Covert Attention with a Spiking Neural Network

Sylvain Chevallier, Philippe Tarroux

Covert Attention with a Spiking Neural Network

Sylvain Chevallier[1,2] and Philippe Tarroux[2,3]

[1] Université Paris-Sud XI, France
[2] LIMSI-CNRS UPR 3251, Orsay, France
[3] École Normale Supérieure, Paris, France
{sylvain.chevallier,philippe.tarroux}@limsi.fr

Abstract. We propose an implementation of covert attention mechanisms with spiking neurons. Spiking neural models describe the activity of a neuron with precise spike-timing rather than firing rate. We investigate the interests offered by such a temporal code for low-level vision and early attentional process. This paper describes a spiking neural network which achieves saliency extraction and stable attentional focus of a moving stimulus. Experimental results obtained using real visual scene illustrate the robustness and the quickness of this approach.

Key words: spiking neurons, precise spike-timing, covert attention, saliency

1 Introduction

Understanding the neural mechanisms underlying early attentional processes can open novel ways of solving some artificial vision problems. Neurobiology and cognitive psychology produce evidence for an early information selection: the brain handles only a small part of the visual scene at a time. Spiking neural networks offer interesting properties since they seem to capture important characteristics of biological neuron with relatively simple models. We propose a bio-inspired spiking neural network (SNN) which selects such a small visual area and focuses on it, using saliency extraction. The focusing mechanism relies on the spatio-temporal continuity of the stimulus and is robust to small movements.

The different approaches used for modeling visual attention and the motivations leading to place this work in a bio-inspired framework are explained in 1.1. The properties of the different spiking neural models that can be used in the present context are detailled in 1.2. The network set up to extract saliencies and focus on a moving stimulus is described in Sect. 2 and experimental results are given in Sect. 3. The experimental validation and the obtained results are given in Sect. 3. The spike coding used in this network is discussed in Sect. 4. Section 5 concludes this paper.

1.1 Modeling Visual Attention and Saliency

Visual attention is a sequential mechanism: the brain concentrates only on a small region at a time as *change blindness* experiments demonstrate (see

O'Regan [1] for example). It is often said that the complexity of the visual world exceeds the brain ability to process a whole visual scene. This hypothesis can be questioned regarding the unmatched computational power of the brain's massively parallel architecture. This sequential process may have been kept by evolution because it brings some benefits. The questions of what these benefits are and whether they are relevant for artificial vision are seldom addressed and have no simple answer. However, for instance, a sequential process endows the system with what Tsotsos calls an "hypothesize-and-test" mechanism [2] leading to inferential abilities that could have been the target of the natural selection process.

The spatial attention is expressed by two mechanisms : overt and covert attention. Overt attention refers to situation where the eye makes a saccade to focus a saliency. It is opposed to covert attention, which involves no saccadic moves, addressed in this paper. The attentional process selects a part of the visual input and is a top-down process, i.e. involving task-dependent or context-dependent influences. Pre-attention is a similar process, but in a bottom-up (BU) or data-driven manner. The rest of this paper deals only with BU processes: there is no assumption on learning or categorizing objects. A specific region of a visual scene, selected by a pre-attentional process, is referred to as a saliency. The saliencies are then gathered on saliency map whose existence is commonly accepted but remains unproven [3]. Many models describe visual attention mechanisms and a significative part rely on saliencies. Saliency-based models can be separated in three main approaches: psychological models, image processing systems and bio-inspired approaches.

Computational models set up by psychologists are closely linked to experimental data, trying to explain or predict the behavior resulting from neuropsychological disorders [4]. Whereas these goals diverge from computational ones, some findings are interesting: Treisman [5] and Wolfe [6] have proved the existence of visual features. These features are the different modalities of vision: color, luminance, movement, orientation, curvature, among others. Treisman proposed a theory [5] in which features are extracted in a parallel way during the pre-attentive stage and then combined on a saliency map. Psychological models help to build a theory of perception although these models cannot be part of an artificial perceptual system dealing with real world situation.

Salient regions can also be seen as "meaningful" descriptors of an image. This formulation leads the image processing community to propose other ways for extracting saliency points. Image processing models use information theory (local complexity or unpredictability, [7]) or pixel distribution (local histograms, [8]). Such descriptors are used to recognize known objects or object classes and give their best performance in controlled environments. Image processing systems can handle real world data but they are often constrained by strong hypothesis and are not suited for generic situation or open environment.

The bio-inspiration way aims at bringing together natural solutions and computational efficiency. Realistic biological or psychological models try to reproduce or explain every observation. Image processing systems seek efficient solutions

and are well suited for specific situations. On the contrary, bio-inspired models seek a compromise and try to capture only key properties. Itti and Koch [9] proposed an image processing based model inspired by low-level biological visual processing. There are many other bio-inspired models (described in [10]) which rely on similar principles and share the notion of a saliency map. The existence of a saliency map neural correlate is broadly discussed [3]. We use the following definition of a saliency: a region in a visual scene which is locally contrasted in terms of visual features and globally rare in the visual scene. Bio-inspired approaches offer a good framework for designing efficient and robust methods for extracting saliencies. In a bio-inspired framework, visual attention problems can be addressed at two levels. At a system level, Bayesian approaches model or reproduce a global behavior. At the opposite, at a unit level, neural-based approaches specify the local properties enabling the emergence of a global behavior. These two approaches follow different methods, explicative model for the former and global analysis for the latter, but can benefit from each other.

We experiment the contribution of a neural-based system, with simple spiking models, not suited for precise modelling but adequate for real-time computation. We choose to investigate the explicative models of bio-inspired visual attention rather than the descriptive ones.

1.2 Spiking Neural Networks

Maass [11] described spiking neurons as the "third generation" of neural models. Spiking neurons capture fundamental aspect of the neural functionality: the ability to code the information as discrete events whereas the underlying equation are reasonably simple. A spiking neuron unit [12, 13] models the variation of the membrane potential and fires a spike if the membrane crosses the threshold. The main difference between a spiking unit and a classical sigmoid unit reside in the way of handling time. The membrane potential V_i of neuron i is driven by a differential equation and takes into account the precise time of the incoming spikes. The learning ability of spiking neural networks are actively investigated [14–17].

A single spiking neuron can exhibit two behaviors: it can either integrate the information over a predefined temporal window or act as a synchrony detector, i.e. emitting spikes when inputs are condensed in a small period of time.

2 Model Description

We use a Leaky Integrate-and-Fire (LIF) model characterized by the following equation:

$$\begin{cases} \tau \dot{V}_i = g_{\text{leak}}(V_i - E_{\text{leak}}) + \text{PSP}_i(t) + \text{I}(t), \text{ if } V \leq \vartheta \\ \text{spike and reset } V \text{ otherwise} \end{cases} \tag{1}$$

where τ is the membrane time constant, g_{leak} is the membrane leak conductance, ϑ is the threshold and E_{leak} is the membrane resting potential [13]. $\text{I}(t)$ represents the influence of an external input current (as in Chap. 4.1.1 of [13]). The $\text{PSP}(t)$

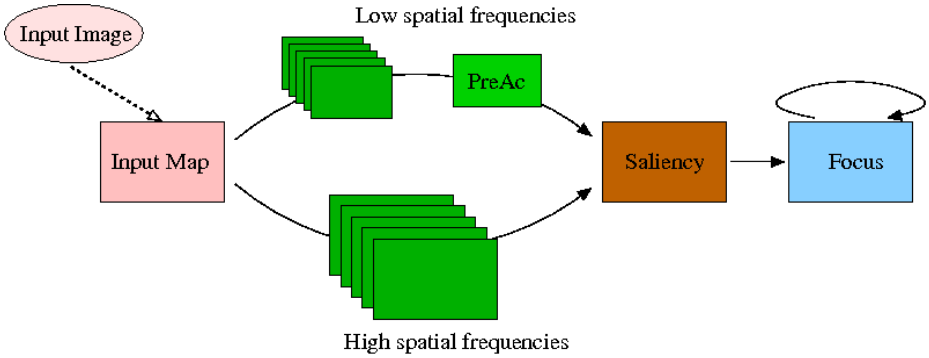

Fig. 1. Architecture of a set of feed-forward connected neural maps, which extract saliencies from an input image and focus on the most salient region. The image is processed both at low spatial frequencies (LSF) and high spatial frequencies (HSF).

is the synaptic input function, describing the influence of incoming spikes on membrane potential. Among the different PSP models that can reflect complex synaptic variations (as in [18, 19]), we have chosen a simple one for achieving fast computation. Thus we use a PSP model without synaptic conductance (as in [20]). Formally, incoming PSPs from neuron j are denoted by:

$$S_j(t) = \sum_f \delta(t - t_j^{(f)} + d_j) \tag{2}$$

where $\delta(x)$ is the Dirac distribution, with $\delta(x) = 0$ for $x \neq 0$ and $\int_{-\infty}^{\infty} \delta(x)\mathrm{d}x = 1$, $t_j^{(f)}$ is the spike emission time and d_j the synaptic delay. Since we use a model without synaptic conductance, the influence of incoming PSPs on membrane potential is given by the simple relation:

$$\mathrm{PSP}_i(t) = \sum_j w_{i,j} S_j(t) \tag{3}$$

These computations was handled in a clock-based sequential simulator which not distributed. This simulator process only the active neurons, i.e. neurons integrating PSPs [21].

The SNN represented on Fig. 1 is a set of neural maps (2D neural layer) which is divided in two main pathways for processing high and low spatial frequencies. Visual modalities of an input image are decomposed with neural filters, explained in Sect. 2.1, and we combine all the obtained visual modalities on the Saliency map, detailed in Sect. 2.2. This combination relies on the temporal processing of spiking neurons, low spatial frequencies being gathered on PreAc map (on Fig. 1), which pre-activates the Saliency map neurons. The Focus map selects the most intense saliency as the focus of attention is described in Sect. 2.3.

2.1 Neural Filter

The input image is translated in spike trains and the network handles luminance and color information. Each neuron of the Input map (Fig. 1) is associated with the corresponding pixel, i.e. the pixel luminance determines the input term I(t) of the corresponding neuron. For a LxM image, there are eight MxM neural maps and six $\frac{L}{2}$x$\frac{M}{2}$ neural maps (see Fig. 1).

Neurons on the Input map project on both the LSF and HSF pathways through connection masks as illustrated on Fig. 2. These masks are static weight matrices with delay and define a generic projection from one neural map to another. The weight matrix values are similar to convolution kernel used in image processing. We use difference of Gaussian (DoG) and four Gabor orientated kernels as connection masks between Input map and maps in both HSF and LSF pathways.

This network achieves an image filtering similar to a classical kernel convolution. However the "convolution" realized by PSPs propagation through connection masks is applied in an order depending on the input value, i.e. the most important filter coefficient being processed first. Furthermore, the lowest input values are not processed: due to the discrete nature of spiking neurons, only above threshold information is propagated into the network (depending on g_{leak} and I(t) values). This functional filtering and the fact that our implementation processes only the neurons receiving PSPs lead to a fast execution (see Sect. 3).

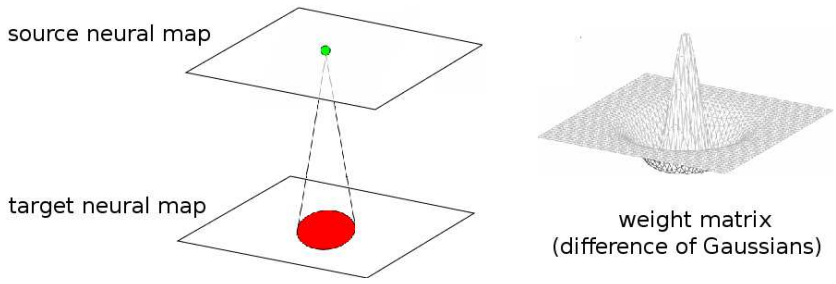

Fig. 2. A neuron (in green or light grey) emits a spike and sends PSPs to all red (or dark grey) neurons. The weight of each PSP is given by the weight matrix, represented on the right. The weight matrix values are similar to convolution kernel used in image processing. Here, a difference of Gaussian.

2.2 Saliency Map

The Saliency map (Fig. 1) gathers all information from visual features on different spatial frequencies. The Saliency map neurons are tuned to implement synchrony detectors, i.e. they emit spikes only if PSPs are gathered in a small

time window. Thus, saliencies are emerging from saliency map only if a spatial location generates spikes on different neural maps coding for different visual features.

A saliency point is represented by a spike emited by a neuron of the Saliency map. Thus saliencies are temporally coded and arise in hierarchical order. The neuron corresponding to the most salient location is the first to emit a spike, and so on. Others bio-inspired approaches extract visual features, combining them on a saliency map, eventually using a Winner-Takes-All (WTA) algorithm to choose the most intense saliency. To find the second saliency, the previous saliency must be inhibited and the whole computation is to be started again. On the contrary, the present implementation uses a recurrent map for implementing selection process and the experimental results show that this kind of network implements a fast and implicit WTA.

2.3 Covert Attention

When a saliency is detected, the output spikes from saliency map are sent to the self-connected Focus map, see Fig. 1. This self-connection mask is a DoG, which excites adjacent neighbours and inhibits distant ones. This self-connection needs the Saliency map spikes to keep a stable activity and is not sufficient to maintain a constant activity alone. As the saliency moves, the activity on the Focus map follows as long as the saliency stays in the positive part of the DoG [22].

3 Experimental results

Real world images were used for the evaluation of this SNN. We used a Sony EVID31 pan-tilt camera for the acquisition and a Khepera robot as the moving stimulus. A 30 frames sequence has been recorded (see Fig. 3). During this sequence, the network focused on the moving stimulus and let the focus change quickly when the camera made a saccade. Note that this saccadic move was driven externally and does not rely on the activity of the network.

The frames are 760x570 pixels wide images and are reduced to 76x56 pixels for the input of the SNN. As one pixel of the input frame corresponds to one neuron, the network was composed of ∼53,000 neurons ($L = 76$ and $M = 56$). In this set of experiment, we only used the luminance information.

The first frame was presented to the network for 20 integration steps. This bootstrap stage let the network activity emerge and the spikes propagate through each neural maps. At the 20^{th} integration step, the Focus map emited spikes. Each frame was then presented during N integration steps. The results for different N values are shown in Fig. 4 and 5.

To check the performance of the SNN, we computed the euclidean distance between the stimulus centroid and the centroid of the activity as an error measure. Stimulus centroid was computed as the centroid of the stimulus pixel, for a given frame. The activity centroid is defined as the centroid of all emitted spikes by the focus map during the N integration steps of the image presentation.

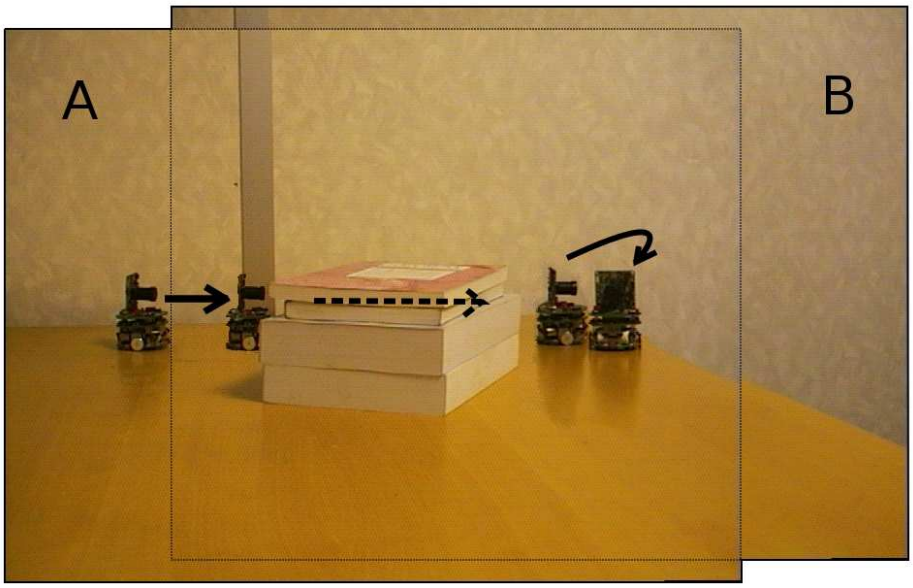

Fig. 3. Representation of the input video sequence. The mobile stimulus (a Khepera robot) moves from left to right. As it reappears, after being hidden behind the books, the camera makes a saccade from frame A to frame B for focusing the stimulus.

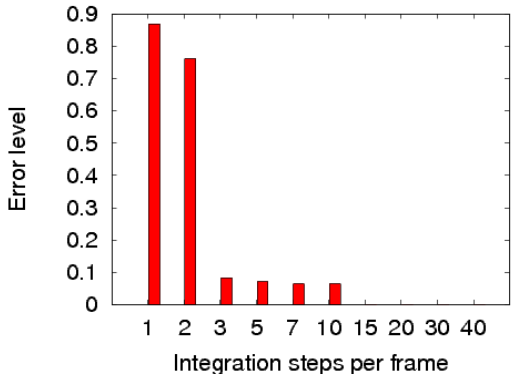

Fig. 4. Error level for the 30 frames video sequence.

Figure 4 shows that the error decreased when the network had more time (integration steps) to process each frame. The error level decreased rapidly when there is a sufficient number of computation steps per frame. One can notice that the error is still very low for only 3 integration steps per frame. This is an interesting result especially given the computation time which are shown on Fig. 5.

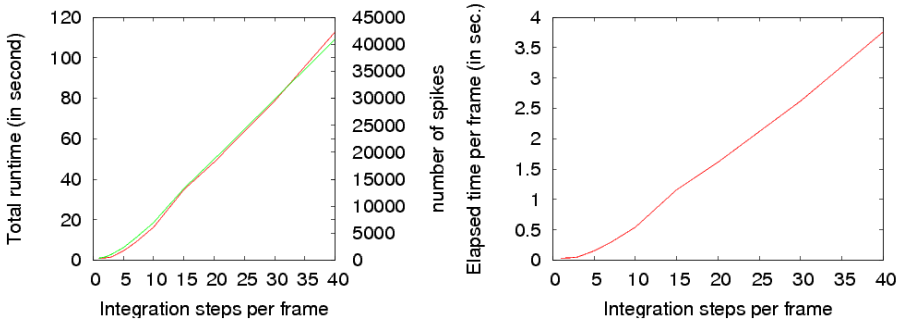

Fig. 5. *Left*: Runtime of our program, including building and ending process, is indicated by the green curve. Overall activity load of the network is displayed by the red curve, as the total number of spikes emitted by the network. *Right*: time taken to process one frame, given the number of integration steps per frame.

All the results presented on Fig. 5 have been obtained with an desktop Intel Core2Duo (1.86GHz). The total run time and the overall activity evolves linearly as the number of steps per frame increases, as shown on left part of Fig. 5. The time needed to process a frame is very promising (Fig 5, right), as with 3 integration steps per frame, the network process ∼20 frames/second. These results confirm that SNN are suitable for visual computation in a real-time framework.

4 Discussion

Thorpe [23] shows that monkeys and humans were able to detect the presence of animals in a visual scene in an extremely short time, leaving neurons just enough time to fire a single spike. Information is condensed in the precise time of each spike and the relative latency between the spikes. This first spike code can been used to recognize a previously learned pattern in real world images [24, 25] or to characterize natural images [26]. The network described in this paper uses spike train for detecting the spatio-temporal continuity of a stimulus, which is not possible with a unique first spike.

5 Conclusion

In this contribution, we build a SNN able to compute saliencies and to focus on the most important one. Thanks to the coincidence detector properties of the Saliency map spiking neurons, we show that this SNN can extract saliencies. The implementation of a covert attention process rely on the temporal computation of the spiking neuron. This network was evaluated on real data (a video sequence) and was able to focus on a moving target. The mesured computation time shows that this network is suitable for a real-time application.

References

1. O'Regan, K.J., Noë, A.: A sensorimotor account of vision and visual consciousness. Behavioral and Brain Sciences **24** (2001) 939–1031
2. Tsotsos, J.K.: On the relative complexity of active v.s. passive visual search. International Journal of Computer Vision **7**(2) (1992) 127–141
3. Fecteau, J.H., Munoz, D.P.: Salience, relevance, and firing: a priority map for target selection. Trends in Cognitive Sciences **10**(8) (2006) 382–390
4. Heinke, D., Humphreys, G.W.: Computational models of visual selective attention: A review. In: Connectionist Models in Cognitive Psychology. Routledge (2005) 273–312
5. Treisman, A., Gelade, G.: A feature-integration theory of attention. Cognitive Psychology **12**(1) (1980) 97–136
6. Wolfe, J.: Visual attention. In: Seeing. 2nd edn. Academic Press (2000) 335–386
7. Kadir, T., Brady, M.: Scale, saliency and image description. International Journal of Computer Vision **45**(2) (2001) 83–105
8. Schmid, C., Mohr, R., Bauckhage, C.: Evaluation of interest point detectors. International Journal of Computer Vision **37**(2) (2000) 151–172
9. Itti, L., Koch, C.: Computational modeling of visual attention. Nature Reviews Neuroscience **2**(3) (2001) 194–203
10. Itti, L., Rees, G., Tsotsos, J., eds.: Models of Bottom-Up Attention and Saliency. In: Neurobiology of Attention. Elsevier, San Diego, CA (January 2005) 576–582
11. Maass, W.: Networks of spiking neurons: the third generation of neural network models. Neural Networks **10** (1997) 1659–1671
12. Maass, W., Bishop, C.M., eds.: Pulsed neural networks. MIT Press, Cambridge, MA, USA (1999)
13. Gerstner, W., Kistler, W.: Spiking Neuron Models: An Introduction. Cambridge University Press, New York, NY, USA (2002)
14. Maass, W., Natschläger, T., Markram, H.: Real-time computing without stable states: A new framework for neural computation based on perturbations. Neural Computation **14**(11) (2002) 2531–2560
15. Jäeger, H., Maass, W., Principe, J.: Introduction to the special issue on echo state networks and liquid state machines. Neural Networks **20**(3) (2007) 287–289
16. Maass, W.: Liquid Computing. In: Computation and Logic in the Real World. Volume 4497/2007 of LNCS. Springer (2007) 507–516
17. Cios, K.J., Swiercz, W., Jackson, W.: Networks of spiking neurons in modeling and problem solving. Neurocomputing **61** (2004) 99–119
18. Rudolph, M., Destexhe, A.: On the use of analytic expressions for the voltage distribution to analyze intracellular recordings. Neural Computation (2006)

19. Brette, R.: Exact simulation of integrate-and-fire models with synaptic conductances. Neural Computation **18**(8) (2006) 2004–2027
20. Brunel, N.: Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. Journal of Computational Neuroscience **8** (2000) 183–208
21. Chevallier, S., Tarroux, P., Paugam-Moisy, H.: Saliency extraction with a distributed spiking neural network. In: Proc. of European Symposium on Artificial Neural Networks, Bruges, Belgium (2006) 209–214
22. Chevallier, S., Tarroux, P.: Visual focus with spiking neurons. In: Proc. of European Symposium on Artificial Neural Networks, Bruges, Belgium (2008)
23. Thorpe, S., Fize, D., Marlot, C.: Speed of processing in the human visual system. Nature **381** (1996) 520–522
24. Delorme, A., Gautrais, J., VanRullen, R., Thorpe, S.J.: Spikenet: a simulator for modeling large networks of integrate-and-fire neurons. Neurocomputing **26-27** (1999) 989–996
25. Thorpe, S.J., Guyonneau, R., Guilbaud, N., Allegraud, J.M., VanRullen, R.: Spikenet: Real-time visual processing with one spike per neuron. Neurocomputing **58-60** (2004) 857–864
26. Perrinet, L.: Finding independent components using spikes : a natural result of hebbian learning in a sparse spike coding scheme. Natural Computing **3**(2) (2004) 159–175