An Efficient Knowledge Transfer Strategy for Spiking Neural Networks from Static to Event Domain

Xiang He1, 2*, Dongcheng Zhao1*, Yang Li1, 2*, Guobin Shen1, 3, Qingqun Kong1, 2, 3†, Yi Zeng1, 2, 3, 4†

1Brain-inspired Cognitive Intelligence Lab, Institute of Automation, Chinese Academy of Sciences, Beijing, China
2School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
3School of Future Technology, University of Chinese Academy of Sciences, Beijing, China
4Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Shanghai, China

Abstract

Spiking neural networks (SNNs) are rich in spatio-temporal dynamics and are suitable for processing event-based neuromorphic data. However, event-based datasets are usually less annotated than static datasets. This small data scale makes SNNs prone to overfitting and limits their performance. In order to improve the generalization ability of SNNs on event-based datasets, we use static images to assist SNN training on event data. In this paper, we first discuss the domain mismatch problem encountered when directly transferring networks trained on static datasets to event data. We argue that the inconsistency of feature distributions becomes a major factor hindering the effective transfer of knowledge from static images to event data. To address this problem, we propose solutions in terms of two aspects: feature distribution and training strategy. Firstly, we propose a knowledge transfer loss, which consists of domain alignment loss and spatio-temporal regularization. The domain alignment loss learns domain-invariant spatial features by reducing the marginal distribution distance between the static image and the event data. Spatio-temporal regularization provides dynamically learnable coefficients for domain alignment loss by using the output features of the event data at each time step as a regularization term. In addition, we propose a sliding training strategy, which gradually replaces static image inputs probabilistically with event data, resulting in a smoother and more stable training for the network. We validate our method on neuromorphic datasets, including N-Caltech101, CEP-DVS, and N-Ommiplot. The experimental results show that our proposed method achieves better performance on all datasets compared to the current state-of-the-art methods. Code is available at https://github.com/Brain-Cog-Lab/Transfer-for-DVS.

Introduction

As the third generation of neural networks, spiking neural networks (SNNs) (Maass 1997) are known for their rich neurodynamic properties in the spatial-temporal domain and event-driven advantages (Roy, Jaiswal, and Panda 2019). Due to the non-differentiable properties of spiking neurons, training SNNs has been a critical area of extensive academic research. The training of SNNs is mainly divided into the following three categories: gradient backpropagation-based methods (Wu et al. 2018, 2019; Zheng et al. 2021; Shen, Zhao, and Zeng 2022a; Li et al. 2022c; Deng et al. 2022), spiking time-dependent plasticity (STDP)-based methods (Diehl and Cook 2015; Hao et al. 2020; Zhao et al. 2020; Dong et al. 2022), and conversion-based methods (Han, Srinivasan, and Roy 2020; Bu et al. 2021; Li and Zeng 2022; Liu et al. 2022; Li et al. 2022b). With these proposed algorithms, SNNs show excellent performance in various complex scenarios (Stagsted et al. 2020; Godet et al. 2021; Sun, Zeng, and Zhang 2021; Chien et al. 2021). In particular, SNNs have shown promising results in processing neuromorphic, event-based data due to their ability to process information in the time dimension (Xing, Di Caterina, and Soraghan 2020; Chen et al. 2020; Viale et al. 2021).

The visual neuromorphic data mainly refers to the dataset collected by Dynamic Vision Sensor (DVS) (Serrano-Gotarredona and Linares-Barranco 2013). DVS is a bio-inspired visual sensor that operates differently from conventional cameras. Instead of capturing images at a fixed rate, the DVS measures intensity changes at each pixel asynchronously and records the time (t), position (x, y), and polarity (p) of the intensity change in the form of an event stream. DVS has been gaining popularity in various applications due to their high dynamic range, high temporal resolution, and low latency (Gallego et al. 2017; Zhu et al. 2018; Stoffregen et al. 2019; Gallego et al. 2020). Despite these advantages, the long and expensive shooting process is still a significant challenge for event cameras, which makes event data acquisition difficult and small in scale, thus limiting its further development. In contrast, static datasets are larger in scale and more accessible. Pre-trained deep neural networks can transfer well to other static datasets. However, applying a pre-trained model on a static dataset directly to event data often yields suboptimal results. This result highlights a sharp challenge: While static images intuitively provide rich spatial information that may benefit event data, exploiting this knowledge remains a difficult problem. For this reason, efficiently uncovering and utilizing the knowledge in static datasets to benefit event data is important for the widespread
deployment of networks for various event data applications. In this paper, we first analyze the domain mismatch problem between networks trained on static and event datasets. We show that the inconsistency of feature distribution is a critical barrier to the effective transfer of static image knowledge to event data. To bridge this gap, we address the challenge from two main aspects: feature distribution and training strategy. Regarding feature distribution, we design the knowledge transfer loss function, which consists of domain alignment loss and spatio-temporal regularization to learn the temporal-spatial domain invariant features between static images and event data. The domain alignment loss learns and acquires domain-invariant spatial features by reducing the marginal distribution distance between static images and event data. The spatio-temporal regularization provides dynamically adjusted coefficients for domain alignment loss to better capture temporal features in the data. In terms of training strategies, we propose the sliding training strategy, in which the static image inputs are gradually replaced with event data probabilistically during the training process, resulting in a smooth reduction of the role of knowledge transfer loss and a smoother learning process. Through the validation on event datasets N-Caltech101, CEP-DVS, and N-Omniglot, our method dramatically improves the performance on these datasets. Overall, the main contributions of this paper can be summarized as follows:

1. We propose a knowledge transfer loss function that learns spatial domain-invariant features and provides dynamically learnable coefficients by regularizing event features in the time dimension. This loss function ensures that the model contains static spatial features and has a comprehensive feature representation in the temporal dimension.
2. We propose the sliding training strategy, in which the static image inputs are gradually replaced with event data probabilistically during the training process, resulting in a smoother and more stable learning process.
3. We conduct experiments on commonly used event datasets to verify the effectiveness of our method. The experimental results show that the proposed method outperforms the state-of-the-art methods on all datasets.

**Related Work**

In order to solve the problem of limited labeled DVS data, previous works endeavored to explore solutions such as domain adaptation, data augmentation and the development of efficient training methods.

**Domain Adaptation Using Static Data.** Using static images to facilitate learning better models in the event domain is an intuitive idea. Messikommer et al. (2022) use a generative event model to classify event features into content and motion features, enabling efficient matching between the latent space of events and images. Zhao, Zhang, and Huang (2022) train a convolutional transformer network for event-based classification tasks using large-scale labeled image data via a passive unsupervised domain adaptation (UDA) algorithm. Sun et al. (2022) introduce event-based semantic segmentation to transfer existing labeled image datasets to unlabeled events for semantic segmentation tasks. These works are related to ours. The difference is that we exploit the spatial domain invariant features between static and event data through domain alignment loss. Further, we use coefficients dynamically adjusted at each time step to better capture the temporal properties in the data. This allows the model to contain not only static spatial features, but also an integrated feature representation of the temporal dimension. These features can provide generalized knowledge for the SNN and enhance the original SNN structure instead of pre-training a new network with more parameters.

**Event-Based Data Augmentation.** Due to the limited amount of event data, directly implementing data augmentation to increase the amount of training data is a feasible strategy. Li et al. (2022c) propose neuromorphic data augmentation to stabilize SNN training and improve generalization. Shen, Zhao, and Zeng (2022b) design an augmentation strategy for event stream data, and perform the mixing of different event streams by Gaussian mixing model, while assigning labels to the mixed samples by calculating the relative distance of event streams. Our method is orthogonal to this category of methods, i.e., these data augmentation strategies can be used together with our proposed method.

**SNN Efficient Training.** Efficient training of SNNs directly is also a way to improve the generalizability of the network. Kim and Panda (2021) propose Spike Activation Lift Training to help the network to deliver information across all levels. Zhan et al. (2021) analyze the plausibility of central kernel alignment (CKA) as a domain distance measure relative to maximum mean difference (MMD) in deep SNNs. A number of subsequent works have contributed to the efficient training of the SNN (Kugele et al. 2020; Fang et al. 2021; Deng et al. 2022; Zhu et al. 2022; Dong, Zhao, and Zeng 2023; Zhao et al. 2023). Nonetheless, the performance of SNN is limited by the small amount of event data. The motivation of this paper is to solve this problem by using static data to provide generalized knowledge transfer for event data and improve the generalization of SNN.

**Preliminaries**

**Neuron Model.** We choose the Leaky Integrate-and-Fire (LIF) neuron model (Dayan and Abbott 2005), the most commonly used neuron model. The update of the membrane potential \( u \) can be written as following discrete form

\[
\begin{align*}
    u^{t+1,l} &= \tau u^{t,l} + W^l s^{t,l-1},
\end{align*}
\]

where \( \tau \) is leaky factor and \( u^{t,l} \) denotes membrane potential of the neurons in layer \( l \) at time step \( t \). \( W^l \) and \( s^{t,l} \) represent the weight parameters of the layer \( l \) and the fired spikes in layer \( l \), respectively. The membrane potential accumulates with the input until a given threshold \( V_{th} \) is exceeded, then the neuron delivers a spike and the membrane potential \( u^{t,l} \) is reset to zero. The equation can be expressed as

\[
\begin{align*}
    s^{t,l} &= H(u^{t,l} - V_{th})
\end{align*}
\]

\[
\begin{align*}
    u^{t+1,l} &= \tau u^{t,l} \cdot (1 - s^{t,l}) + W^l s^{t+1,l-1},
\end{align*}
\]

where \( H \) denotes Heaviside step function. In this paper, leaky factor \( \tau \) is set to 0.5 and threshold \( V_{th} \) to 0.5.
Processing of Neuromorphic Data. The Dynamic Vision Sensor (DVS) triggers an event at a specific pixel point when it detects a significant change in brightness. Formally, it can be expressed as

$$L(x, y, t) - L(x, y, t - \Delta t) \geq pC,$$

(4)

where $x$ and $y$ denote pixel location and $\Delta t$ means the time since last triggered event at $(x, y)$. $p$ is polarity of brightness change and $C$ is a constant contrast threshold. In this way, DVS triggers a number of events $\varepsilon$ during a time interval in the form $\varepsilon = \{(x_i, y_i, t_i, p_i)\}_{i=1}^{N}$. Due to the large number of events, we integrate them into frames to facilitate processing as the previous works (Wu et al. 2019; He et al. 2020; Fang et al. 2021; Shen, Zhao, and Zeng 2022b). Specifically, the events are divided into $T$ slices, and all events in each slice are accumulated. The $j$-th $(0 \leq j \leq T - 1)$ slice event after integration, $E(j, x, y, p)$, can be defined as

$$E(j, x, y, p) = \sum_{i=s}^{j_e-1} 1_{x,y,p}(x_i, y_i, p_i)$$

(5)

$$j_s = \left\lfloor \frac{N}{T} \right\rfloor \cdot j, \quad j_e = \left\lfloor \frac{N}{T} \right\rfloor \cdot (j + 1),$$

(6)

where $1_{x,y,p}(x_i, y_i, p_i)$ is an indicator function. $j_s$ and $j_e$ are the start and end index of event in $j$-th slice.

**Methods**

In this section, we first show the domain mismatch problem that exists for the same network trained on static and event datasets. Then, we introduce our proposed knowledge transfer loss and sliding training strategies correspondingly in terms of feature distribution and training strategy.

**Domain Mismatch**

Compared to static datasets, the scale of event datasets is relatively small, which makes the training more challenging. An intuitive solution strategy is to pre-train on the static dataset and then fine-tune on event dataset. However, this method suffers from a critical problem, i.e., there is a significant domain mismatch between the static and event data. To demonstrate this, we train on static dataset Caltech101 (Fei-Fei, Fergus, and Perona 2004) and its corresponding event dataset N-Caltech101 (Orchard et al. 2015) separately using the same spiking neural network structure. We use the central kernel alignment (CKA) method (Kornblith et al. 2019) to measure the similarity between features and compute CKA heatmap based on 4096 samples following (Nguyen, Raghu, and Kornblith 2020; Li et al. 2023). Moreover, we select LIF neurons of SNN’s first feature layer for membrane potential visualization. The results are shown in Fig. 1.

Fig. 1(a) shows that for the directly trained network, the features of static data are less similar to those extracted from the event dataset. In addition, the membrane potential distribution of neurons in the same layer of SNN is significantly different under different data training, as shown in Fig. 1(b). These results indicate that static data and event data cannot be well fused even under the same network structure. Despite the intuition that static images bring richer texture and edge information to event data, the domain difference between static and event is a hindrance. This makes the strategy of simply using static image pre-training and event fine-tuning ineffective or even counterproductive for feature extraction on event data, as shown in Fig. 1(c). Therefore, we need an efficient method to provide beneficial information for SNN on event data with the help of static images.

**Knowledge Transfer Loss Function**

The knowledge transfer loss function contains domain alignment loss and spatio-temporal regularization.

**Domain Alignment Loss.** For ease of description, we first introduce some notation. We have a labeled source domain $D_s = \{x_i^s, y_i^s\}_{i=1}^{N}$ and a small labeled target domain $D_t = \{x_i^t, y_i^t\}_{i=1}^{M}$ with feature space $X_s$ and $X_t$ respectively. We aim to leverage $D_s$ to assist in learning a better classifier $f_t : X_t \mapsto y_t$ to predict $D_t$ label $y_t \in \hat{Y}_t$.

The model for function $f$ involves a composition of two functions, i.e., $f_t = h_t \circ g_t$. Here $g_t : X \mapsto \hat{X}$ represents an embedding of the input space $X$ into a feature space $\hat{X}$, and $h_t : \hat{Z} \mapsto \hat{Y}$ is a function that predicts outputs from the feature space. We utilize the final classification head of the original model as $h_t$. This function is learned solely through supervised signal update gradients. **Critically, we want to**
provide a generalization of $g_t$ which can pave the way for learning of $h_t$ to improve the generalizability of SNN.

In this paper, the embedding function $g$ is modeled by network sharing between the source and target domains, using all layers before the last classification layer, as shown in Fig. 2. At this point, the shared $g_t = g_s = g$, the optimization objective is to find the satisfied $g$ in its hypothetical space $G$:

$$
\arg \min_{g \in G} \left( d (g (X^s_n), g (X^t_n)) - d (g (X^t_n), g (X^a_n)) \right),
$$

where $X^s_n$ and $X^t_n$ refer to the same data classes in the source and target domains while $X^a_n$ mean the data from different classes. The $d$ is a metric for judging similarity between two domains; we choose CKA here. CKA is a similarity index to better measure neural network representation similarity introduced by (Kornblith et al. 2019).

$$
\text{CKA}(K, L) = \frac{\text{HSIC}(K, L)}{\sqrt{\text{HSIC}(K, K) \times \text{HSIC}(L, L)}},
$$

where HSIC refers to Hilbert-Schmidt Independence Criterion (HSIC) (Gretton et al. 2005) and can be computed as:

$$
\text{HSIC}(K, L) = \frac{1}{(n-1)^2} \text{tr}(K J L J),
$$

where $J$ is the centering matrix $J_n = I_n - \frac{1}{n} 1_n 1_n^T$, here $I_n$ is an $n$ order unit matrix. $\text{tr}$ means trace of matrix.

To compute the CKA, we use a two-stream input paradigm: the inputs come from static image and DVS data, respectively. The closer the value of CKA is to 1 indicates that the two vectors are more correlated. For this reason, we subtract the CKA from 1, minimizing the loss, i.e., maximizing the correlation of the two inputs. We express the samples $x^s_t, x^t_t$ drawn from the whole data $X^s, X^t$. In this way, domain alignment loss (DAL) can be expressed as

$$
\mathcal{L}_d = 1 - \frac{1}{T} \sum_{t=1}^{T} \text{CKA}'(g (x^s_t, t), g (x^t_t, t)),
$$

where we use $g (x^s_t, t)$ to indicate the value of input after shared parameter function $g$, $t$ is brought in to emphasize that here is the output of $g$ at time $t$. Two samples $x^s_t, x^t_t$ are sampled from the same class, expressed by formula $y_t = y_j$, $\text{CKA}'$ represents the computation of the kernel function of the vectors followed by the computation of CKA by Eq. 8.

**Spatio-Temporal Regularization.** Due to the dynamic properties of event data, using only domain alignment loss for spatial feature alignment may miss important information in the temporal dimension. Spatio-temporal regularization provides dynamically learnable coefficients for the domain alignment loss, and such adaptive coefficients ensure specific weight assignments for data features at each time step. To prevent the model from overfitting at a certain time step, we adapt the event data classification loss at each time step (which reflects the contribution of the event frame features to the classification) as the regularization term. In this case, the knowledge transfer loss can be expressed as:

$$
\mathcal{L}_{kt} = 1 - \frac{1}{T} \sum_{t=1}^{T} \sigma(\eta_t) \text{CKA}'(g (x^s_t, t), g (x^t_t, t))
\quad + \frac{1}{T} \sum_{t=1}^{T} (1 - \sigma(\eta_t)) \mathcal{L}_{cls-e},
$$

where $\eta_t$ denotes the learnable coefficient at time step $t$ and $\sigma$ represents the sigmoid function. For classification loss of event data $\mathcal{L}_{cls-e}$, we choose the TET loss, which is proven to compensate the momentum loss of surrogate gradient and make SNN have better generalizability (Deng et al. 2022). $\mathcal{L}_{cc}$ and $\mathcal{L}_{mse}$ are the cross-entropy loss and the mean-squared loss respectively.

We add the knowledge transfer loss $\mathcal{L}_{kt}$ and classification loss of the static image $\mathcal{L}_{cls-s}$ as the total classification loss $\mathcal{L}_{all}$. The total training loss can be expressed as:

$$
\mathcal{L}_{all} = \mathcal{L}_{kt} + \mathcal{L}_{cls-s}.
$$

Figure 2: Proposed knowledge transfer framework for spiking neural network. Static image and event data are input simultaneously and share the network weights except for the last layer. The membrane potential of the neurons in the second-last layer is used to calculate the knowledge transfer loss. MP node in last layer means using membrane potential output.
as $L_{all} = \lambda_{cls-}\cdot L_{cls-} + \lambda_{kt}\cdot L_{kt}$, where $\lambda_{cls-}$ and $\lambda_{kt}$ are manually set parameters that determine the ratio of the two types of losses. The knowledge transfer loss not only learns domain-invariant features spatially, but also provides the network with more generalized knowledge by providing appropriate weighting coefficients temporally. This allows the model to adapt fine-grained to event data characteristics.

**Sliding Training Strategy**

The sliding training strategy aims to modulate the static image input portion of the training process so that the network gradually adapts from relying on domain-invariant features of static images and event data to fully processing event data. Specifically, during the training process, the inputs of static images are replaced by event data with probability, and this substitution probability increases with time steps until the end of the learning phase, by which time event data will replace all static images. Because the substitution process varies over time steps, as if the event data is replacing static images. Because the substitution process varies over time steps, as if the event data is replacing static images in a sliding time frame, we call it “sliding training”.

Separately, with $b_i$ denoting index of training batch, $b_i$ denoting total length of training batch, $e_s$ standing for current epoch and $e_m$ denoting maximum training epoch, then the probability of making a substitution $P_{replacement}$ could be expressed by the following equation

$$P_{replacement} = \left(\frac{b_i + e_s \cdot b_i}{e_s \cdot b_i}\right)^3,$$

(12)

where $e_s$ is a manual settings epoch for the end of the transfer knowledge loss effects. The value of $e_s$ is usually set to $e_m$. In the early training phase, domain invariant features are dominant, providing a stable feature learning base for the model. As time advances, the proportion of event data gradually increases and the domain alignment loss gradually decreases. This gradual transition ensures the stability of the model during the learning process and avoids training instability or convergence difficulties that may result from direct or abrupt data switching.

**Experiments**

We conduct experiments on mainstream event datasets: N-Caltech101 (Orchard et al. 2015) and N-Omniglot to evaluate the effectiveness of the proposed method. For another commonly used event dataset, CIFAR10-DVS (Li et al. 2017), since it is 10000 samples taken from 60,000 static images from the training and test sets together, it cannot be ensured that the event data in the manually delineated test set does not overlap with the static images when using the static images to assist training. To avoid this implicit data leakage, we choose the image-event paired CEP-DVS (Deng et al. 2021) dataset as an alternative.

**Experimental Settings**

We integrate all the event data into frames and then resize to 48x48 for N-Caltech101 and CEP-DVS datasets, and for N-Omniglot dataset, it is resized to 28x28. In terms of network structure, for a fair comparison, we choose VGGSNN (64C3-128C3-AP2-AP2-256C3-AP2-512C3-512C3-AP2-512C3-AP2-FC-AP2-FC) model with step 10 for N-Caltech101, Spiking-ResNet18 with step 6 for CEP-DVS, and SCNN (15C5-AP2-40C5-AP2-FC-FC) with step 12 for N-Omniglot. For the input encoding strategy, we use direct coding for static images and convert the static image to HSV (Hue, Saturation, Value) color space to minimize the mismatch between the two types of input data. To adapt the dual-channel characteristics of the event data, i.e., positive and negative polarity, we replicate the value channel and then duplicate the static image in equal time-step. All experiments are implemented based on the BrainCog framework (Zeng et al. 2023).

**Comparison with the State-of-the-Art**

We first evaluate the proposed method on the N-Caltech101 dataset with VGGSNN network and compare the proposed method with NDA (Li et al. 2022c), EventMix (Shen, Zhao, and Zeng 2022b), TET (Deng et al. 2022), TJCA-TET (Zhu et al. 2022), TKS (Dong, Zhao, and Zeng 2023) and ETC (Zhao et al. 2023). The results are presented in Tab. 1. The experimental results demonstrate that the proposed method can achieve state-of-the-art performance compared with existing methods. In particular, with the proposed method, the VGGSNN network can achieve 93.45% accuracy on the N-Caltech101 dataset. The significant performance improvement validates the effectiveness of knowledge transfer.

As for CEP-DVS and N-Omniglot datasets, there are fewer available results. We re-conducted the baseline experiments on these two datasets and compared them with our proposed method. Experimental results show that our proposed method improves accuracy over the original method. For the N-Omniglot dataset, the improvement of accuracy from knowledge transfer is not as significant as the other two datasets, this is because it is a few-shot dataset with only 20 available static images in each class, so the improvement from knowledge transfer is limited.

**Ablation Study**

In order to verify the effectiveness of the proposed method, in the subsequent ablation experiments, we take the direct training method TET (Deng et al. 2022) as our baseline.

**Knowledge Transfer Loss.** To verify the validity of the domain alignment loss (DAL) and the spatio-temporal regularization (STR) term in the knowledge transfer function, we conduct experiments on N-Caltech101 dataset with VGGSNN. As shown in Fig. 3(a), the baseline, i.e., the TET method, has overfitted at about 100 epochs earlier. Compared to the baseline method, even without employing the knowledge transfer loss in our method, merely using sliding training strategy can achieve certain performance improvement. As it gets better though with the domain alignment loss and spatio-temporal regularization to provide better generalization of the model. In Fig. 3(a), the red line is always at the top in the later training step, indicating that the best results can be achieved with these two terms.

To verify the effect of the spatio-temporal regularization, we also plot the adaptive learning coefficients of the VG-
We conduct experiments on Sliding Training Strategy. More on domain-invariant spatial information. It implies that the beginning and ending moment models focus larger coefficients at the first and last time step, which implies that spatio-temporal regularization are superior to the coefficients that are set to be fixed at each time step, which suggests that sliding training leads to a more stable performance improvement. It is worth mentioning that in the case of without sliding training, the accuracy of our method is 23.70%, which is slightly lower than the accuracy of the baseline method of direct training strategy, which is 25.70%. This is due to the relatively short training epochs for CEP-DVS, which causes the model to have trouble converging in the face of sudden data switches. Despite this, the addition of sliding training strategy solves this problem well.

Summary of Ablation Experiments. We show effectiveness of each part of our proposed method with experiments of VGGSNN on N-Caltech101 dataset and the results are shown in Tab. 4. The top line with no added methods is the baseline (TET) of direct training strategy, which is 23.70%, which is slightly lower than the accuracy of the baseline method of direct training strategy, which is 25.70%. This is due to the relatively short training epochs for CEP-DVS, which causes the model to have trouble converging in the face of sudden data switches. Despite this, the addition of sliding training strategy solves this problem well.

Table 2: Ablation experimental results for sliding training.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Network</th>
<th>Methods</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>N-Caltech101</td>
<td>VGGSNN</td>
<td>baseline</td>
<td>79.66%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KTL w/o DAL &amp; STR</td>
<td>84.14%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KTL w/ DAL</td>
<td>89.31%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KTL w/ DAL &amp; STR</td>
<td>92.64%</td>
</tr>
<tr>
<td>CEP-DVS</td>
<td>ResNet-18</td>
<td>baseline</td>
<td>25.70%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KTL w/o DAL &amp; STR</td>
<td>27.55%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KTL w/ DAL</td>
<td>29.95%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KTL w/ DAL &amp; STR</td>
<td>30.50%</td>
</tr>
</tbody>
</table>

Table 3: Ablation experimental results for sliding training.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Network</th>
<th>Methods</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>N-Caltech101</td>
<td>VGGSNN</td>
<td>w/o sliding training</td>
<td>83.56%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>w/ sliding training</td>
<td>92.64%</td>
</tr>
<tr>
<td>CEP-DVS</td>
<td>ResNet-18</td>
<td>w/o sliding training</td>
<td>23.70%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>w/ sliding training</td>
<td>30.50%</td>
</tr>
</tbody>
</table>

Figure 3: Performance of baseline and knowledge transfer loss methods on the N-Caltech101 dataset.
to assess Visual Explanations from Deep Networks. We conduct a detailed evaluation of our proposed approach on N-Caltech101 dataset using varying amounts of training data, as presented in Fig. 6. Our results show that regardless of training data amount, knowledge transfer loss results in a remarkable performance improvement. This is attributed to the knowledge transfer loss that allows the model to finely adapt to event data features, providing more generalized knowledge to the network.

**Performance of Our Method on Different Amounts of Event Data.** We conduct a detailed evaluation of our proposed approach on N-Caltech101 dataset using varying amounts of training data, as presented in Fig. 6. Our results show that regardless of training data amount, knowledge transfer loss results in a remarkable performance improvement. This is attributed to the knowledge transfer loss that allows the model to finely adapt to event data features, providing more generalized knowledge to the network.

**Conclusion**

In this paper, we explore the challenges faced by spiking neural networks when dealing with event-driven data. By using static images to assist SNN training, we improve the generalization ability of the network. Our proposed domain alignment loss and spatio-temporal regularization support knowledge transfer and alleviate the domain mismatch between static and event datasets. Meanwhile, we propose a sliding training strategy to bring greater stability to network training. Experiments on different event datasets show that our method achieves the best performance. In conclusion, this study not only provides new methods for training SNNs on event-driven datasets but also contributes to further development in the field of neuromorphic computing.

The Thirty-Eighth AAAI Conference on Artificial Intelligence (AAAI-24)

<table>
<thead>
<tr>
<th>DAL</th>
<th>STR</th>
<th>Sliding training</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>79.66%</td>
</tr>
<tr>
<td>✓</td>
<td></td>
<td>✓</td>
<td>82.07%</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>83.56%</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>90.57%</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>92.64%</td>
</tr>
</tbody>
</table>

Table 4: Ablation experimental results overview.

Figure 5: Class Activation Mapping of Caltech101 and N-Caltech101. Three categories are selected for display, the top row under each category represents static images, and the bottom row represents event data integrated into frames. The three columns from left to right represent the results of original picture, baseline and our method, respectively.

Figure 6: Performance on different amounts of event data.

is well illustrated in Fig. 5, where by introducing knowledge transfer loss, for both static pictures and event data, the network pays attention to the contour features of the category. In particular, the results on event data show that our method helps SNNs to move away from the background of the event data and focus on the features of category itself.

**Analysis and Discussion**

**Loss Landscape.** To verify that our method provides SNNs with more generalizability over event data, we utilize 2D loss-landscapes visualization (Li et al. 2018). To this end, we selected the optimal results of the baseline and our method to conduct experiments on N-Caltech101 and CEP-DVS respectively. As depicted in Fig. 4(b) and Fig. 4(d), the lowest loss area becomes flatter compared to Fig. 4(a) and Fig. 4(c), which indicates that the SNN obtains better weights with the knowledge transfer from static images.

**Visual Explanations from Deep Networks.** To assess whether our method learns domain-invariant features of static images and event data, and provides helpful information for SNNs about features of event data, we employ grad-cam++ (Chattopadhay et al. 2018) visualization method. Such visualization allows us to understand which local locations of an original image contributed most significantly to the model’s final classification decision. Ideally, static pictures and event data integrated into frames have similar object contour features when they are in the same class. This baseline. It can be seen that without the knowledge transfer loss function, the performance of model decreases a lot. In addition, the sliding training strategy provides a guarantee for stable convergence. Combined with all the approach, our method can achieve the best performance.

**DAL STR Sliding training Accuracy**

<table>
<thead>
<tr>
<th>DAL</th>
<th>STR</th>
<th>Sliding training</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>79.66%</td>
</tr>
<tr>
<td>✓</td>
<td></td>
<td>✓</td>
<td>82.07%</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>83.56%</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>90.57%</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>92.64%</td>
</tr>
</tbody>
</table>

Table 4: Ablation experimental results overview.

Figure 4: The loss landscape of visualization of our method and baseline on N-Caltech101 and CEP-DVS dataset.

Figure 5: Class Activation Mapping of Caltech101 and N-Caltech101. Three categories are selected for display, the top row under each category represents static images, and the bottom row represents event data integrated into frames. The three columns from left to right represent the results of original picture, baseline and our method, respectively.

Figure 6: Performance on different amounts of event data.

is well illustrated in Fig. 5, where by introducing knowledge transfer loss, for both static pictures and event data, the network pays attention to the contour features of the category. In particular, the results on event data show that our method helps SNNs to move away from the background of the event data and focus on the features of category itself.

**Performance of Our Method on Different Amounts of Event Data.** We conduct a detailed evaluation of our proposed approach on N-Caltech101 dataset using varying amounts of training data, as presented in Fig. 6. Our results show that regardless of training data amount, knowledge transfer loss results in a remarkable performance improvement. This is attributed to the knowledge transfer loss that allows the model to finely adapt to event data features, providing more generalized knowledge to the network.

**Conclusion**

In this paper, we explore the challenges faced by spiking neural networks when dealing with event-driven data. By using static images to assist SNN training, we improve the generalization ability of the network. Our proposed domain alignment loss and spatio-temporal regularization support knowledge transfer and alleviate the domain mismatch between static and event datasets. Meanwhile, we propose a sliding training strategy to bring greater stability to network training. Experiments on different event datasets show that our method achieves the best performance. In conclusion, this study not only provides new methods for training SNNs on event-driven datasets but also contributes to further development in the field of neuromorphic computing.
Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant No. 62372453).

References


