2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Spike-based high energy efficiency and accuracy tracker for Robot

Jinye Qu, Zeyu Gao, Yi Li, Yanfeng Lu, and Hong Qiao

Abstract— **Spiking Neural Networks (SNNs) have gained attention for their apparent energy efficiency and significant biological interpretability, although they also face significant challenges such as prolonged latency and suboptimal tracking accuracy. Recent studies have explored the application of SNNs in object tracking tasks. Dynamic visual sensors (DVS) have become a popular way to implement SNN-based object tracking due to their asynchronous and spiking characteristics similar to SNNs. However, challenges such as the high cost of DVS cameras and the lack of object surface texture information hinder the utility and performance of DVS trackers. In contrast, RGB information has inherent advantages, including low acquisition cost and comprehensive object surface texture representation. However, RGB information is prone to excessive image blurring in low-light conditions or in fast-motion scenes. To address these challenges, we propose the "Motion Feature Extractor" and the "RGB-DVS Fusion Module". The "Motion Feature Extractor" can replace the DVS camera at a very low cost, and the "RGB-DVS Fusion Module" can deeply fuse the feature information of the two to make up for their respective deficiencies. In addition, we adopt a conversion method to obtain a lossless SNN version of the model. Through experiments, our model achieves a 13.6% improvement in the expected average overlap (EAO) index using only 1.47% of the energy consumption of SiamRPN (VOT2016 dataset). In addition, we deployed the model to a robot and then conducted tracking experiments, which confirmed that the model can operate on the robot losslessly with satisfactory results.**

I. INTRODUCTION

Spiking Neural Networks (SNNs) are efficient and low-power models that emulate brain dynamics. The Leaky Integrate-and-Fire (LIF) and IF models are widely used in SNNs to simulate biological neuron behavior and to understand the temporal dynamics of membrane potential changes in response to synaptic inputs [1], [2].

Single-object tracking using Spiking Neural Networks (SNNs) and Dynamic Vision Sensors (DVS) is gaining significant attention [3]–[7], [7]. DVS technology captures pixel-level brightness changes asynchronously, aiding in tracking fast-moving objects and operating in low-light conditions [8]–[10]. Combining DVS with SNNs enhances dynamic object tracking capabilities. Recent advances include twin-network trackers that use the LIF and Siam models for efficient event-based tracking [11], [12]. For instance, STNet [13] uses greyscale maps of RGB frames combined with event streams for tracking, achieving relatively good results. However, it does not globally integrate RGB frame signals, which restricts its tracking capability to dynamic objects only.

Robot tracking is a key area of robotics research, with most current methods relying on particle filtering [14] due to real-time performance needs, but its accuracy in complex scenes is impacted. While deep learning-based approaches offer improved accuracy [15], their high energy demands limit deployment on robots. Spike-based trackers can leverage SNNs to enable high-precision tracking efficiently on robots, combining low energy consumption with speed. However, the challenge lies in the ambiguity of information in dynamic scenes, which exacerbates the accuracy limitations of SNN-based tracking.

In summary, current technologies cannot fully leverage the advantages of both DVS and RGB frames. Additionally, existing SNN tracking methods cannot achieve efficient tracking in dynamic scenes. To address these challenges, our contributions are summarized as follows:

- Considering the high cost of current DVS cameras, we propose a low-cost alternative called Motion Feature Extractor (MFE). Experiments show that MFE effectively replaces DVS functionality with excellent moving object extraction capability.
- We introduce a fusion module that integrates RGB-DVS information, allowing seamless incorporation of both into neural network inputs, combining RGB frame richness with DVS's dynamic response to enhance tracker performance.
- We propose a spike-based high-performance RGB-DVS dual-input tracker with significantly reduced energy consumption.
- Our approach achieves SOTA results on challenging tracking datasets, with a 13.6% improvement in expected average overlap (EAO) while using only 1.47% of the energy of the original siamRPN model. The model was also successfully deployed on a mobile robot platform and tested in real dynamic scenarios with satisfactory results.

II. RELATED WORK

A. Dynamic Vision Sensor (DVS)

The DVS [16] is a bio-inspired optical sensor that captures dynamic changes by outputting events based on changes in pixel intensity, rather than recording video at a fixed frame rate. It generates events with location, timestamp, and polarity data whenever a change in light intensity exceeds a threshold.

Jinye Qu, Zeyu Gao, Yanfeng Lu, and Hong Qiao are with the State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Science (CASIA), Beijing 100190, China, and are also with the University of Chinese Academy of Sciences (UCAS), Beijing 100049, China. Yi Li is with the College of Information Science and Technology, Beijing University of Chemical Technology, Beijing,100029, China, and school of Information Engineering, Nanchang University, Nanchang, 330031, China. Corresponding author: Yanfeng Lu. yanfeng.lv@ia.ac.cn

B. SiamRPN

Siamese Region Proposal Network (SiamRPN) [17] advances visual object tracking by combining a Siamese network for feature extraction with a Region Proposal Network (RPN) for object localization. This approach balances speed and accuracy, making it ideal for real-time applications. The Siamese network learns feature representations, while the RPN localizes objects within the frame.

C. Spiking Neural Networks

Conventional artificial neural networks (ANNs) mimic biological neural networks but are energy-intensive, unlike the human brain, which operates on about 30 watts. Spiking Neural Networks (SNNs) improve energy efficiency by using electrical pulse signals, employing Leaky Integrate-and-Fire (LIF) and Integrate-and-Fire (IF) neurons to replicate electrophysiological properties, as described by Eq.1.

$$\tau_m \frac{dV}{dt} = -(V(t) - V_{\text{rest}}) + RI(t), \tag{1}$$

where τ_m is the neuron's leakage index, constant at 1 for IF neurons and variable for LIF neurons. V represents the membrane voltage, $I(t)$ is the external input current, V_{rest} is the resting potential, and R is the input resistance.

When the membrane potential V hits the threshold V_{thr}, the neuron fires a spike, causing the potential to decrease as described by Eq.2.

$$V = V - s * V_{thr}, \text{if } V \geq V_{\text{th}}, \tag{2}$$

where s refers to the spike. In practical applications, the IF neuron model in SNNs is often paired with a synapse model to simulate neuron connections and information transfer. However, SNNs face challenges such as limited structural support, accessibility issues, low performance, and weak hardware support, leading to their use only in simpler tasks. This paper seeks to address these issues by enhancing the scale and complexity of SNN models through algorithm optimization using a mainstream ANN to SNN conversion method.

D. ANN to SNN

In recent years, the conversion from ANNs to SNNs has become a prominent topic, with numerous methods advancing the technology significantly. The subtraction reset method [18], while retaining spike intensity information, abandons the fixed reset potential. The Temporal Separation (TS) [1] method introduced in 2022, which separates the accumulation and firing phases, effectively reduces errors caused by incorrect spike emission sequences. T. Bu and others proposed a membrane potential initialization theory [19], suggesting that initializing the membrane potential to half of the threshold can achieve lossless conversion. Y. Hu and others developed a conversion method for deep ResNet [20], addressing the challenges posed by bottleneck structures.

III. METHOD

In this section, we describe the generation of motion feature information, the fusion of RGB-DVS information, and the conversion of the siamRPN model.

A. Motion Feature Extractor(MFE)

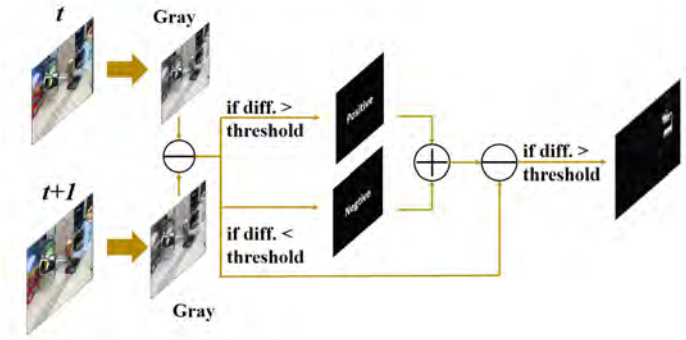

Fig. 1. Motion feature extractor(MFE), which generates low-noise object motion (DVS) information. *pos_threshold* is the positive luminance change threshold, and *neg_threshold* is the negative luminance change threshold.

DVS cameras excel at capturing dynamic objects due to their event-driven design, generating asynchronous information based on pixel brightness changes. However, their high cost limits widespread use at the edge. Most current tracking research [10], [21] relies on accumulating information within a time window to extract motion features. To address this, we introduce the Motion Feature Extractor (MFE), a cost-effective algorithm for motion feature extraction without requiring expensive DVS cameras.

The motion information generation, as detailed in Alg.1 and Fig.1, begins by converting adjacent RGB frames (f_{RGB}) to greyscale maps (f_{Grey}). The global luminance change (G) is derived by subtracting these greyscale maps. Pixels with G above the positive threshold are set to 255 (G_{pos}), those below the negative threshold to -255 (G_{neg}), and others to 0, forming the initial motion feature (F). This feature is then refined by subtracting global luminance variation, followed by thresholding: pixels above the threshold are set to 255 for positive motion features (F_{pos}), and those below are set to -255 for negative motion features (F_{neg}). Only positive motion features are used in this work.

B. DVS-RGB Fusion Module

We aim to leverage the event-driven nature of DVS to enhance the tracker's focus on the target object. Ideally, DVS excels when the target is dynamic and the background static. However, if the background is dynamic and the target static, DVS can introduce significant noise, reducing the effectiveness of region-of-interest delineation. To address this, we propose an RGB-DVS fusion module.

The architecture of the fusion module, detailed in Fig. 2, involves splitting the DVS signal into two branches. Branch

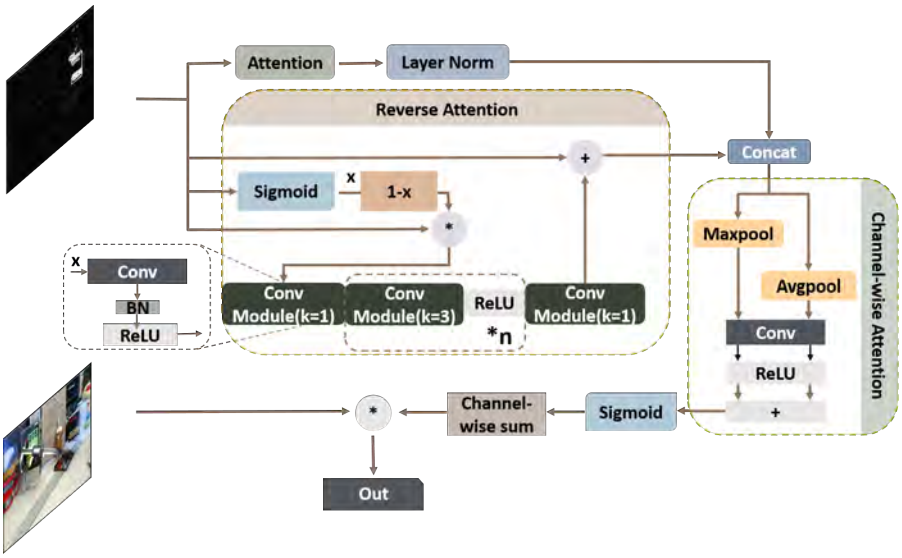

Fig. 2. RGB-DVS Fusion Module Architecture: DVS information is separated into target and background using attention and reverse attention modules, then combined in channels. The channel attention module identifies the focus area, which is then multiplied with RGB information to emphasize the target object.

Algorithm 1 Algorithm of MFE

Set $threshold_{pos}$, $threshold_{neg}$
Input: $f_{RGB}(t)$, $f_{RGB}(t+1)$
Output:
$f_{Grey}(t) \leftarrow Grey(f_{RGB}(t))$
$f_{Grey}(t+1) \leftarrow Grey(f_{RGB}(t+1))$
$G \leftarrow f_{Grey}(t+1) - f_{Grey}(t)$
Initialize all-zero frames G_{pos}, G_{neg}
$G_{pos}[G > threshold_{pos}] \leftarrow 255$
$G_{neg}[G < threshold_{neg}] \leftarrow -255$
$F \leftarrow G_{pos} + G_{neg}$
$F \leftarrow F - G$
Initialize all-zero frames F_{pos}, F_{neg}
$F_{pos}[F > threshold_{pos}] \leftarrow 255$
$F_{neg}[F < threshold_{neg}] \leftarrow -255$
return F_{pos}, F_{neg}

1 uses the Attention module [22] to extract valid target information, as described in Eqs.3 and 4.

$$x_{\text{Att}} = Attention(x), \tag{3}$$

$$x_{\text{LN}} = LN(x_{\text{Att}}), \tag{4}$$

Attention denotes the Attention function, *LN* is the layer norm, and x represents the DVS input, all with an embedding dimension of 127 and an input size of 127x127. Since template and search frames differ in size, event information is upsampled or downsampled using 'nearest' sampling, then restored post-attention. Branch 2 utilizes the Reverse Attention [23] module to determine the probability that a point is background, as shown in Eq.5.

$$x_{\text{rev}} = RA(x), \tag{5}$$

Where *RA* denotes the reverse attention function. The outputs of the two branches are concatenated along the channel dimension and then processed through a Channel-wise attention layer [24] to extract the relevant background and object information, as detailed in Eqs.6 and 7.

$$x_{\text{Cat}} = Concat(x_{\text{LN}}, x_{\text{rev}}), \tag{6}$$

$$x_{\text{CA}} = CA(x_{\text{Cat}}), \tag{7}$$

Concat refers to concatenation, and *CA* denotes channel-wise attention. In Fig. 2, the reverse attention layer includes two pooling layers with an output size of 1. The pooled channel attention values are convolved with $k = 1$ and ReLU, then summed for output. The resulting x_{CA} is processed with a sigmoid function, and the two-channel values are merged into a single channel, as shown in Eqs.8 and 9.

$$x_{\text{Sig}} = Sigmoid(x_{\text{CA}}), \tag{8}$$

$$x_{\text{DVS}} = x_{\text{Sig}}[0] + x_{\text{Sig}}[1]. \tag{9}$$

The processed DVS signal highlights object positions and reduces noise from the dynamic background. Finally, the RGB frames x_{RGB} are multiplied by the processed DVS signals x_{DVS} to emphasize regions of interest based on DVS data while utilizing the rich features of the RGB frames for object recognition, as shown in Eq.10.

$$x_{\text{fea}} = x_{\text{DVS}} * x_{\text{RGB}}. \tag{10}$$

C. Spiking-Based Tracker

Building on the RGB-DVS fusion tracking strategy, we develop a spiking neural network (SNN) model for enhanced compatibility with spiking inputs. SNNs offer faster processing and greater energy efficiency, making them suitable for deployment on mobile robots.

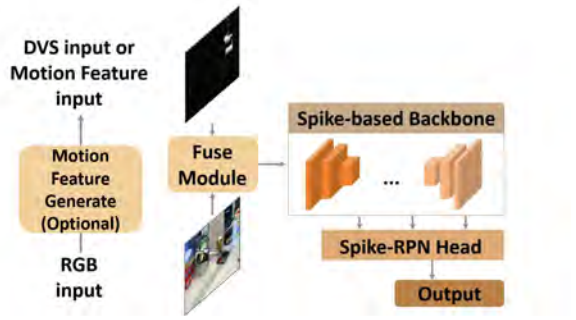

Fig. 3. Fusion tracker framework. Where the pseudo-DVS signal is generated using MFE.

1) Overall Architecture: The overall architecture of the model, shown in Fig.3, includes the fusion module, backbone, and Spike-RPN head. We use Spike-ResNet50 as the backbone. The Spike-RPN head utilizes a bottleneck structure that sums spikes directly. The model's backbone can be adjusted based on efficiency and precision needs. To minimize energy consumption, the model omits an additional encoder, using only the first layer of IF neurons for encoding.

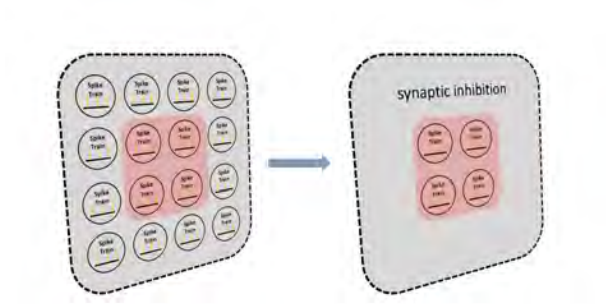

Fig. 4. Synaptic inhibition. When the input feature map is too large, the synaptic inhibitory structure can suppress redundant neurons to transmit spikes. Then we can realize the ANN to SNN conversion of the CenterClip structure in ANN.

2) Acquisition and Training of SNNs: Given the cumbersome and resource-intensive nature of direct SNN training, we use ANN to SNN transformation for obtaining and training SNNs. The main steps are: i) training the ANN to obtain weight data; ii) replacing the ReLU activation function with IF neurons; and iii) using channel normalization to scale weights to the maximum channel activation value and bias, converting incompatible operators. In step i), we use the pre-trained SiamRPN network as a model and initiate training with it. Initially, the pre-trained model is fixed while the fusion module is initialized. After ten rounds, we release the SiamRPN backbone for feature extraction optimization. This study employs ResNet50 or optionally AlexNet as the backbone. The final ten rounds involve releasing all parameters to fully optimize the tracker. Parameter settings

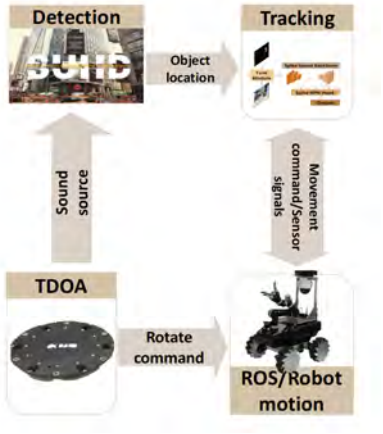

Fig. 5. Flowchart of Wake-on-Speech Tracking: The array microphone receives the wake-up sound and calculates the sound source angle using the TDOA algorithm. ROS commands control the robot's rotation. After rotation, the robot's vision sensor and object detection algorithm detect the sound source position and pass it to the tracker, which is then activated to start tracking the target.

for the training process and model structure are detailed in Tab. I, where 'Used layers' indicates layers used by RPN for feature extraction with each increment representing an additional Resnet layer. 'Weighted' in RPN denotes whether results are weighted against the outcome and serves as an inference component in SiamRPN. All other parameters as well as the model structure are consistent with siamRPN. The

TABLE I
MODEL PARAMETERS SETTING

Module	Parameter	Value
Backbone	Used layers	[2,3,4]
	Type	Resnet50
Adjust	In channels	[512,1024,2048]
	Out channels	[256,256,256]
RPN	Type	'MultiRPN'
	Weighted	False

CentreClip structure within the RPN crops redundant features and reduces the feature map size. Since no existing methods address its transformation, we propose a synaptic inhibition method for conversion, as shown in Fig.4 and described in Eq.11.

$$s_i^l = \begin{cases} s_i^l, & if\ neuron_i^l \in R, \\ 0, & if\ neuron_i^l \notin R, \end{cases} \tag{11}$$

Where s_i^l denotes the spike of neuron i in layer l, $neuron_i^l$ represents neuron i in layer l, and R is the set of neurons allowed for feature transmission. Inhibiting certain synapses and spikes reduces the number of feature neurons and the feature map size.

Fig. 6. Summit-XL robot platform.

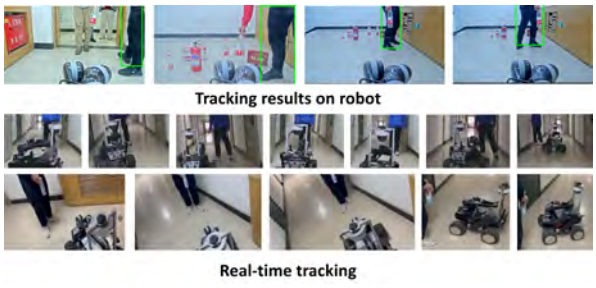

Fig. 7. Tracker tracking effect in dynamic scenes. The robot tracking results show the tracking results of the Summit-XL in the dynamic scene, and the real-time tracking results show the real-time tracking results of the Summit-XL. The upper part of the real-time tracking results shows the results of tracking a person and the lower part shows the results of tracking a cup.

D. Algorithm Deployment and Robot Control

Effective algorithm deployment and robot control are crucial for optimal tracking model performance. We designed a system for voice wake-up and autonomous target selection, as shown in Fig.5. The process involves porting the algorithm to the robot platform and connecting it to the ROS instruction set, with the camera as the input source using the RTSP protocol. The first two frames initialize motion information and the tracker.

There are two target selection methods: i)Acoustic Source Localization. The robot, equipped with an array microphone and an SNN-based target detector [25], wakes up on command, calculates the sound source direction using Time Difference of Arrival (TDOA), and moves towards it. The object detector locates the sound source, and the tracker starts. The robot adjusts its pose based on the angular difference to the target, issuing velocity commands to move and rotate. It stops 1.5 meters from the target for safety and resumes tracking if the target deviates beyond this distance. ii)Manual Selection. The tracker is started directly. After manually selecting the target during the second frame, tracking begins from the third frame. The first two frames are used for initialization. The detailed algorithm has been attached to the appendix.

IV. Experiments

In this section, we performed comparative experiments to validate the effectiveness of our methods and model. The experiments utilized an Intel Core i7-8700K CPU and an NVIDIA RTX2080Ti GPU, with datasets including VOT2016 [26], VOT2018 [27], and OTB2015 (OTB100). Comparison data for other state-of-the-art models were sourced from relevant papers. We also assessed the tracker's real-world object-tracking capability and robot compatibility using the Summit-XL (Fig.6), which features an Intel Core i3-9100 CPU, 7.16 GB of RAM, four independent drive wheels, and an axis gimbal camera.

A. Ablation study and Comparison with Advanced Trackers

We compared our proposed tracker with Spiking SiamFC++, SiamSNN, and SiamRPN to evaluate the fusion module's effectiveness. Metrics include Accuracy (A), Robustness (R), Expected Average Overlap (EAO), Precision, and Success. Results are in Tab.II.

The results show that our model significantly outperforms mainstream methods, with the EAO metric exceeding SOTA by 13.6% and 5.4% from VOT2016 to VOT2018, respectively, and achieving SOTA on OTB2015. Additionally, our model, in comparison with SiamRPN in the ablation study, surpasses it across all metrics, validating the fusion module's effectiveness.

B. Energy Efficiency

Energy consumption is crucial for model inference cost and limits tracking capacity in robotic systems. Following works [2], [29], [30], *FLOPs* measures model complexity, while *energy* denotes model energy cost. Horowitz et al. [29], [31] specify energy costs as 4.6pJ(FLOAT32 MAC)/0.9pJ(FLOAT32 AC)/3.2pJ(INT MAC)/0.1pJ(INT AC). To measure SNN's and ANN's energy consumption more rationally, we use the average FLOPs based on the conversion sample dataset. Specifically, we define SNN/ANN FLOPs as:

$$FLOPs = \sum_{l}\sum_{t=1}^{T} s^{l}(t) \; + \; \sum_{l}\sum p, \tag{12}$$

$s^{l}(t)$ represents the sum of spikes of all neurons in layer l at time t, while $\sum_{l}\sum p$ is the sum of MAC operations across layers. The ANN model involves only MAC operations, so s is 0. The SNN model includes both AC and MAC operations, with neither s nor p being 0. Here, s represents integer AC operations. Energy consumption for SiamRPN is from related papers [17]. As shown in Tab. III, our fusion module significantly increased energy consumption, impacting mobile deployment and practicality. However, after conversion to SNN, our model improved energy efficiency by 67.7 times compared to SiamRPN [32], using only 1.47% of SiamRPN's energy. Thus, our algorithm not only provides superior tracking performance but also enhances energy efficiency.

C. Robotics Experiments

We deployed the algorithmic model to the Summit-XL robotic platform via a removable hard drive or network, configuring the runtime environment. ROS controlled Summit-XL's motion and information transmission, while Rviz mon-

TABLE II
COMPARISON WITH OTHER WORKS UNDER THE VOT AND OTB DATASET

Model	VOT2016			VOT2018			OTB2015	
	A	R	EAO	A	R	EAO	Precision	Success
Spiking siamFC++ [28]	0.600	0.359	0.302	0.556	0.445	0.255	0.854	0.644
SiamSNN [11]	0.497	0.630	0.210	0.460	0.860	0.176	0.528	0.443
SiamRPN [17]	0.560	0.260	0.344	0.490	0.460	0.244	0.851	0.637
Our model	**0.629**	**0.210**	**0.438**	**0.579**	**0.323**	**0.309**	**0.874**	**0.661**

TABLE III
ENERGY EFFICIENCY

	FLOPs	Power(w)	Energy(J)
ANN	9.54E+11	137.2	4.39
SiamRPN [17]	4.33E+09	3.36	2.1E-02
SNN(Ours)	1.25E+08	9.6E-04	3.1E-04
Ratio($\frac{SiamRPN}{Ours}$)	**34.64**	**3500**	**67.7**

itored the system remotely or locally. After deployment, we conducted two experiments on the Summit-XL.

Experiment 1: We tested object tracking algorithms on pre-recorded dynamic scene videos using both the Summit-XL and a computer to compare tracking accuracy. The task was to track a person holding a mineral water bottle in a 1194-frame video featuring stills, slow, medium, and fast movements. Tracking accuracy was calculated based on results from both platforms, using Eq.13.

$$Accuracy = \frac{TP}{TP+FP+NP}, \tag{13}$$

where *Accuracy* denotes accuracy, TP refers to correctly tracked targets, FP refers to incorrectly tracked targets, and FN refers to missed targets. The tracking accuracy on both platforms was 92.7%. Results show no difference in detection ability between the Summit-XL and PC, indicating our model deploys to mobile robot platforms without loss. Some results are shown in Fig.7.

Experiment 2: We deployed the model to Summit-XL to evaluate its tracking capability. After testing in real-world scenarios, Summit-XL tracked the target objects— a moving person and a water bottle—smoothly. Fig.7 shows real-time tracking results, with the top row for the person and the bottom row for the water bottle. For uplink testing, Summit-XL is woken by voice, determines target position using audio signal time differences, rotates towards the target, uses a detection model to identify it, and tracks it. For downlink testing, the tracker is started remotely, targets are framed on screen, and tracking begins. Results show Summit-XL tracks smoothly without needing to reinitialize.

V. CONCLUSION

In summary, we have designed Motion Feature Extractor, that can replace the DVS signal at a lower cost, and We have achieved compatibility with both DVS and RGB inputs for the first time. Additionally, we have proposed a framework for object tracking based on SNNs, utilizing the method of converting ANNs to SNNs for training and acquisition. Compared to other advanced trackers, our tracker has achieved state-of-the-art levels in both accuracy and energy consumption. To the best of our knowledge, this tracker outperforms current state-of-the-art SNN-based object trackers. Furthermore, we have conducted robot experiments to confirm that our tracker can be deployed on mobile robots with edge computing power. Experimental results demonstrate that we can achieve SOTA results at extremely low costs. We believe that this approach holds significant importance for future object tracking on power-efficient neuromorphic electronic and photonic SNN chips, as well as for dynamic real-world object tracking based on edge computing power in practical scenarios.

ACKNOWLEDGEMENTS

This work is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences under (Grants XDA0450200, XDA0450202), Beijing Natural Science Foundation (Grant L211023) and National Natural Science Foundation of China (Grants 91948303, 61627808).

REFERENCES

[1] F. Liu, W. Zhao, Y. Chen, Z. Wang, and L. Jiang, "Spikeconverter: An efficient conversion framework zipping the gap between artificial neural networks and spiking neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, 2022, pp. 1692–1701.

[2] B. Chakraborty, X. She, and S. Mukhopadhyay, "A fully spiking hybrid neural network for energy-efficient object detection," *IEEE Transactions on Image Processing*, vol. 30, pp. 9014–9029, 2021.

[3] Z. Bing, Z. Jiang, L. Cheng, C. Cai, K. Huang, and A. Knoll, "End to end learning of a multi-layered snn based on r-stdp for a target tracking snake-like robot," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 9645–9651.

[4] Z. Yang, Y. Wu, G. Wang, Y. Yang, G. Li, L. Deng, J. Zhu, and L. Shi, "Dashnet: A hybrid artificial and spiking neural network for high-speed object tracking," *arXiv preprint arXiv:1909.12942*, 2019.

[5] Y. Luo, Q. Yi, T. Wang, L. Lin, Y. Xu, J. Zhou, C. Yuan, J. Guo, P. Feng, and Q. Feng, "A spiking neural network architecture for object tracking," in *Image and Graphics: 10th International Conference, ICIG 2019, Beijing, China, August 23–25, 2019, Proceedings, Part I 10*. Springer, 2019, pp. 118–132.

[6] Z. Jiang, R. Otto, Z. Bing, K. Huang, and A. Knoll, "Target tracking control of a wheel-less snake robot based on a supervised multi-layered snn," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 7124–7130.

[7] K. Liu, X. Cui, X. Ji, Y. Kuang, C. Zou, Y. Zhong, K. Xiao, and Y. Wang, "Real-time target tracking system with spiking neural networks implemented on neuromorphic chips," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 4, pp. 1590–1594, 2022.

[8] H. Seok and J. Lim, "Robust feature tracking in dvs event stream using bézier mapping," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 1658–1667.

[9] M. Ji, Z. Wang, R. Yan, Q. Liu, S. Xu, and H. Tang, "Sctn: Event-based object tracking with energy-efficient deep convolutional spiking neural networks," *Frontiers in Neuroscience*, vol. 17, p. 1123698, 2023.

[10] H. Liu, D. P. Moeys, G. Das, D. Neil, S.-C. Liu, and T. Delbrück, "Combined frame- and event-based detection and tracking," in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2016, pp. 2511–2514.

[11] Y. Luo, M. Xu, C. Yuan, X. Cao, L. Zhang, Y. Xu, T. Wang, and Q. Feng, "Siamsnn: Siamese spiking neural networks for energy-efficient object tracking," in *Artificial Neural Networks and Machine Learning – ICANN*, 2021, pp. 182–194.

[12] Y. Luo, H. Shen, X. Cao, T. Wang, Q. Feng, and Z. Tan, "Conversion of siamese networks to spiking neural networks for energy-efficient object tracking," *Neural Computing and Applications*, vol. 34, no. 12, pp. 9967–9982, 2022.

[13] J. Zhang, B. Dong, H. Zhang, J. Ding, F. Heide, B. Yin, and X. Yang, "Spiking transformers for event-based single object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 8801–8810.

[14] B. Jung and G. S. Sukhatme, "Real-time motion tracking from a mobile robot," *International Journal of Social Robotics*, vol. 2, pp. 63–78, 2010.

[15] M.-F. R. Lee and Y.-C. Chen, "Artificial intelligence based object detection and tracking for a small underwater robot," *Processes*, vol. 11, no. 2, 2023. [Online]. Available: https://www.mdpi.com/2227-9717/11/2/312

[16] J. A. Leñero-Bardallo, T. Serrano-Gotarredona, and B. Linares-Barranco, "A 3.6 us latency asynchronous frame-free event-driven dynamic-vision-sensor," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 6, pp. 1443–1455, 2011.

[17] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with siamese region proposal network," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8971–8980.

[18] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, "Conversion of continuous-valued deep networks to efficient event-driven networks for image classification," *Frontiers in neuroscience*, vol. 11, p. 682, 2017.

[19] T. Bu, J. Ding, Z. yu, and T. Huang, "Optimized potential initialization for low-latency spiking neural networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 11–20, 06 2022.

[20] Y. Hu, H. Tang, and G. Pan, "Spiking deep residual networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 8, pp. 5200–5205, 2023.

[21] H. Seok and J. Lim, "Robust feature tracking in dvs event stream using bezier mapping," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[22] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi, and J. Zhong, "Attention is all you need in speech separation," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 21–25.

[23] Q. Huang, C. Xia, C.-H. Wu, S. Li, Y.-C. Wang, Y. Song, and C.-C. J. Kuo, "Semantic segmentation with reverse attention," *ArXiv*, vol. abs/1707.06426, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:30595348

[24] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[25] J. Qu, Z. Gao, T. Zhang, Y. Lu, H. Tang, and H. Qiao, "Spiking neural network for ultra-low-latency and high-accurate object detection," *IEEE Transactions on Neural Networks and Learning Systems*, 2024, doi:10.1109/TNNLS.2024.3372613.

[26] G. Roffo, S. Melzi, *et al.*, "The visual object tracking vot2016 challenge results," in *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II*. Springer International Publishing, 2016, pp. 777–823.

[27] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. ˇCehovin Zajc, T. Vojir, G. Bhat, A. Lukezic, A. Eldesokey, *et al.*, "The sixth visual object tracking vot2018 challenge results," in *Proceedings of the European conference on computer vision (ECCV) workshops*, 2018, pp. 0–0.

[28] S. Xiang, T. Zhang, S. Jiang, Y. Han, Y. Zhang, C. Du, X. Guo, L. Yu, Y. Shi, and Y. Hao, "Spiking siamfc++: Deep spiking neural network for object tracking," *arXiv preprint arXiv:2209.12010*, 2022.

[29] S. Kim, S. Park, B. Na, and S. Yoon, "Spiking-yolo: spiking neural network for energy-efficient object detection," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 11 270–11 277.

[30] S. Narduzzi, S. A. Bigdeli, S.-C. Liu, and L. A. Dunbar, "Optimizing the consumption of spiking neural networks with activity regularization," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 61–65.

[31] M. Horowitz, "1.1 computing's energy problem (and what we can do about it)," in *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, 2014, pp. 10–14.

[32] D. Xing, N. Evangeliou, A. Tsoukalas, and A. Tzes, "A siamese network for real-time object tracking on cpu," *Software Impacts*, vol. 12, p. 100266, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2665963822000252