Towards a more efficient few-shot learning-based human gesture recognition via dynamic vision sensors

Linglin Jing[1]
l.jing@lboro.ac.uk
Yifan Wang[1]
Y.Wang6@lboro.ac.uk
Tailin Chen[3]
t.chen14@newcastle.ac.uk
Shirin Dora[1]
S.Dora@lboro.ac.uk
Zhigang Ji[2]
zhigangji@sjtu.edu.cn
Hui Fang[1]
h.fang@lboro.ac.uk

[1] Department of Computer Science, Loughborough University, Loughborough, UK

[2] The National Key Laboratory of Science and Technology on Micro/Nano Fabrication, Shanghai Jiao Tong University, Shanghai, China

[3] Open Lab, Newcastle University, Newcastle upon Tyne, UK

Abstract

For human gesture recognition task, recent fully supervised deep learning models have achieved impressive performance when sufficient samples of predefined gesture classes are provided. However, these models do not generalise well for new classes, thus limiting the model accuracy on unforeseen gesture categories. Few-shot learning based human gesture recognition (FSL-HGR) addresses this problem by supporting faster learning using only a few samples from new gesture classes. In this paper, we develop a novel FSL-HGR method which enables energy-efficient inference across large number of classes. Specifically, we adapt a surrogate gradient-based spiking neural network model to efficiently process video sequences collected via dynamic vision sensors. With a focus on energy-efficiency, we design two strategies, spiking noise suppression and emission sparsity learning, to significantly reduce the spike emission rate in all layers of the network. Additionally, we introduce a dual-speed stream contrastive learning to achieve high accuracy without increasing computational burden associated with inference using dual stream processing. Our experimental results demonstrate the effectiveness of our approach. We achieve state-of-ate-art 84.75%, and 92.82% accuracy on 5way-1shot and 5way-5shot learning task with 60.02% and 58.21% reduced spike emission number respectively compared to a standard SNN architecture without using our learning strategies when processing the DVS128 Gesture dataset.

1 Introduction

Vision-based human gesture recognition is an established topic in computer vision research since it is the key component of various applications involving human-computer interac-

tion [41] like sign-language translation [27], robot control [30], virtual manipulation [16], daily care assistance [14] and entertainment [34]. Despite impressive performance achieved by deep learning algorithms in recognising pre-defined gesture classes [16, 19, 22, 25], fully supervised models hardly generalise well to new classes of gestures that have not been observed during training.

Few-Shot Learning Human Gesture Recognition (FSL-HGR) methods are proposed to address the challenges to build more generalised models [28, 41]. FSL algorithms adapt the parameters of a pre-trained model using limited training steps and a small number of training samples to perform well on new but related tasks. In the context of the gesture recognition problem, a model trained using FSL has the capability to reliably recognise N newly defined gestures when only K samples are available for each new gesture (termed as N-way K-shot learning).

While significant progress has been made in the development of FSL-HGR models, their deployment in the real-world is inhibited by the limited computation supported by edge devices. Thus, model efficiency has emerged as a crucial factor when developing advanced FSL approaches [38]. Recently, several attempts have been made to improve model efficiency. For example, Dynamic Vision Sensor (DVS) camera has been applied as a privacy-preserving and energy-saving device for gesture recognition applications [2]. DVS camera is designed to capture illumination changes and generate asynchronous events using much lesser energy compared to RGB cameras [4, 23]. Another attempt in this direction is the development of an energy-efficient architecture, e.g., Spiking Neural Networks (SNN) [38]. Different from conventional neural networks, SNNs transmit and process information using binary sequences of spikes over time, which underlies their lower energy requirements for computation [39]. However, the discontinuity characteristics of SNN hinder the training of its network via back-propagation method. To solve this issue, surrogate-gradient approach [24] enables achieving performance comparable to deep neural networks.

In this paper, we propose a new FSL-HGR approach using surrogate gradient based SNNs which are optimized for both performance and energy-efficiency. As described in [12], the network is trained using two videos streams with different sampling rates which enables capturing both high- and low-frequency variations in the input. To overcome the higher computational burden imposed by multiple streams, we devleop a contrastive learning strategy that enforces consistency between the feature representations obtained from different streams during training, thus enabling inference using a single stream. To further improve the model efficiency, we developed strategies of motion noise suppression and spike emission sparsity. Both of these strategies encourage generation of fewer spikes in all layers of the network. Based on the comprehensive evaluation, we report the state-of-the-art results for the DVS based FSL-HGR in terms of both model accuracy and its energy-efficiency. Our key contributions are as follows:

- A new contrastive learning strategy is introduced to support model reliability of the proposed FSL-HGR SNN model without increasing computational burden at inference stage.
- We present a data pre-processing strategy, including spiking noise suppression and motion trajectory enforcement, to improve model efficiency.
- We embed a channel-wise spike sparsity loss term in our loss function to further reduce spike emissions to improve the model efficiency.

2 Related work

Human Gesture Recognition has been an active research topic in the field of computer vision since it is a crucial component of human-computer interaction systems [41]. To achieve reliable recognition performance, some work focuses on designing more effective feature representations, e.g., [7, 8, 9, 40], while many explore the advantages from various modality or multi-modality fusion, including the use of RGB-D camera [11], infrared sequences [36], event-stream sensors [2, 21], or from skeleton tracking [6]. Although recent deep learning based methods have reported impressive results when models are trained in a strongly supervised manner [10, 45], it is still under extensive investigation how to build more generalised HGR model that can be scalable to related tasks in which new gestures are involved [15].

Few-shot Learning methods are proposed to improve the capability of model generalisation so that model could achieve better accuracy when only few samples of new classes are provided. Few-shot learning approaches can be categorized into three classes: transfer learning, meta learning and metric learning. Transfer learning and fine-tuning is the simplest means by re-learning a pre-trained model with these few samples of new classes to avoid overfitting [18]. While meta learning is composed of two learning stages, named base-learner and meta-learner to obtain across-task meta-knowledge to enhance model generalisation. Two typical meta learning examples are MAML [13] and R2D2 [3]. In contrast to the above-mentioned training methods, metric-learning based approaches, e.g., SiameseNet [17] and ProtoNet [35], enforce the distances between query images and support classes to enlarge the classification boundary margins. Since the few-shot learning based methods can be easily adapted to new tasks, they are well-suited to the real-world deployment of HGR models.

Model efficiency has become more and more important for the purpose of deploying deep learning models on affordable edge devices and reducing energy consumption [44]. Motivated by these reasons, spiking neural network (SNN), a bio-inspired and energy efficient network architecture, has attracted more research interest [32]. The classical unsupervised SNNs, e.g., Hebbian learning [1] and Spike- Timing-Dependent Plasticity (STDP) [20], are still inferior to recent supervised DL models although they have the advantages of high training efficiency and are more device-friendly. Consequently, supervised SNNs has been increasingly investigated to improve their model accuracy without compromising their efficiency [10]. Surrogate gradient back-propagation method is a popular SNN model that has achieved comparable accuracy with other DL based models [24].

In this paper, we focus on designing a SNN based FSL-HGR model to support the real-time HGR system deployment on affordable devices. It is worth to mention that there are several works which share similarity to our idea, including [15, 31, 37]. However, our work differs from these methods on: (i) we adapt an advanced SNN model architecture from [10] to improve the baseline performance significantly; and (ii) we have designed several novel training strategies to further boost both the model accuracy and efficiency.

3 Proposed Method

The overview of our proposed FSL-HGR model is depicted in Figure 1. During training, like ProtoNet [35], we randomly select two DVS sequences from the same class as the support and query sequence in each iteration, where the support and query pairs are assumed to have closer distance in the feature embedding space when model converges.

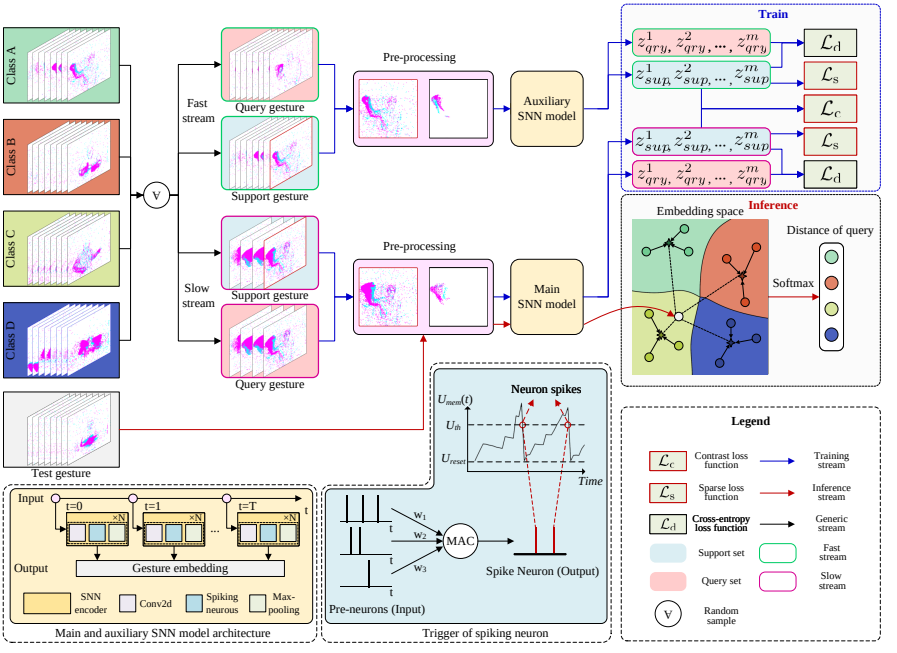

Figure 1: The overview of our proposed FSL-HGR method with both the training and inference stages.

After extracting motion representations from the data and suppress the noise emissions via the preprocessing module (Sec. 3.1), the signals activate neurons in the SNN model (Sec. 3.2) to produce responses for the classification task. Regarding to the loss function (Sec. 3.3), in addition to the classification loss term, we embed two extra loss terms in our training to further improve the model reliability and efficiency. Specifically, a contrastive loss term is used to enforce feature consistency between the main SNN model and an auxiliary model that are deployed to process sequences with different frame rates, while a sparsity loss term is defined to reduce spike emission numbers by decoupling spikes across channels.

At the inference stage, we only deploy the main SNN model and follow the process of the few shot learning paradigm to compare the distances of the activated patterns between a test sequence and the support gesture sequences. Given the few-shot classification task, the gesture support set can be described as $\mathcal{S} = \{(x_i, y_i), i = 1, \ldots, n\}$, where x_i is a sequence of an event stream from few-shot gesture samples, and $y_i \in \{1, \ldots, K\}$ is its corresponding label. As illustrated in Figure 1, the support samples are mapped into the embedding space and the prototype of class k is generated by averaging the support samples embedding vectors. The test gesture is then classified into its closest class by using the following equation:

$$p(y = k \mid m) = \frac{\exp\left(-d(f_\phi(m), c_k)\right)}{\sum_{k'} \exp\left(-d(f_\phi(m), c_{k'})\right)}, \tag{1}$$

where $d(\cdot)$ denotes the Euclidean distance between the query gesture embedding $f_\phi(m)$ and the prototype of class c_k.

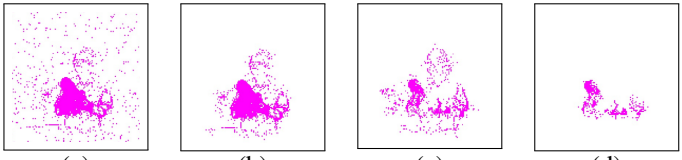

Figure 2: Visualisation of an event frame representation in the positive polarity channel with (a) original events; (b) with spatial noise suppression; (c) with spatial and temporal noise suppression; and (d) with noise suppression and motion trajectory capture.

3.1 Pre-processing of DVS Sequences

DVS is a vision sensor that converts changes in brightness to binary dynamic signals with enhanced privacy preservation [29]. The binary sparse nature of signals generated by DVS sensors allows natural energy-efficient processing using SNNs. This motivated us to focus on developing gesture recognition models that are tailored towards processing DVS sequences.

A DVS sequence can be represented as (h,w,p,t) where h,w are the spatial coordinates, t is the timestamp and p refers to the binarised values of brightness changes between two timestamps. Due to the high temporal resolution of original DVS sequences, we sub-sample the DVS sequences following the process in [10] as expressed in the following equation:

$$F_t = q\left(F'_{t'}\right), t' \in [\beta \cdot t, \beta \cdot (t+1) - 1], \tag{2}$$

where F' denotes the original continuous events stream. t' is the original time stamp, F denotes the new events representation, t is the new time step, β is the temporal resolution factor and $q(\cdot)$ is an aggregation function. We utilise the same aggregation function as in [10] which involves stacking spikes in the two DVS channels and convert the signals back to spikes by using a convolution layer as the spiking encoder. An example frame representation from a DVS sequence is shown in Figure 2(a).

Many spikes in a DVS event stream are generated by illumination changes in the background [42]. These spikes lead to higher activation in shallower layers of SNNs leading to higher compute requirements. Further, they may affect the performance of the model negatively due to presence of noisy signals in the input. To deal with this issue, we added a noise suppression stage during the preprocessing of the DVS sequences. Inspired by the classical DBSCAN algorithm [33], we assume that spatially or temporally sparse spiking activity corresponds to noisy signals in the DVS sequence. The neighbourhood radius R and number of spikes N to identify regions of dense spiking activity are set heuristically in this work. It was observed that the final performance is relatively insensitive to these two hyperparameters as long as they are set in a reasonable range. After noise suppression, the absolute difference between frames [42] is presented to the network to highlight spikes representing the motion trajectories. These preprocessing steps reduce the number of spikes generated significantly thereby improving the energy-efficiency of the model (Figure 2). This is further demonstrated in the ablation study in Section (4.3).

3.2 Network Architecture

Our SNN, illustrated in Figure 1, consists of four sequentially connected blocks, each of which has the same structure. Each block consists of a convolution layer with 64 filters of

size 3×3 followed by a Max-pooling layer of size 2×2. The spiking neurons used in the network are Leaky Integrate-and-Fire (LIF) neurons. The membrane potential of an LIF neuron at time t is given by the following equation:

$$\tau \frac{dU(t)}{dt} = -U(t) + X(t), \tag{3}$$

where $U(t)$ represents the membrane potential of the neuron at time and $X(t)$ represents the input to the neuron and τ is the time constant of the neuron. When the membrane potential exceeds a pre-defined threshold U_{th}, the neuron transmits a spike to all connected downstream neurons and the membrane potential is reset to zero. The output of the spiking neurons in the n^{th} convolution layer can be expressed through the following equations

$$H_{t,n} = f\left(U_{t-1,n}, h\left(W_n, A_{t,n-1}\right)\right), \tag{4}$$

$$Z_{t,n} = \Theta\left(H_{t,n} - U_{th}\right), \tag{5}$$

$$U_{t,n} = H_{t,n}\left(1 - Z_{t,n}\right) + U_{\text{reset}} \cdot Z_{t,n}, \tag{6}$$

where n and t are indices for the layer and time-step. H and U are the the accumulation process of the internal membrane potential and the initial value after firing the spike, respectively. Z is a binarized output, it equals to 1 if the membrane potential $H > U_{th}$, which is a pre-set threshold value, otherwise 0. W is the weight value. $h(\cdot)$ is the convolutional layer or fully connected layer operation. $f(\cdot)$ is the computational equation of the spike neuron model, and different models have their corresponding equations. $\Theta(\cdot)$ is a Heaviside step function which equals to 1 if its internal calculation is greater than 0, otherwise it is 0. Combining Equation (4)(5)(6), the output of the current layer $A_{t,n}$ can be described by the following equation. In our work, we also report the performance of using several alternative SNN neuron types, including IF, LIF, PLIF and LIAF, since they have different levels of balance on efficiency and accuracy.

$$A_{t,n} = \begin{cases} \Theta\left(U_{t-1} + h\left(\omega_n, A_{t,n-1}\right) - U_{th}\right) & \text{for IF,} \\ \Theta\left(U_{t-1} + \frac{1}{\tau}\left(-\left(U_{t-1}\right) + h\left(\omega_n, A_{t,n-1}\right)\right) - U_{th}\right) & \text{for LIF,} \\ \Theta\left(U_{t-1} + \frac{1}{1+\exp(-a)}\left(-\left(U_{t-1}\right) + h\left(\omega_n, A_{t,n-1}\right)\right) - U_{th}\right) & \text{for PLIF,} \\ ReLU\left(U_{t-1} + \frac{1}{\tau}\left(-\left(U_{t-1}\right) + h\left(\omega_n, A_{t,n-1}\right)\right) - U_{th}\right) & \text{for LIAF,} \end{cases} \tag{7}$$

where τ is a pre-set time constant, a is a trainable parameter. The first three neuron models are binarized outputs, while the LIAF model has floating outputs.

3.3 Training strategies and the loss function

Dual-speed stream contrastive loss: Performance of gesture recognition from video sequences is usually sensitive to the magnitude of movement in these videos. To improve recognition of gestures with different magnitude of movements, two-stream or slow-fast networks have been proposed [17]. Slow-fast networks integrate feature representations from two different networks which estimate probabilities of gesture classes based on inputs with different frame rates. However, this design significantly increases the model parameters and

doubles the computation required for inference. For this purpose, we propose a contrastive loss that enables inference using a single network while retaining the performance-related advantages of slow-fast networks. We refer the network trained using input at a higher frame rate as the main network and the network trained using input at a lower frame rate is termed as the auxiliary network. At the end of training, the auxiliary network is dropped and the main network is used for inference. During training, the contrastive loss minimizes the difference between the feature representations inferred by the main network and the auxiliary network. The contrastive loss term between the features from these two networks are expressed in the following equation:

$$\mathcal{L}_c = -\frac{1}{N}\sum_{i=1}^{N} \cos\mathrm{sim}\left(f_\phi^F(x_i), f_\zeta^S(x_i)\right), \tag{8}$$

where $\cos\mathrm{sim}(.)$ is the cosine similarity, $f_\phi^F(x_i)$ and $f_\zeta^S(x_i)$ denotes the support gesture embeddings in fast/slow streams, ϕ and ζ is the model paramaters of the fast/slow network, N is the dimension of the feature representations.

Channel-wise sparsity loss: Inspired by the observation that spikes in channels become more salient in deeper layers, we introduce a channel-wise sparsity loss to improve sparsity and decouple the activation across channels. This helps reduce the number of spike emitted in deeper layers of the network. The sparsity loss is expressed as follows:

$$\mathcal{L}_s = -\frac{1}{HW}\sum_{i=1}^{HW} \mathrm{MSE}\left(Max(f_\phi^F(x_i)), Avg(f_\phi^F(x_i))\right), \tag{9}$$

where Max and Avg is the maximum/average responses of each pixel in all channels of the last layer, and H and W are the height and width of its output respectively.

Classification Loss: The classification loss we use is the traditional cross-entropy loss between the prototype and query gesture. According to Eq. (10), the negative log-probabilities of the class in both streams are minimized:

$$\mathcal{L}_d^F = -\log p(y=k \mid m^F), \quad \mathcal{L}_d^S = -\log p(y=k \mid m^S), \tag{10}$$

where $\mathcal{L}_d^F$ and $\mathcal{L}_d^S$ are the fast and slow stream loss respectively, m^F and m^S are the query gesture sample in fast/slow streams.

The total loss function of the proposed model is defined in Eq. (11). Notably, we weigh each of the loss terms equally to avoid overfitting and keep the training process simple.

$$\mathcal{L} = \mathcal{L}_d^F + \mathcal{L}_d^S + \mathcal{L}_c + \mathcal{L}_s. \tag{11}$$

4 Experiments

4.1 Experimental Setup

We use DVS128 Gesture dataset [2] to evaluate the performance of our method. This dataset contains 11 classes of gesture tracks, e.g., waving and clapping, performed by 29 participants. Each gesture class has 122 video clips recorded under three lighting conditions, i.e. natural light, fluorescent and LED. Each clip can be represented as (h,w,p,t), where h and w are the height and width of the scene and represents the spatial information in the scene.

Proposals	Methods	Fine Tune	Aug.	5w1s Acc.	5w5s Acc.
LPR [37]	Transfer	Y	Y	40.00%	43.30%
MAML [13]	Meta	Y	F	45.50%	53.70%
SOEL [38]	Transfer	Y	Y	64.70%	65.10%
MTO [15]	Meta	N	N	63.20%	73.30%
PLIF [10]	Metric	N	N	80.21%	88.53%
This Work	Metric	N	N	**84.75 %**	**92.82 %**

Table 1: 6+5-WAY Few-shot accuracy on DVS-gesture.

Method	Operations	Latency(s)	Energy(mJ)	Energy ratio
Protot(TPU) [35]	$3.89E^{10}$	$5.40E^{-1}$	45.55	$\times$240
Protot(Memristor) [35]	$3.89E^{10}$	$6.80E^{-3}$	1.56	$\times$8
PLIF [10]	$2.04E^{7}$	$7.14E^{-2}$	0.48	$\times$2.5
Ours	$1.29E^{7}$	$4.52E^{-2}$	0.31	$\times$1.6
Ours(+N)	$8.94E^{6}$	$3.13E^{-2}$	0.21	$\times$1.1
Ours(+N+S)	$\mathbf{8.23E^{6}}$	$2.88E^{-2}$	**0.19**	$\times$**1**

Table 2: Power consumption comparison results on 5w1s task, N is noise identification unit, S is adaptive feature sparse loss.

p denotes the event polarity (positive and negative), and t is the recording duration which is set to 300ms. The surrogate gradient [24] back-propagation method for SNNs with Adam optimizer is used to update the network parameters.

4.2 Comparison to the state-of-the-art models

We evaluate our model with two few shot learning configurations, namely 5-way-1-shot task (5w1s) and 5-way-5-shot task (5w5s). We randomly select 6 of the 11 gestures from DVS128 gesture dataset for model pre-training, and the remaining 5 gesture classes are used as the few-shot test set to form the 6+5 way gesture recognition task. At the testing stage, the sequences are obtained from 1000 randomly generated samples from the test data set, which follows the evaluation protocol in [15, 35].

Total accuracy. Table 1 shows the accuracy of our method on 6+5 way few-shot gesture recognition task compared to four benchmark methods. In addition to four FSL-HGR models, we also deploy the SNN model from [10] which is an SNN architecture originally trained in a fully supervised manner. Our method achieves 84.75% and 92.82% in the 5-way 1-shot and 5-way 5-shot tasks respectively without any augmentation and fine tuning procedures. This is the best performance when compared to the other five methods.

Efficiency. To estimate the compute requirements on a supporting hardware, we compute energy usage on a 5-way 1-shot gesture recognition task. Table 2 shows the results of power comparison with other similar algorithms. The table 2 shows a comparison with the energy consumption of a CNN prototypical network with TPU [26], on-chip implementation (CNN on memristor) [43], base SNN model from [10] and several variants of our model, i.e. our model without using any preprocessing and learning strategies, our model with only noise suppression and our full model. Our results are from a deployment simulation on neuromorphic hardware (Loihi) for SNN, TPU and hybrid analogue-digital chip (Memristor)

for CNN. For a fair comparison, we only calculate the inference time of the model while ignoring the energy consumed by peripheral circuits. The energy calculations are based on the officially published parameters of Loihi and analogue-digital chip [6, 43]. These results clearly demonstrate the great advantages of the two new efficiency-enhanced strategies, which reduces the energy consumption significantly.

4.3 Ablation study

To investigate the influence of different time frequency on gesture events representation in Eq. (2), we conduct several ablation studies on 5w1s and 5w5s task. In Figure 3, T denotes the accumulations of event stream into T temporal bins. we test the performance in the following scales $T \in \{5, 6, 8, 12, 16, 20\}$. According to Figure 3, it can be observed that as T increases to a certain level, the performance gains become insignificant with a slight drop for very high values. Further, a high value of T implies that the network consumers more energy since one process cycle requires computations on more frames.

To investigate the impact of contrastive loss and channel-wise sparsity loss on performance, table 3 shows the weight comparision of three loss terms. It can be observed that the contrastive loss improves the performance on both 5w1s and 5w5s tasks, and the sparsity loss has an insignificant accuracy drop, i.e., 0.1% (5w1s) and 0.08% (5w5s).

To investigate the role of efficient learning strategies for each convolutional layer, Table 4 shows several metrics before and after optimization on the number of spikes generated. It can be observed that the preprocessing step and the terms for sparsity loss reduce spiking emissions across all layers, especially on first, second and fourth convolutional layers.

To investigate the impact of SNN neuron types, we compare several popular spiking neuron models in another ablation study. In Table 5, we list the performance of four spiking neuron models: Integrate-and-Fire(IF), Leaky Integrate-and-Fire(LIF), Parametric Leaky Integrate-and-Fire(PLIF) and Leaky Integrate-and-Analog-Fire(LIAF) model. IF and LIF are the most classical models, which accumulate internal voltages through inputs in the time range. PLIF changes the time constant of the LIF formula to a learnable variable to improve the model flexibility. While LIAF accumulates internal voltages like LIF but transmit analog values instead of binarised spikes. It is found that LIAF achieves best performance but consumes more energies due to its floating output. In contrast, LIF achieves a better trade-off when considering both accuracy and efficiency.

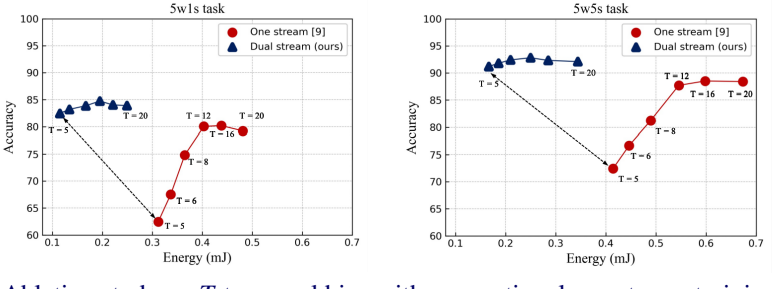

Figure 3: Ablation study on T temporal bins with conventional one stream training [10] and our proposed dual stream training with pre-processing step and the sparsity loss term on 5w1s and 5w5s tasks.

α	β	γ	5w1s Accuracy	5w5s Accuracy
1	0	0	81.68%	89.97%
1	1	0	84.85%	92.90%
1	0	1	81.57%	89.89%
1	1	1	84.75%	92.82%

Table 3: Comparison of Loss weight. α, β, γ is the weight parameter of $\mathcal{L}_d$, $\mathcal{L}_c$ and $\mathcal{L}_s$ respectively.

Layers	Nrg.(mJ)	EFF. Nrg.(mJ)	diff.
Conv1	$4.53E^{-1}$	$1.79E^{-1}$	$\times$0.39
Conv2	$2.11E^{-2}$	$1.21E^{-2}$	$\times$0.57
Conv3	$7.03E^{-3}$	$5.45E^{-3}$	$\times$0.77
Conv4	$3.37E^{-4}$	$1.42E^{-4}$	$\times$0.41

Table 4: Comparison of energy on each layer. (EFF.) is our efficient learning strategy.

Model	5w1s	Nrg.(mJ)
IF	77.15%	0.68
LIF	84.75%	0.19
PLIF	83.24%	0.25
LIAF	85.94%	2.26

Table 5: Comparison of different SNN Neuron types.

5 Conclusion

In this paper, we presented a SNN model to recognize human gestures from DVS videos when only few samples of gesture classes are provided. Our main focus in this work was designing a more energy efficient FSL-GHR to facilitate the application deployment on affordable devices. To achieve this goal, we proposed a pre-processing step and embeded a sparsity loss term to reduce spike emission rates in the entire network. Without compromising the efficiency, we leveraged an auxiliary model to enforce a contrastive learning constraint to improve our model reliability. In our future work, we will further investigate how to make our model working under lifelong learning scenario for on-chip neuromorphic computing.

References

[1] Larry F Abbott and Sacha B Nelson. Synaptic plasticity: taming the beast. *Nature neuroscience*, 3(11):1178–1183, 2000.

[2] Arnon Amir, Brian Taba, David Berg, Timothy Melano, Jeffrey McKinstry, Carmelo Di Nolfo, Tapan Nayak, Alexander Andreopoulos, Guillaume Garreau, Marcela Mendoza, et al. A low power, fully event-based gesture recognition system. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7243–7252, 2017.

[3] Luca Bertinetto, Joao F Henriques, Philip HS Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018.

[4] Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. A 240$\times$ 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits*, 49(10):2333–2341, 2014.

[5] Tailin Chen, Desen Zhou, Jian Wang, Shidong Wang, Yu Guan, Xuming He, and Errui Ding. Learning multi-granular spatio-temporal graph network for skeleton-based action recognition. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 4334–4342, 2021.

[6] Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, et al. Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee Micro*, 38(1): 82–99, 2018.

[7] Minwei Deng. Robust human gesture recognition by leveraging multi-scale feature fusion. *Signal Processing: Image Communication*, 83:115768, 2020.

[8] Abdessamad Elboushaki, Rachida Hannane, Karim Afdel, and Lahcen Koutti. Multid-cnn: A multi-dimensional feature learning approach based on deep convolutional networks for gesture recognition in rgb-d image sequences. *Expert Systems with Applications*, 139:112829, 2020.

[9] Hui Fang, Jeyarajan Thiyagalingam, Nik Bessis, and Eran Edirisinghe. Fast and reliable human action recognition in video sequences by sequential analysis. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3973–3977. IEEE, 2017.

[10] Wei Fang, Zhaofei Yu, Yanqi Chen, Timothée Masquelier, Tiejun Huang, and Yonghong Tian. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2661–2671, 2021.

[11] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016.

[12] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019.

[13] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[14] Jesse Hoey, Pascal Poupart, Axel von Bertoldi, Tammy Craig, Craig Boutilier, and Alex Mihailidis. Automated handwashing assistance for persons with dementia using video and a partially observable markov decision process. *Computer Vision and Image Understanding*, 114(5):503–519, 2010.

[15] Runhao Jiang, Jie Zhang, Rui Yan, and Huajin Tang. Few-shot learning in spiking neural networks by multi-timescale optimization. *Neural Computation*, 33(9):2439–2472, 2021.

[16] Cherdsak Kingkan, Joshua Owoyemi, and Koichi Hashimoto. Point attention network for gesture recognition using point cloud data. In *In British Machine Vision Conference (BMVC)*, 2019.

[17] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2, page 0. Lille, 2015.

[18] Wenbin Li, Chuanqi Dong, Pinzhuo Tian, Tiexin Qin, Xuesong Yang, Ziyi Wang, Jing Huo, Yinghuan Shi, Lei Wang, Yang Gao, et al. Libfewshot: A comprehensive library for few-shot learning. *arXiv preprint arXiv:2109.04898*, 2021.

[19] Dan Liu, Libo Zhang, and Yanjun Wu. Ld-congr: A large rgb-d video dataset for long-distance continuous gesture recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3304–3312, 2022.

[20] Timothée Masquelier and Simon J Thorpe. Unsupervised learning of visual features through spike timing dependent plasticity. *PLoS computational biology*, 3(2):e31, 2007.

[21] Nico Messikommer, Stamatios Georgoulis, Daniel Gehrig, Stepan Tulyakov, Julius Erbach, Alfredo Bochicchio, Yuanyou Li, and Davide Scaramuzza. Multi-bracket high dynamic range imaging with event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 547–557, 2022.

[22] Yuecong Min, Xiujuan Chai, Lei Zhao, and Xilin Chen. Flickernet: Adaptive 3d gesture recognition from sparse point clouds. In *In British Machine Vision Conference (BMVC)*, volume 2, page 5, 2019.

[23] Elias Mueggler, Henri Rebecq, Guillermo Gallego, Tobi Delbruck, and Davide Scaramuzza. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *The International Journal of Robotics Research*, 36(2): 142–149, 2017.

[24] Emre O Neftci, Hesham Mostafa, and Friedemann Zenke. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6):51–63, 2019.

[25] Xuan Son Nguyen, Luc Brun, Olivier Lézoray, and Sébastien Bougleux. A neural network based on spd manifold learning for skeleton-based hand gesture recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12036–12045, 2019.

[26] Thomas Norrie, Nishant Patil, Doe Hyun Yoon, George Kurian, Sheng Li, James Laudon, Cliff Young, Norman Jouppi, and David Patterson. The design process for google's training chips: Tpuv2 and tpuv3. *IEEE Micro*, 41(2):56–63, 2021.

[27] Andres Jessé Porfirio, Kelly Laís Wiggers, Luiz ES Oliveira, and Daniel Weingaertner. Libras sign language hand configuration recognition based on 3d meshes. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1588–1593. IEEE, 2013.

[28] Elahe Rahimian, Soheil Zabihi, Amir Asif, Dario Farina, Seyed Farokh Atashzar, and Arash Mohammadi. Fs-hgr: Few-shot learning for hand gesture recognition via electromyography. *IEEE transactions on neural systems and rehabilitation engineering*, 29:1004–1015, 2021.

[29] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE transactions on pattern analysis and machine intelligence*, 43(6):1964–1980, 2019.

[30] Isidoros Rodomagoulakis, Nikolaos Kardaris, Vassilis Pitsikalis, E Mavroudi, Athanasios Katsamanis, Antigoni Tsiami, and Petros Maragos. Multimodal human action recognition in assistive human-robot interaction. In *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 2702–2706. IEEE, 2016.

[31] Bleema Rosenfeld, Bipin Rajendran, and Osvaldo Simeone. Fast on-device adaptation for spiking neural networks via online-within-online meta-learning. In *2021 IEEE Data Science and Learning Workshop (DSLW)*, pages 1–6. IEEE, 2021.

[32] Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda. Towards spike-based machine intelligence with neuromorphic computing. *Nature*, 575(7784):607–617, 2019.

[33] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21, 2017.

[34] Jamie Shotton, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *CVPR 2011*, pages 1297–1304. Ieee, 2011.

[35] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

[36] Scott Sorensen, Philip Saponaro, Stephen Rhein, Chandra Kambhamettu, and MWJ Xianghua Xie. Multimodal stereo vision for reconstruction in the presence of reflection. In *In British Machine Vision Conference (BMVC)*, pages 112–1, 2015.

[37] Kenneth Stewart, Garrick Orchard, Sumit Bam Shrestha, and Emre Neftci. On-chip few-shot learning with surrogate gradient descent on a neuromorphic processor. In *2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, pages 223–227. IEEE, 2020.

[38] Kenneth Stewart, Garrick Orchard, Sumit Bam Shrestha, and Emre Neftci. Online few-shot gesture learning on a neuromorphic processor. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 10(4):512–521, 2020.

[39] Aboozar Taherkhani, Ammar Belatreche, Yuhua Li, Georgina Cosma, Liam P Maguire, and T Martin McGinnity. A review of learning in biologically plausible spiking neural networks. *Neural Networks*, 122:253–272, 2020.

[40] Xianlun Tang, Zhenfu Yan, Jiangping Peng, Bohui Hao, Huiming Wang, and Jie Li. Selective spatiotemporal features learning for dynamic gesture recognition. *Expert Systems with Applications*, 169:114499, 2021.

[41] Jun Wan, Guodong Guo, and Stan Z Li. Explore efficient local features from rgb-d data for one-shot learning gesture recognition. *IEEE transactions on pattern analysis and machine intelligence*, 38(8):1626–1639, 2015.

[42] Yanxiang Wang, Bowen Du, Yiran Shen, Kai Wu, Guangrong Zhao, Jianguo Sun, and Hongkai Wen. Ev-gait: Event-based robust gait recognition using dynamic vision sensors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6358–6367, 2019.

[43] Zhongrui Wang, Can Li, Wenhao Song, Mingyi Rao, Daniel Belkin, Yunning Li, Peng Yan, Hao Jiang, Peng Lin, Miao Hu, et al. Reinforcement learning with analogue memristor arrays. *Nature electronics*, 2(3):115–124, 2019.

[44] Xiurui Xie, Hong Qu, Zhang Yi, and Jürgen Kurths. Efficient training of supervised spiking neural network via accurate synaptic-efficiency adjustment method. *IEEE transactions on neural networks and learning systems*, 28(6):1411–1424, 2016.

[45] Man Yao, Huanhuan Gao, Guangshe Zhao, Dingheng Wang, Yihan Lin, Zhaoxu Yang, and Guoqi Li. Temporal-wise attention spiking neural networks for event streams classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10221–10230, 2021.