Simultaneous neuromorphic selection of multiple salient objects for event vision

Amélie Gruel, Jean Martinet, Michele Magno

Simultaneous neuromorphic selection of multiple salient objects for event vision

Amélie Gruel
i3S / CNRS
Université Côte d'Azur
Sophia Antipolis, France
amelie.gruel@univ-cotedazur.fr

Jean Martinet
i3S / CNRS
Université Côte d'Azur
Sophia Antipolis, France
jean.martinet@univ-cotedazur.fr

Michele Magno
PBL Center
ETH Zürich
Zürich, Switzerland
michele.magno@pbl.ee.ethz.ch

***Abstract*—The combined use of spiking neural networks and event cameras is gaining momentum in the field of embedded computer vision as they promise to reduce latency and computational resource requests. However, state-of-the-art embedded neuromorphic models show little interest in modifying input data to optimise model performance, memory usage, latency, and power consumption. This work addresses this optimisation trade-off by implementing a neuromorphic model of salient selection, which simultaneously outputs multiple segregated objects of interest detected in an event-based scene. This work extends previous ones and identifies regions of interest as those corresponding to a high spatiotemporal density of events. Without any training and with a limited number of neurons, the proposed model is able to simultaneously detect different objects with a delay of only 14ms at most, and filtered objects maintain 73% of the original data's classification performance. We are thus confident that the method proposed in this paper will allow for improving the subsequent neuromorphic processing of event data on embedded systems. To the best of our knowledge, it is the first neuromorphic model able to simultaneously select multiple objects of interest. Our code can be found here: `github.com/amygruel/FoveationStakes_DVS/`.**

***Index Terms*—Visual attention, Spiking neural network, Event camera, Saliency, Neuromorphic**

I. Introduction

Spiking neural networks (SNNs) [1] are bio-inspired artificial neural networks aiming to mimic the dynamics of biological neuronal circuits by receiving and processing information in the form of spike trains (see Fig. 1B). Event cameras [2] are increasingly popular for capturing fine-grained dynamics of a scene, with a native SNN-friendly encoding. Instead of measuring the intensity of every pixel in a fixed time interval like standard cameras, they generate events of significant pixel intensity changes (see Fig. 1A). Every such event is represented by its position, sign of change, and time-stamp, accurate to the microsecond. Because of their asynchronous operation principle, they are a natural match for SNNs. Their combined use is of such high interest from the point of view of biological inspiration, energy savings, decision latency, and memory use that it is gaining momentum in the field of embedded computer vision [3].

However, embedded systems and even early realised neuromorphic embedded processors are quite limited in terms of memory bandwidth. Although event cameras produce less heavy data with less redundant information than a conventional RGB camera, the visual scene may in some cases of fast motion and highly textured objects produce a flow of events too dense to be correctly processed by the state-of-the-art low-power embedded system [6]. The latter is likely to saturate and drop incoming events thus missing potentially relevant information, without any human control. In such cases, it is thus important to focus the treatment on relevant information to fasten and better it. We believe this comes from reducing the size of the input data while maintaining the quality of the information conveyed in order to optimise the embedded system performance.

Previous work has attempted to address this issue by proposing neuromorphic [7] or non-neuromorphic [6], [8] spatial reduction techniques. However, the trade-off between the quantity and the quality of the reduced event data is not optimal (as explained in [7], [8]), of which a possible explanation may be the non-detection of salience during data reduction. Indeed, we believe that detecting Regions of Interest (RoIs) in the original visual scene to select the corresponding Objects of Interest (OoIs — i.e. the events taking place in the detected RoIs) is a more promising approach. A significant number of computer vision tasks (classification, object tracking, autonomous navigation, etc) could rely on small

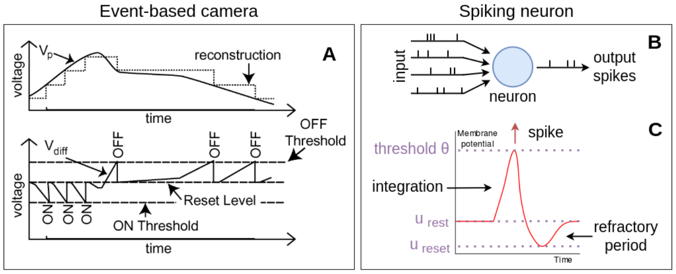


Fig. 1: (A) Principle of operation of an event-based camera, from [4]. (B) Behaviour of a spiking neuron, which receives spike trains as input and processes this information to produce a new sequence of activations. (C) Evolution of the neuron's membrane potential over time when activated by input spikes.

This work was supported by the European Union's ERA-NET CHIST-ERA 2018 research and innovation programme under grant agreement ANR-19-CHR3-0008. The authors are grateful to the OPAL infrastructure from Université Côte d'Azur for providing resources and support.

salient items in the global scene [9]. This work presents and demonstrates the assumption that selecting only the objects of interest will simultaneously reduce the size of data to be processed, the number of events and spatiotemporal densities while maintaining significant information quality.

According to [10], visual attention can be defined as the behavioural and cognitive process of selectively focusing on a discrete aspect of sensory cues while disregarding other perceivable information. The RoI detection we propose in this work consists ofp a neuromorphic visual attention model applied to event data. However, works that address all of these constraints are rare in the literature (see [11]): for example, the models introduced in [12]–[14] detect saliency on event data with traditional neural networks while [15], [16] implement neuromorphic models of visual saliency in RGB data. Additionally, neuromorphic saliency detection models applied to event data are most often than not derived from existing models implemented with traditional neural networks and applied to RGB or grayscale images: [17] is for example adapted from [13], itself originally adapted from the grouping mechanism estimating the location and spatial scale of proto-objects in RGB data implemented in [18]. Only a few models truly take advantage of the intrinsic dynamics of SNNs and the uniqueness of event data: in particular, [19] make use of the mathematical model of Dynamic Neural Field [20] as a soft Winner-Takes-All (WTA) to implement salient tracking of pre-activated objects.

Following this limited number of existing visual attention models, this paper will first present in more detail the existing architectures our work is based on and describe our contribution; then it will extensively outline the different experimental validations of the original architecture and our new contribution. In summary, the proposed model of multi-OoIs attentional selection:

- is able to simultaneously select multiple OoIs present more than 50% of the time, with no training phase;
- selects at least one OoI with a delay of less than $15ms$, and reaching $5ms$ in the best cases;
- filters out OoIs with a quality leading to a classification performance reaching 73% of the original data;
- achieves all the above with a reduced number of neurons and synapses compared to existing methods.

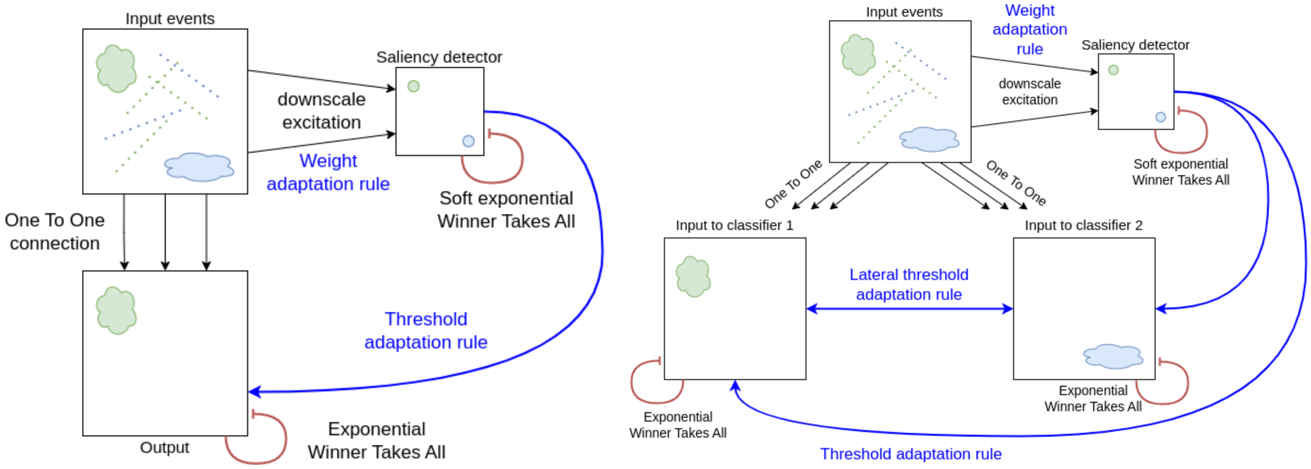
II. Neuromorphic models for OoIs selection

The detection of OoIs is a little-explored issue regarding event data. This work extends the neuromorphic model we first introduced in [5] to multi-OoIs selection. Our original mechanism relies solely on intrinsic SNN dynamics and dynamic adaptation rules applied to synaptic weights and population thresholds. These are crucial features as they lead to minimising the latency since it does not require the conversion of spiking events into frames. The saliency detection and multi-OoIs selection model proposed in this new work is not specialised for any specific context or any specific shape (in other words, there is no training phase) which allows for a good generalisation ability of the network.

The original architecture, presented in Fig. 2A, and our contribution, presented in Fig. 2B, are designed to be lightweight enough to enable embedded simulations in real-time. These models are implemented using the ”Leaky Integrate-And-Fire” SNN model because of its simplicity: the membrane potential is at rest when there is no input; otherwise, it increases according to the incoming spikes and slowly decays towards the resting value when the input stops (leak). If the membrane potential overcomes a threshold, an output spike is produced and the membrane potential is reset.

A. Saliency detection

As described in [21], the saliency detector integrates the events produced by each pixel at a low resolution and outputs a set of coordinates for one or multiple RoIs. In this case, the RoI would be a region where the amount of events received over a certain amount of time is more important than elsewhere


Input events
Saliency detector
downscale excitation
Weight adaptation rule
Soft exponential Winner Takes All
One To One connection
Threshold adaptation rule
Exponential Winner Takes All
Output
Input events
Weight adaptation rule
Saliency detector
downscale excitation
Soft exponential Winner Takes All
One To One
One To One
Input to classifier 1
Input to classifier 2
Lateral threshold adaptation rule
Exponential Winner Takes All
Exponential Winner Takes All
Threshold adaptation rule

(a) Architecture of our initial neuromorphic model detecting saliency by event density and filtering out one OoI.)

(b) Architecture of our main contribution. The maximum number of OoIs (i.e. the number of output layers) has been set to 2 for clarity.

Fig. 2: Overview of our initial model introduced in [5] (A) and our contribution (B).

over the whole scene, i.e. a region where the events are numerous in a small spatiotemporal window. The visual attention mechanism implemented in [21] is thus *bottom up* (i.e. independent from any previously set motivation or rule) and *covert* (i.e. without simulated saccadic eye movements) [11]. The saliency detector is formed by the ”Input events” and ”Saliency detector” layers and their interconnections depicted in Fig. 2A.

1) Input events: The input layer translates relative changes in the illumination from the sensor (or events) into spikes, which are sent to the saliency detector via an excitatory downscaling connection. This corresponds to a convolutional layer with a kernel size $S \times S$, a stride S, without padding.

2) Saliency detector: The saliency detection aggregates the active regions into distinct segments using a soft exponential WTA strategy by laterally inhibiting neurons in the same layer (see Eq. 1): each neuron activation leads to the inhibition of the others, without autapses (self-connections). A soft WTA strategy is adopted as it leads to the activation of multiple neurons in the layer thus the detection of multiple RoIs.

$$\omega_{WTA} = \min(\frac{e^{d}}{w \times h}, \omega_{max}) \tag{1}$$

where d corresponds to the Euclidean distance between the active and target neuron subject to inhibition, and w and h to the width and height of the layer. The upper bound ω_{max} of the weight ω_{WTA} is a tunable parameter for optimising the saliency detection, depending on the input data (see Fig. 8).

3) Weight adaptation rule: Finally, the adaptive detection of saliency in this layer is enabled by a dynamic weight adaptation rule between the input layer and the saliency detector, inspired by Hebb's rule: ”cells that fire together wire together” [22]. This rule is implemented by increasing or decreasing the weights of synapses that have recently fired, as described in Eq. 2.

$$\omega(t+1) = \begin{cases} \omega(t) + \Delta\omega & \text{if } ft_{saliency} \geq t \\ \omega_{init} & \text{if } ft_{saliency} < t - t_{\delta} \\ \omega(t) & \text{otherwise} \end{cases} \tag{2}$$

where $\omega(t)$ is the weight at the simulation step t of the synapse undergoing the dynamic weight adaptation rule, $\Delta\omega$ is the positive weight variation at each simulation step, ω_{init} is the initial weight of the synapse (homogeneously initialised), $ft_{saliency}$ is the firing time of the last spike emitted by the saliency detector and t_{δ} the delay before the synaptic weight decays back to ω_{init}.

B. Attentional selection of one OoI

The saliency detection described above was extended to a first attentional model by the authors of [5] (see Fig. 2A).

1) Output layer: An output layer of the same size as the input layer is added in this model. It receives the activity of the input layer through one-to-one connectors (i.e. each input neuron is solely connected to the neuron located at the same coordinates in the output layer). Only the spikes corresponding to one OoI are emitted by this layer thanks to an exponential WTA mechanism similar to the one described in Eq. 1.

2) Threshold adaptation rule: The attentional filtering implemented in [5] aims to maintain the same information as the OoI identified in the original data. In order to do so, the authors set up a dynamic threshold adaptation rule aiming to facilitate the neuronal spiking at the salient coordinates and to hinder it in other neurons of the output layer. This is respectively translated into the decrease (closer to the resting value) and increase (further away from it) of the neuronal threshold of each neuron depending on the activity at the corresponding region in the saliency detector, as described in Eq. 3 and Eq. 4.

$$\theta(t+1) = \begin{cases} \theta(t) - \Delta\theta & \text{if } ft_{saliency} \geq t \\ \theta(t) + \Delta\theta & \text{if } ft_{saliency} < t - t_{\delta} \\ \theta(t) & \text{otherwise} \end{cases} \tag{3}$$

where $\theta(t)$ is the threshold at the simulation step t of the neuron to which is applied the dynamic weight adaptation rule, $\Delta\theta$ is the positive threshold variation at each simulation step, $ft_{saliency}$ corresponds to the firing time of the last spike transmitted by the saliency detector in the corresponding neuronal regions and t_{δ} is the delay before the neuronal spiking is hindered i.e. before the neuronal threshold is increased. Note that the $ft_{saliency}$ used here corresponds to the $ft_{synapse}$ used in Eq. 2. However, while the weight adaptation rule applied to the synapse linking the input and the saliency detector relies on the activity of its post-synaptic neurons, here the thresholds modifications carried out by the adaptation rule do not rely on the post-synaptic activity (i.e. the output layer's activity) but on the saliency detector acting as an external supervisor.

$$\theta(t+1) = \begin{cases} \theta_{max} & \text{if } \theta(t+1) > \theta_{max} \\ \theta_{reset} & \text{if } \theta(t+1) < \theta_{reset} \\ \theta(t+1) & \text{otherwise} \end{cases} \tag{4}$$

where θ_{max} and θ_{reset} are respectively the upper and lower bound (i.e. the reset value) of the neuronal threshold. Note that this equation (as well as Eq. 6 and Eq. 7) updates θ, which was first calculated in Eq. 3, according to various conditions at $t+1$, which explains the repeated use of the term $\theta(t+1)$.

The authors of [5] demonstrated that such a plasticity rule is highly preferred to a simple system of synaptic activations and inhibitions. Indeed, activating the neurons corresponding to the salient regions identified by the saliency detector would compete with the input activation and saturate these neurons. This would blur the spikes from the input data, causing the filter to lose spatiotemporal accuracy.

C. Attentional selection of multiple OoIs

The main contribution of this work is the implementation of the following proposed neuromorphic model, an extension of our previous work to simultaneously detect and filter out multiple OoIs in an event-based visual scene using intrinsic SNN dynamics. This model is designed to detect n OoIs — however, in an effort to simplify the reader's comprehension, it is depicted in Fig. 2B under an architecture which would allow the detection of two OoIs at most.

This new attentional selection of n OoIs features the same saliency detector, exponential WTA and weight adaptation rule we introduced in [5]; however, we propose here a new dynamic threshold adaptation rule grounded on the activity of both the saliency detector and the lateral output layers. At each simulation timestep, for each neuron of each output layer, this dynamic rule will first identify the lateral neuronal activity at the same coordinates (see Eq. 5) and maximise the threshold if the corresponding lateral neurons are activated (see Eq. 6 and Eq. 7). The rule will then modify the threshold depending on the activation of the saliency detector in the corresponding neuronal regions (as described in Eq. 3 and 4).

$$\forall \lambda \in \lambda_{\text{lateral layers}}, \forall n \in n_{neighbourhood},$$

$$\alpha_n(t+1) = \begin{cases} 0 & \text{if } t \geq \delta_\lambda \times i_\lambda \text{ and } ft_\lambda > t_\delta \\ 1 & \text{otherwise} \end{cases} \qquad (5)$$

where $\alpha_n(t)$ is the binary mask applied to the neuron n depending on the activity of the lateral layer λ at the simulation step t and ft_λ corresponds to the firing time of the lateral layer λ's neuron at the same coordinates. δ_λ is the delay applied to each output's influence on its lateral populations: as the output layers are implemented sequentially, i_λ is the arbitrary identifier given to each layer and by which the δ_λ is multiplied to calculate λ's delay. $\alpha_n(t)$ is null if the corresponding neuron in at least one of the lateral layers is activated — thus hindering the selection of spikes emitted in other layers.

$$\theta(t+1) = \begin{cases} \theta_{max} & \text{if } \alpha(t+1) = 0 \\ \theta(t+1) & \text{otherwise} \end{cases} \qquad (6)$$

where $\theta(t)$ is the neuronal threshold at the simulation step and θ_{max} is the upper bound of the neuronal threshold. If $\alpha(t+1)$ is null, then the corresponding neurons in at least one of the lateral layers λ are active and the neuron of the current layer cannot fire.

$$\forall \lambda \in \lambda_{\text{lateral layers}},$$

$$\theta(t+1) = \begin{cases} \theta(t+1) - \Delta\theta & \text{if } ft_\lambda < t - t_\delta \\ \theta(t+1) & \text{otherwise} \end{cases} \qquad (7)$$

where $\Delta\theta$ is the positive threshold variation at each simulation step and $ft_\lambda(t)$ the firing time of the corresponding neuron in the lateral layer λ.

III. Experimental validation

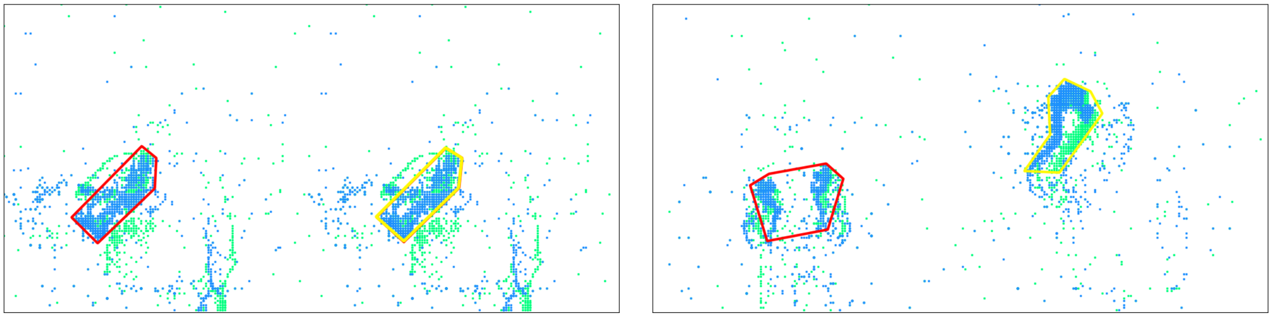
Both architectures described above were implemented with PyNN, a simulator-independent Python interface for SNN simulators [24], combined with NEST (NEural Simulation Tool) [25], a Python simulator for SNN on CPUs. We used these libraries to simulate architectures of 4,224 neurons interacting *via* approximately 8,400,000 synaptic connections and two dynamic adaptation rules[1].

A. Visual input data

This work aims to demonstrate the neuromorphic models' efficient saliency detection and attentional selection of OoIs in an event-based visual scene by verifying their quantitative and qualitative accuracies. To this end, we need an event dataset with a controlled number of OoIs, with known spatiotemporal coordinates. Additionally, as we wish to assess a qualitative aspect by evaluating the performance of a classification task on the output OoIs, the dataset must include labelled objects corresponding to the output OoIs.

Since (to the best of our knowledge) such a dataset does not exist, we artificially created one meeting the criteria described above by combining together various samples from DVS 128 Gesture [23] according to a protocol described below.

1) DVS 128 Gesture: The DVS128 Gesture dataset [23] has now become a standard benchmark in event data classification. It features 29 subjects recorded (with a 128 × 128 pixels DVS128 camera) performing 11 different hand gestures under 3 kinds of illumination conditions. A total of 133 samples are available for each gesture, each composed roughly of 400K events, for a duration of 6 seconds approximately.



A) Control dataset — (left and right) gesture "right arm clockwise" performed by user 5.

B) Random symmetric dataset — (left) gesture "hand clap" performed by user 16 and (right) gesture "right arm counter-clockwise" performed by user 5.

Fig. 3: Samples from control and random symmetric datasets with $n = 2$ and $shift = 0$, constituted from DVS 128 Gesture [23].

[1] This number of neurons and synaptic connections was calculated for two output layers. It was obtained by computing the corresponding values in the saliency detection (see the equations described in Tab. II) added to the number of synaptic connections between the input and the output layers ($2 \times w \times h$) and the number of output neurons ($2 \times w \times h$). We ran the experiments on the custom-made datasets described in Section III.A, where each sample is spatially downsized by 4 (thus $w = 64$ and $h = 32$) using the *eventcount* method introduced in [8].

2) Control and random symmetric combinations: A first custom-made dataset called ”control” was created by randomly selecting 50 DVS 128 Gesture samples, copying them n times and combining them together side-by-side by offsetting their x and y coordinates accordingly. This leads to a pool of 50 samples on which to detect one to n OoIs (i.e. the hand doing the gesture), whose spatiotemporal localisation is approximately known.

A second custom-made dataset was similarly created by randomly selecting $50 \times n$ DVS 128 Gesture sample and combining them together to produce a final pool of 50 samples. As the intrinsic activity measures of each sample differ according to their class and may affect the saliency detection (see Fig. 5), for each combination $n-1$ others are created by permuting the order of the samples in order to create a random but symmetric custom-made dataset of $50 \times n$ elements.

The two datasets described above will be respectively referred to as ”control” and ”random symmetric” in the rest of this paper. An example of both custom-made datasets is shown in Fig. 3, where n is arbitrarily set to 2 to facilitate the reader’s comprehension. It is to be noted that the actual data used as input in the following sections have been spatially reduced by 4 using the event count method introduced in [8] (see Fig. 6 and Fig. 7), due to the limitations of PyNN in terms of the maximum number of simultaneously simulatable neurons and connections. Furthermore, only the first $100ms$ (or $1s$ for any classification task) of each sample was used to reduce the computation time.

3) Shift: Additionally, we introduce some variations within the two kinds of custom-made datasets presented above: the n samples are combined together according to a certain temporal shift, where the $m+1$ sample’s temporal coordinate

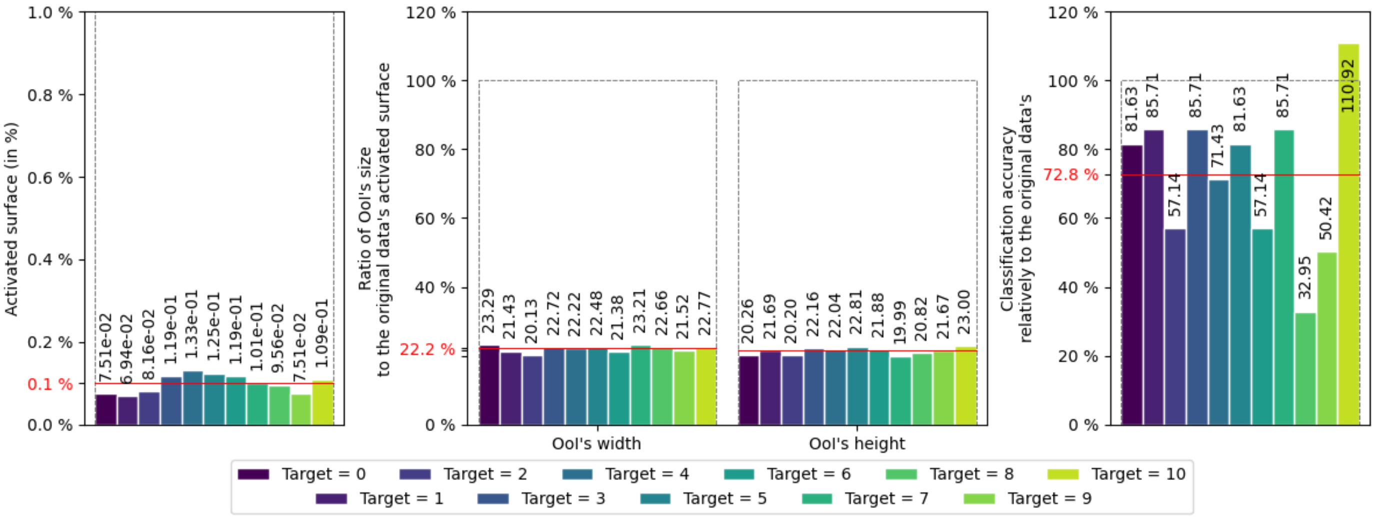


Fig. 4: Evolution of quantitative and qualitative properties of the DVS 128 Gesture dataset after attentional filtering, according to the dataset’s various *targets* i.e. various labelled gestures.

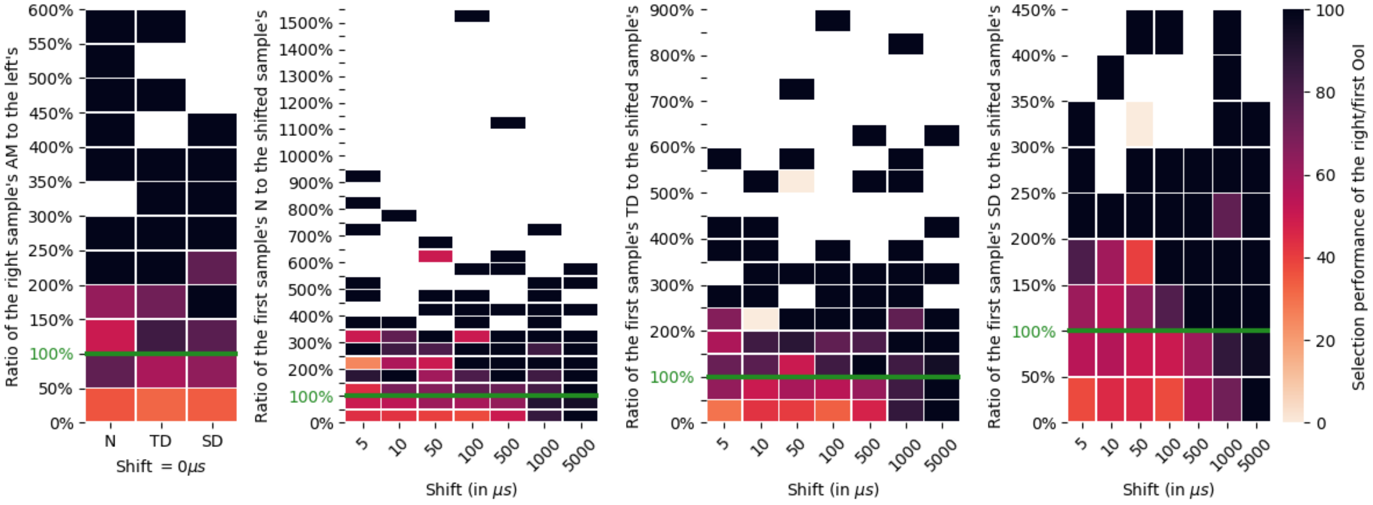
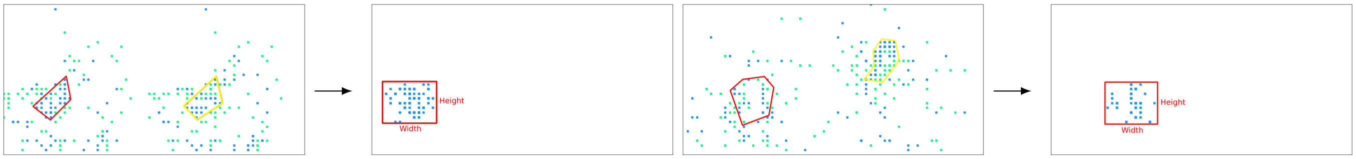


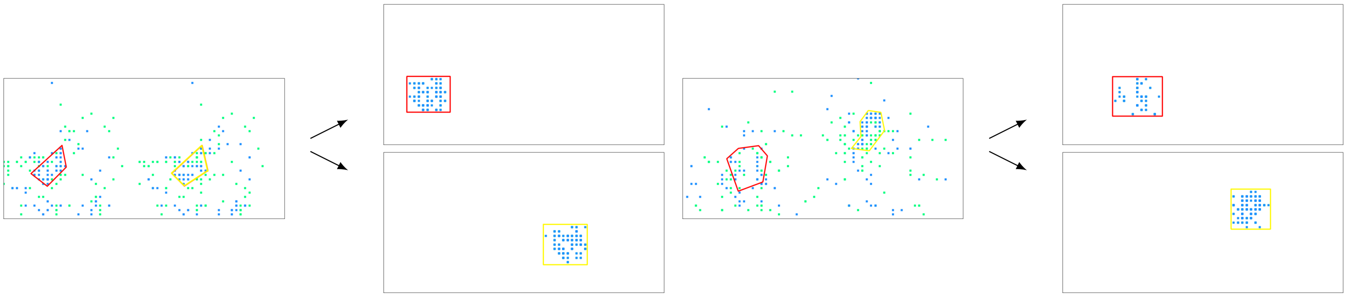
Fig. 5: Performance of the attentional filtering introduced in [5] according to the shift and diverse activity measures. N corresponds to the number of events, TD to the temporal density and SD to the spatial density.



(a) Input reduced data to attentional selection of a "control" sample (see Fig. 3A).

(b) Input reduced data to attentional selection of a "random symmetric" sample (see Fig. 3B).

Fig. 6: Samples from control and random symmetric datasets with $n = 2$ and $shift = 0$, after spatial reduction using the event count method introduced in [8] then attentional selection of one OoI.



(a) Input reduced data to attentional selection of multi-OoIs in a "control" sample (see Fig. 3A).

(b) Input reduced data to attentional selection of multi-OoIs in a "random symmetric" sample (see Fig. 3A).

Fig. 7: Samples from control and random symmetric datasets with $n = 2$ and $shift = 0$, after spatial reduction then attentional selection of one OoI.

is shifted by a factor $shift$ to the m^{th} sample. In the following experimental results, $shift$ varies between 0 (all the combined samples start simultaneously) and $5000\mu s$.

B. Validation of the attentional selection of one OoI

1) One OoI in input data: Firstly we aimed to validate our original attentional mechanism by assessing the evolution of the quantitative and qualitative properties of DVS 128 Gesture [23] before and after attentional filtering. Fig. 4 presents the ratio of various quantitative properties' values of the model's output to those of the original dataset, averaged for each class; as well as the classification performance performed by the Parametric Leaky Integrate-and-Fire (PLIF) classifier [26] on the original and output datasets.

We aim to implement an attentional selection of multiple OoIs that on one side significantly reduce the input data to handle, while on the other side maintaining relatively good quality. Fig. 4 assures us that using our initial saliency detection model allows for an attentional selection of events answering to our needs, with a repartition of events in space reduced by 80% and a classification accuracy maintained at 70% of the original one's.

2) Multiple OoIs in input data: After verifying the performance of the saliency detection model with one OoI in the input data, we now wish to assess its behaviour and its bio-plausibility. Indeed, when confronted with a visual scene with multiple OoIs and asked to select only one, we expect a human being to always select either the one with the highest density of information (i.e. highest number of events and spatiotemporal density) or the one that comes first when the shift between the multiple OoIs is big enough. A bio-plausible saliency detection model would follow a similar behaviour; we thus present our previous model with the pool of custom-made "random symmetric" samples (see Section III.A). Fig. 5 displays the results of this experiment: we can see that as expected, the performance of detection of the first sample increases strongly with the $shift$, reaching nearly 100% for a $shift$ of $1000\mu s$ and higher. On the other side, this performance is strongly degraded when the value of the activity measure of the first sample is smaller than the second one (i.e. where the ratio of the first sample's value to the second is smaller than 100%, highlighted by a green line on the figure). This is confirmed by comparing these results to the one obtained in case of no shift (plot on the left).

We can thus conclude that detecting the saliency in event data according to the event density is bio-plausible: the detected OoI corresponds either to the first object appearing in the scene (with at least a $1000\mu s$ delay compared to the others) or the one with the highest spatial density of events. We can thus indeed call such a model a spatiotemporal attention mechanism. Additionally, Fig. 6 allows for a visual assessment of the quality of the OoI selection on reduced input event data,

TABLE I: Hyperparameters used to implement our contribution, the multi-OoI selection model.

Parameter	**Values**	
	Saliency detector	Outputs
Resting membrane potential	$-65mV$	$-65mV$
Reset membrane potential	$-100mV$	$-65mV$
Neuronal threshold	$-25mV$	$-20mV$
Membrane time constant	$2.5ms$	$25ms$
Refractory period	$0.1ms$	$0.1ms$
Excitatory decay time	$5ms$	$5ms$
Inhibitory decay time	$5ms$	$5ms$
ω_{WTA}	0.5	/
$\Delta\theta$	/	$12mV$
t_δ	/	$50ms$

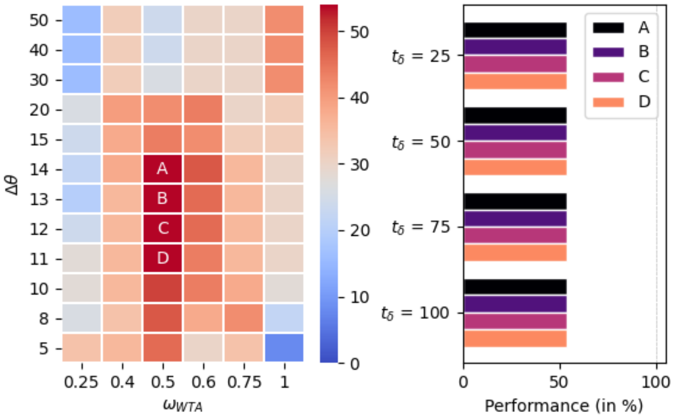


Fig. 8: Performance of simultaneous multi-OoIs detection according to the parameters ω_{WTA}, $\Delta\theta$ and t_δ. The impact of t_δ variation was observed for the parameterizations A ($\omega_{WTA} = 0.5$, $\Delta\theta = 14$), B ($\omega_{WTA} = 0.5$, $\Delta\theta = 13$), C ($\omega_{WTA} = 0.5$, $\Delta\theta = 12$) and D ($\omega_{WTA} = 0.5$, $\Delta\theta = 11$).

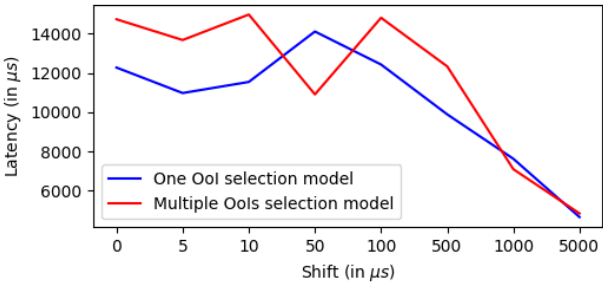


Fig. 9: Latency of the OoI selection in [5]'s (in blue) and our novel architecture according to the shift (in red). With the exception of the dip observed for a $50\mu s$ shift, the overall latency of OoI selection increases with the number of OoIs to be detected (from one in blue to two in red).

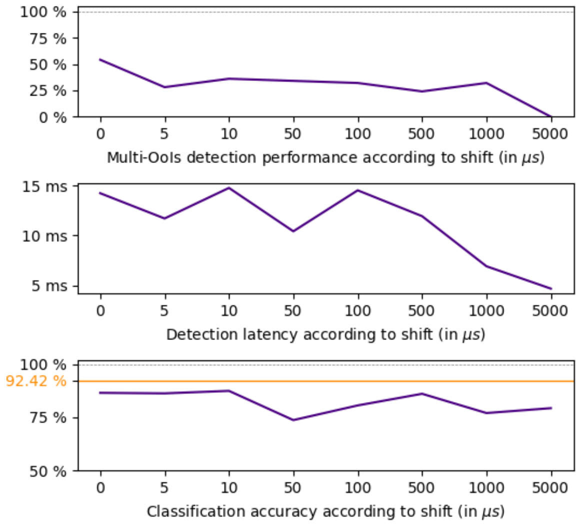


Fig. 10: Multi-OoIs detection performance (top), latency of the OoIs detection (middle), and classification accuracy of the selected OoIs (bottom) of the multi-OoIs selection model on a ”control” dataset.

whether the dataset used is ”control” (Fig. 6a) or ”random symmetric” (Fig. 6b).

C. Quality of the simultaneous attentional selection of multiple OoIs

Fig. 8 presents the evolution of the performance of detection of two different OoIs depending on the tuning of synaptic and neuronal parameters: ω_{WTA} which influences the number and size of detected RoIs (see Eq. 1), $\Delta\theta$ which moderates the impact of salient activity on the output's thresholds (see Eq. 3) and t_δ which delays the impact of the lateral output layers (see Eq.7). It highlights the importance of parameter tuning in SNN, and the difficulty to identify the correct parameters for each dataset — although we can conclude that it seems that t_δ has little impact on the selection performance. According to Fig. 8, in order to optimise the multi-OoI detection performance, this work uses the hyperparameters defined in Table I.

We aim to assess the multi-OoIs selection performance as well as the quality of the output OoIs. Fig. 10 shows the evolution of the performance of multi-OoIs detection according to the temporal shift in the custom-made ”control” datasets, as well as the detection latency and classification accuracy of PLIF [26] compared to the original (in orange). The drop in accuracy compared to the original can be explained by the smaller number of events contained in the multiple output OoIs (see Fig. 7). Indeed, the PLIF classifier [26] accumulates events in frames and therefore performs better on a dataset rich in events.Finally, Fig. 7 allows for a visual assessment of the quality of the OoI selection on reduced input event data, whether the dataset used is ”control” (Fig. 7a) or ”random symmetric” (Fig. 7b).

D. Latency of OoI selection

On an embedded system, the latency of the model's decision is crucial to limit any risk of an accident. We found in our previous work that our ”Neuromorphic Event-Based Spatio-temporal Attention Model rejects more than 50% of incoming unwanted events occurring only 20 ms after activity onset” [5]. One might fear that extending this model to multiple OoI might lead to a significant increase in decision latency — however, Fig. 9 demonstrates that our contribution maintains this latency performance, which is revised to the value of $14ms$ maximum without shift and decreases down to $5ms$ for an increasing shift.

E. Comparison with State-of-the-Art

Table II compares our contribution with the neuromorphic saliency models implemented in [17], [19] as well as with our initial model [5]. The data presented here were either retrieved directly from the information given by the authors or calculated from the description of each model. Those different metrics enhance our proposed model, whose implementation is resource-efficient while implementing additional features with lower latency.

TABLE II: Comparison between our contribution and the state-of-the-art, with input data of size $w \times h$ (w the width and h the height), OL the overlapping percentage described in [17] and div the dividing factor between the input layer and the saliency detector in [5]. A numerical value was calculated for each theoretical estimation, for $w = h = 128$, $OL = 5\%$ and $div = 16$.

	Renner et al., 2019 [19]	D'Angelo et al., 2022 [17]	Our initial model [5]	**Our contribution**
Saliency detection				
n_{layers}	2	10	2	2
$n_{neurons}$	$2 \times w \times h$	$w \times h \times (1 + \frac{4}{OL^2}) + 5$	$w \times h \times (1 + \frac{1}{div^2})$	$w \times h \times (1 + \frac{1}{div^2})$
For $S = 128$:	32,768	19,010	17,408	17,408
$n_{synapses}$	$w \times h \times (2 \times w \times h - 1)$	$w \times h \times (4 + \frac{20}{OL^2}) + 4$	$w \times h \times (1 + \frac{w \times h}{div^4} - \frac{1}{div^2})$	$w \times h \times (1 + \frac{w \times h}{div^4} - \frac{1}{div^2})$
For $S = 128$:	536,854,528	65,540	1,063,936	1,063,936
Selection of OoIs				
Selection of OoI	No	No	Yes	Yes
Simultaneous multi-OoI selection	No	No	No	Yes
Detection latency	NA	$16\mu s$	$13\mu s$	$14\mu s$

IV. Conclusion

This work significantly extends our original preliminary SNN architecture introduced in [5] to attentionally and simultaneously select multiple OoIs in event data. This innovative proposed architecture is able to accurately filter out n objects of interest out of n initially present more than 50% of the time; selects at least one OoI with a delay of less than $15ms$ at most and reaching $5ms$ in the best cases; filters out OoIs with a quality leading to a classification performance reaching 73% the original data's; and achieves all the above with no training phase and a reduced number of neurons and connections compared to state-of-the-art salient detection methods.

In future works, we wish to validate this novel architecture on additional event-based datasets such as the One Megapixel Detection Dataset [27], to an increasing number of n OoIs as well as implement it on neuromorphic hardware, such as SpiNNaker [28] or Kraken, an academic platform that includes a Spiking Neural Accelerators and RISC-V cores [29], directly interfaced with an event camera. We believe in its usefulness in embedded multi-object tracking or scene segmentation.

References

[1] H. Paugam-Moisy and S. M. Bohte, "Computing with Spiking Neuron Networks," in *Handbook of Natural Computing*, Springer-Verlag, 2012.

[2] C. Posch, T. Serrano-Gotarredona, B. Linares-Barranco, and T. Delbruck, "Retinomorphic event-based vision sensors: Bioinspired cameras with spiking output," *Pr. of the IEEE*, vol. 102, no. 10, 2014.

[3] G. Gallego *et al.*, "Event-based vision: A survey," *PAMI*, 2020.

[4] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 128x128 120 db 15 us latency asynchronous temporal contrast vision sensor," *IEEE Journal of Solid-State Circuits*, 2008.

[5] A. Gruel *et al.*, "Neuromorphic event-based spatio-temporal attention using adaptive mechanisms," in *AICAS*, 2022.

[6] D. Gehrig and D. Scaramuzza, "Are high-resolution cameras really needed?," *arXiv*, 2022.

[7] A. Gruel *et al.*, "Performance comparison of dvs data spatial downscaling methods using spiking neural networks," in *WACV*, 2023.

[8] A. Gruel *et al.*, "Event data downscaling for embedded computer vision," in *VISAPP*, 2022.

[9] F. Moosmann, D. Larlus, and F. Jurie, "Learning saliency maps for object categorization," 2006.

[10] A. Borji and L. Itti, "State-of-the-art in visual attention modeling," *IEEE PAMI*, vol. 35, no. 1, pp. 185–207, 2013.

[11] A. Gruel and J. Martinet, "Bio-inspired visual attention for silicon retinas based on spiking neural network applied to pattern classification," *CBMI*, 2021.

[12] M. Cannici, M. Ciccone, A. Romanoni, and M. Matteucci, "Attention mechanisms for object recognition with event-based cameras," WACV, 2018.

[13] M. Iacono, G. D'Angelo, A. Glover, V. Tikhanoff, E. Niebur, and C. Bartolozzi, "Proto-object based saliency for event-driven cameras," in *IROS*, pp. 805–812, 2019.

[14] S. Ghosh *et al.*, "Event-driven proto-object based saliency in 3d space to attract a robot's attention," *Scientific Reports*, vol. 12, no. 7645, 2022.

[15] C. S. Thakur *et al.*, "Neuromorphic visual saliency implementation using stochastic computation," in *ISCAS*, pp. 1–4, 2017.

[16] J. Molin, C. Thakur, E. Niebur, and R. Etienne-Cummings, "A neuromorphic proto-object based dynamic visual saliency model with a hybrid fpga implementation," in *TBioCaS*, vol. 15, p. 580–594, 2021.

[17] G. D'Angelo, A. Perrett, M. Iacono, S. Furber, and C. Bartolozzi, "Event driven bio-inspired attentive system for the icub humanoid robot on spinnaker," *Neuromorphic Computing and Engineering*, 2022.

[18] A. Russell, S. Mihalas, R. von der Heydt, E. Niebur, and R. Etienne-Cummings, "A model of proto-object based saliency," *Vision research*, vol. 94, 2013.

[19] A. Renner, M. Evanusa, and Y. Sandamirskaya, "Event-based attention and tracking on neuromorphic hardware," *IEEE CVPRW*, 2019.

[20] Y. Sandamirskaya, "Dynamic neural fields as a step toward cognitive neuromorphic architectures," *Fr. in Neurosciences*, vol. 7, 2014.

[21] A. Gruel *et al.*, "Stakes of neuromorphic foveation: a promising future for embedded event cameras," *Resarch Square*, 2022. Unpublished, under review at *Biological Cybernetics*.

[22] D. Hebb, "The organization of behavior: A neuropsychological theory," *Journal of the American Medical Association*, vol. 143, no. 12, 1949.

[23] A. Amir *et al.*, "A low power, fully event-based gesture recognition system," in *CVPR*, 2017.

[24] A. P. Davison *et al.*, "PyNN: A Common Interface for Neuronal Network Simulators.," *Fr. in Neuroinformatics*, vol. 2, 2009.

[25] M.-O. Gewaltig and M. Diesmann, "Nest (neural simulation tool)," 2007.

[26] W. Fang *et al.*, "Incorporating learnable membrane time constant to enhance learning of spiking neural networks," in *ICCV*, 2021.

[27] E. Perot, P. de Tournemire, D. Nitti, J. Masci, and A. Sironi, "Learning to detect objects with a 1 megapixel event camera," *CoRR*, vol. abs/2009.13436, 2020.

[28] S. B. Furber *et al.*, "Overview of the spinnaker system architecture," *IEEE Transactions on Computers*, 2013.

[29] A. Di Mauro, M. Scherer, D. Rossi, and L. Benini, "Kraken: A direct event/frame-based multi-sensor fusion soc for ultra-efficient visual processing in nano-uavs," in *HCS*, pp. 1–19, 2022.