

MIT Open Access Articles

Optimization of a Hardware Implementation for Pulse Coupled Neural Networks for Image Applications

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

Citation: Sarciada, Jesus Gimeno, Horacio Lamela Rivera, and Cardinal Warde. "Optimization of a hardware implementation for pulse coupled neural networks for image applications." Independent Component Analyses, Wavelets, Neural Networks, Biosystems, and Nanoengineering VIII. Ed. Harold H. Szu & F. Jack Agee. ©2010 SPIE

As Published: <http://dx.doi.org/10.1117/12.850778>

Publisher: SPIE

Persistent URL: <http://hdl.handle.net/1721.1/58611>

Version: Final published version: final published article, as it appeared in a journal, conference proceedings, or other formally published context

Terms of Use: Article is made available in accordance with the publisher's policy and may be subject to US copyright law. Please refer to the publisher's site for terms of use.



Optimization of a hardware implementation for Pulse Coupled neural Networks for image applications

Jesús Gimeno Sarciada^a, Horacio Lamela Rivera^a, Cardinal Warde^b

^aGrupo de trabajo de Optoelectrónica y Tecnología Láser de la Universidad Carlos III de Madrid
c/ Butarque 15 28911 Leganés, Madrid, Spain

^bEECS Department, MIT, 77 Massachusetts Avenue Cambridge, MA USA 02139-4307

ABSTRACT

Pulse Coupled Neural Networks are a very useful tool for image processing and visual applications, since it has the advantages of being invariant to image changes as rotation, scale, or certain distortion. Among other characteristics, the PCNN changes a given image input into a temporal representation which can be easily later analyzed for pattern recognition. The structure of a PCNN though, makes it necessary to determine all of its parameters very carefully in order to function optimally, so that the responses to the kind of inputs it will be subjected are clearly discriminated allowing for an easy and fast post-processing yielding useful results. This tweaking of the system is a taxing process.

In this paper we analyze and compare two methods for modeling PCNNs. A purely mathematical model is programmed and a similar circuitual model is also designed. Both are then used to determine the optimal values of the several parameters of a PCNN: gain, threshold, time constants for feed-in and threshold and linking leading to an optimal design for image recognition. The results are compared for usefulness, accuracy and speed, as well as the performance and time requirements for fast and easy design, thus providing a tool for future ease of management of a PCNN for different tasks.

Keywords: Pulse Coupled Neural Networks, PCNN, image processing, Hardware implementation

1. INTRODUCTION

Based on his experiments with the visual cortex of cats, Eckhorn¹ and his team have proposed a neural model that more closely resembles biological neurons that typically use artificial neural models. His model has proved capable of reproducing some of the responses observed in the cortex of cats. For the Eckhorn neuron design to be useful in image processing applications, Johnson and Padgett², modified the design of the original model removing of the existing constraints. This amended neuron model is called Pulse Coupled Neuron (PCN).

The Pulse Coupled Neuron is a nonlinear integrating system, which consists of three functional units: a feeding field formed by the external input and outputs of the neurons in its field, connected through a leaky integrator, a linking field, which modulates the feeding field to form the internal activity of the neuron and a spike generator formed by a threshold generator and a pulse function generator. The schematic of a Pulse Coupled Neuron is shown in Figure 1.

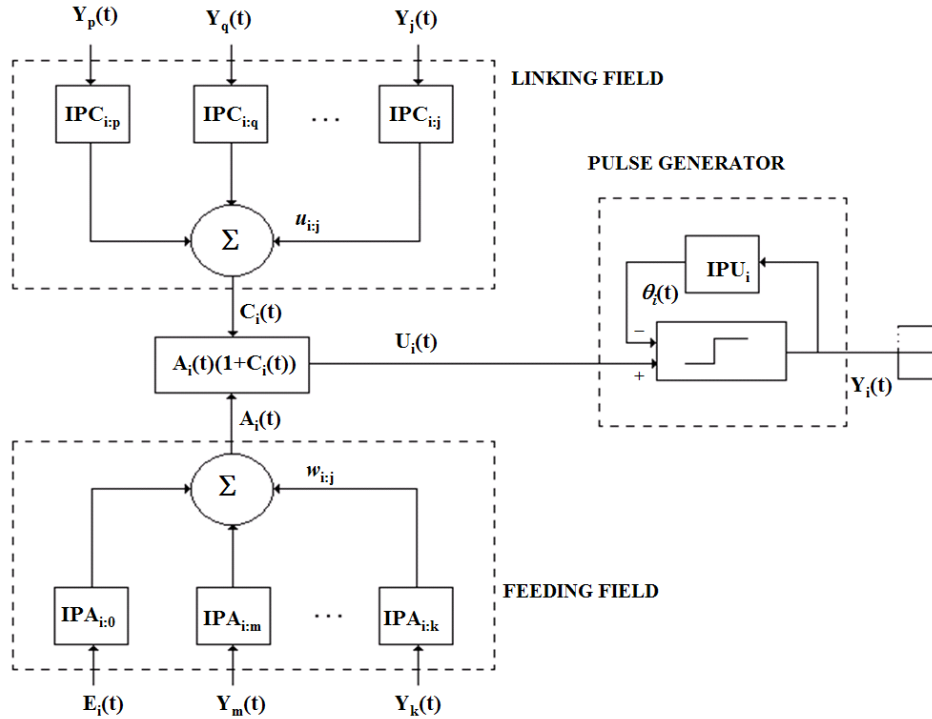


Figure 1. Design of a Pulse Coupled Neuron. The internal activity $U_i(t)$ comprises the feeding field $A_i(t)$ modulated through the linking field $C_i(t)$. A spike is generated by the impulse function generator, whenever the internal activity $U_i(t)$ is greater than the threshold $\theta_i(t)$.

Each time the Pulse Coupled Neuron pulses, its threshold signal generator is loaded to a preset value regardless of the value it had prior to the pulsing of the neuron.

$$\theta_i(t) = \begin{cases} V_i & t = t_1, \\ V_i e^{-(t-t_1)/\tau_i} & t_1 < t < t_2, \\ V_i & t = t_2 \end{cases} \quad (1)$$

If we take two neurons connected to each other and with constant inputs and make them pulse simultaneously at $t = 0$, then, in a given time corresponding to its natural frequency, the neuron N_i with the biggest external input will pulse. This pulse is transmitted by the linking or feeding field to the other neuron N_j , so that the internal activity increases. If, thanks to this increase, the internal activity exceeds the θ_j and pulses, it is said that this neuron has been captured by the other and both neurons continue pulsing together indefinitely if the system remains unchanged.

These characteristics of the PCNN make such a network useful for tasks such as segmentation³, where it performs better separating alike pixels in an image than many other methods. This segmentation capability allows in turn for other applications as object⁴ and movement⁵ detection leading to improved image classification⁶. Before a direct hardware implementation of a PCNN is built, it is convenient to analyze the operation that such a network with certain characteristics will have and if it fits the specifications sought for the system. This is especially desirable given the high number of parameters that need to be adjusted.

This article will present a mathematical model of a simplified Pulse Coupled Neuron and a hardware model of the same neuron and the behavior of both will be simulated, showing the similarity of the responses and the consistency of the simulation parameters. In Section 2 we study the mathematical model of the simplified neuron and the output of a neural network. Section 3 presents the hardware implementation of the same design and the response of both, an individual neuron and a whole network. Section 4 presents the conclusions reached in this work.

2. MATHEMATICAL MODEL

The goal is to obtain a neuron model that is able to faithfully provide the results obtained subsequently when physically implemented, so that all desirable tests can be performed, knowing that the results can be reproduced by building a physical circuit that uses the already known design.

The equations that implement the design shown in Figure 1, are simple to program. It is only necessary to define the operation as discrete, instead of continuous. The equations are then:

$$\begin{aligned}
 A_{ij}(t) &= e^{-\alpha_a \Delta t} A_{ij}(t-1) + A_{ij,0} + \sum_{k,l} w_{ijkl} Y_{kl}(t-1) \\
 C_{ij}(t) &= e^{-\alpha_c \Delta t} C_{ij}(t-1) + \sum_{k,l} u_{ijkl} Y_{kl}(t-1) \\
 U_{ij} &= A_{ij}(t)(1 + \beta C_{ij}(t)) \\
 Y_{ij}(t) &= \begin{cases} 1 & \text{si } U_{ij}(t) \geq \Theta_{ij}(t) \\ 0 & \text{En caso contrario} \end{cases} \\
 A_{ij}(t) &= e^{-\alpha_e \Delta t} \Theta_{ij}(t-1) + VY_{ij}(t-1)
 \end{aligned} \tag{2}$$

The first step when considering a design of Pulse Coupled Neurons is to choose the type of connectivity that the neurons will have between themselves, since the most important feature of these neurons is precisely the ability to capture others in their feeding or linking field, so the pulses of the neurons couple. When choosing the design characteristics of a PCNN, we have chosen a simple interconnection model, in which surrounding neurons in a radius '1', i.e. the four neurons of up, down, left and right have its output connected to the input of the central neuron, adding to the external input.

In this design, the effect of surrounding neurons is limited to the feeding field, leaving the linking field void, thus leaving the feeding field as the internal activity of the neuron. Therefore, the individual linking coefficients and of total linking coefficient are not necessary. All neurons have the same feeding coefficient. That is, none of the surrounding neurons have greater effect than any other. Similarly the time constants of all neurons are identical. Therefore, the only parameters to model in neurons of our network are: the time constant of the feeding field, the time constant of the threshold generator, the feeding coefficient and threshold values. The result of the mathematical model of the modified design of Pulse Coupled Neuron, gives the response shown in Figure 2. It can be seen that the pulse frequency changes according to the value of the entry, which is a key feature of the neurons.

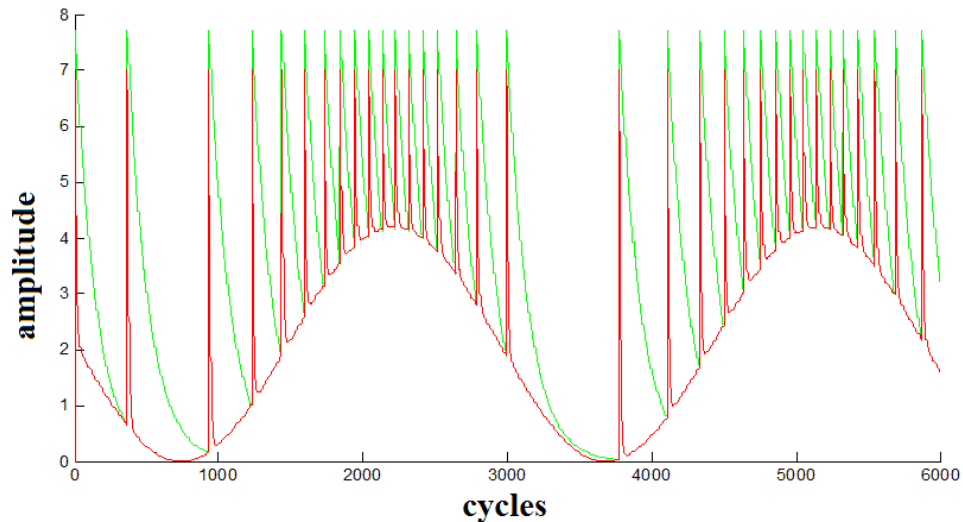


Figure 2. Response of a pulse coupled neuron to a sine input. The pulse frequency increases with the intensity of the external input, while the value of the pulse remains constant.

The mathematical model of the individual neuron is, nevertheless, be the easiest part to develop, since the proper interconnection between neurons is quite complex to perform and requires repeated adjustments of the interconnection weights. These adjustments are complex, because the mathematical equations allow solutions that would be easy to avoid or outright impossible in the real circuit. A clear example shows in Figure 3, where the neural network and the internal parameters cause the value of the internal activity of the neuron to exceed the reference value, thus the pulse occurs, the neuron reduces its internal activity, rather than increase it.

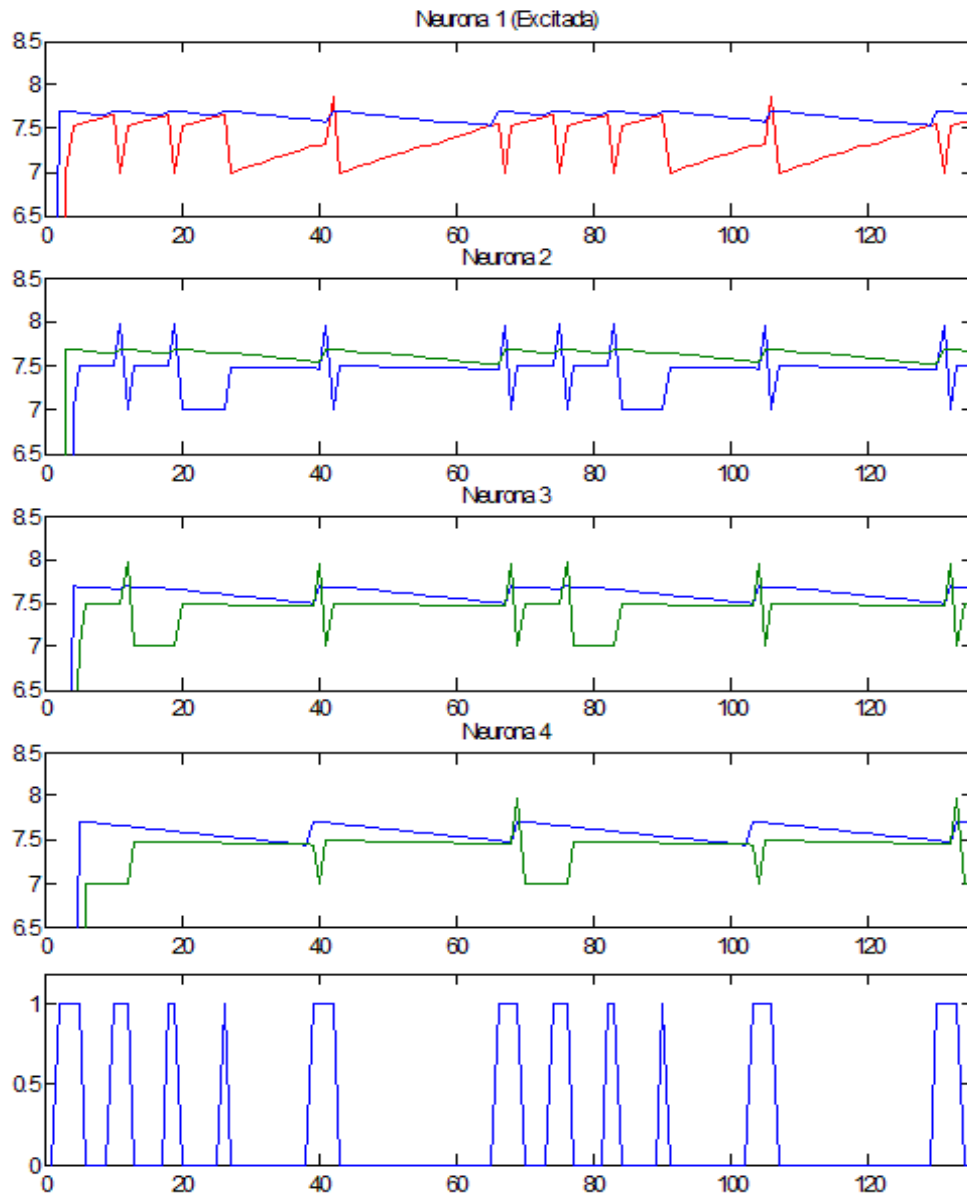


Figure 3. Response of a PCNN formed by four neurons in a row with invalid parameters.

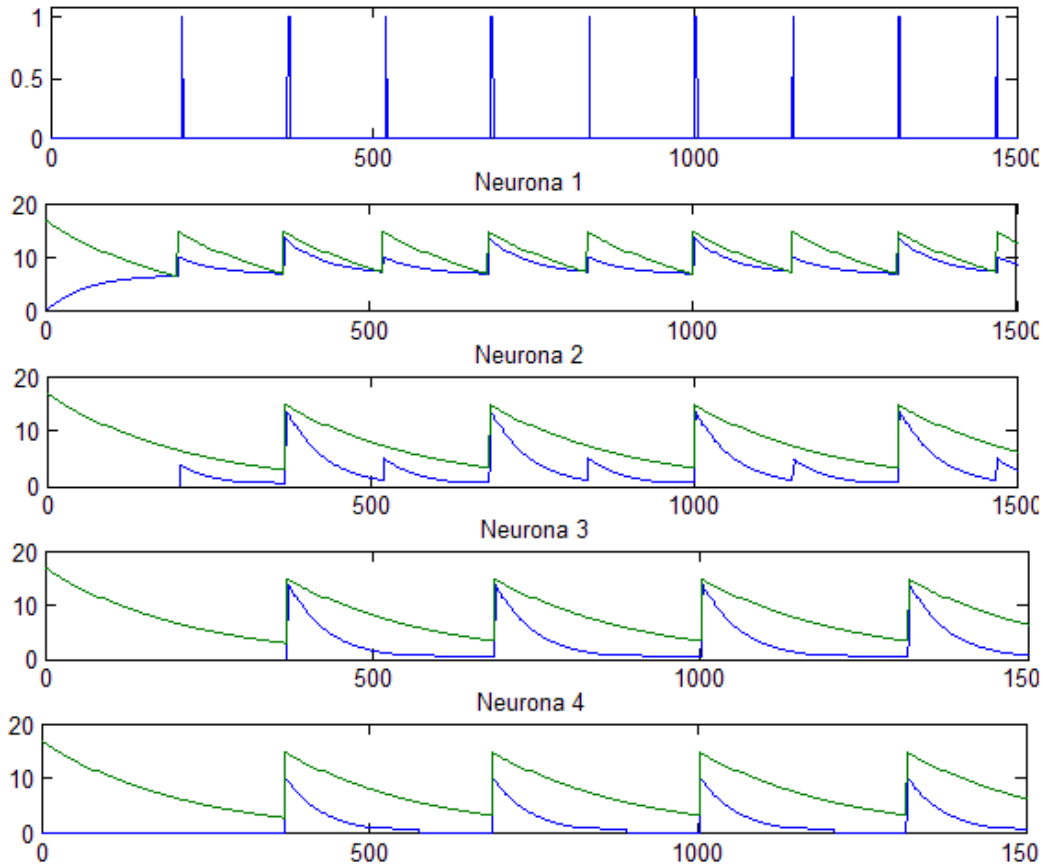


Figure 4. Response of a PCNN formed by four neurons in a row with valid parameters.

For this reason, it is necessary to carefully adjust the model to obtain the results in Figure 4, more suitable for the study of PCNNs, since they do not exhibit the problems described above.

3. DIGITAL CIRCUIT

The schematic of this neuron design, shown in Figure 5, which has the same characteristics as discussed in the previous section in the software simulation is based on the design of Ota and Wilamowski⁷ and consists of a leaky integrator characterized by resistor R1 and capacitor C1, which represents the internal activity of the neuron and other leaky integrator, characterized by R2 and C2, which serves as the threshold. The comparison between the two is made by a transistor with its base connected to the leaky integrator of the feeding field and the emitter to the threshold.

Thus, when the voltage corresponding to the internal activity overcomes the threshold voltage plus the base-emitter voltage of transistor Q1, the transistor changes state to the active region, closing the circuit, so that the current flows in the circuit producing a pulse. This current is provided by mirror of the current that is activated upon activation of the first transistor.

The current mirror formed by Q2 and Q3, feeds simultaneously both sides of the circuit, producing on the side of the internal activity the output pulse, since the connection with the surrounding neurons is done in that side, and on the other side, the threshold increases which raises the threshold value above the internal activity.

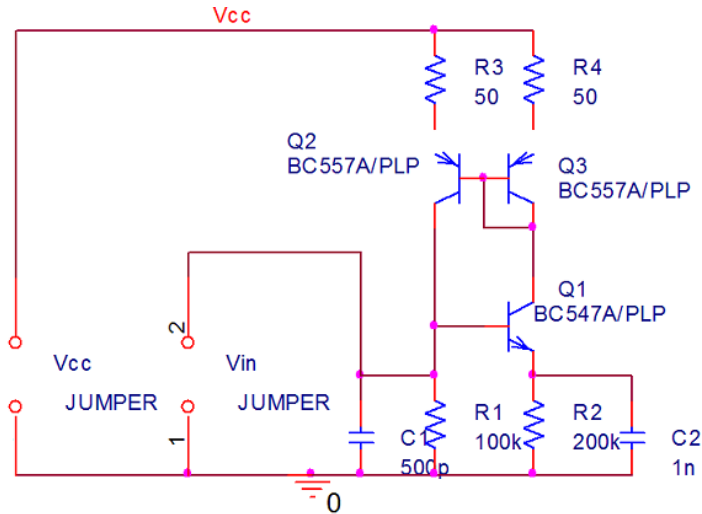


Figure 5. Electronic circuit of the Pulse Coupled Neuron. The parallel R1C1 circuit forms the feeding field, the parallel R2C2 circuit forms the threshold generator and the transistor Q1 is the pulse function generator, comparing the voltages of the feeding and threshold. The current mirror Q2 and Q3 provide the current when the neuron pulses.

This design, although not conforming exactly to the equations described above, has certain advantages that make it well suited for studying these neurons. First, it is simple, easy to make and model, with a limited number of parameters (the values of R and C, which give the feeding field and threshold time constants and interconnection resistances with other neurons) and which nevertheless meets the fundamental characteristics theory.

Second, it has the great ability to generate easily, both excitatory and inhibitory inputs. The only difference between them lies in the connection point of the neuron to the signals. An excitatory signal increases the value of internal activity and accordingly, it approaches the value of the threshold. An inhibitory signal reduces the internal activity and increases the distance to the threshold. This can be accomplished also by raising the threshold value. Thus, if the signals are connected at the input side, raise the internal activity and are considered excitatory. On the other hand, connected on the side of the threshold, raise its value, which is equivalent to reducing internal activity and are considered inhibitory. Since the time constant of the threshold is higher than the input field, the effect of inhibitory signals is longer than the excitation, as they will decay slower.

If we want to limit the effect of inhibitory signals, a way of doing this is by reducing the intensity of inhibitory pulses, limiting the current. The temporary effect would be longer, but their power would be limited.

Figure 6 shows the response of the neuron to a sinusoidal input. We see how the pulse frequency increases with the intensity of the input, while the pulse amplitude remains constant at all times. This response is identical to that obtained in the previous section in the software simulation of Figure 2. The only difference comes from the base-emitter voltage, for the reasons explained previously. If the base-emitter voltage had been modeled in hardware design, the response of both systems would be identical in all respects. It is precisely this equality of both responses what confirms that the design behaves in accordance with the theoretical behavior which allows us to take this neuron model as a basis for implementing a complete PCNN.

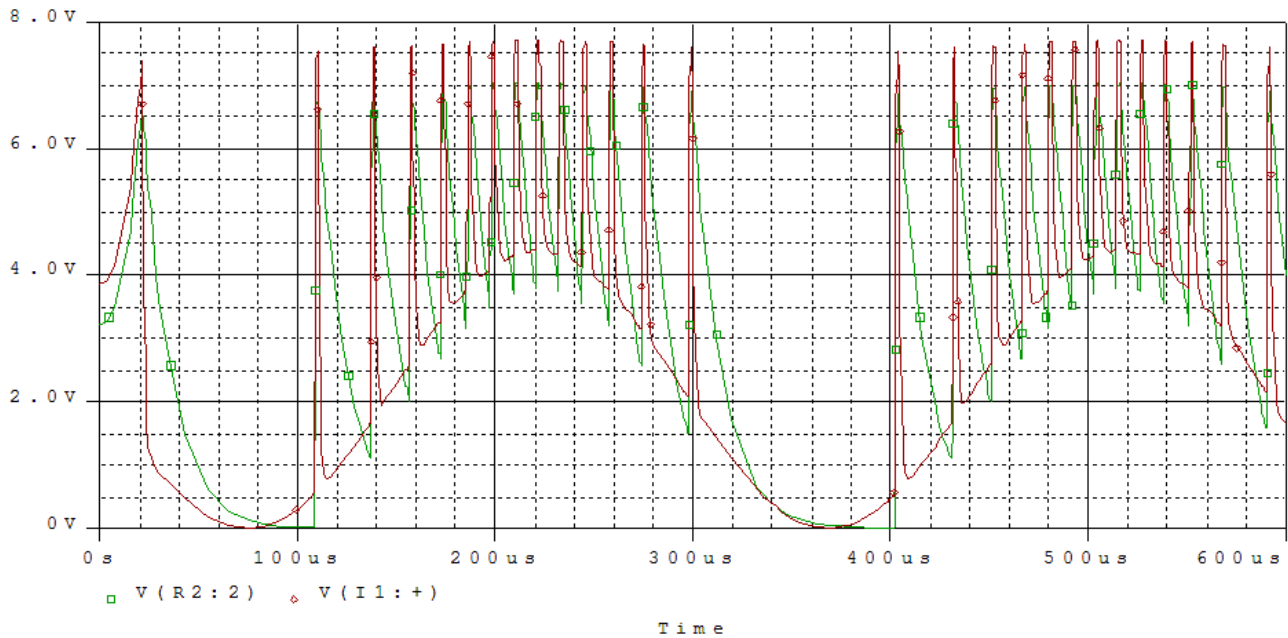


Figure 6. Response of the circuit pulse coupled neuron to a sine input. The response is similar to that of the simulated neuron, except for the V_{BE} offset of the feeding value.

The design of the interconnection between neurons is the one shown in Figure 7, which includes an output buffer and a coupling resistance for each neuron, which allows the interconnection current to come not from the current mirror, but be independent and the same for each neuron interconnection, thus making the position of an object in any position of the image not modify the response⁴.

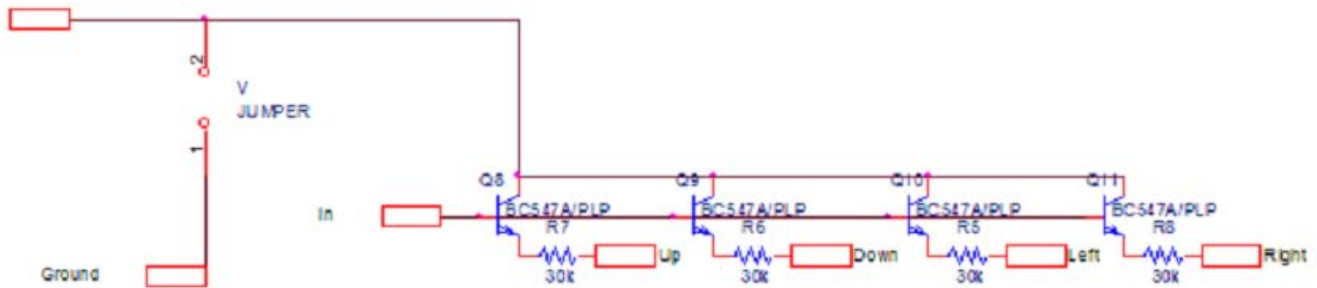


Figure 7. Electronic circuit of the coupling between neurons. There is a transistor for providing the current to each connected neuron and a resistance to limit the effect.

The coupling resistance gives the value of feeding coefficient.

We have created 8x8 pixel images, according to the schematic shown in Figure 8, with single-bit codification. The reason for a binary coding is that the first conclusions that are to be drawn from the network responses concern, not the possible classifications of the images but the capacity of the network to respond in the same way to the same input image regardless of position and angle.

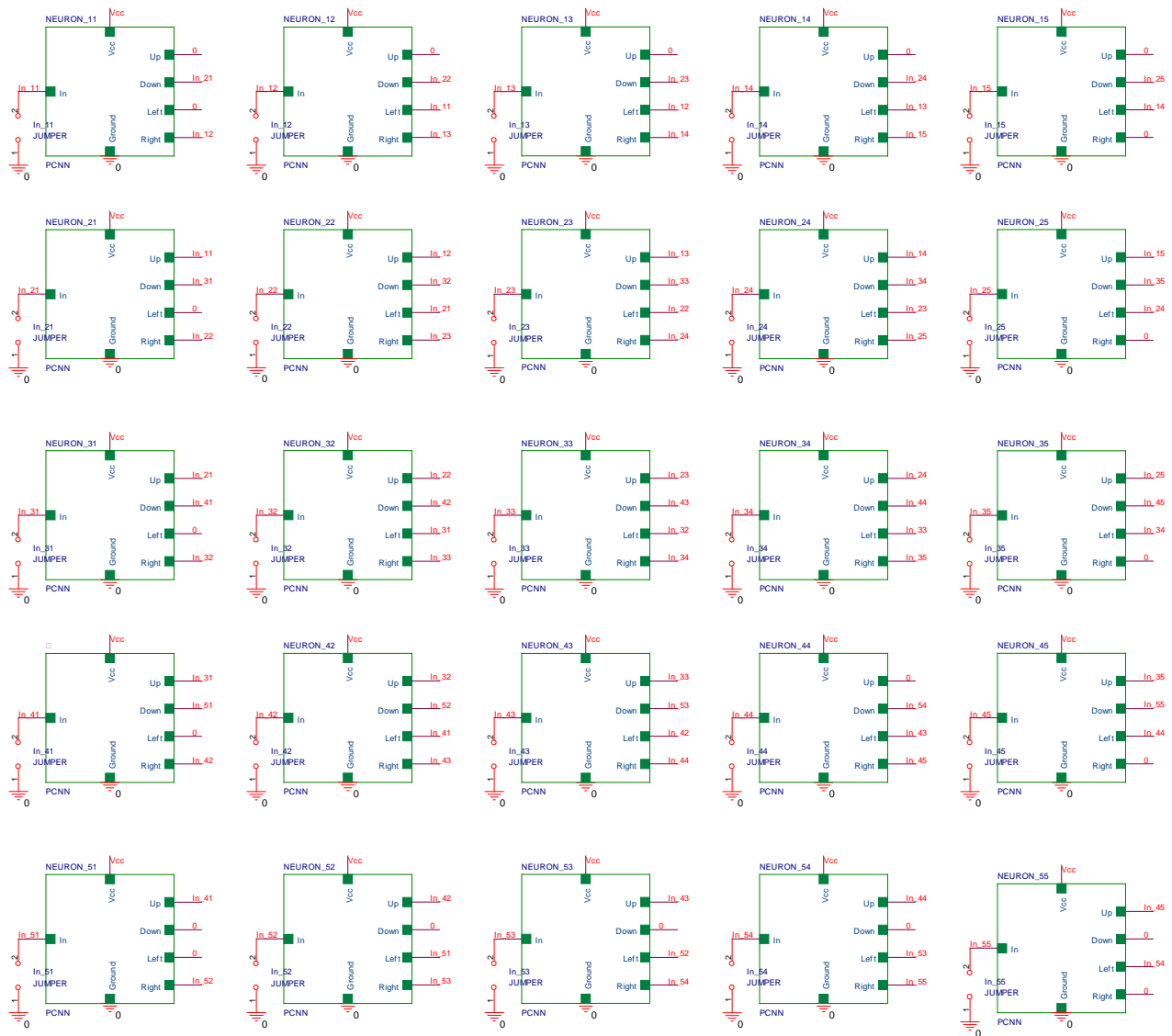


Figure 8. Part of the neuron array used for testing the response of the neural network.

Figures 9, 10 and 10 show images introduced in the circuit and its response for the first 500 μ s. The input images consist of arrays of 8x8 pixels, coded in black and white. The images are color inverted, i.e. a black dot represents a '1' and a white pixel represents a '0'. The response is the weighted sum of the output voltages with the amplitude changing 1V for each neuron that fires and the graph ranging from 0V to 64V which would indicate that all neurons in the network have fired simultaneously, so that the peaks observed represent the number of neurons that jump at a certain point and width of the peaks, the propagation time of pulses. The images have been introduced as input parameters in the circuit and are constant in time.

These results show that even if they could not allow us to classify the input images, they could provide an output invariant to position and rotation⁸ that could be classified by means of another system that did not have these characteristics. The output patterns suggest that also in the case of image scaling, the outcome of the neural network would be similar to the larger or smaller control image, since the only parameter due to change would be the amplitude of the signal, which could be normalized, thus making it a system invariant to variations in size⁹. That could be also the case for intensity variations, where the frequency of the pulses would change, but not the relative rate of pulses. The next step is to obtain individual patterns for each input signal that serve to classify the images directly from the output of the

PCNN using a neural network¹⁰. Some results are observed in figure 10, which also can be compared with those of Figure 9 and indicate a pattern-output relationship.

It should be noted, however, that the response to the inputs of Figure 11 seems to show that this relationship is not univocal, since in this example, similar patterns lead to identical temporary responses, so the pattern thus obtained is not sufficient by itself to provide a satisfactory classification.

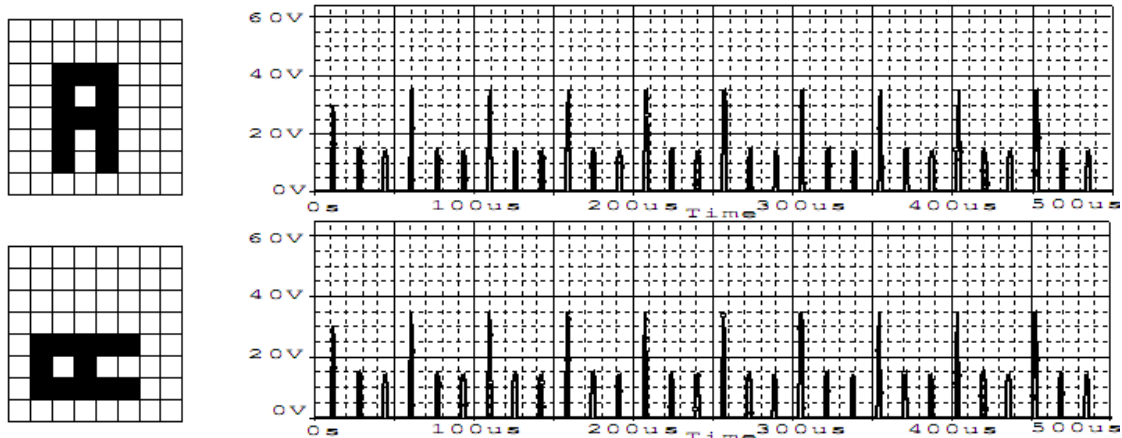


Figure 9. Responses of the PCNN to two similar input patterns. The figure shows angle and position invariance of the neuron, since both signals are equal despite the displacement of the input image..

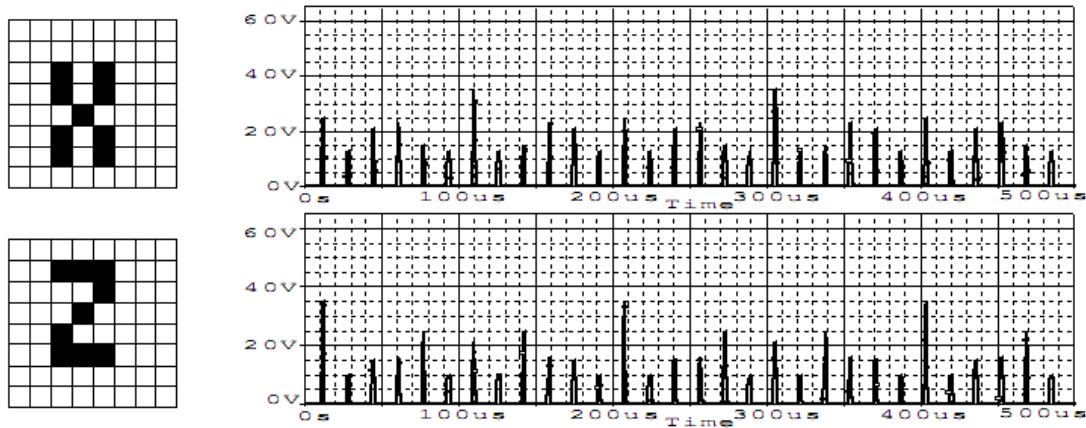


Figure 10. The figure shows the difference between outputs from different input patterns that could lead to a classification. The output patterns are also different from those of the figure 9.

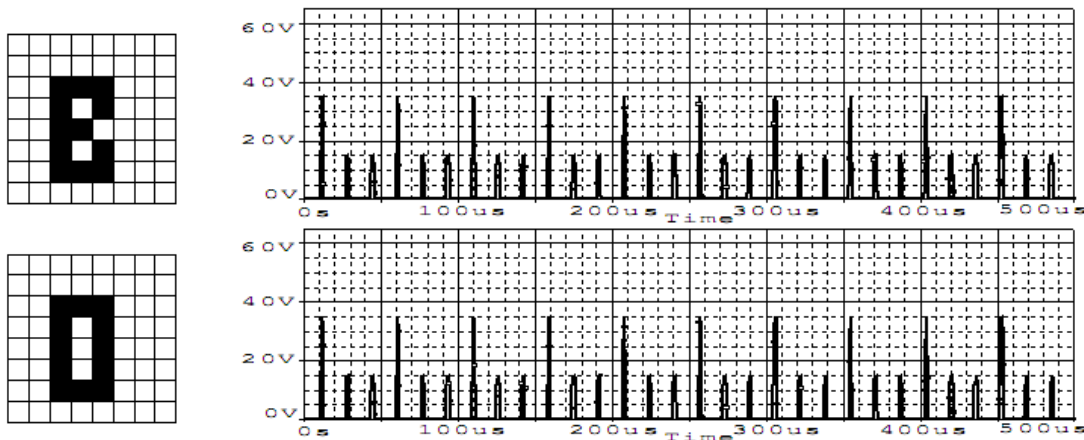


Figure 11. Responses of the PCNN to two input patterns. The figure shows the similar responses of similar but not identical patterns, showing some possible limitations of the classification ability.

4. CONCLUSIONS

In this work we have implemented a design of Pulse Coupled Neural Networks via a mathematical simulation and a hardware circuit. There is evidence that the performance of both implementations is identical, allowing interchangeability to optimize the design of a PCNN.

This way, the parameters can be adjusted quickly and easily to the specific application sought. This can lead to the development of complete visual applications¹¹, where a PCNN is a stage extracting the main characteristics of the input images.

A fast parameter design simplifies the creation of new applications, limiting the difficulties of finding the right values for a given application, or multiple ones. The adaptations of a PCNN system to image smoothing, segmentation, object detection and classification, with the correct settings to optimize the use of the network in a given situation is greatly simplified through the use of easy simulation, leading to the development of new and more efficient vision applications.

REFERENCES

1. R. Eckhorn, "Neural mechanisms of scene segmentation: recordings from the visual cortex suggest basic circuits for linking field models" *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 464–479, 1999.
2. J. L. Johnson and M. L. Padgett, "PCNN models and applications," *IEEE Transactions on neural networks*, vol. 10, no. 3, pp. 480–498, 1999.
3. Kuntimad G. Kuntimad and H. S. Ranganath, "Perfect image segmentation using pulse coupled neural networks," *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 591-598, 1999.
4. H. S. Ranganath and G. Kuntimad, "Object detection using pulse coupled neural networks," *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 615-620, 1999.
5. B. Yu and L. Zhang, "Pulse coupled neural network for motion detection," in *Neural Networks, 2003. Proceedings of the International Joint Conference on*, vol. 2, pp. 1179-1184, 2003.
6. P. E. Keller, D. L. McMakin, T. E. Hall and D. M. Sheen, "Use of a neural network to identify man-made structure in millimeter-wave images for security screening applications," in *IJCNN*, pp. 2009-2014, 2006.
7. Y. Ota and B. M. Wilamowski, "Analog implementation of pulse-coupled neural networks," *IEEE Transactions on Neural Networks*, vol. 10, no. 3, pp. 539–544, 1999.
8. H. Lamela, M. Ruiz-Llata, M. Jiménez, M. González, J. Gimeno and C. Warde, "PCNN pre-processor stage for the Optical Broadcast Neural Network Processor," *Proceedings of SPIE*, vol. 6247, 2006.
9. J. L. Johnson, "Pulse-coupled neural nets: translation, rotation, scale, distortion and intensity signal invariance for images," *Applied Optics*, vol. 33, no. 26, pp. 6239-6253, 1994.
10. H. Lamela and M. Ruiz-Llata "Image identification system based on an optical broadcast neural network and a pulse coupled neural network preprocessor stage," *Appl. Opt.*, vol. 47, no. 10, pp. B52–B63, 2008.

11. M. González, H. Lamela, M. Jiménez, J. Gimeno and M. Ruiz-Llata “Hardware implementation of a neural vision system based on a neural network using integrated and fire neurons” Independent Component Analyses, Wavelets, Unsupervised Nano-Biomimetic Sensors, and Neural Networks V. Proc. SPIE, Vol. 6576- 65760Y, 2007.