

Event Camera Data Pre-training

Yan Yang¹* Liyuan Pan²†* Liu Liu³

¹BDSI, ANU ²BITSZ & School of CSAT, BIT ³Cyberverse Dept., Huawei

Yan.Yang@anu.edu.au liyuan.pan@bit.edu.cn liuliu33@huawei.com

Abstract

This paper proposes a pre-trained neural network for handling event camera data. Our model is a self-supervised learning framework, and uses paired event camera data and natural RGB images for training. Our method contains three modules connected in a sequence: i) a family of event data augmentations, generating meaningful event images for self-supervised training; ii) a conditional masking strategy to sample informative event patches from event images, encouraging our model to capture the spatial layout of a scene and accelerating training; iii) a contrastive learning approach, enforcing the similarity of embeddings between matching event images, and between paired event and RGB images. An embedding projection loss is proposed to avoid the model collapse when enforcing the event image embedding similarities. A probability distribution alignment loss is proposed to encourage the event image to be consistent with its paired RGB image in the feature space. Transfer learning performance on downstream tasks shows the superiority of our method over state-of-the-art methods. For example, we achieve top-1 accuracy at 64.83% on the N-ImageNet dataset. Our code is available at <https://github.com/Yan98/Event-Camera-Data-Pre-training>.

1. Introduction

An event camera asynchronously captures the time, location, and polarity of pixel-wise changes in brightness as a sequence of events. Event cameras are widely used in many applications, e.g., recognition [22], detection [30, 27], segmentation [3], optical flow estimation [43], and SLAM [40]. Compared with conventional RGB cameras which record all pixel intensities at a fixed frame rate, event cameras enjoy a high dynamic range and temporal resolution, and are robust to lighting changes and motion blur [22, 36, 25].

This paper studies the problem of event camera data pre-training. Our model is pre-trained in a self-supervised man-

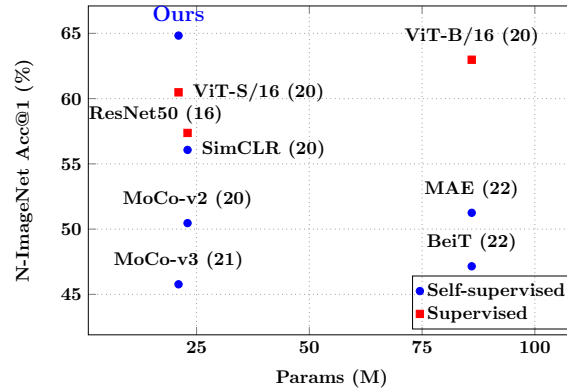


Figure 1: Comparison of our methods and state-of-the-art methods on N-ImageNet dataset [22]. The Blue circles and red squares separately denote the self-supervised and supervised pre-training methods. We show top-1 accuracy (%), i. e., $acc@1$, with respect to the number of model parameters (M). We include the publication year of each method in the brackets beside the method names.

ner, only using paired event data and RGB images for training. One can simply transfer our pre-trained model for diverse downstream tasks.

In self-supervised learning (SSL), significant progress has been made in pre-training with RGB images [19, 2, 10]. However, it is non-trivial to replicate the success on event camera data, as there is a domain gap between RGB images and event data. An RGB image records all pixel intensities of a scene and is spatially dense, while the event data only records scene changes and is spatially sparse.

For network training in the SSL framework, image augmentations (e.g., Gaussian Blur, ColorJitter, RandomResizedCrop) are one of the most important parts. The sparse event camera data can be commonly represented as an event image [22]. One may directly and wrongly perform these augmentations on event images, e.g., blurring a binary event image (0/1 valued pixels) generates a meaningless event image. In contrast, we study how to perform event data augmentations before converting to an event image.

We formulate our learning problem as a contrastive learning task. Taking event images as inputs, one may directly perform a random masking strategy to sample a fixed

† Corresponding author. * Equal contribution.

number of event patches for encouraging the model to capture the spatial layout and accelerating training. However, an event image is spatially sparse, and random masking would generate non-informative patches, leading to training instability. To mitigate this problem, we propose a conditional masking strategy to sample informative patches.

With event patches, we are able to learn discriminative event embeddings, i. e., pulling together embeddings from similar event images while pushing away embeddings from dissimilar ones. Surprisingly, we find that simply performing metric learning in the event embedding space leads to model collapse, producing over-similar embeddings. The reason comes from the spatial sparsity of event images. To solve this problem, we find that embeddings from paired RGB images can be used as a regularizer, and we propose an embedding projection loss to solve the collapse.

With paired event data and RGB images, we also aim to pull together embeddings from matched pairs. This is motivated by the fact that many well pre-trained RGB networks are available, and an event image is less informative than its paired RGB image. Therefore, the RGB network serves as a teacher for our event network, and we propose a probability distribution alignment loss for the learning.

Our contributions are summarized as follows:

- A self-supervised framework for event camera data pre-training. The pre-trained model can be transferred to diverse downstream tasks;
- A family of event data augmentations, generating meaningful event images;
- A conditional masking strategy, sampling informative event patches for network training;
- An embedding projection loss, using paired RGB embeddings to regularize event embeddings to avoid model collapse;
- A probability distribution alignment loss for aligning embeddings from the paired event and RGB images.
- We achieve state-of-the-art performance in standard event benchmark datasets (e.g., Fig. 1).

2. Related Work

The SSL frameworks can be generally divided into two categories: contrastive learning and masked modeling. We briefly review their recent achievements, and then introduce event datasets for diverse computer vision tasks.

Contrastive learning. This approach generally assumes augmentation invariance of images [7, 20]. Two or more views of each image are generated for instance discrimination that enforces embedding similarity and dissimilarity among the views [7, 20, 8, 10, 41]. Only enforcing the embedding similarity is also possible and has been studied in [9, 18]. In addition to model design and optimization objectives, contrastive learning approaches usually rely

on strong augmentations over images to boost model performance [7, 20, 8, 10, 4, 9]. Under certain tasks, contrastive learning has shown better performance than supervised pre-training [14, 10]. However, one notable drawback of contrastive learning is suffering from model collapse and training instability. Diverse methods including asymmetric network designs (e.g., maintaining a momentum network) [7, 20], partial weight freezing [10], and group-based discrimination [4, 5] are introduced to avoid the model collapse and instability issue.

Masked modeling. Reconstructing masked inputs from the (i. e., unmasked) visible ones is a popular self-supervised learning objective motivated by the idea of auto-encoding. The pioneer works can be specified to the natural language processing domain, e.g., Bert [13] and GPT [31]. Recently, masked modeling has been formulated in the image domain, where the objective is defined in a similar vein, and the masking of images is done pixel-wisely or patch-wisely [6, 19, 14, 2, 42]. Some works [2, 42] turn the masked modeling into a classification problem by predicting discrete indices assigned to the masked patches by a tokenizer, e.g., pre-trained discrete VAE [32, 15] or self-distilled network [42, 5]. One could also target to directly regress the pixel intensity of masked patches [6, 19, 14].

Event datasets. The event camera is a novel sensor that asynchronously captures the time, location, and polarity (i. e., direction) of per-pixel brightness change as a sequence of events. With growing interest in event-based computer vision tasks, researchers have collected a wide range of datasets for object recognition [22, 35, 28, 11], semantic segmentation [3], optical flow estimations [43], and so forth [33, 26, 39]. To leverage existing computer vision algorithms, e.g., CNN and ViT, the majority of event-based vision frameworks convert event data into image/video-like grid representations, where the conversion is done either learnable [34] or by directly using the position and time of each event [22]. This paper leverages the event image representation to study the event-based SSL algorithm that benefits diverse event-based downstream tasks.

3. Method

We start with a brief overview of background knowledge, and then present our self-supervised learning framework, in this section. The overall architecture is shown in Fig. 2.

Preliminary. Contrastive learning aims to learn an embedding space, where similar image pairs stay close to each other while dissimilar ones are far apart. Specifically, images are embedded into vectors to collect a query set $\{\mathbf{q}\}$ and a key set $\{\mathbf{k}\}$. For each query \mathbf{q} , we have a matching key \mathbf{k}_+ and non-matching keys $\{\mathbf{k}_-\}$. Usually, \mathbf{q} and \mathbf{k}_+ are generated from views of the same instance, while \mathbf{q} and

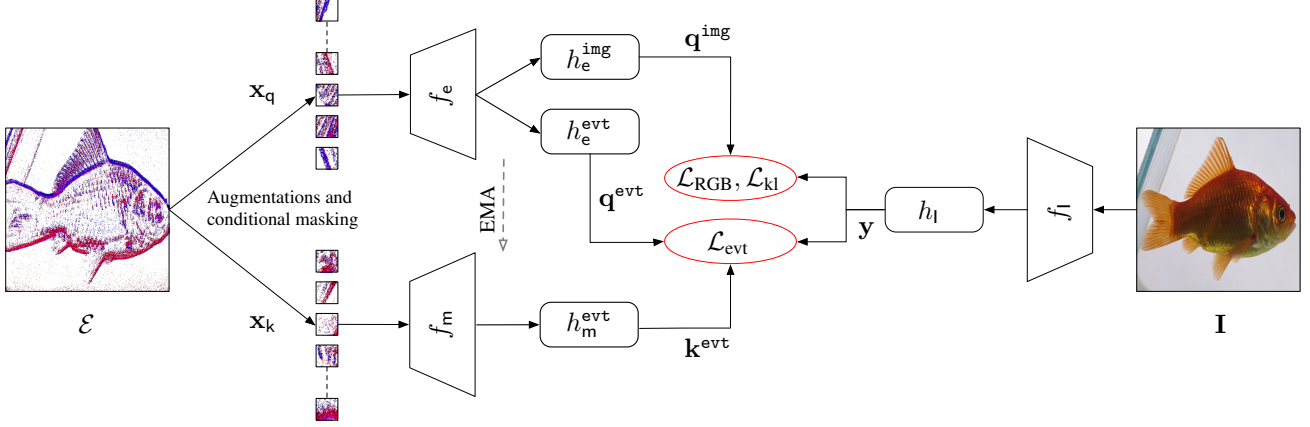


Figure 2: The overall architecture. For pre-training, our method takes event data \mathcal{E} and its paired natural RGB image \mathbf{I} as inputs, and outputs a pre-trained network f_e . Given \mathcal{E} (its abstract representation is used for visualization purposes), we first consecutively perform data augmentations, event image generation, and conditional masking to obtain two patch sets ($\mathbf{x}_q, \mathbf{x}_k$). Second, f_e extracts features from event patch set \mathbf{x}_q , and h_e^{img} and h_e^{evt} separately project features from f_e to latent embeddings \mathbf{q}^{img} and \mathbf{q}^{evt} . f_m and h_m^{evt} are the momentum of f_e and h_e^{evt} , and are updated by the exponential moving average (EMA). The momentum network takes patch set \mathbf{x}_k as input and generates an embedding \mathbf{k}^{evt} . At the same time, the natural RGB image \mathbf{I} is embedded into $\mathbf{y} = f_1(h_1(\mathbf{I}))$. Finally, we perform event discrimination, and event and natural RGB image discrimination to train our model. We optimize the network by \mathcal{L}_{evt} (Eq. 3), \mathcal{L}_{RGB} (Eq. 4), and \mathcal{L}_{kl} (Eq. 6). \mathcal{L}_{evt} is an event embedding projection loss aiming to pull together paired event embeddings \mathbf{q}^{evt} and \mathbf{k}^{evt} , for event discrimination. \mathcal{L}_{RGB} aims to pull together paired event and RGB embeddings \mathbf{q}^{evt} and \mathbf{y} , for event and natural RGB image discrimination. \mathcal{L}_{kl} aims to drive f_e learning discriminative event embeddings, towards well-structured embedding space of natural RGB images.

$\{\mathbf{k}_-\}$ are generated from views of different instances. Contrastive learning aims to pull together embeddings \mathbf{q} and \mathbf{k}_+ , and pushes away embeddings \mathbf{q} and $\{\mathbf{k}_-\}$. In this paper, we use the InfoNCE loss [37],

$$\mathcal{L}_{\text{ncc}}(\mathbf{q}, \{\mathbf{k}\}) = -\log \frac{\exp(\mathbf{q} \cdot \mathbf{k}_+ / \tau)}{\exp(\mathbf{q} \cdot \mathbf{k}_+ / \tau) + \sum_{\mathbf{k}_-} \exp(\mathbf{q} \cdot \mathbf{k}_- / \tau)},$$

where \mathbf{q} and \mathbf{k} are L_2 normalized to a metric space, and the similarity between them is then measured by the cosine similarity using dot-product (\cdot). τ is a temperature hyperparameter [10].

Overall architecture. Given an event data $\mathcal{E} = (u_i, t_i, p_i)_{i=1}^{\mathcal{N}}$ and a paired natural RGB image \mathbf{I} , where u_i , t_i , and p_i separately denotes spatial location, time, and polarity of each event, and \mathcal{N} is the length of the event data.

We aim to pre-train a neural network f_e , such that f_e can generate discriminative features for benefiting diverse event-based downstream tasks. Our method is self-supervised and has three components: i) event image patch generation. Given input \mathcal{E} , it generates matching patches ($\mathbf{x}_q, \mathbf{x}_k$) on \mathcal{E} ; ii) event discrimination. It aims to pull together embeddings of ($\mathbf{x}_q, \mathbf{x}_k$); iii) event and RGB image discrimination. It aims to pull together embeddings of \mathbf{x}_q and \mathbf{I} . Details of the above three components are given in the following paragraphs.

Event image patch generation. To convert \mathcal{E} into two matching patches ($\mathbf{x}_q, \mathbf{x}_k$), we consecutively apply our data

augmentations, event image generation, and conditional masking strategy. We perform event data augmentations before converting them to event images. Please refer to the supplementary material for details, i.e., how to perform RandomResizedCrop, GaussianBlur, and ColorJitter for \mathcal{E} . With augmented \mathcal{E} , we first generate an event image by applying the event histogram algorithm [23], and then use our conditional masking strategy to obtain patches \mathbf{x}_q and patches \mathbf{x}_k .

Given an event image, considering its sparsity, using a random masking strategy to sample patches is prone to generate meaningless/non-informative patches. Therefore, a conditional masking strategy is proposed to sample patches. Let $\{\mathbf{p}_i\}_{i=1}^{\mathcal{P}}$ be a patch set of an event image, \mathbf{p}_i is the i -th patch, and \mathcal{P} is the cardinality of the set. After vectorizing \mathbf{p}_i , we calculate the information quantity d_i of each patch,

$$d_i = |\mathbf{p}_i| \cdot \mathbf{1}, \quad \forall i \in [1, \dots, \mathcal{P}], \quad (1)$$

where $\mathbf{1}$ denotes a vector of ones. Collecting \mathcal{P} information quantities and L_1 normalizing them, we obtain a probability distribution. A patch probability describes how likely it contains meaningful information. We randomly sample a fixed number ($\ll \mathcal{P}$) of patches according to the probability distribution, resulting in \mathbf{x}_q . Then, the same process is performed to generate \mathbf{x}_k .

Event discrimination. With patches \mathbf{x}_q and patches \mathbf{x}_k , we show how to pull together embeddings of them. \mathbf{x}_q is fed to network f_e to extract features, and features from f_e

are fed to a projection head h_e^{evt} to extract an embedding \mathbf{q}^{evt} , $\mathbf{q}^{\text{evt}} = h_e^{\text{evt}}(f_e(\mathbf{x}_q))$. For self-supervised training, \mathbf{x}_k is fed to f_m and h_m^{evt} to extract an embedding \mathbf{k}^{evt} , $\mathbf{k}^{\text{evt}} = h_m^{\text{evt}}(f_m(\mathbf{x}_k))$, where f_m and h_m^{evt} are the momentum [20] of f_e and h_e^{evt} , respectively.

To enforce the similarity between embeddings \mathbf{q}^{evt} and \mathbf{k}^{evt} , one may directly optimize the InfoNCE loss $\mathcal{L}_{\text{ncc}}(\mathbf{q}^{\text{evt}}, \{\mathbf{k}^{\text{evt}}\})$. However, we find that optimized embeddings collapse, i. e., they are over-similar. The reason would be the sparsity of event images, and the sparsity decreases the discriminativeness of event embeddings.

To solve this collapse problem, interestingly, we find that the embedding $\mathbf{y} = h_l(f_l(\mathbf{I}))$ of the paired natural RGB image \mathbf{I} is a basis vector and provides good regularization. f_l is an image feature extraction network, and h_l projects features to an embedding. We have the event embedding projection loss,

$$\mathcal{L}_{\text{evt}} = \mathcal{L}_{\text{ncc}}(\zeta(\mathbf{q}^{\text{evt}}, \mathbf{y}), \{\zeta(\mathbf{k}^{\text{evt}}, \mathbf{y})\}), \quad (2)$$

$$\zeta(\mathbf{v}_1, \mathbf{v}_2) = \mathbf{v}_1 \cdot \mathbf{v}_2 \frac{\mathbf{v}_2}{\|\mathbf{v}_2\|^2}, \quad (3)$$

where $\zeta(\mathbf{v}_1, \mathbf{v}_2)$ is the projection function. Here, $\zeta(\mathbf{q}^{\text{evt}}, \mathbf{y})$ and $\zeta(\mathbf{k}^{\text{evt}}, \mathbf{y})$ separately projects event embeddings \mathbf{q}^{evt} and \mathbf{k}^{evt} to embedding \mathbf{y} . We do not perform L_2 normalization on $\zeta(\mathbf{q}^{\text{evt}}, \mathbf{y})$ and $\zeta(\mathbf{k}^{\text{evt}}, \mathbf{y})$ for calculating \mathcal{L}_{evt} .

Event and RGB image discrimination. Considering the sparsity of the event image, a single event image is less informative than an RGB image, possessing difficulty for self-supervised event network training. In contrast, many well-trained RGB networks are available. We aim to teach our event network f_e , using well pre-trained RGB network f_l . We pull together embeddings of paired event and RGB images, \mathbf{x}_q and \mathbf{I} . Features from f_e are fed to a projection head h_e^{img} to extract an event image embedding \mathbf{q}^{img} . Given embeddings \mathbf{q}^{img} and \mathbf{y} , we enforce their similarity by optimizing the InfoNCE loss,

$$\mathcal{L}_{\text{RGB}} = \mathcal{L}_{\text{ncc}}(\mathbf{q}^{\text{img}}, \{\mathbf{y}\}). \quad (4)$$

To better align event and RGB embedding spaces, we first separately fit two probability distributions in the event and RGB embedding spaces, and then use Kullback–Leibler divergence to minimize the distribution mismatch.

Specifically, given a batch of event embeddings $\{\mathbf{q}^{\text{img}}\}$, we first compute the pairwise embedding similarity and then fit an exponential kernel to the similarities to compute probability scores. The probability score of the (i, j) -th pair is given by,

$$s_{i,j}^{\mathbf{q}} = \frac{\exp(\mathbf{k}_i \cdot \mathbf{k}_j / \tau)}{\sum_j \exp(\mathbf{k}_i \cdot \mathbf{k}_j / \tau)}, \quad (5)$$

where \mathbf{k}_i and \mathbf{k}_j are the i -th and j -th embedding of the batch $\{\mathbf{q}^{\text{img}}\}$. The probability score of \mathbf{y} is obtained in the

same way and is denoted as $s_{i,j}^{\mathbf{y}}$. Our probability distribution alignment loss is given by,

$$\mathcal{L}_{\text{kl}} = \sum_i \sum_j s_{i,j}^{\mathbf{q}} \cdot \log\left(\frac{s_{i,j}^{\mathbf{q}}}{s_{i,j}^{\mathbf{y}}}\right) \quad (6)$$

Losses. Our network is trained end-to-end, and the total loss is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{evt}} + \mathcal{L}_{\text{RGB}} + \lambda_1 \mathcal{L}_{\text{kl}}, \quad (7)$$

where λ_1 is a hyper-parameter for balancing the losses.

4. Experiments

4.1. Experimental Setup

Pre-training dataset. We use the N-ImageNet [22] and ImageNet-1K [12] datasets for pre-training. The N-ImageNet dataset is built from the ImageNet-1K dataset, where a moving event camera observes natural RGB images displayed by a monitor. Similar to the ImageNet-1K, it contains 1,781,167 samples of event data, covering 1,000 object classes. All event samples are recorded in 480×640 resolution. We resize them to 224×224 resolution, and use the official training set for pre-training.

Implementation. We explore two backbones ViT-S/16 and ResNet50 for our method, and separately report our pre-training performance. We use the backbone for f_e and f_m , and the same projection head as MoCo-v3 for h_e^{evt} , h_m^{evt} , and h_e^{img} . We use SSL pre-trained ViT-B/32 for the RGB image backbone f_l , and set h_l to a single linear layer. The hyper-parameters λ_1 is set to 2. Please refer to the supplement material for optimization schemes, ablation of f_e , f_l and h_l . Our method is implemented in Pytorch [29].

Transfer learning tasks. We evaluate our method and state-of-the-art methods on three downstream tasks: object recognition (Sec. 4.2), optical flow estimation (Sec. 4.3), and semantic segmentation (Sec. 4.4).

Baselines. Our method is compared with four groups of methods: i) Previous best. We compare with state-of-the-art methods for each task; ii) Training from scratch. We train state-of-the-art methods with random weight initialization; iii) Transfer learning of supervised pre-training. The initial weights of state-of-the-art methods are obtained in a supervised manner using the ImageNet-1K dataset; iv) Transfer learning of self-supervised pre-training. The initial weights of state-of-the-art methods are obtained in a self-supervised manner using the ImageNet-1K dataset.

4.2. Object Recognition

We first show our object recognition performance on the large-scale N-ImageNet [22] dataset and then report our performance on three small-scale datasets, N-Cars [35], N-Caltech101 [28], and CIFAR-10-DVS [11].

Table 1: Comparison of object recognition accuracies on the N-ImageNet dataset [22]. We show the top-1 and top-5 accuracies (i. e., acc@1 and acc@5 (%)) of state-of-the-art methods.

Method	Architecture	Parameters	Pre-training Epoch	Fine-tuning	
				acc@1	acc@5
<i>The best performance in the literature.</i>					
EST [16]	-	21M	-	48.93	-
<i>Training from scratch, i. e., random weight initialization.</i>					
ViT [14]	ViT-S/16	21M	-	46.70	69.89
ViT [14]	ViT-B/16	86M	-	51.23	74.50
ResNet [21]	ResNet50	23M	-	50.07	74.83
<i>Transfer learning of supervised pre-training methods, i. e., initial weights learned in a supervised manner.</i>					
ViT [14]	ViT-S/16	21M	300	60.48	83.02
ViT [14]	ViT-B/16	86M	300	62.98	84.75
ResNet [21]	ResNet50	23M	90	57.37	80.93
<i>Transfer learning of self-supervised pre-training methods, i. e., initial weights learned in a self-supervised manner.</i>					
SimCLR [7]	ResNet50	23M	100	56.07	80.49
MoCo-v2 [8]	ResNet50	23M	200	50.46	75.67
MoCo-v3 [10]	ViT-S/16	21M	300	45.77	68.89
BeiT [2]	ViT-B/16	86M	800	47.15	69.27
iBoT [42]	ViT-S/16	21M	800	19.55	38.72
MAE [19]	ViT-B/16	86M	800	51.25	72.64
Ours	ResNet50	23M	300	59.80	82.04
Ours	ViT-S/16	21M	300	64.83	86.30

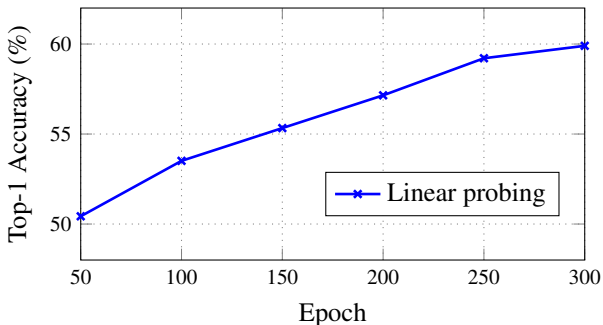


Figure 3: The linear probing accuracy of our method with respect to the number of training epochs.

Results on the large-scale N-ImageNet dataset. The comparisons are given in Tab. 1. It shows that fine-tuning our pre-trained model with a ViT-S/16 backbone achieves a top-1 accuracy at 64.83%, outperforming all other methods. Additionally, we examine the linear probing performance of this pre-trained model, and it achieves a top-1 accuracy at 59.90%, outperforming methods in the self-supervised group. The linear probing accuracies of our method with respect to the number of training epochs are given in Fig. 3.

For methods (except ours) in the self-supervised group, we find that they overfit easily (even achieving a near-perfect top-1 training accuracy) when fine-tuning on the N-ImageNet dataset, though we have tried our best to use di-

verse regularization techniques. This further demonstrates the value of this paper – a self-supervised learning framework for event camera data pre-training.

Results on other small-scale datasets. The comparisons on N-cars [35], N-Caltech101 [28], and CIFAR-10-DVS [11] datasets are given in Tab. 2. Note that the N-Caltech101 and CIFAR-10-DVS have not provided training and testing splits. We therefore randomly split them for generating training and testing datasets (please refer to the supplementary materials). Our pre-trained model with a ViT-S/16 backbone outperforms all other methods, with 97.93%, 87.66%, and 78.00% top-1 accuracy on N-Cars, N-Caltech101, and CIFAR-10-DVS datasets, respectively.

4.3. Optical Flow Estimation

We show our optical flow estimation performance on the MVSEC dataset [43]. Following [19, 2], we simply append a decoder network to pre-trained networks to estimate the optical flow. Please refer to the supplementary material for architecture details and train-test splitting. The comparisons on the ‘indoor_flying1’, ‘indoor_flying2’, and ‘indoor_flying3’ scenes are given in Tab. 3.

Compared with other methods, our method with a ViT-S/16 backbone has the lowest AEEs and outlier ratios, showing the effectiveness of our pre-trained model for the optical flow estimation task. We show optical flow prediction examples of our method in Fig. 4.

Table 2: Comparison of object recognition accuracies on the N-Cars [35], N-Caltech101 [28], and CIFAR-10-DVS [11] datasets. We show the top-1 accuracy for clarity.

Method	Architecture	N-Cars	N-Caltech101	CIFAR-10-DVS
<i>The best performance in the literature.</i>				
N-ImageNet [22]	-	94.73	86.81	73.72
<i>Training from scratch, i. e., random weight initialization.</i>				
ViT [14]	ViT-S/16	89.14	55.63	52.45
ViT [14]	ViT-B/16	93.09	67.11	55.15
ResNet [21]	ResNet50	91.20	62.69	56.65
<i>Transfer learning of supervised pre-training methods, i. e., initial weights learned in a supervised manner.</i>				
ViT [14]	ViT-S/16	96.76	85.02	76.10
ViT [14]	ViT-B/16	97.56	86.45	77.45
ResNet [21]	ResNet50	97.61	86.51	73.40
<i>Transfer learning of self-supervised pre-training methods, i. e., initial weights learned in a self-supervised manner.</i>				
SimCLR [7]	ResNet50	97.10	86.57	75.15
MoCo-v2 [8]	ResNet50	96.64	84.16	74.65
MoCo-v3 [10]	ViT-S/16	95.33	76.59	68.40
BeiT [2]	ViT-B/16	90.61	53.10	53.15
iBoT [42]	ViT-S/16	92.30	47.36	56.10
MAE [19]	ViT-B/16	95.34	67.68	68.65
Ours	ResNet50	98.01	87.08	74.75
Ours	ViT-S/16	97.93	87.66	78.00

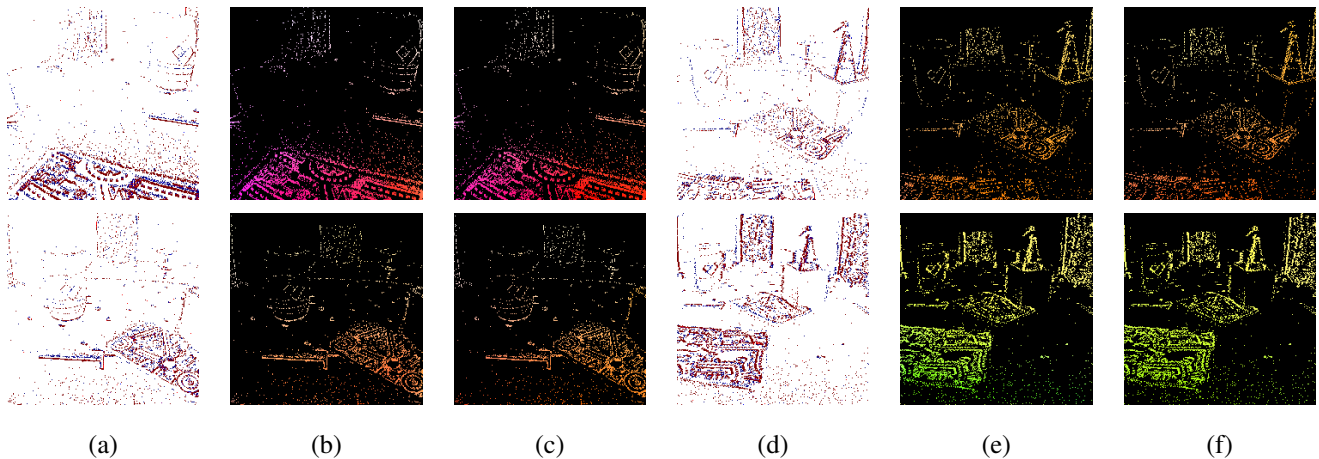


Figure 4: Optical flow prediction examples of our method on the MVSEC dataset [43]. (a)/(d) are event images, where red and blue indicate positive and negative events. (b)/(e) are ground-truth optical flows. (c)/(f) are our predicted optical flows.

4.4. Semantic Segmentation

We show our semantic segmentation performance on the DDD17 [3, 1] and DSEC datasets [17, 36]. Following [2], we simply append a decoder network to pre-trained networks to estimate semantic labels, and use the mean intersection over union (mIoU) metric to evaluate methods. The comparisons are given in Tab. 4.

The performance of our method with a ResNet50 backbone is comparable with respect to the state-of-the-art method ESS [36], which uses additional RGB images and their semantic labels for training. For methods only using

event data and semantic labels for training, our method outperforms the state-of-the-art method EV-SegNet [1]. Please refer to Fig. 5 for our semantic segmentation examples.

4.5. Discussion

Analysis of attention maps. We visualize attention maps of our pre-trained model in Fig. 6, where features from the last layer of our pre-trained model are used to compute the attention map. The results show that our pre-trained model successfully focuses on semantic meaningful objects on the noisy event images (e.g., spider on the second row of Fig. 6).

Table 3: Comparison of optical flow estimation on the MVSEC dataset [43]. We use average end-point error (AEE) and percentage of outliers (%) for evaluation. Similar to the KITTI benchmark [24], the outlier measures the percentage of pixels that has end-point error larger than three units and 5% of the ground truth optical flow.

Method	Backbone	<i>indoor_flying1</i>		<i>indoor_flying2</i>		<i>indoor_flying3</i>	
		AEE	Outlier	AEE	Outlier	AEE	Outlier
<i>The best performance in the literature.</i>							
EST [16]	-	1.24	5.09	2.05	19.90	1.71	11.67
DCEIFlow [38]	-	0.75	0.60	1.39	8.01	1.13	5.29
<i>Training from scratch, i. e., random weight initialization.</i>							
ViT [14]	ViT-S/16	0.68	0.13	1.38	7.58	1.08	3.76
ViT [14]	ViT-B/16	0.64	0.19	1.36	7.21	1.05	3.86
ResNet [21]	ResNet50	0.73	0.66	1.55	9.81	1.23	5.77
<i>Transfer learning of supervised pre-training methods, i. e., initial weights learned in a supervised manner.</i>							
ViT [14]	ViT-S/16	0.88	3.06	1.79	16.63	1.49	8.66
ViT [14]	ViT-B/16	0.65	0.45	1.34	7.65	1.11	4.96
ResNet [21]	ResNet50	0.60	0.23	1.37	8.76	1.15	5.34
<i>Transfer learning of self-supervised pre-training methods, i. e., initial weights learned in a self-supervised manner.</i>							
SimCLR [7]	ResNet50	0.65	0.49	1.45	9.33	1.19	5.51
MoCo-v2 [8]	ResNet50	0.61	0.46	1.36	8.68	1.13	5.20
MoCo-v3 [10]	ViT-S/16	0.66	0.35	1.41	8.23	1.17	5.10
BeiT [2]	ViT-B/16	0.64	0.29	1.32	7.34	1.07	4.32
iBoT [42]	ViT-S/16	0.80	0.81	1.47	8.77	1.16	5.43
MAE [19]	ViT-B/16	0.61	0.17	1.29	6.95	1.11	4.64
Ours	ResNet50	0.60	0.35	1.35	8.57	1.12	5.26
Ours	ViT-S/16	0.61	0.05	1.26	6.69	1.00	3.11

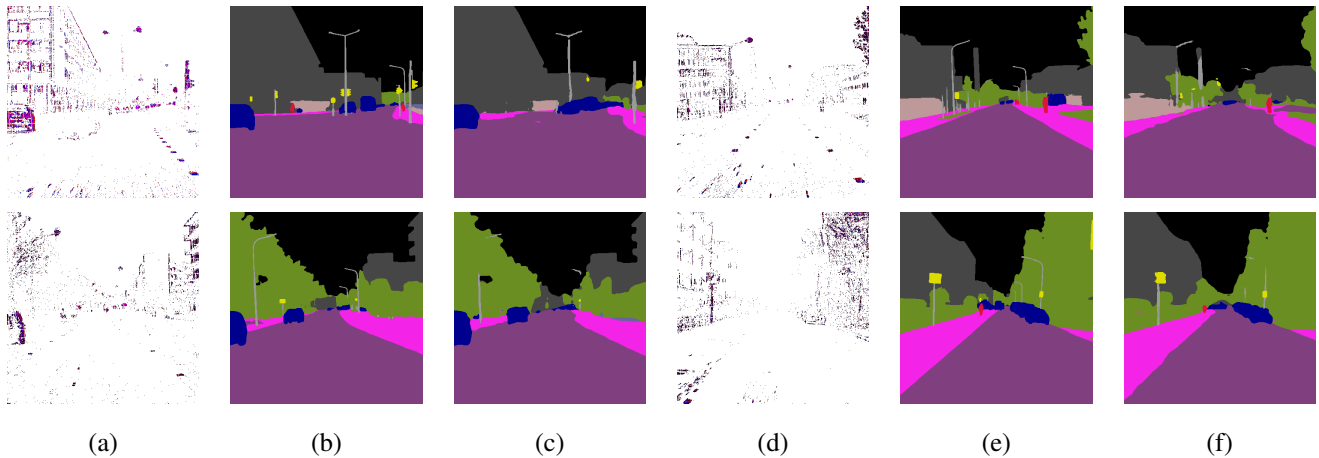


Figure 5: Semantic segmentation prediction examples of our method on the DSEC dataset [43]. (a)/(d) are event images, where red and blue indicate positive and negative events. (b)/(e) are ground-truth segmentation images, and pixel colors denote semantic classes. (c)/(f) are our predicted segmentation images.

This strong pattern discovery ability of our method potentially explains the effectiveness of our pre-trained model when transferring to diverse downstream tasks.

Effectiveness of data augmentations. Generating different views of the same data is one of the most important parts of the self-supervised learning framework. We compare two methods: i) previously commonly used methods [10, 19]; ii)

our event data augmentations. By pre-training our method with the above two different augmentation methods for 100 epochs on the N-ImageNet, we obtain 46.1% and 53.5% top-1 accuracy with linear probing, respectively.

Pre-train MAE using event data? Can we pre-train the state-of-the-art self-supervised method MAE [19] using event data? To check the feasibility, we perform a bi-

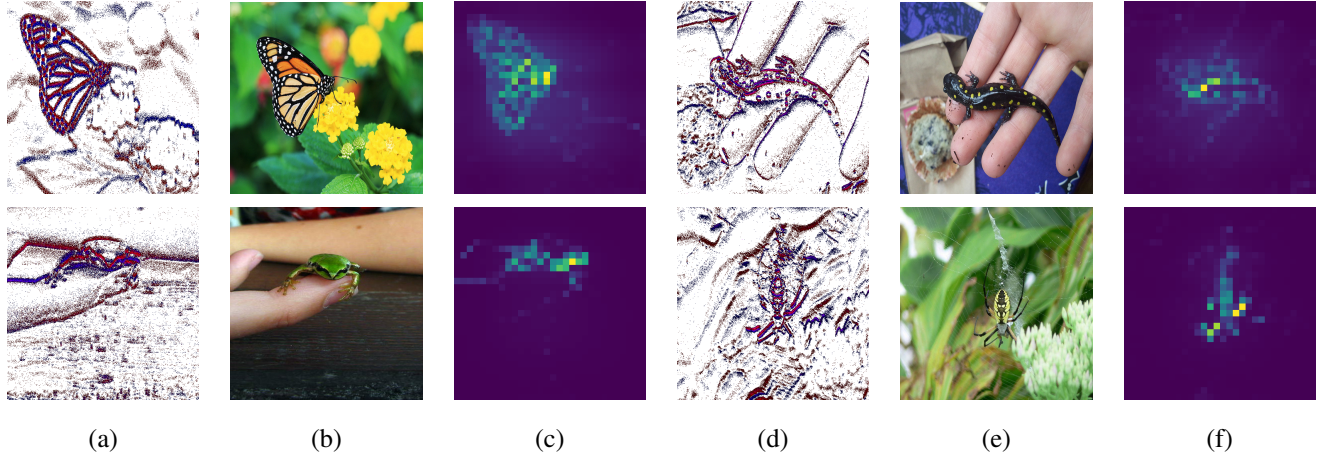


Figure 6: Attention maps of our pre-trained model (without any fine-tuning) on sample data from the N-ImageNet dataset [22]. (a)/(d) are event images. Similarly, we use red and blue to indicate positive and negative events. (b)/(e) are corresponding natural RGB images used for visualization assistance. (c)/(f) are our attention maps.

Table 4: Comparison of semantic segmentation on the DDD17 [3, 1] and DSEC datasets [17, 36]. Following [2], we use the mean intersection over union (mIoU (%)) for comparison. Our method and EV-SegNet [1] only use event data and corresponding semantic labels for training. ESS [36] uses additional RGB images and semantic labels in the training stage.

Method	Backbone	DDD17	DSEC
<i>The best performance in the literature.</i>			
EV-SegNet [1]	-	54.81	51.76
ESS [36]	-	61.37	53.29
<i>Training from scratch.</i>			
ViT [14]	ViT-S/16	48.76	40.53
ViT [14]	ViT-B/16	43.89	38.24
ResNet	ResNet50	56.96	57.60
<i>Transfer learning of supervised pre-training methods.</i>			
ViT [14]	ViT-S/16	54.12	42.92
ViT [14]	ViT-B/16	54.06	45.55
ResNet	ResNet50	59.25	58.50
<i>Transfer learning of self-supervised pre-training methods.</i>			
SimCLR [7]	ResNet50	57.22	59.06
MoCo-v2 [8]	ResNet50	58.28	59.09
MoCo-v3 [10]	ViT-S/16	53.65	49.21
BeiT [2]	ViT-B/16	52.39	46.52
IBoT [42]	ViT-S/16	49.94	42.53
MAE [19]	ViT-B/16	52.36	47.56
Ours	ViT-S/16	54.66	47.91
Ours	ResNet50	59.15	59.16

nary search to find the best masking ratio and optimization schema for MAE. MAE with a ViT-B/16 backbone obtains top-1 accuracy at 55.45% after fine-tuning on the N-ImageNet dataset. In comparison, our method with a ViT-S/16 backbone achieves top-1 accuracy at 64.83%.

Table 5: Scaling the number of parameters of f_e and f_m .

Backbone	Linear Probing		Fine-tuning	
	acc@1	acc@5	acc@1	acc@5
ViT-S/16	59.90	82.26	64.84	86.30
ViT-B/16	61.75	82.53	68.31	88.02
ViT-L/16	64.53	84.90	71.05	89.86

Model Scalability. We scale our backbone (f_e and f_m) from ViT-S/16 to ViT-L/16. The results are given in Tab. 5. The accuracy of our method improves with respect to the increasing number of model parameters of ViT.

5. Conclusion

In this paper, we have trained a neural network for processing event camera data, as a self-supervised learning framework. The method contains three key components: a family of event data augmentations, a conditional masking strategy, and a contrastive learning approach. Our key insight is enforcing the similarity of embeddings between matching event images and between paired event and RGB images to pre-train our model. Extensive experiments on downstream tasks (i. e., object recognition, optical flow estimation, and semantic segmentation) demonstrate the superiority of our method over past methods.

Broader impacts. Our model is promising to be extended for zero-shot learning by using vision-language methods from the RGB image domain to drive event data network. We hope this paper will inspire future work.

Acknowledgment. This work was supported in part by the Beijing Institute of Technology Research Fund Program for Young Scholars and National Natural Science Foundation of China 62302045.

References

- [1] Iñigo Alonso and Ana C. Murillo. Ev-segnet: Semantic segmentation for event-based cameras. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 1624–1633. Computer Vision Foundation / IEEE, 2019. 6, 8
- [2] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: BERT pre-training of image transformers. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. 1, 2, 5, 6, 7, 8
- [3] Jonathan Binas, Daniel Neil, Shih-Chii Liu, and Tobi Delbrück. DDD17: end-to-end DAVIS driving dataset. *CoRR*, abs/1711.01458, 2017. 1, 2, 6, 8
- [4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 2
- [5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 9630–9640. IEEE, 2021. 2
- [6] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pre-training from pixels. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1691–1703. PMLR, 2020. 2
- [7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 2020. 2, 5, 6, 7, 8
- [8] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2, 5, 6, 7, 8
- [9] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 15750–15758. Computer Vision Foundation / IEEE, 2021. 2
- [10] Xinlei Chen*, Saining Xie*, and Kaiming He. An empirical study of training self-supervised vision transformers. *arXiv preprint arXiv:2104.02057*, 2021. 1, 2, 3, 5, 6, 7, 8
- [11] Wensheng Cheng, Hao Luo, Wen Yang, Lei Yu, and Wei Li. Structure-aware network for lane marker extraction with dynamic vision sensor. *CoRR*, abs/2008.06204, 2020. 2, 4, 5, 6
- [12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 248–255. IEEE Computer Society, 2009. 4
- [13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. 2
- [14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 2, 5, 6, 7, 8
- [15] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 12873–12883. Computer Vision Foundation / IEEE, 2021. 2
- [16] Daniel Gehrig, Antonio Loquercio, Konstantinos G. Derpanis, and Davide Scaramuzza. End-to-end learning of representations for asynchronous event-based data. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 5632–5642. IEEE, 2019. 5, 7
- [17] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. DSEC: A stereo event camera dataset for driving scenarios. *IEEE Robotics Autom. Lett.*, 6(3):4947–4954, 2021. 6, 8
- [18] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 2
- [19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 15979–15988. IEEE, 2022. 1, 2, 5, 6, 7, 8

- [20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9726–9735. Computer Vision Foundation / IEEE, 2020. [2](#), [4](#)
- [21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. [5](#), [6](#), [7](#)
- [22] Junho Kim, Jaehyeok Bae, Gangin Park, Dongsu Zhang, and Young Min Kim. N-imagenet: Towards robust, fine-grained object recognition with event cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2146–2156, October 2021. [1](#), [2](#), [4](#), [5](#), [6](#), [8](#)
- [23] Ana I. Maqueda, Antonio Loquercio, Guillermo Gallego, Narciso García, and Davide Scaramuzza. Event-based vision meets deep learning on steering prediction for self-driving cars. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 5419–5427. Computer Vision Foundation / IEEE Computer Society, 2018. [3](#)
- [24] M. Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W5:427–434, 08 2015. [7](#)
- [25] Nico Messikommer, Daniel Gehrig, Mathias Gehrig, and Davide Scaramuzza. Bridging the gap between events and frames through unsupervised domain adaptation. *IEEE Robotics Autom. Lett.*, 7(2):3515–3522, 2022. [1](#)
- [26] Nico Messikommer, Daniel Gehrig, Antonio Loquercio, and Davide Scaramuzza. Event-based asynchronous sparse convolutional networks. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part VIII*, volume 12353 of *Lecture Notes in Computer Science*, pages 415–431. Springer, 2020. [2](#)
- [27] Anindya Mondal, Shashant R, Jhony H. Giraldo, Thierry Bouwmans, and Ananda S. Chowdhury. Moving object detection for event-based vision using graph spectral clustering. In *IEEE/CVF International Conference on Computer Vision Workshops, ICCVW 2021, Montreal, BC, Canada, October 11-17, 2021*, pages 876–884. IEEE, 2021. [1](#)
- [28] Garrick Orchard, Ajinkya Jayawant, Gregory Cohen, and Nitish V. Thakor. Converting static image datasets to spiking neuromorphic datasets using saccades. *CoRR*, abs/1507.07629, 2015. [2](#), [4](#), [5](#), [6](#)
- [29] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035, 2019. [4](#)
- [30] Etienne Perot, Pierre de Tournemire, Davide Nitti, Jonathan Masci, and Amos Sironi. Learning to detect objects with a 1 megapixel event camera. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. [1](#)
- [31] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI*, 2019. [2](#)
- [32] Jason Tyler Rolfe. Discrete variational autoencoders. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. [2](#)
- [33] Cedric Scheerlinck, Nick Barnes, and Robert E. Mahony. Continuous-time intensity estimation using event cameras. In C. V. Jawahar, Hongdong Li, Greg Mori, and Konrad Schindler, editors, *Computer Vision - ACCV 2018 - 14th Asian Conference on Computer Vision, Perth, Australia, December 2-6, 2018, Revised Selected Papers, Part V*, volume 11365 of *Lecture Notes in Computer Science*, pages 308–324. Springer, 2018. [2](#)
- [34] Cedric Scheerlinck, Henri Rebecq, Daniel Gehrig, Nick Barnes, Robert E. Mahony, and Davide Scaramuzza. Fast image reconstruction with an event camera. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2020, Snowmass Village, CO, USA, March 1-5, 2020*, pages 156–163. IEEE, 2020. [2](#)
- [35] Amos Sironi, Manuele Brambilla, Nicolas Bourdis, Xavier Lagorce, and Ryad Benosman. HATS: histograms of averaged time surfaces for robust event-based object classification. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1731–1740. Computer Vision Foundation / IEEE Computer Society, 2018. [2](#), [4](#), [5](#), [6](#)
- [36] Zhaoning Sun, Nico Messikommer, Daniel Gehrig, and Davide Scaramuzza. ESS: learning event-based semantic segmentation from still images. In Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXXIV*, volume 13694 of *Lecture Notes in Computer Science*, pages 341–357. Springer, 2022. [1](#), [6](#), [8](#)
- [37] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018. [3](#)
- [38] Zhexiong Wan, Yuchao Dai, and Yuxin Mao. Learning dense and continuous optical flow from an event camera. *IEEE Trans. Image Process.*, 31:7237–7251, 2022. [7](#)
- [39] Lin Wang, S. Mohammad Mostafavi I., Yo-Sung Ho, and Kuk-Jin Yoon. Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long*

- Beach, CA, USA, June 16-20, 2019*, pages 10081–10090. Computer Vision Foundation / IEEE, 2019. [2](#)
- [40] David Weikersdorfer, David B. Adrian, Daniel Cremers, and Jörg Conradt. Event-based 3d SLAM with a depth-augmented dynamic vision sensor. In *2014 IEEE International Conference on Robotics and Automation, ICRA 2014, Hong Kong, China, May 31 - June 7, 2014*, pages 359–364. IEEE, 2014. [1](#)
- [41] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3733–3742. Computer Vision Foundation / IEEE Computer Society, 2018. [2](#)
- [42] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan L. Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. [2](#), [5](#), [6](#), [7](#), [8](#)
- [43] Alex Zihao Zhu, Dinesh Thakur, Tolga Özaslan, Bernd Pfrommer, Vijay Kumar, and Kostas Daniilidis. The multi vehicle stereo event camera dataset: An event camera dataset for 3d perception. *CoRR*, abs/1801.10202, 2018. [1](#), [2](#), [5](#), [6](#), [7](#)