Journal of Network and Innovative Computing
ISSN 2160-2174 Volume 1 (2013) pp. 87 - 98

Traffic Sign Recognition with WiSARD and VG-RAM Weightless Neural Networks

Mariella Berger, Avelino Forechi, Alberto F. De Souza, Jorcy de Oliveira Neto, Lucas Veronese, Victor Neves, Edilson de Aguiar and Claudine Badue

Departamento de Informática, Universidade Federal do Espírito Santo,
Av. Fernando Ferrari, 514, 29075-910, Vitória, ES, Brazil
mberger@lcad.inf.ufes.br

***Abstract*: We present two biologically inspired approaches to traffic sign recognition based on Weightless Neural Networks (WNN): one based on Virtual Generalizing Random Access Memory (VG-RAM) neurons and the other on the Wilkes, Stonham and Aleksander Recognition Device (WiSARD) neurons. Both approaches employ the same neural architecture that models the transformations suffered by the images captured by the eyes from the retina to the primary visual cortex (V1) of the mammalian brain. We evaluated the performance of both approaches on the German Traffic Sign Recognition Benchmark (GTSRB). Our system based on VG-RAM neurons achieved a performance significantly better than the one based on WiSARD neurons and was ranked fifth in the GTSRB (the third and fourth places were human classifiers) with a recognition rate of 98.42%.**

***Keywords*:** Traffic Sign Recognition, VG-RAM Weightless Neural Networks, WiSARD, Log-Polar Mapping from Retina to Primary Visual Cortex (V1), German Traffic Sign Recognition Benchmark**.**

I. Introduction

Automatic traffic sign identification has many practical applications, such as traffic sign regulation, driver assistance and automated intelligent driving, and has been a challenging and active research topic in computer vision in the last years. However, the identification of traffic signs with large variations in visual appearance—due to deterioration, illumination changes, partial occlusions, rotation, weather conditions, etc.—remains still an interesting machine learning and pattern recognition problem.

The problem of traffic sign identification can be formulated as follows: given an image of a scene, try and identify one or more traffic signs in the scene using a priori information about the shape, color or features present in the traffic signs. The current solutions in the literature typically involve segmentation of traffic signs from the scenes (traffic sign detection), feature extraction from the traffic signs, and recognition. In this paper, we examined the recognition part of the identification problem only.

Traffic sign recognition is a multi-class classification problem with unbalanced class frequencies. The challenge lies in the fact that, even though there is a wide range of variations between classes in terms of color, shape and presence of pictograms or text, there exist classes very similar to each other (see Figure 1).

Figure 1: Samples of very similar traffic sign classes.

In this paper, we present two biologically inspired approaches to traffic sign recognition: one based on Virtual Generalizing Random Access Memory Weightless Neural Networks (VG-RAM WNN [1]) and the other based on the Wilkes, Stonham and Aleksander Recognition Device (WiSARD [2]). Both approaches employ the same architecture, which models the transformations suffered by the images captured by the eyes from the retina to the primary visual cortex (V1) of the mammalian brain.

We developed systems for traffic sign recognition using both approaches and evaluated their performances on the German Traffic Sign Recognition Benchmark (GTSRB) (http://benchmark.ini.rub.de) [3, 4]. Our system based on VG-RAM neurons achieved a performance significantly better than the one based on WiSARD neurons and was ranked fifth in the GTSRB (the third and fourth places were human classifiers) with a recognition rate of 98.42%.

This paper is organized as follows. After this introduction, in Section II, we present related work. In Section III, we briefly discuss the mammalian brain's mapping from the

retina to V1. In Section IV, we present both types of WNN neurons examined and the biologically inspired neural architecture for traffic sign recognition we developed. In Section V, we describe our experimental methodology and analyze our experimental results. Our conclusions and directions for future work follow in Section VI.

II. Related Work

Many methods have been proposed in the literature for traffic sign recognition (see overviews in [5, 6, 7]). These methods can be grouped in three main categories, depending on the main attributes used in the recognition process, namely color, shape and other features.

Color-based methods perform color segmentation in order to detect and classify (recognize) image regions into specific types of traffic signs [8]. Earlier techniques used adaptive thresholding [9] or fixed color thresholding [10] to identify traffic sign pixels. Other approaches used color indexing and region growing [11], fuzzy logic [12] or color distance transform [13] to determine the borders of the traffic sign and the corresponding pictograms. However, the efficiency of color-based methods is usually affected by outdoor illumination. This can be reduced by converting the RGB image to the Hue-Saturation-Intensity (HSI) or Hue-Saturation-Value (HSV) spaces, which are, to some extent, invariant to changes in illumination conditions [14].

Shape-based methods are more robust to changes in illumination conditions, if compared to color-based methods. Most shape-based approaches first apply robust edge detection to an input image and, thereafter, the result is grouped or compared against relevant geometrical shapes. Many techniques can be used for classifying edges into geometrical shapes corresponding to specific traffic signs: distance transform matching [15], hierarchical spatial feature matching [16], similarity detection [9], Hough transform [17], and template matching [18]. Alternatively, radial symmetry [19] can be employed to detect regular shapes like triangles, squares and octagons in the images, which can be later classified as a specific traffic sign according to its shape.

For many reasons—such as changes in illumination, the appearance of traffic signs in cluttered scenes, imperfect shape of signs, as well as differences in scale and size of traffic signs—, the detection and recognition of traffic signs is a challenging problem for methods based only on color and shape. Feature-based methods rely on special features detected in the images that are invariant with respect to viewing and environmental conditions (e.g. color SIFT [20], Haar-like features [21], HOG features [22]). These features are classified into specific traffic signs using genetic algorithms [23], histographic recognition [24], decision trees [16], nearest neighbor method [25], support vector machines [26, 27], AdaBoost methods [28], neural networks [29, 30], random forest of trees and kd-trees [31], and Virtual Generalizing Random Access Memory Weightless Neural Networks (VG-RAM WNN [32]).

In this paper, we evaluate the performance of two WNN systems for traffic sign recognition: one based on VG-RAM neurons [1] and the other based on WiSARD neurons [2]. WNN are effective machine learning tools that offer simple implementation and fast training and test. Our experimental evaluation shows that VG-RAM neurons outperform WiSARD neurons on the German Traffic Sign Recognition Benchmark (GTSRB).

III. Mapping from the Retina to V1

The images captured by the eyes are transformed into electrical impulses by the retina and, through the optic nerve, are projected into the primary visual cortex (V1) and other areas of the mammalian brain [33]. The neural projection from the retina to V1 follows a retinotopic mapping, i.e., neighboring regions in the retina are projected onto neighboring regions of the V1 [34].

Figure 2 shows how an image containing concentric circles is projected to V1 of the macaque monkey [34]. As Figure 2 shows, circles 1, 2 and 3 in the left image become approximately straight lines in V1, and the regions circumscribed by the inner circles in the left image occupy a much larger area in V1. This mapping from the retina to V1 follows a **log-polar function**.

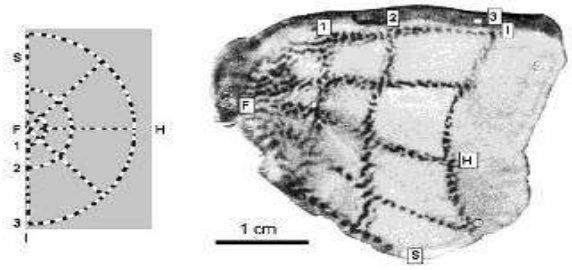


Figure 2: Retinotopic mapping of an image from the retina to V1 [34].

Figure 3 shows the log-polar transform of an image, centered at the point (x_c, y_c)—this point corresponds to the *fovea* of the model. The fovea is the central region of the retina, has the highest density of receptors and, thus, affords the greatest visual acuity [33]. Note that the circle (in red) in the left image of Figure 3 becomes a straight line in the right image, and the regions around the circle's center (the fovea of the model) in the left image occupy a much larger area in the right image. The mathematical modeling of the log-polar transform commonly used in the literature is given by:

$$R = \sqrt{(x - x_c)^2 + (y - y_c)^2} \rightarrow \rho \alpha \log(R) \text{ and} \qquad \textbf{(1)}$$

$$\theta = \arctan\left(\frac{(y - y_c)}{(x - x_c)}\right) \rightarrow \phi \alpha \theta . \qquad \textbf{(2)}$$

In this paper, we did not employ the log-polar transform exactly as shown above, but a variant that was created to emulate more precisely the mapping from the retina to V1. Figure 4 shows this variant of the log-polar transform.

As Figure 4 shows, neighboring regions in the image around the circle's center (the fovea of the model) are also neighbors in the log-polar transform (retinotopy), as occurs in V1. This does not occur in the transform depicted in Figure 3.

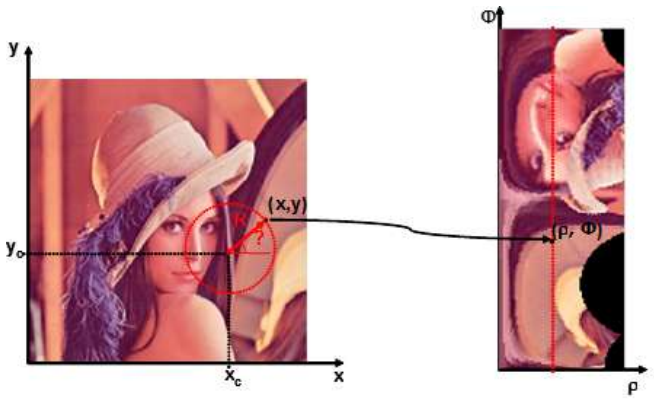


Figure 3: Log-polar transform.

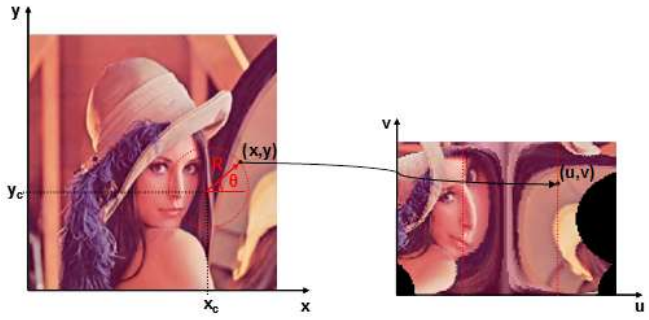


Figure 4: Our variant of the log-polar transform.

IV. Traffic Sign Recognition with WiSARD and VG-RAM WNN

A. WiSARD WNN

RAM-based neural networks, also known as n-tuple classifiers or weightless neural networks, do not store knowledge in their connections but in Random Access Memories (RAM) inside the network's nodes, or neurons. These neurons operate with binary input values and use RAM as lookup tables: the synapses of each neuron collect a vector of bits from the network's inputs that is used as the RAM address, and the value stored at this address is the neuron's output. Training can be made in one shot and basically consists of storing the desired output in the address associated with the neuron input vector [35].

In spite of their remarkable simplicity, RAM-based neural networks are very effective as pattern recognition tools, offering fast training and test, in addition to easy implementation [1]. However, if the network input is too large, the memory size becomes prohibitive, since it must be equal to 2^n, where n is the input size.

The WiSARD proponents [2] tackled this problem by dividing the n-sized input into m segments, each one addressing a RAM memory module of size $2^{n/m}$, as shown in Figure 5(a).

As Figure 5(a) shows, in a WiSARD WNN, each segment of the n-sized input is used to select one position of a RAM memory module of size $2^{n/m}$; the WiSARD output is the most voted output by the m modules. Thanks to this organization, the total amount of memory required is reduced from 2^n to $m \times 2^{n/m}$.

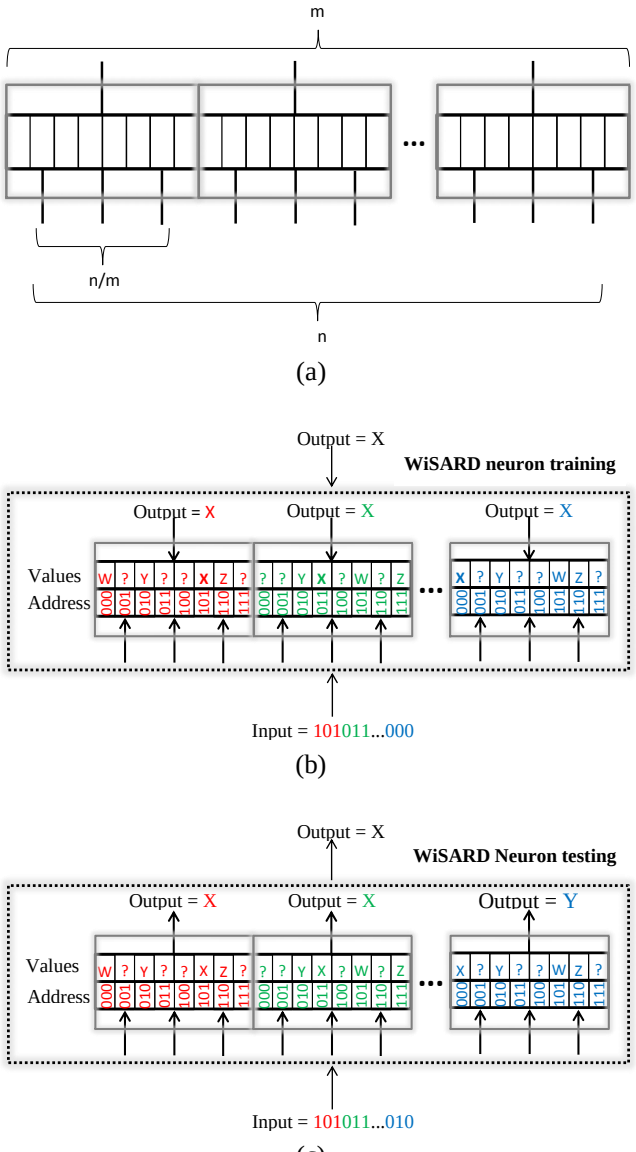


Figure 5: WiSARD WNN. (a) General architecture. (b) Training. (c) Testing.

Figure 5(b) and (c) show a training and a testing instance of a WiSARD neuron, respectively, where $n/m = 3$. As these figures show, in this case ($n/m = 3$), one needs three bits to address each module. During training (Figure 5(b)), a n-sized training input pattern (101011…000) is divided by m and each n/m-sized subpattern (101, 011, …, 010) is used to address a RAM position of the corresponding RAM module. The addressed position in each RAM module stores the output pattern (X) associated with the input pattern.

It is important to note that, if the *n*/*m* value is too small (say 3 or 4), depending on the number of training patterns collisions may occur, i.e., several input-output pairs may address the same RAM module's memory position, hurting learning performance. Therefore, the *n*/*m* value should be chosen taking into consideration the number of input-output pairs in the learning set. Please refer to [36] for an in depth analysis in the context of nearest neighbor search on binary codes.

During test (Figure 5(b)), the *n*-sized input test pattern (101011…010) is also divided by *m* (101, 011, …, 010) and each *n*/*m*-sized input subpattern is used to address a RAM position of the corresponding RAM module. The neuron's output is given by the output pattern (X) with the largest count (2). If a tie occurs, the neuron's output is chosen randomly among the tied output patterns.

B. VG-RAM WNN

Virtual Generalizing Random Access Memory (VG-RAM) Weightless Neural Networks (WNN) are RAM-based neural networks that only require memory capacity to store the data related to the training set [37]. In the neurons of these networks, the memory stores the input-output pairs shown during training, instead of only the output. In the test phase, each neuron searches associatively its memory by comparing the input presented to the network with all inputs in the input-output pairs learned. The output of each VG-RAM WNN neuron is taken from the pair whose input is nearest to the input presented—the distance function employed by VG-RAM WNN neurons is the Hamming distance. If there is more than one pair at the same minimum distance from the input presented, the neuron's output is chosen randomly among these pairs.

Lookup Table	X_1	X_2	X_3	Y
entry #1	1	1	0	*label 1*
entry #2	0	0	1	*label 2*
entry #3	0	1	0	*label 3*
	↑	↑	↑	↓
input	1	0	1	*label 2*

Figure 6: VG-RAM WNN neuron's lookup table.

Figure 6 shows the lookup table of a VG-RAM WNN neuron with three synapses (X_1, X_2 and X_3). This lookup table contains three entries (input-output pairs), which were stored during the training phase (*entry #1*, *entry #2* and *entry #3*). During the test phase, when an input vector (*input*) is presented to the network, the VG-RAM WNN test algorithm calculates the distance between this input vector and each input of the input-output pairs stored in the neuron's lookup table. In the example of Figure 6, the Hamming distance from the *input* to *entry #1* is two, because both X_2 and X_3 bits do not match the input vector. The distance to *entry #2* is one, because X_1 is the only non-matching bit. The distance to *entry #3* is three, as the reader may easily verify. Hence, for this input vector, the algorithm evaluates the neuron's output, *Y*, as *label 2*, since it is the output value stored in *entry #2*.

C. WNN Architecture for Traffic Sign Recognition

Our WNN architecture for traffic sign recognition has a single two-dimensional array of $m \times n$ neurons, N, where each neuron, $n_{i,j}$, has a set of synapses, $W = (w_1, w_2, \ldots w_{|W|})$, which are connected to the network's two-dimensional input, Φ, of $u \times v$ pixels, $\varphi_{k,l}$ (Figure 7). The neurons, $n_{i,j}$ (Figure 7), can be WiSARD neurons, as shown in Figure 5, or VG-RAM WNN neurons, as shown in Figure 6. The mapping of the elements of Φ onto the center of the *receptive field* of each neuron of N follows a log-polar function, which models the mapping from the retina to V1 (Section III).

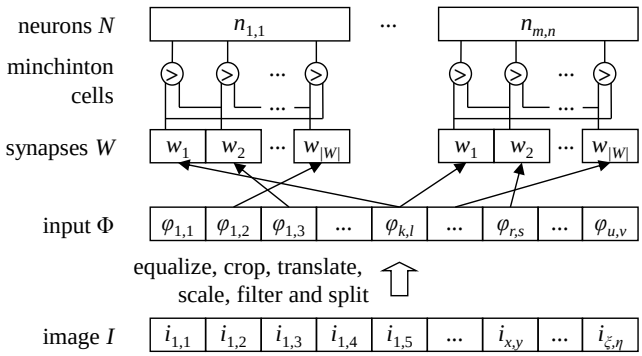


Figure 7: Schematic diagram of our WNN architecture for traffic sign recognition.

The synaptic interconnection pattern of each neuron $n_{i,j}$ (which consubstantiates its receptive field), $\Omega_{i,j,\sigma}(W)$, follows a two-dimensional Normal distribution with variance σ^2 centered at φ_{μ_k,μ_l}, where the coordinates μ_k and μ_l of Φ are given by the inverse log-polar function of the coordinates i and j of N; i.e., the distribution of coordinates k and l of the pixels of Φ to which $n_{i,j}$ connects via W follow the probability density functions:

$$\omega_{\mu_k,\sigma^2}(k) = \frac{1}{\sigma\sqrt{2\Pi}} e^{-\frac{(k-\mu_k)^2}{2\sigma^2}} \text{ and} \quad \textbf{(3)}$$

$$\omega_{\mu_l,\sigma^2}(l) = \frac{1}{\sigma\sqrt{2\Pi}} e^{-\frac{(l-\mu_l)^2}{2\sigma^2}}, \quad \textbf{(4)}$$

where σ is a parameter of the architecture, and the coordinates μ_k and μ_l of the pixel of Φ where the Normal distribution is centered at are calculated by:

$$\mu_k = \frac{u}{2} + d \cdot \cos(\theta) \text{ and} \quad \textbf{(5)}$$

$$\mu_l = \frac{v}{2} + d \cdot sen(\theta), \quad \textbf{(6)}$$

where

$$d = \frac{u}{2} \cdot \left(\frac{\alpha^{\left| \frac{i-m/2}{m/2} \right|} - 1}{\alpha - 1} \right) \text{ and} \quad \textbf{(7)}$$

$$\theta = \begin{cases} \pi \cdot \left(\frac{3n}{2} - \frac{j}{n} \right) + \frac{\pi}{2n}; \text{if } k < \frac{m}{2} \\ \pi \cdot \left(\frac{3n}{2} + \frac{j}{n} \right) + \frac{\pi}{2n}; \text{ if } k > \frac{m}{2} \end{cases}, \quad \textbf{(8)}$$

where α is the log factor of the log-polar function and is also a parameter of the architecture.

This synaptic interconnection pattern mimics that observed in many classes of biological neurons [33]. It is randomly created when the network is built and does not change afterwards; furthermore, although random, it is the same for all neurons.

WNN synapses can only get a single bit from the network input, Φ. Thus, in order to allow our WNN to deal with images, in which a pixel may assume a range of different values, we use *minchinton cells* [38]. In the proposed WNN architecture, each neuron's synapse, w_t, forms a minchinton cell with the next, w_{t+1} ($w_{|W|}$ forms a minchinton cell with w_1). The type of the minchinton cell we have used returns 1 (one) if the synapse w_t of the cell is connected to an input element, $\varphi_{k,l}$, whose value is larger than the value of the element $\varphi_{r,s}$ to which the synapse w_{t+1} is connected, i.e., $\varphi_{k,l} > \varphi_{r,s}$; otherwise, it returns zero (see the synapses w_1 and w_2 of the neuron $n_{1,1}$ of Figure 7).

The input traffic sign images, I, of $\xi \times \eta$ pixels (Figure 8(a)), are transformed before being copied to Φ. They are first equalized by the Contrast-Limited Adaptive Histogram Equalization (CLAHE) [39] for improving the contrast (Figure 8(b)). They are also cropped to keep only the region of interest (traffic sign region) using the GTSRB ground truth bounding box (Figure 8(c)). Following, they are translated to try and bring the region of interest's center closer to the input image's center (Figure 8(d)). The distance (in pixels) and the direction (right, left, up, or down) of the translation are chosen randomly. After that, they are scaled to fit into Φ (Figure 8(e)) and filtered by a Gaussian filter to smooth out artifacts produced by the transformations (Figure 8(f)). Finally, each transformed (color) image is split into its three RGB components, which yields three separate grayscale images representing the red (Figure 8(g)), green (Figure 8(h)) and blue color channels (Figure 8(i)). **We use a separate neural network for each color channel.** This RGB image decomposition helps the system discriminate traffic sign classes of different colors.

Even after cropping the input traffic sign images using the ground truth bounding box, they still present part of the background scene (Figure 8(c)). This impacts negatively the WNN performance because the neurons' synapses connected to the background regions would collect non-relevant information, which could generate ambiguous classification results. To minimize this effect, we assigned weights to the neurons' output. Higher weights are attributed to the output of neurons monitoring regions near the image center, while lower ones are assigned to the output of neurons monitoring regions near the image borders, since background regions typically appear on image corners. The weights of neurons' output in each row of the array of $m \times n$ neurons, N, follow a one-dimensional Normal distribution with variance σ_w^2 and mean $m/2$, where σ_w is a parameter of the architecture. The same weight is attributed to all neurons' output in the same column of N. Figure 9 shows the distribution of weights of the neurons' output in N, where colors varying from black to white correspond to increasing weights.

During training, the input traffic sign image is transformed, i.e., equalized, cropped, translated, scaled, filtered and split into its three RGB components. The pixels of the red image component are copied to the input Φ of the first of the three networks—the red network—and all neurons' outputs are set to the value of the label (class identifier) associated with the image. All neurons are then trained to output this label with this input image. This procedure is repeated for the green and blue networks and, likewise, for all traffic sign images in the training dataset.

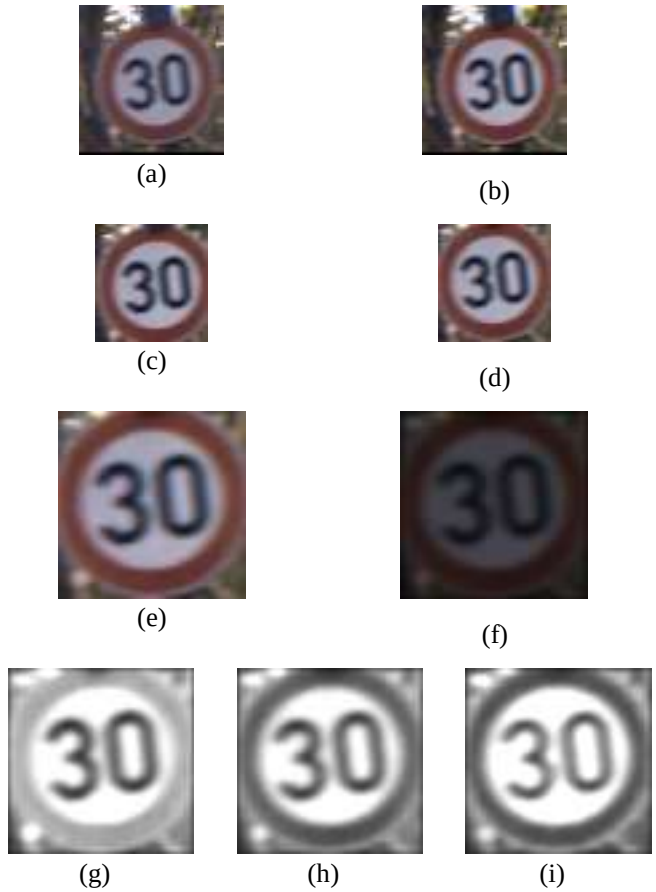


Figure 8: Traffic sign image and its preprocessing. (a) Original image; (b) equalized image; (c) cropped image; (d) translated image; (e) scaled image; (f) filtered image; and image split into its three RGB components: (g) red, (h) green and (i) blue color channels.

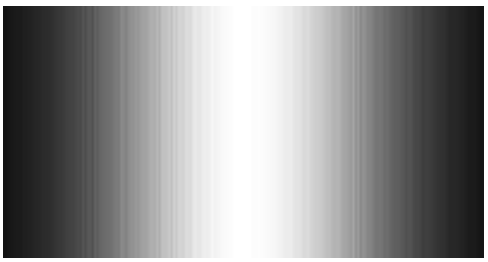


Figure 9: Distribution of the weights of neurons' output in N.

During testing, the input traffic sign image is also transformed and split into its three RGB components, and the pixels of the red image component are copied to the input Φ of the red network. All neurons' outputs are then computed. After that, the number of votes to each label is calculated as the sum of the weights associated with the neurons outputting that label, and the labels are ranked by the number of weighted votes. The degree of belief of the system that (the red component of) the input image belongs to the class identified by the first ranked label is estimated by the relative difference between the number of votes received by the first and second ranked labels, i.e., the difference between the number of votes received by the first and second ranked labels divided by the number of votes received by the first ranked label. This procedure is repeated in the green and blue networks. The system's output is given by the label with the largest degree of belief among the three top ranked labels for the red, green and blue components of the input image.

V. Experimental Evaluation

A. Experimental Methodology

1) Dataset

To evaluate the performance of our approaches for traffic sign recognition based on WNN, we used the German Traffic Sign Recognition Benchmark (GTSRB) (http://benchmark.ini.rub.de) [3, 4]. The GTSRB consists of 51,839 images of German traffic signs classified into 43 classes. These images contain a border of about 10% around the actual traffic sign (at least 5 pixels) and the traffic sign is not necessarily centered within the image; a more precise ground truth bounding box of the traffic sign is part of the provided annotations. Image sizes vary between 15×15 and 250×250 pixels. The GTSRB is divided into a training dataset, which contains 39,209 images, and a test dataset, with 12,630 images. Figure 10 shows representatives of the 43 traffic sign classes, which were selected randomly from the GTSRB training dataset.

2) Parameters Search Space

For tuning the parameters of our approaches for traffic sign recognition based on WNN, we generated a training subset and a validation subset, composed respectively of 860 and 430 images randomly selected from the GRSRB training dataset. We trained the networks with the images of the training subset and evaluated their performance in terms of the recognition rate (i.e., the percentage of correctly recognized traffic sign images) on the validation subset, while varying their parameters.

The WNN architecture used in both approaches we studied has six parameters (Section IV.C): (i) the number of neurons, $m \times n$; (ii) the number of synapses per neuron, $|W|$; (iii) the size of the network input, $u \times v$; (iv) the standard deviation, σ, of the two-dimensional Normal distribution followed by the synaptic interconnection pattern of the neurons, Ω; (v) the concentration factor, α, of the log-polar function that maps Φ onto N; and (vi) the standard deviation, σ_w, of the one-dimensional Normal distribution followed by the weights of the neurons' output.

We tested our WNN approaches with: (i) number of neurons equal to 5 × 3, 9 × 5, 18 × 10, 34 × 18, 51 × 27 and 68 × 36; (ii) number of synapses per neuron equal to 8, 16, 32, 64, 128 and 256; (iii) size of the network input equal to 70 × 70 (we did not vary the size of the network input to reduce the parameter search space); (iv) σ equal to 1, 3, 5, 7 and 9; (v) α equal to 2, 4, 6, 8, and 10; and (vi) σ_w equal to 2, 2.5, and 3.

3) WiSARD Traffic Sign Recognition System Tuning

Figure 11 presents the results of the experiments we carried out to tune the parameters of our WiSARD WNN. In the graphs of Figure 11, the x-axis is the number of neurons and the y-axis is the recognition rate; please note that the y-axis scale changes from one graph to another for better visualization. In the legends of the graphs, the first number denotes the number of synapses, the second σ, the third α, and the fourth σ_w.

As Figure 11(a) shows, the performance of our WiSARD system improves as the number of neurons increase; the performance also improves with the number of synapses per neuron, but reaches a plateau at about 128 synapses. Actually, with 128 synapses or more, even with a very little amount of neurons (5 × 3) the performance is very high (about 96%). As we are interested in finding a set of parameters that allows the highest performance but with a reasonable sized architecture, we selected 256 as the number of synapses for the following (a higher number of synapses would not improve the performance).

The graph of Figure 11(b) shows the impact of the standard deviation, σ, of the two-dimensional Normal distribution followed by the synaptic interconnection pattern of the neurons, Ω, on the performance of our WiSARD system. As the graph shows, the system's performance increases with σ, reaching a plateau at about 5; the highest performance is achieved with σ equal to 7. Figure 11(c) shows the impact of the concentration factor, α, of the log-polar function that maps Φ onto N, on the WiSARD system's performance. As the graph in Figure 11(c) shows, in the range of values examined, this parameter does not affect the system's performance much (note the y-axis scale). We selected α equal to 4. Finally, in Figure 11(d), we present the impact of the standard deviation, σ_w, of the one-dimensional Normal distribution followed by the weights of the neurons' output on the WiSARD system's performance. Again, this parameter does not affect the performance much; we select σ_w equal to

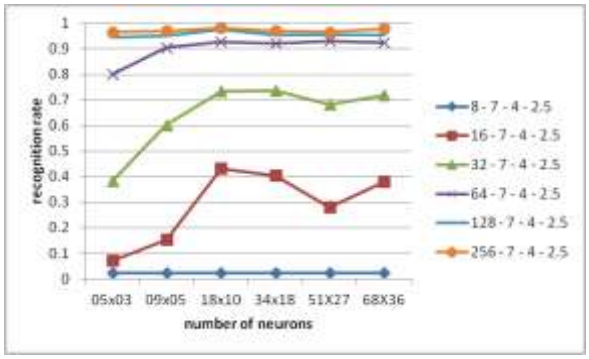
2.5. The highest performance achieved with the WiSARD system was 98.13%, with $m \times n = 18 \times 10$, $|W| = 256$, $\sigma = 7$, $\alpha = 4$ and $\sigma_w = 2.5$.

4) VG-RAM WNN Traffic Sign Recognition System Tuning

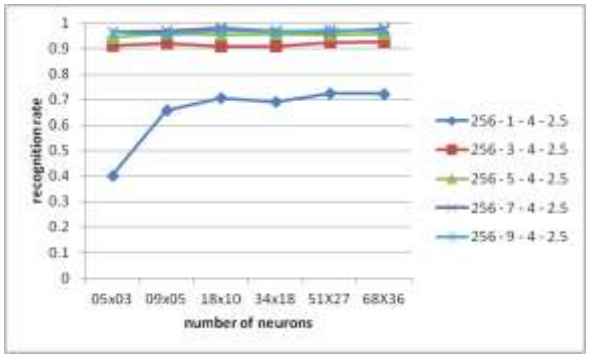
Figure 12 presents the results of the experiments we carried out to tune the parameters of our VG-RAM WNN system. The graphs of Figure 12(a-d) are equivalent to those of Figure 11(a-d) and allow examining the effect of the various parameters of our neural architecture on the VG-RAM WNN system's performance.

As the graphs of Figure 12(a-d) show, an in a way similar to the WiSARD system, the performance of the VG-RAM WNN increases with the number of neurons, the number of synapses, and σ, and it is not much affected by α and σ_w. The highest performance achieved with the VG-RAM WNN system was 99.53%, with $m \times n = 51 \times 27$, $|W| = 64$, $\sigma = 7$, $\alpha = 2$ and $\sigma_w = 2.5$. Again, it is important to note that a much smaller architecture ($m \times n = 5 \times 3$) can achieve a very good performance (about 97%, see Figure 12(d)).

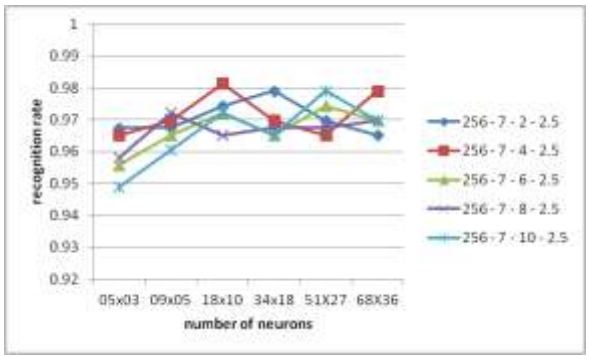
Figure 10: Representatives of the 43 traffic sign classes in the GTSRB.



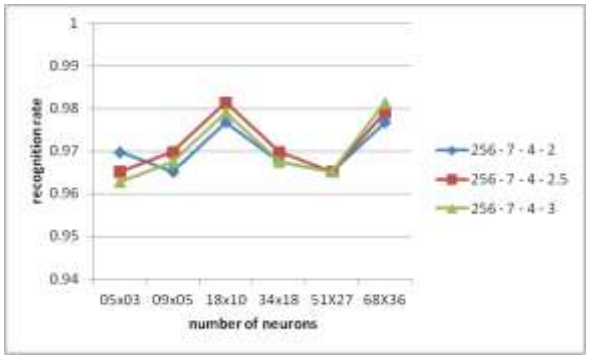
(a)



(b)

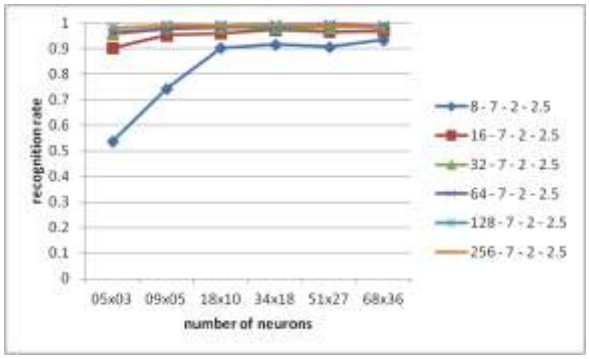


(c)

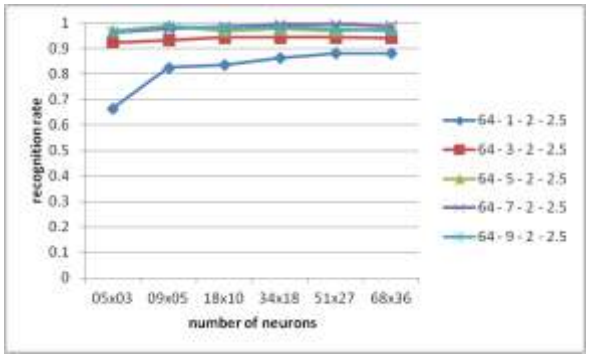


(d)

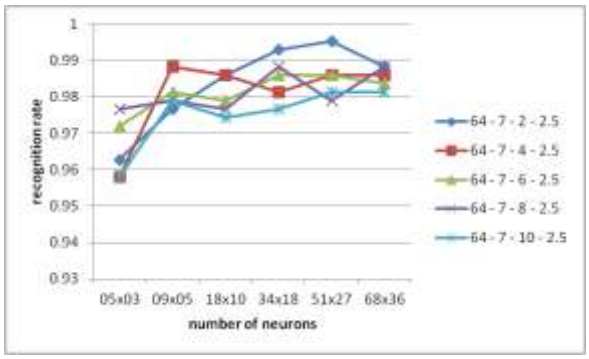
Figure 11: Parameter tuning of our WiSARD WNN.



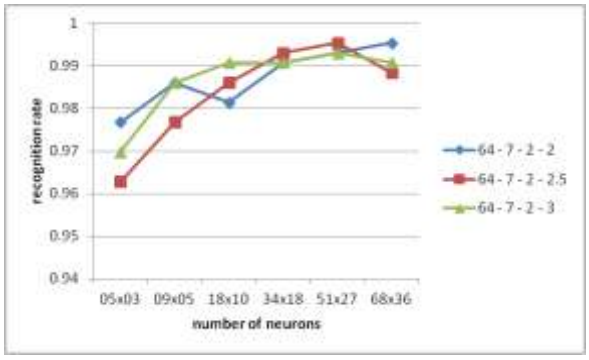
(a)



(b)



(c)



(d)

Figure 12: Parameter tuning of our VG-RAM WNN.

B. Experimental Results

For evaluating the performance of our approaches to traffic sign recognition based on WNN, we set their parameters with the tuned values (WiSARD: 18 × 10 neurons, 256 synapses, σ = 7, α = 4, and σ_w = 2.5; VG-RAM: 51 × 27 neurons, 64 synapses, σ = 7, α = 2, and σ_w = 2.5), trained them with the images in the GTSRB training dataset, and evaluated their performance in terms of recognition rate on the GTSRB test dataset. We submitted the results of both the WiSARD and VG-RAM WNN to the GTSRB website on March 20th 2013. Our WiSARD WNN was ranked ninth in the GTSRB with a recognition rate of 94.74%, while our VG-RAM WNN achieved a higher performance, being ranked fifth in the GTSRB with a recognition rate of 98.42%. It is important to note that the third and fourth places were human classifiers. Figure 13 show the results of these submissions for the WiSARD and VG-RAM WNN systems.

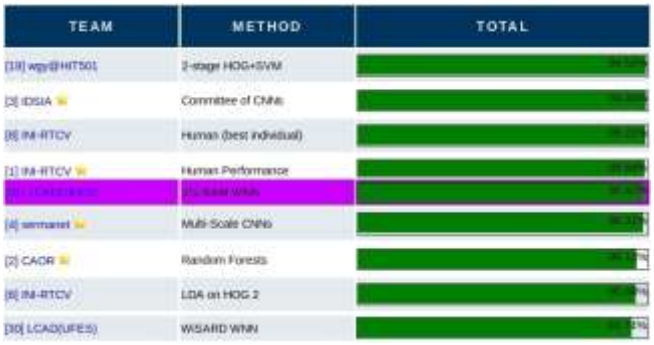


Figure 13: Result of the submission of our WiSARD and VG-RAM WNN's results to GTSRB on March 20th 2013.

VI. Conclusions and Future Work

In this paper, we present two biologically inspired approaches to traffic sign recognition based on WNN: one using VG-RAM neurons and the other using WiSARD neurons. We evaluated the performance of both approaches on the German Traffic Sign Recognition Benchmark (GTSRB). Our system based on VG-RAM neurons outperformed the one based on WiSARD neurons and was ranked fifth in the GTSRB (the third and fourth places were human classifiers) with a recognition rate of 98.42%.

As directions for future work, we plan to evaluate the performance of our VG-RAM WNN system on traffic signs of Brazil's road environment, which we expect will be more challenging, and the real time performance of both the VG-RAM WNN system and the WiSARD system. We believe that the systems' performances can be improved further by using Bayesian inference over several images of the same traffic sign acquired in sequence, as would be possible in a real time system.

Acknowledgment

We would like to thank Conselho Nacional de Desenvolvimento Científico e Tecnológico-CNPq-Brasil (grants 552630/2011-0, 308096/2010-0, and 314485/2009-0) and Fundação de Amparo à Pesquisa do Espírito Santo-FAPES-Brasil (grant 48511579/2009) for their support to this research work.

References

[1] I. Aleksander, "From WISARD to MAGNUS: A family of weightless virtual neural machines", in *RAM-Based Neural Networks*, J. Austin, (ed.), Singapore: World Scientific, pp. 18-30, 1998.

[2] I. Aleksander, W. V. Thomas, and P. A. Bowden, "WISARD: a radical step forward in image recognition", *Sensor Review*, vol. 4, no. 3, pp. 120-124, 1984.

[3] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The german traffic sign recognition benchmark: a multi-class classification competition", in *Proceedings of the 2011 International Joint Conference on Neural Networks*, pp. 1453-1460, 2011.

[4] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: benchmarking machine learning algorithms for traffic sign recognition", *Neural Networks*, vol. 32, pp. 323-332, 2012.

[5] D. Gavrila, "Traffic sign recognition revisited", in *Proceedings of the 21 DAGM Symposium fur Mustererkennun*, pp. 86-93,1999.

[6] Y. Li, "Real-time traffic sign detection: an evaluation study", in *Proceedings of the 20th International Conference on Pattern Recognition (ICPR)*, pp. 3033-3036, 2010.

[7] S. Escalera, X. Baró, O. Pujol, J. Vitrià, and P. Radeva, *Traffic Sign Recognition Systems*, Springer, 2011.

[8] Y.-Y. Nguwi and A. Z. Kouzani, "Detection and classification of road signs in natural environments", *Neural Computing and Applications*, vol. 17, no. 3, pp. 265-289. 2008.

[9] S. Vitabile, G. Pollaccia, G. Pilato, and F. Sorbello, "Road signs recognition using a dynamic pixel aggregation technique in the HSV colour space", in *Proceedings of the International Conference on Image Analysis and Processing*, pp. 572-577, 2001.

[10] A. de la Escalera, J. M. Armingol, and M. Mata, "Traffic sign recognition and analysis for intelligent vehicles", *Image and Vision Computing*, vol. 21, pp. 247-258, 2003.

[11] M. Lalonde and Y. Li, "Road sign recognition-survey of the state of art", Technical Report for Sub-Project 2.4, Centre de Recherche Informatique de Montreal, CRIM-IIT-95/09-35, 1995.

[12] H. Fleyeh, S. Gilani, and M. Dougherty, "Road sign detection and recognition using Fuzzy ARTMAP: a case study swedish speed-limit signs", in *Proceedings of Artificial Intelligence and Soft Computing*, pp. 242-249, 2006.

[13] A. Ruta, Y. Li, and X. Liu, "Real-time traffic sign recognition from video by class-specific

discriminative features", *Pattern Recognition*, vol. 43, pp. 416-430, 2010.

[14] L. Priese, J. Klieber, R. Lakmann, V. Rehrmann, and R. Schian, "New results on traffic sign recognition", in *Proceedings of the Intelligent Vehicles Symposium*, pp. 249-254, 1994.

[15] D. M. Gavrila and V. Philomin, "Real-time object detection for smart vehicles", in *Proceedings of IEEE International Conference on Computer Vision*, pp. 87-93, 1999.

[16] P. Paclik and J. Novovicova, "Road sign classification without colour information", in *Proceedings of 6th Conference of Advanced School of Imaging and Computing*, pp. 84-90, 2000.

[17] G. Loy and N. Barnes, "Fast shape-based road sign detection for a driver assistance system", in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol 1, pp. 70-75, 2004.

[18] J. Torresen, J. Bakke, and L. Sekanina, "Efficient recognition of speed limit signs", in *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems*, pp. 652-656, 2004

[19] N. Barnes, A. Zelinsky, and L.S. Fletcher, "Real-time speed sign detection using the radial symmetry detector" *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 2, pp. 322-332, 2008.

[20] M. C. Kus, M. Gokmen, and S. Etaner-Uyar, "Traffic sign recognition using scale invariant feature transform and color classification", in *Proceedings of the 23rd International Symposium on Computer and Information Sciences*, pp. 1-6, 2008.

[21] J. Jiao, Z. Zheng, J. Park, Y. L. Murphey, and Y. Luo, "A robust multi-class traffic sign detection and classification system using asymmetric and symmetric features", in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pp. 3421-3427, 2009.

[22] G. Overett and L. Petersson, "Large scale sign detection using HOG feature variants", in *Proceedings of the 2011 IEEE Intelligent Vehicles Symposium*, pp. 326-331, 2011.

[23] Y. Aoyagi and T. Asakura, "Detection and recognition of traffic sign in scene image using genetic algorithms and neural networks", in *Proceedings of the 35th SICE Annual Conference*, pp. 1343-1348, 1996.

[24] L. Estevez and N. Kehtarnavaz, "A real-time histographic approach to road sign recognition", in *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 95-100, 1996.

[25] A. de la Escalera and P. Radeva, "Fast greyscale road sign model matching and recognition", *Recent Advances in Artificial Intelligence Research and Development*, J. Vitrià, P. Radeva, I. Aguiló (eds.), IOS Press, pp. 69-76, 2004.

[26] S. Maldonado-Bascon, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gomez-Moreno, and F. Lopez-Ferreras, "Road-sign detection and recognition based on support vector machines", *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 264-278, 2007.

[27] R. Timofte, K. Zimmermann, and L. Van Gool, "Multi-view traffic sign detection, recognition, and 3D localisation", in *Proceedings of the Workshop on Applications of Computer Vision*, pp. 1-8, 2009.

[28] X. Baro, S. Escalera, J. Vitria, O. Pujol, and P. Radeva, "Traffic sign recognition using evolutionary adaboost detection and forest-ECOC classification", *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 1, pp.113-126, 2009.

[29] D. Ciresan, U. Meier, J. Masci, and J. Schmidhuber, "A committee of neural networks for traffic sign classification", in *Proceedings of the 2011 International Joint Conference on Neural Networks*, pp. 1918-1921, 2011.

[30] P. Sermanet and Y. LeCun, "Traffic sign recognition with multiscale convolutional networks", in *Proceedings of the 2011 International Joint Conference on Neural Networks*, pp. 2809-2813, 2011.

[31] F. Zaklouta, B. Stanciulescu, and O. Hamdoun, "Traffic sign classification using k-d trees and random forests", *Proceedings of the 2011 IEEE International Joint Conference on Neural Networks*, pp. 2151-2155, 2011.

[32] M. Berger, A. F. Silva, A. F. De Souza, J. Oliveira Neto, L. P. Veronese, C. Badue, "Traffic sign recognition with VG-RAM weightless neural networks", in *Proceedings of the 12th International Conference on Intelligent Systems Design and Applications*, pp. 315-319, 2012.

[33] E. R. Kandel, J. H. Schwartz, and T. M. Jessell, *Principles of Neural Science*, 4th ed., McGraw-Hill, 2000.

[34] R. B. Tootell, M. S. Silverman, E. Switkes, and R. L. Valois, "Deoxyglucose analysis of retinotopic organization in primate striate córtex", *Science*, vol. 218, pp. 902-904, 1982.

[35] I. Aleksander, "Self-adaptive universal logic circuits", *IEEE Electronic Letters*, vol. 2, no. 8, pp. 231-232, 1966.

[36] M. N. A. Punjani and D. J. Fleet, "Fast search in Hamming space with multi-index hashing", in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3108-3115, 2012.

[37] T. B. Ludermir, A. C. P. L. F. Carvalho, A. P. Braga, and M. D. Souto, "Weightless neural models: a review of current and past works", *Neural Computing Surveys*, vol. 2, pp. 41-61, 1999.

[38] R. J. Mitchell, J. M. Bishop, S. K. Box, and J. F. Hawker, "Comparison of some methods for

processing grey level data in weightless networks", *RAM-based Neural Networks*, J. Austin (ed.), Singapore: World Scientific, pp. 61-70, 1998.

[39] K. Zuiderveld, "Contrast limited adaptive histogram equalization", *Graphics Gems IV*, P. S. Heckbert (ed.), San Diego:Academic Press Professional, Inc., pp. 474-485, 1994.

Author Biographies

Mariella Berger was born in Colatina, ES, Brazil, on January 18, 1982. She received the B.Sc. degree in computer science in November 2004 and the M.Sc. degree in computer science in August 2007, both from the Universidade Federal do Espírito Santo (UFES), in Vitória, ES, Brazil. Nowadays, she is a Ph.D. student and member of the Laboratório de Computação de Alto Desempenho (LCAD – High Performance Computing Laboratory), both at UFES.

Avelino Forechi was born in Colatina, ES, Brazil, on August 26, 1978. He received the B.Sc. in computer science from Universidade Federal de Viçosa (UFV), in Viçosa, MG, Brazil, in 2003 and the M.Sc. degree in computer science from Universidade Federal do Espírito Santo (UFES), in Vitória, ES, Brazil, in 2012. Nowadays, he is a PhD student and member of the Laboratório de Computação de Alto Desempenho (LCAD – High Performance Computing Laboratory), both at UFES.

Alberto F. De Souza was born in Cachoeiro de Itapemirim, ES, Brazil, on October 27, 1963. He received the B. Eng. (Cum Laude) degree in electronics engineering and M.Sc. in systems engineering in computer science from Universidade Federal do Rio de Janeiro (COPPE/UFRJ), in Rio de Janeiro, RJ, Brazil, in 1988 and 1993, respectively; and Doctor of Philosophy (Ph.D.) in computer science from the University College London, in London, United Kingdom, in 1999. He is a professor of computer science and coordinator of the Laboratório de Computação de Alto Desempenho (High Performance Computing Laboratory) at the Universidade Federal do Espírito Santo (UFES), in Vitória, ES, Brazil. He has authored/co-authored one USA patent and over 90 publications. He has edited proceedings of four conferences (two IEEE sponsored conferences), is a Standing Member of the Steering Committee of the International Conference in Computer Architecture and High Performance Computing (SBAC-PAD), Senior Member of the IEEE, and Comendador of the Order of Rubem Braga.

Jorcy de Oliveira Neto was born in Linhares, ES, Brazil, on August 1, 1989. He received the B.Sc. degree in computer engineering in 2011 from the Departamento de Informática of the Universidade Federal do Espírito Santo (UFES), in Vitória, ES, Brazil. Nowadays, he is working as an Electronic Equipment Engineer at Petrobras, Rio de Janeiro, RJ, Brazil.

Lucas was born in Natividade, RJ, Brazil, on March 07, 1985. He received the B.Sc. degree in computer science in 2008 and the M.Sc. degree in computer science in 2011, both from the Universidade Federal do Espírito Santo (UFES), in Vitória, ES, Brazil. Nowadays, he is a PhD. student and member of the Laboratório de Computação de Alto Desempenho (LCAD – High Performance Computing Laboratory), both at UFES.

Victor Neves was born in Vitória, ES, Brazil, on May 27, 1988. He received the B.Sc. degree in computer science in 2012 from the Departamento de Informática of the Universidade Federal do Espírito Santo (UFES), in Vitória, ES, Brazil. Nowadays, he is working as a System Analyst in Vitória, ES, Brazil.

Edilson de Aguiar was born in Vila Velha, ES, Brazil, on June 11, 1979. In March 2002, he received the B.Sc. degree in computer engineering from the Universidade Federal do Espírito Santo (UFES), in Vitória, ES, Brazil. He received the M.Sc. degree in computer science in December 2003 and the Ph.D. degree in computer science in December 2008, both from the Saarland University and the Max-Planck Institute for Computer Science, in Saarbrücken, Saarland, Germany. After that he worked as researcher from 2009 to 2010 at the Disney Research laboratory in Pittsburgh, USA. Since then, he has been with the Departamento de Computação e Eletrônica of UFES, in São Mateus, ES, Brazil, where he is an Associate Professor and researcher at the Laboratório de Computação de Alto Desempenho (LCAD – High Performance Computing Laboratory). His research interests are in the areas of Computer Graphics, Computer Vision, Image Processing and Robotics. He has been involved in research projects financed through Brazilian research agencies, such as State of Espírito Santo Research Foundation (Fundação de Apoio a Pesquisa do Estado do Espírito Santo – FAPES). He has also been in the program committee and organizing committee of national and international conferences in computer science.

Claudine Badue was born in Anápolis, GO, Brazil, on November 26, 1975. In March 1998, she received the B.Sc. degree in computer science from the Instituto de Informática of the Universidade Federal de Goiás (UFG), in Goiânia, GO, Brazil. She received the M.Sc. degree in computer science in March 2001 and the Ph.D. degree in computer science in February 2007, both from the Departamento de Ciência da Computação of the Universidade Federal de Minas Gerais (UFMG), in Belo Horizonte, MG, Brazil. Since then, she has been with the Departamento de Informática of the Universidade Federal do Espírito Santo (UFES), in Vitória, ES, Brazil, where she is an Associate Professor and researcher at the Laboratório de Computação de Alto Desempenho (LCAD – High Performance Computing Laboratory). Her research interests are in the areas of Artificial Cognitive Vision, Robotics, Information Retrieval and High-Performance Computing. She has been involved in research projects financed through Brazilian research agencies, such as Brazilian National Council for Scientific and Technological Development (Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq) and State of Espírito Santo Research Foundation (Fundação de Apoio a Pesquisa do Estado do Espírito Santo – FAPES). She has also been in the program committee and organizing committee of national and international conferences in computer science.