Movement Classification and Segmentation Using Event-Based Sensing and Spiking Neural Networks

Paul Kirkland
Neuromorphic Sensor Signal Processing Lab
University of Strathclyde
Glasgow, Scotland
paul.kirkland@strath.ac.uk

Gaetano Di Caterina
Neuromorphic Sensor Signal Processing Lab
University of Strathclyde
Glasgow, Scotland
gaetano.di-caterina@strath.ac.uk

***Abstract*—The development of Spiking Neural Networks (SNN) and the discipline of Neuromorphic Engineering has resulted in a paradigm shift in how Machine Learning (ML) and Computer Vision (CV) problems are approached. At the heart of this shift is the adoption of event-based sensing and processing methods. The production of sparse and asynchronous events that are dynamically connected to the scene is possible with an event-based vision sensor, allowing for the acquisition of not just spatial data but also high-fidelity temporal data. In this work, we describe a novel method for performing instance segmentation of objects, only using their spatio-temporal movement patterns, by utilising the weights of an unsupervised Spiking Convolutional Neural Network that was originally trained for object recognition and extending it to instance segmentation. This takes advantage of the network's spatial and temporal characteristics encoded within its internal feature representation, to offer this additional discriminative ability. We demonstrate this through a track path identification problem, where 6 identical blobs complete complex movement patterns within the same area at the same time. The network is able to successfully identify all 6 individual movements and segment the movement patterns belonging to each. The work then also explains how these methods map into the more complex Track before Detect problem. A complex track initiation problem, where detection can only be completed after an integration period, due to the low signal, high noise environment. These problem characteristics seem to complement the properties of event-based sensing and processing and initial test results are shown.**

***Index Terms*—Neuromorphic Engineering, Neuromorphic Algorithms, SNN, STDP, Computer Vision, Unsupervised Learning, Instance Segmentation, Event-Based Vision**

I. Introduction

In most defence applications, identification of any target is a time-sensitive and crucial function. However, it is not only detection and identification that is vital, as the exact location is also an important consideration. With the recent take over of deep learning (DL) in the computer vision domain, much research and effort have gone into turning the state of the art in object detection [1], [2], into the instance recognition of video information [3]. However, the reality of the situation in a defence scenario is that the target object is often extremely small (one or few pixels), and it contains no relevant spatial information to discriminate it from background noise and clutter. This then rules out the idea of performing frame-based detection. In cases like this, the requirement for a recurrent approach to allow the accumulation of information over time is required [4]. However, the drawback to this solution is that the longer the integration period, the higher the computational overhead required. Once this issue gets into the low signal or low signal high noise realm, where methods such as Track before Detect (TBD) are used, then DL approaches appear to have had a minimal impact [5].

Neuromorphic Engineering introduces a new paradigm to the sensing and processing domain with the use of event-driven asynchronous sparse binary information. Taking inspiration from biological systems, Neuromorphic sensor signal processing aims to take methods from the breadth of the machine learning community, including DL, and to combine them with the new event-based method of sensing data. This way of thinking is driven by innate abilities that exist in nature. For instance, even in the presence of various background and foreground distractors, human vision has the natural ability to recognise, localise, and discriminate items of interest. This is all done in real-time within a minimal power budget, usually while also completing a number of other complex tasks. Neuromorphic simply means brain-like, in that biological inspiration is taken in how to handle information. Specifically, this makes use of event-driven binary spikes, rather than numerical values, to sense and process data. This means the information precision lies in the timing or rate of the spikes rather than in their magnitude. Neuromorphic sensors give a high temporal resolution without the computational burden, while the event-driven nature of the sensing means the processing would naturally accumulate information over time. This results in high fidelity spatio-temporal patterns to be resolved, where detection and localisation are computed simultaneously.

Neuromorphic, or event-based, sensors have matured over recent years, with vision sensors becoming particularly popular. So much so even consumer products are available, as for example the asynchronous time-based image sensor (ATIS [6]), backed by Sony and sold by Prophesee, and the Dynamic Vision Sensor (DVS [7]), backed by Samsung and sold by Inivation. Event-based sensing is done typically

This work is funded by the Defence Science and Technology Laboratory (DSTL) under the DASA Advanced Vision 2020 Project (ACC6010078 - Neuromorphic processing detection and tracking of fast moving targets) contract DSTLX1000147830.

This is a peer-reviewed, accepted author manuscript of the following article: Kirkland, P., & Di Caterina, G. (Accepted/In press). Movement classification and segmentation using event-based sensing and spiking neural networks. In Sensor Signal Processing for Defense IEEE.

through change detection, where a large enough change in the signal causes the sensor to output spikes. The level of this change can be set on the sensor to ensure a suitable output. This change detection greatly helps to sparsify the output. The spikes output by the sensor then represent a high resolution and asynchronous temporal record of the changes occurring in the scene. Even though there is a high degree of spatial and temporal resolution, the data is still sparse compared to a traditional frame-based imaging approach, since not every pixel changes at the same time. This means the sensor has a dynamical relationship to the scene. To exploit this feature, we pair the sensing with a processing method that has a variable integration period, thus capturing the movement period precisely and collecting the relevant information.

Neuromorphic processing is typically carried out using the 3rd generation of neural networks, referred to as Spiking Neural Networks (SNN). The SNN exhibits properties such as asynchronous and event-driven processing, fast inference, low power consumption, massive parallelism and online learning. All of which makes it an interesting prospect in many applications, and ideal for processing information that requires integration over time. In this sense, it means the SNN benefits from not requiring recurrency to extract sequential or temporal information, as such networks are naturally time-dependent. Another benefit of the SNN is that it can exploit being related to Artificial Neural Networks (ANN), as methods of feature extraction can be ported from one to the other. One such method, the Convolutional Neural Networks (CNN), is an efficient and effective method for both learning and extracting features, due to the natural local continuity of objects in both space and time.

II. Algorithmic Development

The main theme to the algorithmic work is to exploit the SNN in the application of instance segmentation. The detection and pixel-wise delineation of each separate item of interest present in a picture is known as instance segmentation. In essence, instance segmentation is a mixture between object recognition and semantic segmentation, two important computer vision problems. Detecting instances of things belonging to a specific class, while simultaneously determining their physical position, usually using a bounding box, is known as object detection. Semantic segmentation, on the other hand, is the challenge of grouping areas of an image that belong to the same object class together, resulting in a considerably more thorough pixel-wise localisation. This problem only gets more complex when attempted on video instead of images, as now the processing time must be less than the time interval available until the next frame, otherwise extra latency is added to the system. For fast-moving objects or scenes, this only becomes more difficult as the rate at which one senses must increase, i.e. higher frame rate, thus forcing a shorter processing interval available.

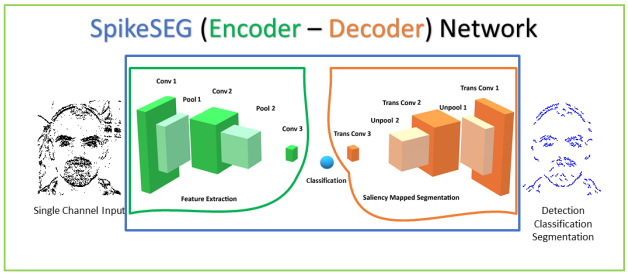

Fig. 1. The SpikeSEG network used to segment spiking images. The encoder is featured in green and the decoder is featured in orange.

A. Spatial Scene Understanding

To initially approach the extraction of useful spatial features from a spiking event-based scene, this work borrows from the previous own SpikeSEG [8], which details how a convolutional encoder-decoder network can be utilised to extract commonly occurring spatial features in a scene (within the encoder). Then it maps this semantically contextualised information into the pixel space again (through the decoder). This in essence allows semantic segmentation to be performed on spiking event data within an unsupervised regime. An example of the network along with an input/output is shown in Fig. 1. The network architecture illustrated here is made up of two main sections seen in green and orange, that relate to the encoding and decoding layers respectively. The network is split into these two sections where training only occurs on the encoding side, while the weights are tied to the mirrored decoding layers. This allows an integrate and fire neuron with layer-wise STDP mechanism, and with adaptive thresholding and pruning, to be used to help represent spatial features of the input. These features are then learned through the encoder, which in turn allows the decoder to segment the image based on the *Conv3 / Trans Conv3* pseudo classification layers.

This encoding-decoding structure symbolises a feature extraction and then a shape generation process. The learning of the encoding process aims to extract common spatial structures as useful features, then it decodes those learned features over to the shape generation process, unravelling the latent space classification representation, although with a reduction in spiking activity due to the max-pooling process. The network has 9 computational layers *(Conv1-Pool1-Conv2-Pool2-Conv3-TransConv3-UnPool2-TransConv2-UnPool1-TransConv1)* as seen in Fig. 1. Between the *Conv3* and *TransConv3* layers, there is a user-defined attention inhibition mechanism / classification, which can operate in two manners: 'No Inhibition', which allows semantic segmentation of all recognised classes from the pseudo classification layer; or 'With Inhibition', which only allows one class to propagate forward to the decoding layers. This attention not only provides a reduction in the amount of computation, but also simplifies the output of the network, for simpler handover to downstream systems. For further

information contained within this section regarding the process of encoding, decoding, thresholding and pruning, see [9]. Fig. 2 helps visualise the internal working of the network. This illustration details the internal network dynamics, with each coloured pixel representing the corresponding region's feature map activation. Classification is the joint representation of *Conv3* and *TransConv3*, which in this case would be the same, as only one class is present.

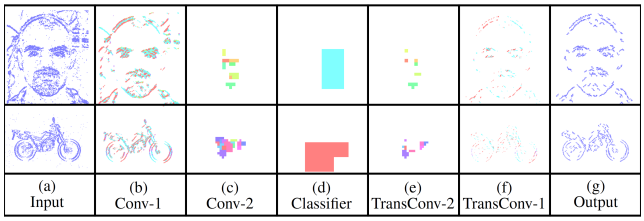

Fig. 2. The internal network representation of SpikeSEG for two class examples.

B. Featural-Temporal Decomposition

Building upon the successful feature extraction of the SpikeSEG network, it was noted that items within particular classes seemed to exhibit rather unique temporal patterns in which the features (neurons) inside the network would be active. This can be simply explained due to the STDP process of learning by looking for the most salient and occurring features. Therefore the more salient the feature, the larger likelihood it would be activated earlier in time. From this hypothesis, the Hierarchical Unravelling of Linked Kernels (HULK) and Similarity Matching through Active Spike Hashing (SMASH) algorithms were designed.

HULK is the process of taking each spiking instance from the last layer of the encoder and unravelling its path through the decoder, no longer at a semantic level, but at the instance level. So for each spike in that feature map, one can track it back to the pixel space, rather than doing it from all the spikes in any given feature map, as was shown previously in Fig. 2. Instead, there is a more granular process now as shown within Fig. 3, which depicts a flow chart of the HULK SMASH process, along with examples from sections of the process [9]. The image highlights the process starting with the SpikeSEG network, but looks at each spike within the last convolution layer leading to the HULK ASH image. Another representation for this featural-temporal representation is shown just below with the red and blue spike trains, which highlight the differences more clearly. The final example image depicts the SMASH process, where the similarity and proximity scores are combined to decide on the number of instances present in the image. Further details regarding specific parts of the process can be found in the [9].

Overall the HULK SMASH process was able to show that not only is the spatial feature information useful in identifying objects within the image, but also that the temporal sequencing in which the features occur can be utilised for more specification identifications. This finding underpins the importance of the temporal nature of the spiking event data: it is the ability to encode the saliency of features, simply by allowing them to occur earlier than less salient features.

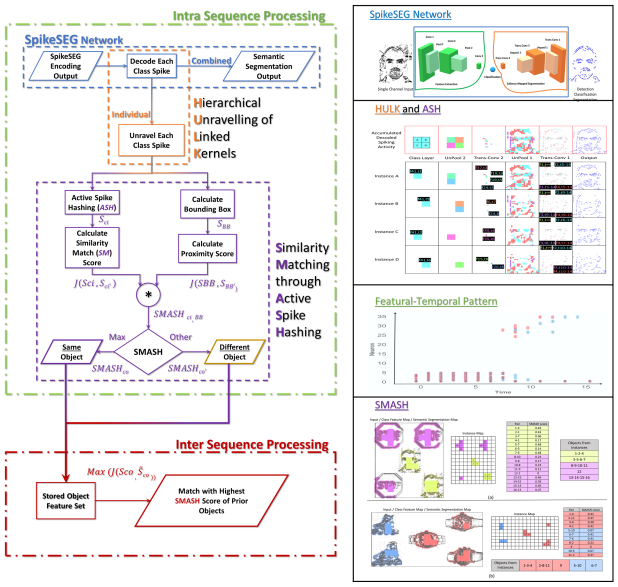

Fig. 3. Flow chart for HULK SMASH with examples for each section.

C. Spatio-Temporal-Featural Decomposition

Once it was established that featural-temporal information could be extracted from the spatial features of the spiking event data, the next step was to test the feature extraction ability on spatio-temporal information. As such, the spatial information alone is not representative of anything meaningful, so a longer integration period is required to ascertain if there is a temporal component to the spatial information presented. This was tested under the assumption of an unknown object (small dot) completing a set number of movement patterns, as seen in Fig. 4. It would then be the movement pattern that would be the identifying feature of the data. The SpikeSEG network allows a temporally invariant classification of known movement patterns to be determined, while the HULK process re-enables the temporal variance to further determine the temporal aspect of the feature occurrence. In essence, it allows the system to further resolve if the movement was completed fast or slow. An example of the feature breakdown is shown per layer in the encoder and decoder in Fig. 5. This is a time integrated view of the accumulation of features showing the mapping from pixel to classification latent space and back to pixel domain. The HULK and ASH process ensures the temporal continuity is also captured to be used for further comparing and contrasting of event sequences.

III. Use Case Demonstration

The demonstrator envisages a particularly challenging tracking example, where the three previously mentioned movement

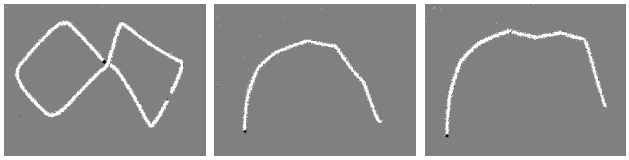

Fig. 4. Movement patterns.

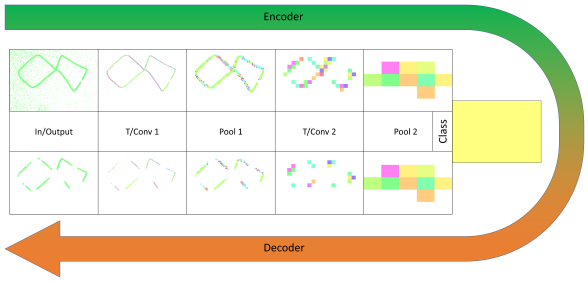

Fig. 5. HULK breakdown of spatio-temporal features.

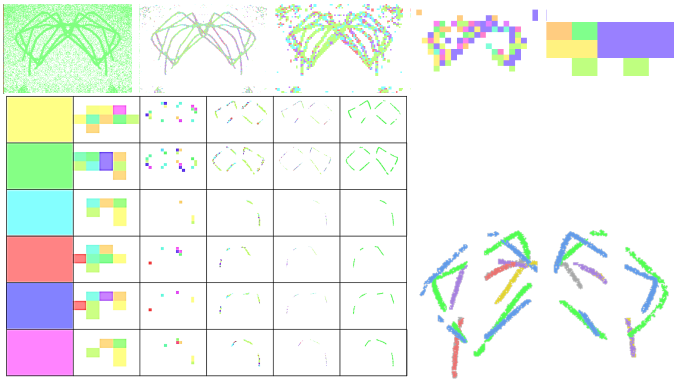

Fig. 6. Breakdown of the features used to segment one movement from the multi-movement scene, with final coloured segmentation

patterns are occurring in close proximity to one another and at about the same time. This happens along with the mirrored (over the y-axis) version of the events. The resultant integrated over time image for this scene is shown in top left of Fig. 6. The difficulty in this task is that spatially the target object that is moving around it the same in all examples. It also is occluded and crosses paths of the other targets, together with some unpredictable movements (i.e. figure of eight). This demo is supposed to rule out the possibility of just simple identifying each of the moving targets as individuals, instead meaning one relies on the movement of the target, to be able to classify it.

Testing of this complex scenario highlights the strength of the SpikeSEG and HULK SMASH methods. A breakdown of the integrated feature extraction process for the whole multi-target movement scene is illustrated in Fig. 6, where there is a high degree of spatial overlap from the scene which is represented within all the feature extraction layers. However, due to the high temporal resolution of the event data from the scene, the spatio-temporal overlap of the target is rather minimal. This results in only minimal overlap of features allowing the movement patterns to be resolved, as shown in Fig. 6.

The accumulated result of this is that the 6 movement patterns can be distinguish between as seen in Fig. 6, where although there was a large overlap in the spatial location of the movements, each movement path could be classified and segmented.

The image appears fragmented as the pooling layers are still active on the decoding side, meaning only the most relevant information passes through to the pixel space again. This was to ensure the output of the network was more specific than it was sensitive. The high degree of spatial overlap means that certain regions were not the most salient in terms of the classification process and therefore are not shown in the segmentation. The segmentation is quite literally a saliency mapping of the found features. However, now the output of the network is an instance and semantically contextualised version of the input. Meaning, that if only wanting to look for a figure of eight movements, one could inhibit all the other classes, and the output from the demonstrator would only show the two figure of eight movements. A number of spatial and temporal variations of this demo were tested (i.e. X,Y displacement, time displacement, temporal continuity). The SpikeSEG network was able to semantically classify each movement successfully, while the HULK SMASH algorithm was able to determine instances within the classes. As such, it was possible to notice changes in the temporal structure of the spiking event data (i.e. the scene was faster/slower than the previous, and if the features occurred in a different order). This means the system is invariant to movement and the location of the movement within the scene does not matter, while being temporally variant, as the timing of the occurrence of the features does matter. This clarifies that the SpikeSEG network is invariant to both space and time, while the HULK SMASH algorithm adds the variance to the feature data. This is only permitted due to the SpikeSEG network being an asynchronous processing Spiking Convolutions Neural Network, which maintains the temporal continuity of the incoming data due to the neurons firing, even though the network itself is invariant to time.

A. *Track before Detect Problem*

This section covers the initial testing that has been carried out using the same network as described above, but in the situation of a low signal to noise ratio (SNR). This problem is highly related to the principle behind Track before Detect (TBD), as detection is based on tracking or accumulating information on any objects of interest within a scene. However, the time scales required for movement detection are far shorter than that required in the previous classification task. Regardless, it became clear that neuromorphic sensing and processing could be utilised to great effect in the more challenging TBD domain. The neuromorphic event-

based sensor allows for the accumulation of spatio-temporal information on higher fidelity and variable/incoherent scale, due to its high temporal resolution and asynchronous readout. This means the sensor can accumulate small enough amounts of time to detect pixel motion, while mitigating the effects of the sensor noise and clutter. Fig. 7 illustrates how a longer integration time has lower levels of SNR (left), compared to a less noisy shorter integration time (right). It is in this non-linear relationship between the signal and noise where the benefits of an asynchronous approach are most seen, which is somewhat similar to the benefits of incoherence in randomly sampling for PF. This asynchronous sensing also allows a dynamical relationship to movement in the scene, meaning those moments when movement is detected can be extracted as needed, exploiting the ability to collect high SNR values over this short period. This is in contrast to the fixed temporal rate in a traditional frame-based imaging sensor, which will just accumulate over a set period irrespective of signal movement.

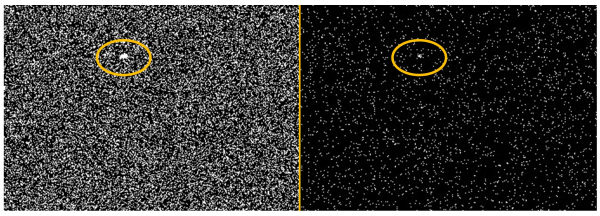

Fig. 7. High and low noise due to integration time.

Preliminary testing of our previously designed systems on an example of a TBD problem has resulted in very encouraging results, with a similar laser pointer example as shown earlier, creating a non-distinct moving blob, but with a high level of noise present due to the closing of the aperture of the sensor. This results in a very low SNR value of around -21dB for the movement sequence (based on signal strength captured in a relatively noise-free environment compared to a signal-free noise environment). This scenario was initially tested against simple implementations of a Kalman filter and a particle filter in both the high and low SNR scenarios. All three systems are not optimised for the task, but manage to perform tracking very well on the clean data. However, when tested on the highly noisy data, only the neuromorphic processing can extract the moving point, as illustrated in Fig. 8. The Kalman filter case shows two predicted points, one of which is close to the object, including briefly tracking the point, but then losing it. The particle filter case shows the particles as a red plus and the mean point a yellow star, and none of the particles are aligned with the object. The SNN case shows the output of the system with only the pixels that were first activated in the system (time to first spike), so operating on a single layer spiking convolution process with a matching encoding and decoding layer, then inhibiting all other feature neurons to produce this output. For this to work in a continuous asynchronous manner, the time to first spike method would need to be changed to a rate-based approach based on spatio-temporal correlation neurons firing.

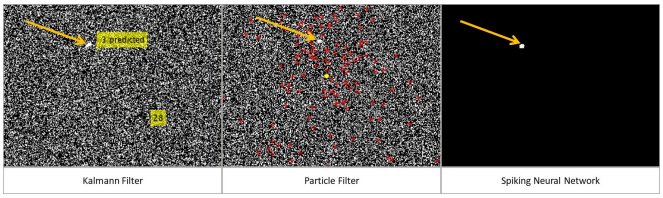

Fig. 8. Output from noisy data for Kalman Filter, Particle Filter and Spiking Neural Network.

IV. Conclusion

In this paper, we have presented how the paradigm of neuromorphic engineering and its event-based sensing and processing can provide an efficient and effective method of extracting complex spatio-temporal patterns from a visual scene, without the requirement for recurrency. This method is then also shown to have promise in TBD, a more relevant defence scenario of low SNR track initiation. Here engineering and its event-based sensing and processing can allow recovering the movement pattern from a highly noisy scene by exploiting the non-linear relationship between the noise distribution and the movement induced signal.

References

[1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[2] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*, pp. 213–229, Springer, 2020.

[3] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "Yolact: Real-time instance segmentation," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9157–9166, 2019.

[4] G. Ning, Z. Zhang, C. Huang, X. Ren, H. Wang, C. Cai, and Z. He, "Spatially supervised recurrent convolutional neural networks for visual object tracking," in *2017 IEEE international symposium on circuits and systems (ISCAS)*, pp. 1–4, IEEE, 2017.

[5] E. Peters and J. Roecker, "Hybrid tracking of low snr targets," in *2021 IEEE Aerospace Conference (50100)*, pp. 1–6, IEEE, 2021.

[6] C. Posch, D. Matolin, and R. Wohlgenannt, "A QVGA 143 dB Dynamic Range Frame-Free PWM Image Sensor With Lossless Pixel-Level Video Compression and Time-Domain CDS," *IEEE JOURNAL OF SOLID-STATE CIRCUITS*, vol. 46, no. 1, p. 259, 2011.

[7] P. Lichtsteiner, C. Posch, and T. Delbruck, "A 120 dB 15micro s Latency Asynchronous Temporal Contrast Vision Sensor," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 2, pp. 566–576, 2008.

[8] P. Kirkland, G. Di Caterina, J. Soraghan, and G. Matich, "Spikeseg: Spiking segmentation via stdp saliency mapping," in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2020.

[9] P. Kirkland, G. Di Caterina, J. Soraghan, and G. Matich, "Perception understanding action: adding understanding to the perception action cycle with spiking segmentation," *Frontiers in Neurorobotics*, vol. 14, 2020.