Fig. 2. Vertical line training set (left) and associated RAM-discriminator memory contents (right).

induce a different behaviour with respect to regularly trained RAM-discriminators if the Σ device counts the number of addressed memory locations whose content differs from "0".

One may take advantage of the new values stored in the RAM nodes in order to produce "mental" images [4,16]. This behaviour is significantly related (but not identical) to the exact input/output reversibility exhibited by bidirectional associative memories (BAM) introduced by Kosko [5]. The form of bidirectional behaviour we want to obtain from a RAM-discriminator D, trained with the new algorithm to pick out the elements of class C, must satisfy the following conditions:

(a) in one direction, D has to perform the usual classification process of RAM-discriminators;
(b) in the opposite direction, D has to provide, given the name of class C as input, an example of C.

It is not required that the example be identical to any member of the training set for D. Furthermore, we regard (b) as satisfied for an example if it is correctly classified by the multi-discriminator. The solution herein outlined involves the construction of grey level (rather than black and white) images, in an internal retina having the same dimensions of the input field, by exploiting the information held in the modified RAM memory locations. (A mathematical framework approaching the reversibility problem for weightless systems is briefly sketched in [6].)

The procedure for constructing grey level images is the following. Let b_1, b_2, and b_3 be the first, second, and third bit forming the address of a memory location (for instance, $b_1 = 0$, $b_2=1$ and $b_3=1$ represent the address of the "011" memory location). A particular pixel of the image is associated with each of these bits (see the mapping in Fig. 2). For a given RAM node, let B_i, $i=1, 2, 3$, be the sum of all memory location contents for which b_i is "1" and the value stored is not equal to "0". As an illustration, consider the values computed for the ◦—discriminator in Fig. 2: $B_1=1$, $B_2=7$ and $B_3=1$. Applying this condition to every RAM in Fig. 2 we obtain: $\forall j : j \in \{\bullet, *, \circ, \diamond\}, B_{1j} = 1, B_{2j} = 7, B_{3j} = 1$. This regularity over the four RAMs depends on the fact that each pixel in the left-hand and right-hand columns of the matrix assumes value "1" (black) only once in the training set, whereas each pixel in the central column assumes value "1" (black) seven times in the training set.

Now, one can set the grey intensity level of each pixel associated with bit b_{ij} in such a way that it is proportional to the corresponding value B_{ij}: the higher the value of B_{ij} the darker will be its grey intensity level. The result of this procedure applied to the modified RAM-discriminator trained on the feature vertical line is shown in Fig. 3.

Let us now consider a generic WiSARD trained on a wider domain of simple visual features. In recognising these features, the WiSARD system (trained by the modified algorithm described in this section) works in a canonical way, i.e., just as any regular multi-discriminator system. Moreover, the system may also provide, upon request, an example of each visual feature it can classify.

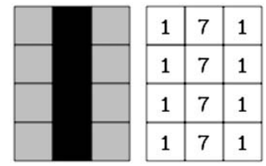

Fig. 3. Two equivalent views of the resulting vertical line "mental" image: grey level (left) and frequency counting (right).

Table 1
RAM contents membership.

a	b	c	d	Σ	Norm.
0	0	0	0	0	0
0	0	0	1	0	0
0	0	1	0	0	0
0	0	1	1	3	0.6
.	.	.	.	…	…
1	1	0	0	0	0
1	1	0	1	2	0.4
1	1	1	0	0	0
1	1	1	1	0	0

4. "Mental" images as fuzzy rules

This section describes the idea of using a *fuzzy discriminator* for retro-classification, in the same spirit of DRASiW's "mental" images, introduced in [14]. One possible benefit of this approach would be to provide a way for the classification system to pipe its output to yet another rule-based automated system. This way, the semantic gap between the symbolic logical reasoning and the fuzzy discriminator based system could be shortened since intercommunication would be performed via the exchange of a set of rules, rather than explicit "mental" images. If desired, it would still be possible to generate prototypes of the stored training set from this set of rules.

The fuzzy discriminator consists of a set of fuzzy OR rules (t-conorms), each representing one of DRASiW's original RAMs. The output of the fuzzy discriminator is the sum of the outputs of each fuzzy OR rules. Each of the fuzzy OR rules is a fuzzy OR between a set of fuzzy AND rules (t-norms), generated by DRASiW's RAM contents.

By analysing the contents of each individual RAM, one could notice that each entry may be regarded as a rule stating that, for example, if input a equals "1" and input b equals "0", then this RAM outputs "1". Since each DRASiW's RAM node position has an associated integer value, if this value is normalized with respect to the highest output value, it may be said that it represents the membership of this entry within the set of firing rules for this RAM node. The same apply to the input lines themselves. Normalized accordingly, they produce a table that represents the membership value of that specific input to the RAM node. As an input may be either "0" or "1", both values must be taken into account when creating a membership table. Table 1 illustrates a possible contents membership table of an individual RAM node, absolute and normalized counting values. Table 2 illustrates an input bit table that could be associated with that of Table 1, also including absolute and normalized membership values.

By targeting the memory locations that have a stored value (set during training) different from zero, a fuzzy AND rule is created taking into account the input membership value and the correlation of those individual inputs membership values. For example, considering the membership of training pattern "0011" as 0.6 (see Table 1) and the normalized membership value of inputs a, b, c, and d (see Table 2), a corresponding fuzzy AND rule would be

$$[(\neg a \times 0.6) \wedge (\neg b \times 0.6) \wedge (c \times 0.6) \wedge (d \times 1.0)] \times 0.6 \tag{1}$$

The procedure that derives the expected fuzzy OR rules like the one in Eq. (2) is straightforward and consists of concatenating the several fuzzy AND rules generated by (1).

$$\{[(\neg a \times 0.6) \wedge (\neg b \times 0.6) \wedge (c \times 0.6) \wedge (d \times 1.0)] \times 0.6 \vee \cdots \vee [(a \times 0.4) \wedge (b \times 0.4) \wedge (c \times 0.4) \wedge (d \times 1.0)] \times 0.4\} \tag{2}$$

For the fuzzy AND function, the MIN t-norm was used and, the fuzzy OR function, the MAX t-conorm was applied to. Once the membership value for the input (through t-norm) is chosen, it is multiplied by the membership value of the chosen rule (t-conorm). This way, each set of rules provided by a discriminator produces a fuzzy output value. Those results are then added as a WiSARD discriminator would have done.

The extraction of a mental image from the fuzzy discriminator involves testing all possible inputs for each of the fuzzy OR rules. The one that produces the highest output yields the mental image bitmap. Once the mental image is extracted, its output represents the maximum output value of the fuzzy discriminator, because each of the chosen sub-patterns constitutes the one that maximizes the value of a fuzzy OR rule. Considering that the fuzzy discriminator output is the sum of those rules, the output result is also maximum. A mental image is not unique since several patterns may maximize one of the fuzzy OR rules. The number of possible mental images may be calculated as the product of the number of patterns that maximize each independent fuzzy OR rule.

Table 2
RAM input membership.

Input	Σ	Norm.
a	2	0.4
$\neg a$	3	0.6
b	2	0.4
$\neg b$	3	0.6
c	3	0.6
$\neg c$	2	0.4
d	5	1
$\neg d$	0	0

5. Inventing pattern examples

The automatic generation of pattern examples, or prototypes, has several potential applications such as communicating with users, verifying the correctness of a training procedure, checking the efficiency of the learning process (sufficient differences among classes), etc. As already shown in Section 3, DRASiW is able to produce "mental" images through a very simple method. In this section we use "mental" images to generate possible, but not necessarily real, examples of a given set.

The domain considered is that of handwritten numbers, more specifically, the MNIST database [15]. The MNIST handwritten numbers database counts with a training set having 60,000 samples (6000 for each numeral class) and a 10,000 (1000 for each numeral class) patterns test set. In the experiments carried out, a 784-bit bitmap, 28×28, input retina was used.

The grey level view of the "mental" image integrates information about all training patterns in such a way one could not have a clear idea about "what" had been trained if not told beforehand. Understanding a "mental" image can be interpreted as obtaining an example, or an instance, of the trained class. Two approaches to the generation of prototypes, or examples, via processing the mental image stored in a trained discriminator, are investigated in this paper.

The first approach consists of halftoning DRASiW's (grey level) mental image (in the internal retina), via the use of a luminance threshold (e.g., the average images luminance), so that the most relevant pixels, i.e., above the luminance threshold, are set to black, and the less relevant ones set to white, resulting in a (black and white) image of a "prototype" of the target class. The second approach involves the direct use of a fuzzy discriminator to extract a "prototype". As "mental" images produced by fuzzy discriminators are not in greyscale (see the last paragraph of Section 4), the prototype generation process consists on testing all input combinations and setting pixels to black ("1") or white ("0") according to the pattern producing the highest fuzzy output.

Fig. 4 shows "mental" images produced from the MNIST training set, having 60,000 handwritten numbers. Prototypes generated via thresholding DRASiW's mental images are shown in Fig. 5. Prototypes generated via selecting binary patterns maximizing the highest fuzzy output are shown in Fig. 6.

Fig. 4. "Mental" images of the MNIST handwritten numbers training set.

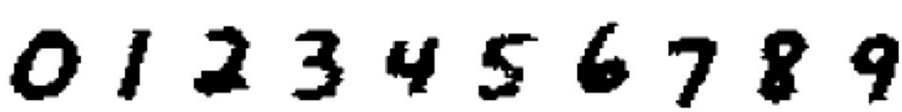

Fig. 5. Prototypes generated via DRASiW's "mental" images.

Fig. 6. Prototypes generated via "mental" images of fuzzy discriminators.

One could see the similarity between the two approaches, i.e., DRASiW fuzzy and prototypes are very similar to each other, although a few differences may be spotted at a closer observation. Fig. 7 shows one example of a possible handwritten number "5" derived by each method.

6. Improving classification skills with "mental" images

Having extracted representative examples of the classes of a given domain from mental images, one would wonder if there would be a way of further profiting from this condensed information. In [14], simple vertical lines were used as training patterns for demonstrating the equivalence of WiSARD/DRASiW and fuzzy discriminator responses. Vertical lines were also used in Section 4 just to explain the formation of mental images. This section introduces a new way of exploring the mental image formation process for the improvement of the classification abilities of the WiSARD model. It is possible that some of the "mental" images produced by DRASiW are contour-saturated images, predominantly composed by dark grey/black pixels, as illustrated in Fig. 4. Such an effect is strongly related to the size of the WiSARD/DRASiW architectures considered on the experiments presented in this section, which were carried out with the MNIST training set. In such a scenario, WiSARD's discriminators generate ambiguous responses, i.e., a draw between true and false winner responses.

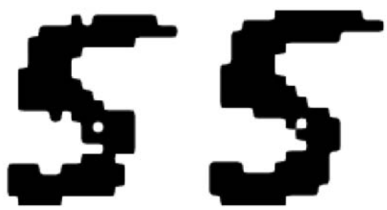

Fig. 7. Prototypes of handwritten "5". Left: via DRASiW's mental image. Right: via fuzzy discriminators.

It is expected that saturation should occur more intensely in an inverted order of the size of RAM neurons: smaller RAM neurons get saturated (most RAM positions written) sooner as the number of patterns used in the training phase increases. Four RAM neurons sizes were tested: 4, 8, 12 and 16 bits input addresses. Fig. 8 presents how the percentage of draws among discriminator responses vary according to the number of training samples per class used in the four WiSARD versions (RAM neuron sizes) considered. The percentage of draws is calculated via counting the number of times during the testing phase, through the use of the MNIST test set, at least one discriminator presents a false positive response of the same magnitude as the discriminator presenting the true positive response. By applying the full MNIST training set, i.e., 6000 patterns per class, the 4-bits RAM nodes WiSARD version produced the highest amount of draws (83.7%) while 16-bits RAM nodes produced the lowest amount of draws (19.2%), as expected.

In order to improve WiSARD's classification performance, RAM-discriminator responses were multiplied by the output of the equivalent fuzzy discriminators formulation. Fig. 9 shows the effect of the new combination compared to the four versions of the original WiSARD architecture. A "correct response" is assumed here when one gets either (i) a single winner discriminator response or (ii) the first winner discriminator response chosen among other winner discriminator responses is a match (draws may occur). While the 4-bits RAM nodes WiSARD took very little profit of the new strategy (from 66.74% to 67.61% of correct responses), the 16-bits RAM nodes version improved from 66.22% to 73.48% of correct responses on the full MNIST training set.

Although combining WiSARD and fuzzy discriminators responses produces better results, saturation of RAM neurons contents still occurs, affecting the overall performance of the classification mechanism. Tackling exactly this issue, a novel and simple way of improving the classification performance of the WiSARD model is introduced here. By taking advantage of

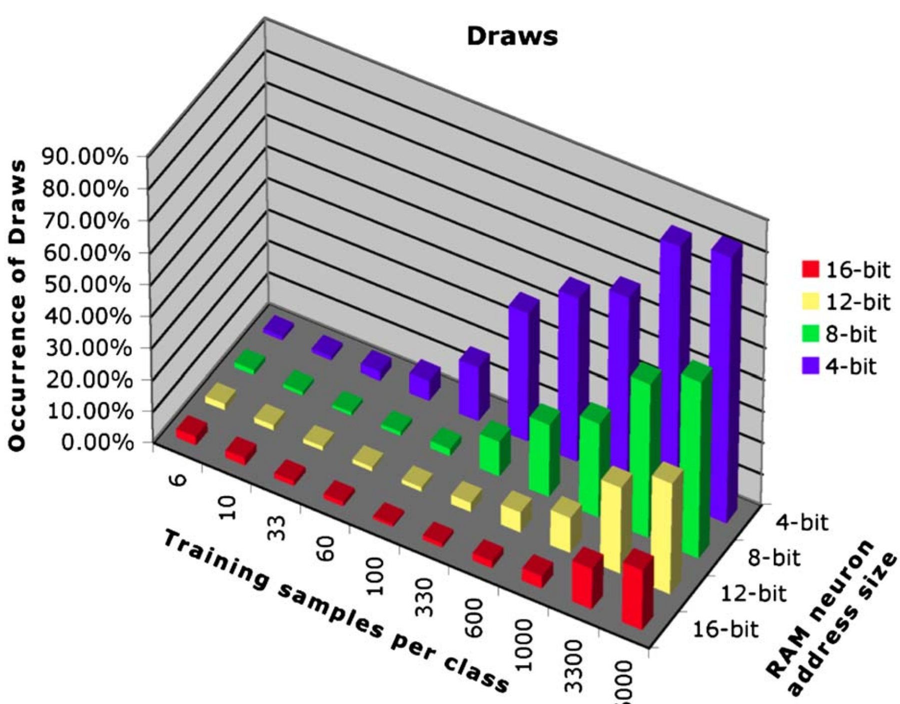

Fig. 8. Occurrence of draws vs. number of training samples per class vs. RAM neurons size.

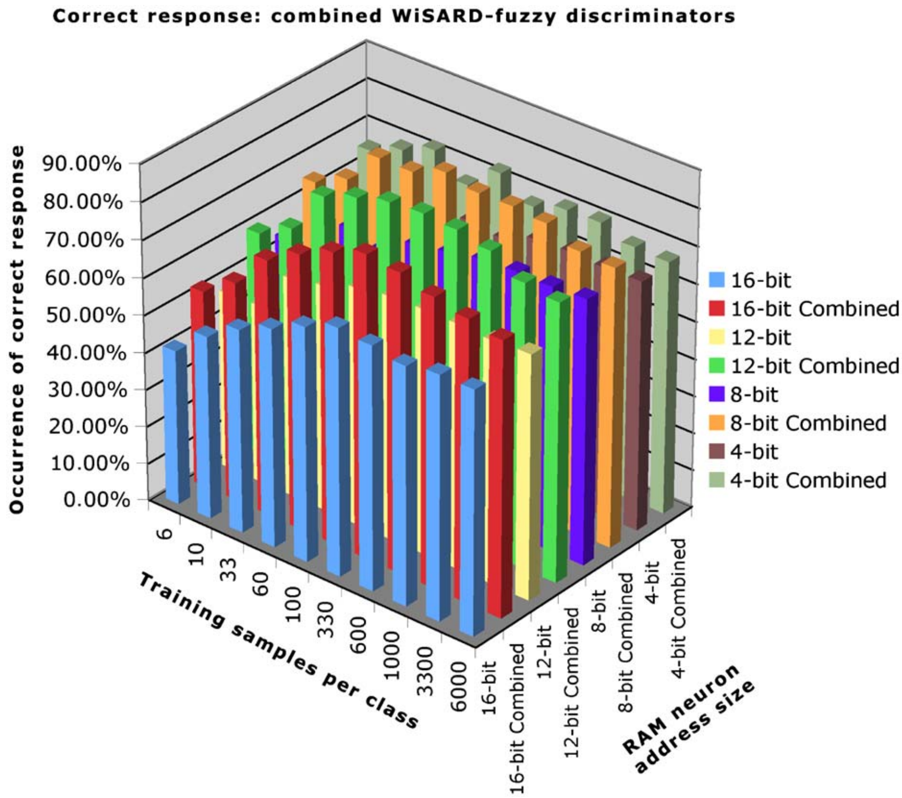

Fig. 9. Occurrence of correct responses vs. number of training samples per class vs. RAM neurons size, with and without combined WiSARD–fuzzy discriminator responses.

DRASIW's prototype generation idea, one can avoid ambiguous discriminator responses upon the presentation of a target test pattern to the WiSARD perceptron. This is obtained through introducing an integer variable threshold b, $b \geq 1$, over all RAM neurons contents of all discriminators. At the beginning of a pattern test, $b=1$ and, if a draw between discriminator responses happens, i.e., $d=0$, one keeps incrementing b so that only RAM neuron contents above b are taken into account by Σ units. A simple convergence policy is to have b incremented until just one of the discriminators produce a winner response, i.e., $d \geq 0$, thus breaking the draw.

Notice that this process is directly related to the way pattern examples are produced from "mental" images in the DRASiW internal retina (Section 5). Fig. 10 shows that the occurrence of draws dropped drastically from 83.7% (see Fig. 8) to 24.12%, in the case of 4-bits RAM neurons version with the full MNIST training set. It can be seen, as illustrated by Fig. 11, that the number of correct responses remained relatively stable in situations where saturation of RAM neurons contents was a problem. The best classification performance, 89.99% of correct responses, was obtained by the 16-bits RAM neurons version when 330 patterns per class were used for training. In fact, the 16-bits RAM nodes version outperformed all other versions, including the combined 8-bits WiSARD–fuzzy. It is worth noticing that the profile of the 16-bits RAM neurons version concerning occurrence of draws, shown in Fig. 10, is coherent with the observation that the best classification performance figures follows the region (100–600 training patterns per class) were the occurrence of draws is minimized.

7. Conclusions

Two approaches for generating unseen examples, or prototypes, of learnt pattern classes in the WiSARD perceptron were introduced in this paper. Both methods were based on the idea of associating the training frequency of each RAM neuron position with the "mental" image of a learnt class. Thus, the DRASiW model constitutes a backward-classification extension to the WiSARD weightless neural model. The first strategy towards generating prototypes consisted of using a threshold in order to filter the most relevant pixels of the "mental" image, constructed from the history of accessed RAM node positions. In the second approach it was shown, through the use of fuzzy discriminators, that it was also possible to produce unseen examples from already learned classes.

A novel way to improve the classification abilities of the WiSARD perceptron was introduced. Based on the same underlying idea of thresholding a "mental" image to generate a prototype, it was shown that, by re-evaluating the set of RAM-discriminator responses upon the detection of a draw, higher rates of correct classification could be obtained. It is possible to draw a comparison between the "mental" image approach and other overfitting avoidance techniques, such as in [21]. The main difference between the two types of approaches lies on the following: classic overfitting control is performed during the training phase and the new approach is independent from how the learning phase was conducted and can be carried out in a continuous fashion.

It is worth mentioning that a probable unfolding of this work is looking for new ways of using "mental" images in the context of neurosymbolic systems, such as in the ones proposed in [12,16,17]. Prototypes obtained via"mental" images may serve as valuable information for hybrid neural rule-based reasoning systems. When fed-back from online pattern classifiers, pattern examples could help rules to be derived in an online fashion.

Finally, it must be noticed that all experimental results were produced over a quite simple and small weightless neural

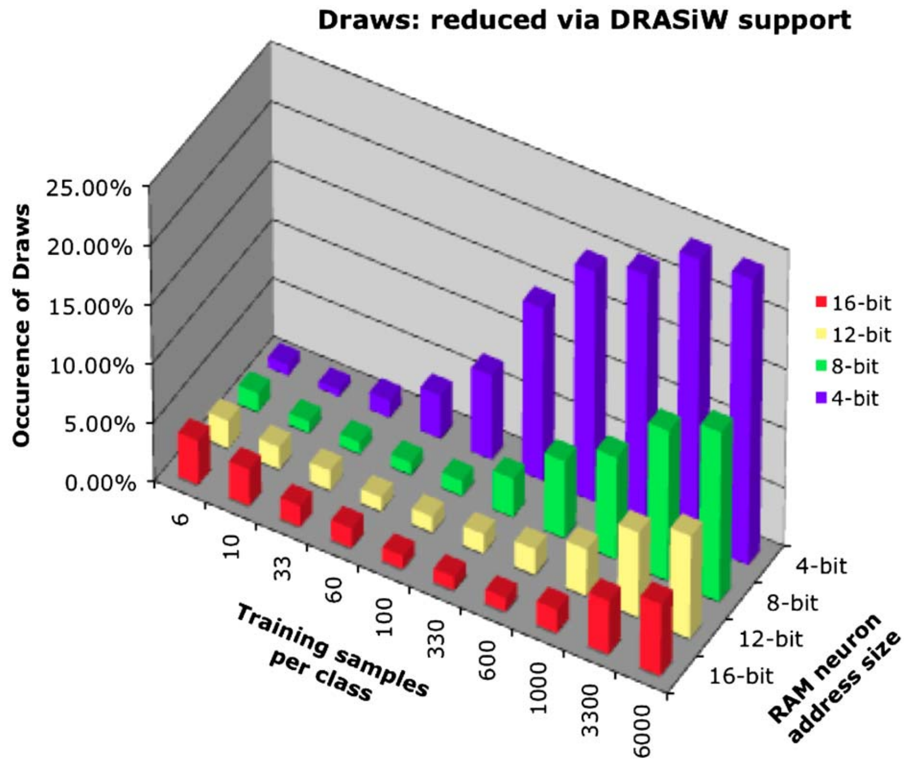

Fig. 10. Occurrence of draws (reduced via using DRASiW's internal retina information) vs. number of training samples per class vs. RAM neurons size.

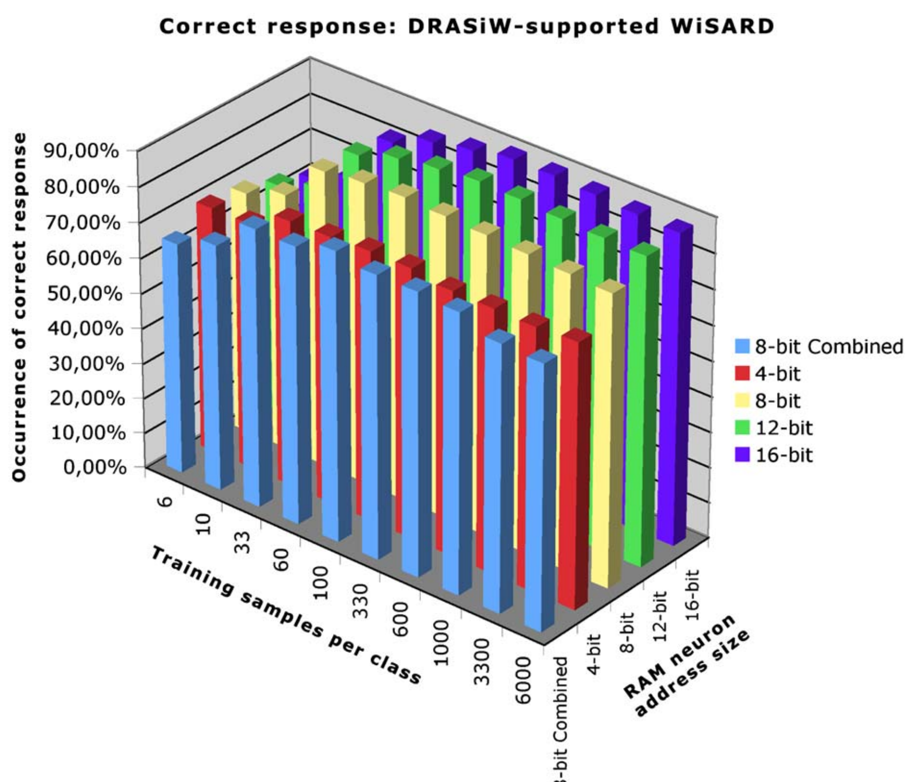

Fig. 11. Correct responses (increased via using DRASiW's internal retina information) vs. number of training samples per class vs. RAM neurons size, with and without combined WiSARD–fuzzy discriminator responses.

architecture. This was so because one of the main intentions of this research was to exercise most of the few parameters of our target neural architecture. Even so, good qualitative and quantitative results were obtained. Investigating the domain and the scalability of the benefits of "mental" images in more complex and larger weightless architectures, where the distribution of patterns in the Boolean space is a focus of concern, such as in [22], is also left for future work.

References

[1] I. Aleksander, W.V. Thomas, P.A. Bowden, WiSARD: a radical step forward in image recognition, Sensor Review (1984) 120–124.
[2] I. Aleksander, An Introduction to Neural Computing, Chapman and Hall, London, 1990.
[3] I. Aleksander, Logical connectionist systems, in: R. Eckmiller, C. von der Malsburg (Eds.), Neural Computers, Springer, Berlin, 1988, pp. 189–197.
[4] M. De Gregorio, On the reversibility of multidiscriminator systems, Technical Report 125/97, Istituto di Cibernetica—CNR, Italy, 1997.
[5] B. Kosko, Bidirectional associative memories, IEEE Transaction on System, Man, and Cybernetics 18 (1988) 49–60.
[6] A. Redgers, I. Aleksander, Digital neural networks, in: K. Warwick, et al. (Eds.), Neural Networks for Control and Systems, Peter Peregrinus, London, 1992, pp. 13–30.
[7] C.M. Soares, C.L.F. da Silva, M. De Gregorio, F.M.G. França, Uma Implementaç ão em Software do Classificador WiSARD, in: Proceedings of the V Simpósio Brasileiro de Redes Neurais (SBRN 1998), vol. 2, December 1998, pp. 225–229 (in Portuguese).
[8] J.A. Tome, J.P. Carvalho, Fuzzy Boolean networks learning behaviour, in: Proceedings of the 7th International Conference on Intelligent Systems Design and Applications (ISDA 2007), October 2007, pp. 889–894.
[9] J.A. Tome, J.P. Carvalho, Decision validation and emotional layers on fuzzy Boolean networks, in: Proceedings of the IEEE Annual Meeting of the Fuzzy Information (NAFIPS '04), vol. 1, June 2004, pp. 136–139.
[10] T.B. Ludermir, M.C.P. de Souto, W.R. de Oliveira, Weightless neural networks: knowledge-based inference systems, in: Proceedings of the 10th Brazilian Symposium on Neural Networks (SBRN 2008), November 2008, pp. 207–212.
[11] R.K. Brouwer, Fuzzy rule extraction from a feed forward neural network by training a representative fuzzy neural network using gradient descent, in: Proceedings of the 2004 IEEE International Conference on Industrial Technology (IEEE ICIT '04), vol. 3, December 2004, pp. 1168–1172.
[12] E. Burattini, P. Coraggio, M. De Gregorio, M. Staffa, Agent WiSARD in a 3D world, in: J. Mira, J.R. Álvarez (Eds.), Proceedings of the 1st International Work-Conference on the Interplay Between Natural And Artificial Computation (IWINAC 2005), Lecture Notes in Computer Science, vol. 3562, Springer, Berlin, 2005, pp. 272–280.
[13] I. Aleksander, M. De Gregorio, F.M.G. França, P.M.V. Lima, H. Morton, A brief introduction to weightless neural systems, in: Proceedings of the 17th European Symposium on Artificial Neural Networks (ESANN 2009), Bruges, Belgium, 2009, pp. 299–305.
[14] B.P.A. Grieco, P.M.V. Lima, M. De Gregorio, F.M.G. França, Extracting fuzzy rules from "mental" images generated by modified WiSARD perceptrons, in: Proceedings of the 17th European Symposium on Artificial Neural Networks (ESANN 2009), Bruges, Belgium, 2009, pp. 313–318.
[15] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (11) (1998) 2278–2324.
[16] E. Burattini, M. De Gregorio, G. Tamburrini, Mental imagery in explanation of visual object classification, in: Proceedings of the 6th Brazilian Symposium on Neural networks (SBRN 2000), Rio de Janeiro, Brazil, 2000, pp. 137–143.
[17] E. Burattini, M. De Gregorio, G. Tamburrini, Generation and classification of recall images by neurosymbolic computation, in: Proceedings of the 2nd European Conference on Cognitive Modelling (ECCM98), Notthingam, UK, April 1998, pp. 127–134.
[18] W.W. Bledsoe, I. Browning, Pattern recognition and reading by machine, in: Proceedings of the Eastern Joint Computer Conference, Boston, USA, 1959, pp. 225–232.
[19] I. Aleksander, Self-adaptive universal logic circuits, IEE Electronic Letters 2 (1966) 321.
[20] T.B. Ludermir, A. Carvalho, A.P. Braga, M.C.P. Souto, Weightless neural models: a review of current and past works, Neural Computing Surveys 2 (1999) 41–61.
[21] G. Monari, G. Dreyfus, Local overfitting control via leverages, Neural Computation 14 (2002) 1481–1506.
[22] A.P. Braga, I. Aleksander, Geometrical treatment and statistical modelling of the distribution of patterns in the n-dimensional Boolean space, Pattern Recognition Letters 16 (1995) 507–515.

Bruno P.A. Grieco graduated in Technology Information from the Pontifical Catholic University of Rio de Janeiro (PUC-RJ), where his interest on artificial intelligence and games started in 1992. He received his M.Sc. degree on Informatics from the Postgraduate Programme on Informatics of Universidade Federal do Rio de Janeiro (UFRJ) in 2007. He is currently pursuing a doctorate in neuro-symbolic reasoning at the Systems Engineering and Computing Postgraduate Programme of COPPE-UFRJ.

Priscila M.V. Lima graduated in Computer Science (magna cum lauda) at Universidade Federal do Rio de Janeiro (UFRJ). She received her M.Sc. in Computer Science from COPPE-UFRJ, in 1987, and her Ph.D. in Computing (Artificial Intelligence) from Imperial College London, in 2000. She is currently with the Department of Mathematics, Universidade Federal Rural do Rio de Janeiro. Among her research interests are logical reasoning with artificial neural networks, applications of artificial and computational intelligence and distributed systems.

Massimo De Gregorio got his Laurea in Fisica (CIbernetica) from University of Naples, in 1990. Since 1994 he has been a researcher at the Istituto di Cibernetica, Consiglio Nazionale delle Ricerche (CNR) in the "Theory and Techniques of Knowledge Representation" research group. Between 1999 and 2003, he held visiting professor positions at University of Naples "Federico II" (2000–2003) and at the Second University of Naples (1999–2000). His research interests include artificial intelligence, neuro-symbolic hybrid systems, with extensive applications in expert systems, pattern recognition, active video surveillance and behaviour based robotics.

Felipe M.G. França is an associate professor at the Systems Engineering and Computer Science Program, COPPE, Universidade Federal do Rio de Janeiro (UFRJ). He received his BEE (Electronics Engineering, 1981) and his M.Sc. (Computer Science, 1987) from UFRJ, and his Ph.D. (Neural Systems Engineering, 1994) from the Imperial College London. His research and teaching interests includes computational intelligence models and all aspects of parallel and distributed computing.