Bridging the Gap between Events and Frames through Unsupervised Domain Adaptation

Nico Messikommer, Daniel Gehrig, Mathias Gehrig and Davide Scaramuzza
Dept. Informatics, Univ. of Zurich and
Dept. of Neuroinformatics, Univ. of Zurich and ETH Zurich

Abstract

Event cameras are novel sensors with outstanding properties such as high temporal resolution and high dynamic range. Despite these characteristics, event-based vision has been held back by the shortage of labeled datasets due to the novelty of event cameras. To overcome this drawback, we propose a task transfer method that allows models to be trained directly with labeled images and unlabeled event data. Compared to previous approaches, (i) our method transfers from single images to events instead of high frame rate videos, and (ii) does not rely on paired sensor data. To achieve this, we leverage the generative event model to split event features into content and motion features. This feature split enables to efficiently match the latent space for events and images, which is crucial for a successful task transfer. Thus, our approach unlocks the vast amount of existing image datasets for the training of event-based neural networks. Our task transfer method consistently outperforms methods applicable in the Unsupervised Domain Adaptation setting for object detection by 0.26 mAP (increase by 93%) and classification by 2.7% accuracy.

Multimedia Material

Additional qualitative results can be viewed in this video: https://youtu.be/fZnBSqni6PY

1. Introduction

Humans can effortlessly identify objects depicted in paintings even if there is a large visual gap between drawn and real objects. This is because humans possess incredible *transductive* abilities which allow them to separate medium-specific (painting/real world) from content-specific attributes [3]. This transductive transfer ability comes in quite handy as it enables us to reason about objects in different drawing styles without much additional effort.

Achieving such abilities in machine vision has the potential to enable effortless multimodal sensing. Moreover, it promises to solve a persistent issue for novel sensor modalities: the lack of labeled datasets. Contemporary perception systems heavily rely on high-quality, large-scale datasets that are tedious to create. Many such image or video datasets have been produced with frame-based cameras due to their abundance in everyday life and availability. However, novel sensors such as event cameras cannot directly benefit from these efforts and available event datasets remain scarce (Fig. 2).

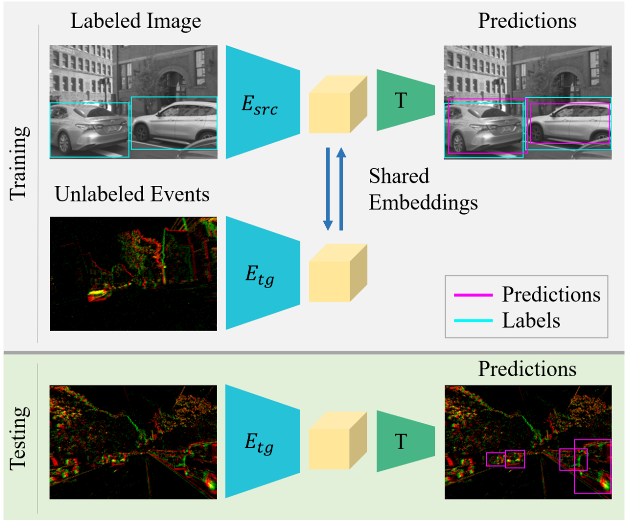

Figure 1. Our approach can teach a network to detect cars in event frames even though it was never told how cars look in the event domain. This unsupervised domain adaption is possible by leveraging labeled grayscale images and unlabeled events. During testing, our approach consists of a standard network and thus has no computational overhead of first translating events to images.

Event cameras possess outstanding properties such as high dynamic range (up to 140 dB), very high temporal, resolution and low latency. Instead of capturing images at a fixed rate, event cameras measure changes of intensity asynchronously per pixel. This results in a stream of events that encodes the time, location, and polarity of the intensity change. For a more in-depth survey, we refer to [9].

While the working principle of event cameras is radically different from conventional frame-based cameras, the output of event and frame-based cameras still contains a significant information overlap, as both cameras share the underlying principle of capturing the scene irradiance through an optical system [33]. In this work, we show how this information overlap can be leveraged for *Unsupervised Domain Adaptation (UDA)* of event-based networks, in which labeled source and unlabeled target data are available to transfer a task to the target domain.

The unpaired UDA setting allows our method to directly leverage large-scale datasets containing still images. This is in contrast to recent work that either relies on paired sensor data [20, 51], which requires side-by-side recording or video datasets [13, 51]. Furthermore, by tackling the transfer in a data-driven way, we do not rely on hand-designed generative models for video-to-events generation [4, 13, 31]. Finally, our approach has no overhead during inference by converting events first to intensity images [33].

To bridge the gap between frames and events without paired data, we introduce a novel *single-image-to-event* translation technique based on the event generation model combined with standard image-to-image translation techniques [23, 49, 53]. Crucially, instead of learning image-to-event translation directly, it only learns to correct an initial guess from the generative model. However, the event construction from a single image is ill-posed due to the missing motion information. We solve this problem by explicitly extracting motion features from events in addition to the shared features that contain domain invariant information about the scene. To achieve this split, we introduce a shared embedding discriminator and enforce shared feature consistency using sensor-specific knowledge.

Our experiments validate that our approach can successfully transfer from frames to events by leveraging large-scale datasets for object detection and classification. We show this by achieving state-of-the-art for both object detection and classification in the UDA setting on N-Caltech101 [30] and MVSEC [50].

Our contributions can be summarized as follows.

1. We propose a transfer learning method that uses labeled frame-based datasets together with unlabeled events recorded in a target environment to train event-based networks. As one application, we show that networks trained on labeled daylight images can be transferred to challenging nighttime scenarios, where event cameras outperform standard cameras thanks to their higher dynamic range.

2. Our approach leverages prior sensor knowledge based on the generative event model. Additionally, we introduce a mapping of events into motion and content embeddings. This enables us to transfer from single images to events, which opens up all of the existing frame-based datasets for event cameras.

3. Our task transfer method can train neural networks that outperform state-of-the-art object detection methods applicable in the UDA setting by 0.27 mAP (an increase of 93%) and achieve a 2.7% accuracy increase in the classification task outperforming even some supervised approaches.

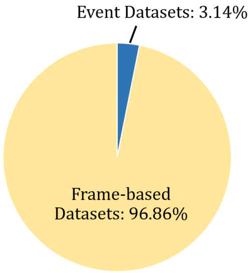

Figure 2. Event-based datasets represent only a small fraction of the existing frame-based datasets (source: [8, 20]).

2. Related Work

Unsupervised Domain Adaptation The general problem of transfer learning based on labeled source data and unlabeled target data has been an active research field for many years, which has accumulated vast literature. For a survey, we refer the reader to [46]. Early deep-learning-based methods use discriminators [37] or gradient reversal layers [11, 12] to align the embedding space of source and target domain samples. The resulting shared embedding spaces were applied for *task transfer*, where dedicated task networks were trained on features from the shared embedding space together with labels from the source domain [5, 15], thereby enabling direct transfer to the target domain.

Analogously, *image-to-image translation* methods can be used to induce shared embedding spaces by mapping samples from target to source domain, or vice versa. While the former enables the reuse of pre-trained networks trained in the source domain [49], the latter can provide a source of labeled datasets in the target domain by converting datasets from the source domain [24, 38].

While these methods show promising results, they are not designed for task transfer and thus do not find optimal mappings for source and target data for a given task. For this reason, recent work [42, 19] proposed to jointly learn the task and mapping from input to domain-shared and domain-invariant features. However, they consider a smaller domain gap since they work only with images, which means that they can mostly work with shared network layers between source and target domain.

One work that addresses a large domain gap in the context of biomedical imaging is [47], where domain-invariant features are constructed from source and target characteristics. They then train a task network on the resulting features. Similarly, our task-transfer approach addresses the large domain gap between two complementary but different vision sensors: standard cameras, which output dense intensity frames at a fixed rate, and events cameras, which only measure changes in intensity in the form of sparse and asynchronous *events*. To do this, we leverage cycle consistency constraints, which were introduced in image-to-image translation literature [49, 53] and extended for transfer learning for specific tasks [18, 28, 49, 2], but only use a single source to target translation for aligning the embedding space for the downstream task. This reduces the computational complexity and training time significantly.

Event-based Approaches The lack of labeled datasets is one of the main limiting factors hindering the progress of event cameras. To address this challenge, a recent class of methods seeks to convert events to high dynamic range (HDR) image reconstruction through supervised [33] or adversarial training [26, 27, 45]. With these images, standard pre-trained neural networks trained on images can be used. However, despite this advantage, these methods impose a computational overhead by first generating image reconstructions. Instead, [20] simply adapt the first few layers of pre-trained frame-based networks to event data by enforcing feature consistency between the two separate sensor encoders. While this eliminates the need for costly event preprocessing, it requires paired images and events, *i.e.* events and images recorded on the same sensor and scene to adapt a given network. By contrast, the method in [48] is designed to work with unpaired data but only converts between events in different illumination conditions. The first works to leverage existing frame-based datasets were *video-to-event translation* methods. These methods either rely on model-based [31, 33, 13, 4] or data-driven [51] approaches to convert video sequences into artificial events, which can be used to directly train neural networks on event data. This opened up the possibility of training networks for event data on larger and more diverse datasets. However, these methods are still limited to translating video to events, thus ignoring the majority of existing datasets that comprise images. The work most similar to ours leverages affinity graphs to perform the task transfer from frames to events [44]. In comparison, our approach splits the embedding space into shared and sensor-specific features and leverages the event generation model to align both domains.

In this work, we introduce a novel method that addresses the limitations of previous approaches by performing unsupervised domain adaptation, which *(i)* maps unpaired images and events to a shared embedding space, *(ii)* leverages single-image-to-event instead of video-to-event translation, and *(iii)* performs task-transfer by jointly training a task-specific network on the shared embedding. We introduce a novel single-image-to-event translation module that combines the event generation model [10] with standard translation methods. Moreover, our method maps event data into separate content and sensor-specific features and only matches content features across modalities. In doing so, we take inspiration from style-transfer techniques [21, 23, 49].

3. Method

Our goal is to train a network on labeled images for a specific task and transfer the network to events, such that the network successfully performs the task in the event-domain without requiring any labeled events nor paired images and event data, see Fig 1. This setting of transferring a task from a labeled source domain (image domain Y_{img}) to an unlabeled target domain (event domain Y_{event}) is generally defined as Unsupervised Domain Adaptation, short UDA. The task transfer is possible since event and frame-based cameras share the underlying principle of capturing the scene irradiance through an optical system. Therefore, an information overlap exists on which the task can be learned on images and transferred to events.

In Sec. 3.1, we present the general network architecture and the latent space split into shared and sensor-specific features. The alignment of the shared latent space is enforced through multiple losses, which are explained in Sec. 3.2. As a common constraint in the UDA literature [2, 18, 28, 42, 47, 49], we perform domain translation to generate pseudo pairs, which are used to align the shared latent space on which the task is learned. However, compared to those approaches, we only use a one-sided translation from images to events to achieve a better embedding alignment. Sec. 3.4 introduces our novel event construction from a single image based on the event generation model [10], which is explained in Sec. 3.3. Since the event generation model constitutes a relation between events to the image gradient and optical flow, we can strongly constraint the image-to-event translation and thus significantly improve the task transfer. It is important to state that the image-to-event translation is only applied as an auxiliary task to help to transfer the task from images to events. As we use directly optimize the task transfer from images to events, our method consistently outperforms pure translation methods [13, 32, 33, 51].

3.1. General Model Architecture

In our framework, events $\mathbf{y}_{\text{event}}$ and images $\mathbf{y}_{\text{img}}$ are processed with separate encoders E_{img} and E_{event} due to the large domain gap between Y_{img} and Y_{event}. As event cameras also capture motion information because of their asynchronous output signal, they measure specific features ζ_{event} about the scene, which standard cameras can not perceive in a single frame. This non-overlapping information, however, hinders the image-to-event task transfer as it is im-

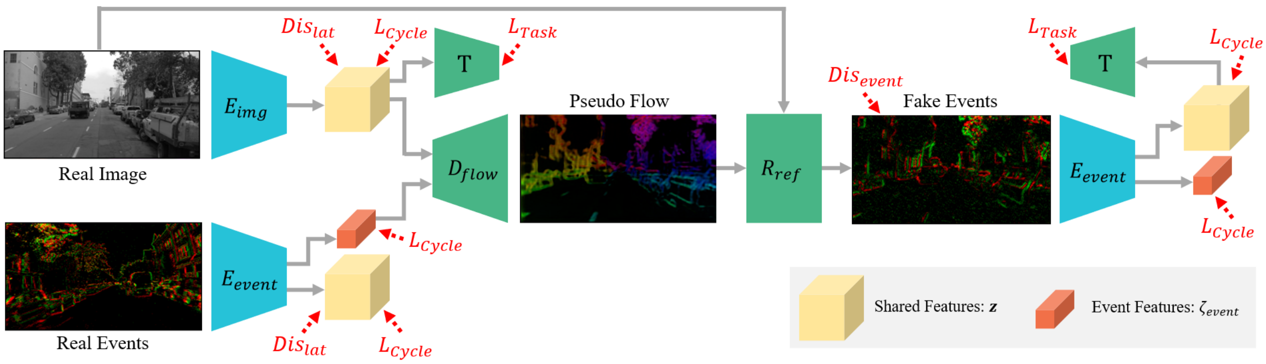

Figure 3. During training, our method uses single-image-to-event translation to transfer a task from the image to the event domain. As there is a large domain gap between events and grayscale images, we use two separate encoders E_{img} and E_{event} to process unpaired images and event frames. Shared features $\mathbf{z}$ and event-specific features ζ_{event} are extracted from the event frame. Both features are given as input to the event decoder D_{flow}, which creates a pseudo flow map. This flow map is combined with the image to create clean events using the event generation module. To model sensor noise, the refinement module takes additionally random input feature maps. The applied constraints are visualized with red arrows.

possible to fully align the embedding space. We solve this by separating event features into *sensor specific* features ζ_{event}, which contain motion information, and *content* features $\mathbf{z}_{\text{event}}$, which carry information shared in both domains Y_{img} and Y_{event}.

$$\begin{aligned} \mathbf{z}_{\text{img}} &= E_{\text{img}}(\mathbf{y}_{\text{img}}) \\ \mathbf{z}_{\text{event}} = E_{\text{event}}(\mathbf{y}_{\text{event}}) &\qquad \zeta_{\text{event}} = E_{\text{event, attr}}(\mathbf{y}_{\text{event}}). \end{aligned} \tag{1}$$

The resulting shared features $\mathbf{z}_{\text{img}}$ and $\mathbf{z}_{\text{event}}$ are given as input to the task branch T, which computes the task-specific output. To generate pseudo event and image pairs, shared features from an image $\mathbf{z}_{\text{img}}$ are combined with event-specific features ζ_{event} from a random event sample to compute a pseudo-flow field using a flow decoder D_{flow}. The resulting pseudo-flow and the input image are then converted to events $\hat{\mathbf{y}}_{\text{event}}$ in the refinement network R_{ref},

$$\hat{\mathbf{y}}_{\text{event}} = R_{\text{ref}}(D_{\text{flow}}(\mathbf{z}_{\text{img}}, \zeta_{\text{event}})). \tag{2}$$

The single-image-to-event translation is explained in more detail in Sec. 3.4. The overall architecture is depicted in Fig. 3.

3.2. Shared Latent Space Constraints

The unsupervised task transfer from images to events requires multiple constraints as there is neither task supervision in the event domain nor paired sensor data. Therefore, multiple losses are applied to enforce a shared latent space of $\mathbf{z}_{\text{img}}$ and $\mathbf{z}_{\text{event}}$, which ensures that the task branch T successfully performs the task in both domains. As a first constraint, we apply adversarial training [16] with a PatchGAN discriminator network Dis_{lat} [22] to the latent features $\mathbf{z}_{\text{img}}$ and $\mathbf{z}_{\text{event}}$

$$\begin{aligned} \mathcal{L}_{\text{lat.,disc.}} =& \mathbb{E}_{\mathbf{y}_{\text{img}}}[\max(0, 1 - Dis_{\text{lat}}(\mathbf{z}_{\text{img}}))] \\ &+ \mathbb{E}_{\mathbf{y}_{\text{event}}}[\max(0, 1 + Dis_{\text{lat}}(\mathbf{z}_{\text{event}}))] \\ \mathcal{L}_{\text{lat.,gen.}} =& \mathbb{E}_{\mathbf{z}_{\text{img}}}[Dis_{\text{lat}}(\mathbf{z}_{\text{img}})] - \mathbb{E}_{\mathbf{y}_{\text{event}}}[Dis_{\text{lat}}(\mathbf{z}_{\text{event}})]. \end{aligned} \tag{3}$$

Similar to [51], we use a hinge-loss [43] and optimize the above loss functions in an alternating fashion. The above objective forces the latent space to be indistinguishable to the discriminator Dis_{lat}, and thus, the latent space becomes aligned.

As an additional constraint, we generate pseudo sensor pairs using a one-sided translation from single images to events. These pseudo-pairs are used to formulate consistency losses on the latent variables $\mathbf{z}_{\text{img}}$ and ζ_{event}, as summarized below:

$$\begin{aligned} \mathcal{L}_{\text{cycle}} =& |\mathbf{z}_{\text{img}} - E_{\text{event}}(R_{\text{ref}}(D_{\text{flow}}(\mathbf{z}_{\text{img}}, \zeta_{\text{event}})))|_1 \\ &+ |\zeta_{\text{event}} - E_{\text{event, attr}}(R_{\text{ref}}(D_{\text{flow}}(\mathbf{z}_{\text{img}}, \zeta_{\text{event}})))|_1 \end{aligned} \tag{4}$$

To generate realistic events from a single image, the following adversarial loss is applied on the reconstructed events $\hat{\mathbf{y}}_{\text{event}}$ using an event discriminator Dis_{event}

$$\begin{aligned} \mathcal{L}_{\text{recons.,disc.}} =& \mathbb{E}_{\hat{\mathbf{y}}_{\text{event}}}[\max(0, 1 - Dis_{\text{event}}(\hat{\mathbf{y}}_{\text{event}}))] \\ & \mathbb{E}_{\mathbf{y}_{\text{event}}}[\max(0, 1 + Dis_{\text{event}}(\mathbf{y}_{\text{event}}))] \\ \mathcal{L}_{\text{recons.,gen.}} =& \mathbb{E}_{\hat{\mathbf{y}}_{\text{event}}}[Dis_{\text{event}}(\hat{\mathbf{y}}_{\text{event}})] \end{aligned} \tag{5}$$

The used constraints are visualized with red arrows in Fig. 3. For more specific details about the loss functions and training procedure, we refer the reader to the supplementary A1. Overall, these general UDA methods represent a solid basis for closing the domain gap between events and images. However, as shown in our experiments in Sec. 4, current UDA methods fall short in transferring task knowledge between the large gap of events and images. The next section shows how the event generation model can be leveraged to improve the task transfer between the sensor domains.

3.3. Event Generation Model

The underlying principle of an event and frame camera can be exploited to guide the single-image-to-event translation. As discussed in Sec 3, event and frame cameras are both optical sensors, which capture the scene irradiance through lenses. Due to this shared principle of measuring the light intensity, images can approximately be translated to events through the theoretical concept of the event generation model [10]. This generative model describes the behavior of an ideal event camera under the assumption of constant brightness and of small time differences Δt. In Eq. 6, $\tilde{I}_{(x,y)} = \log(I_{(x,y)})$ expresses the measured intensity in log space and α represents the angle between the optical flow vector $\mathbf{v}_{(x,y)}$ and image gradient $\nabla \tilde{I}_{(x,y)}$.

$$\begin{aligned} \Delta \tilde{I}_{(x,y)} &\approx -\langle \nabla \tilde{I}_{(x,y)}, \mathbf{v}_{(x,y)} \Delta t \rangle \\ &= -|\nabla \tilde{I}_{(x,y)}||\mathbf{v}_{(x,y)}|\Delta t \cos \alpha \end{aligned} \tag{6}$$

An event is triggered if the log intensity change $\Delta \tilde{I}$ is above a predefined contrast threshold C. Thus, the number N of events at a pixel (x, y) can be approximated according to Eq. 7.

$$N_{(x,y)} \approx \lfloor \Delta \tilde{I}_{(x,y)} / C \rfloor \tag{7}$$

The event generation model enables the transfer from a single image to events, if motion information in form of optical flow $\mathbf{v}_{(x,y)}$ and time difference Δt is provided. By considering only single frames, which most frame-based datasets consist of, the event generation is an ill-posed problem due to the missing motion information. To account for that, we split the events $\mathbf{y}_{\text{event}}$ into two domains: latent space $\mathbf{z}_{\text{event}}$ shared with image features and an additional sensor-specific space ζ_{event}, in which the motion information in the events $\mathbf{y}_{\text{event}}$ is stored. Thus, we can reconstruct artificial events from frames by combining content and sensor-specific features.

3.4. Event Generation based on Pseudo-Flow

Following Eq. 7, we observe that the predicted events relate to the image gradient $\nabla \tilde{I}_{(x,y)}$ via optical flow. Instead of optical flow, we propose to directly predict pseudo-flow vectors $\hat{\mathbf{v}}_{(x,y)}$, which implicitly contain the unknown parameters Δt, $\cos \alpha$ and C. Thus, we do not need to compute these parameters explicitly.

$$\begin{aligned} \hat{\mathbf{v}}_{(x,y)} &= \mathbf{v}_{(x,y)} \frac{1}{C} \Delta t \cos \alpha \\ \hat{N}_{(x,y)} &\approx \langle \Delta \tilde{I}_{(x,y)}, \hat{\mathbf{v}}_{(x,y)} \rangle \end{aligned} \tag{8}$$

It can be observed in Eq. 8 that the number of predicted events $\hat{N}_{(x,y)}$ can either be changed by the pseudo-flow magnitude or by the angle between the two vectors $\Delta \tilde{I}_{(x,y)}$ and $\hat{\mathbf{v}}_{(x,y)}$. Thus, our pseudo-flow is not equivalent to optical flow as the adversarial training only enforces realistic events by either adjusting the direction or the magnitude of $\hat{\mathbf{v}}_{(x,y)}$.

The pseudo-flow is constructed from a combination of *sensor-specific* and *shared* features. The resulting pseudo-flow field adheres to the content extracted from an image $\mathbf{z}_{\text{img}}$ but with the general motion information of the event data, encoded in the *sensor-specific* feature ζ_{event}. The event generation based on pseudo-flow constrains the image-to-event translation and supports the adversarial training since it provides good event predictions early during training.

We propose a novel refinement block R_{ref}, which computes the inner product of the predicted pseudo-flow and image gradients as in Eq. (8) to obtain an initial guess of the translated events. In the next step, the refinement net uses three convolutional layers to predict residual event representations, which are added to the initial guess. These residual events correct for overlapping polarity regions and event noise in the initial reconstruction.

As the target domain is mostly known, we can augment the event generation by adding an artificial flow field according to the motion present in the target event data. This flow augmentation is crucial to enforce the split into sensor-shared and sensor-specific features. It is essential to include augmentation consistent with the target event domain. Otherwise, the discriminator easily distinguishes between the translated and real events, thus degenerating the adversarial training and the task transfer. The augmented pseudo-flow field consists of vectors with a magnitude of the pseudo-flow predictions and the directions of the target-specific motion distributions. Events are then generated in the refinement module R_{ref} from this augmented flow. These events are then processed again by the event encoder E_{event} to obtain event-specific features $\mathbf{z}_{\text{event}}$, which are combined with the original shared feature to construct new events. This event representation should be identical to the events obtained by the augmented flow as only the event-specific features $\mathbf{z}_{\text{event}}$ contain motion information. Thus, an L1 loss is applied between those two event representation, as visualized in Fig. 4.

4. Experiments

We validate our transfer learning approach on two different tasks: image classification (Section 4.1) and object detection (Section 4.2). Furthermore, we justify our design choices with ablation experiments included in the supplementary A2.

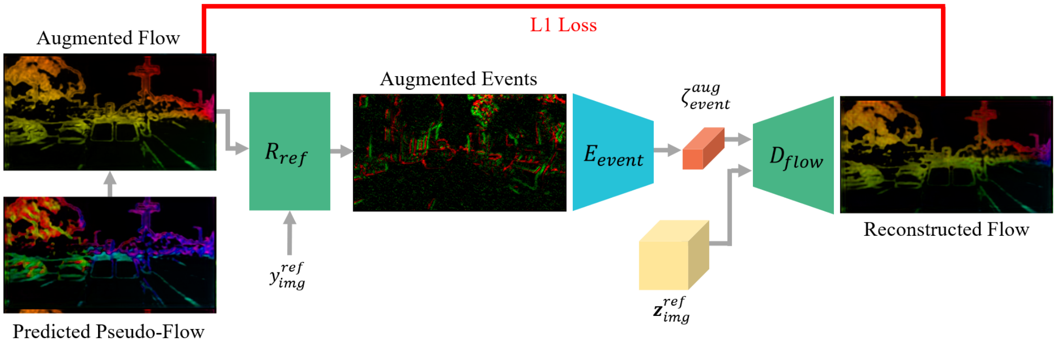

Figure 4. To enforce the split into sensor shared features and event-specific features, i.e., motion features, we propose to augment the pseudo-flow predictions. Specifically, we take a pseudo-flow prediction and augment the flow with the target domain-specific motions. In the shown car driving case, we sample an augmentation based on random epipoles in the image. Event-specific features $\zeta_{\text{event}}^{\text{aug}}$ are extracted from the events constructed based on the augmented flow. These event specific features $\zeta_{\text{event}}^{\text{aug}}$ are then combined with the shared features $z_{\text{img}}^{\text{ref}}$ from the reference frame $y_{\text{img}}^{\text{ref}}$ to reconstruct the pseudo-flow. An L1 loss is then applied between the augmented and the reconstructed loss. By doing so, the networks can only adapt the motion features $\zeta_{\text{event}}^{\text{aug}}$ as the content features are fixed.

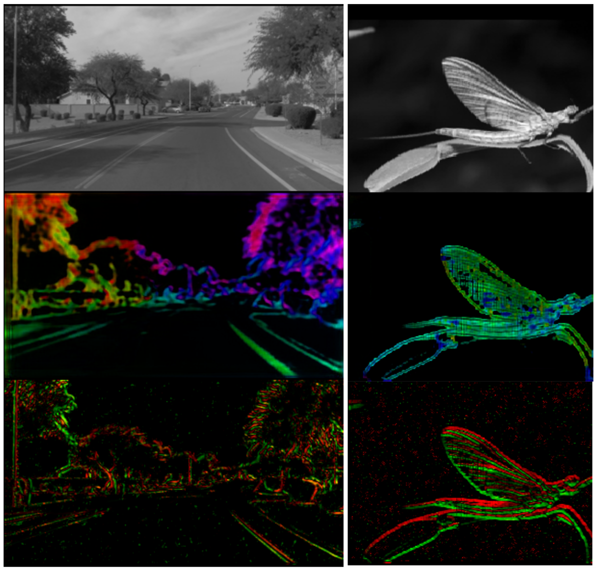

Figure 5. Example of events generated from a single image. The top row shows the grayscale images, from which the content features are extracted and combined with event-specific motion features to construct a pseudo flow map (middle row). These pseudo flow maps are then combined with the image gradients in the refinement module R_{ref} to construct the fake events. The refinement module R_{ref} extends the events with realistic noise based on a random feature input.

4.1. Classification

Experimental Setup We validate our approach for event classification on the Neuromorphic-Caltech101 (N-Caltech101) [30] dataset, which is a common benchmark for event-based classification. N-Caltech101 was recorded by tilting an event camera in front of a screen, on which image samples from Caltech101 [7] are projected. It is, therefore, a straightforward choice to use Caltech101 as a labeled source dataset. It is important to point out that we do not use paired sensor data even though it is available for N-Caltech101 and Caltech101. As our approach can leverage unpaired single images for event-based training, we extend the frame-based Caltech101 with a set of additional images showing the 101 classes. These additional images were also used in the baseline method VID2E [13] for training a classification network on events, which are generated based on simulated motion.

Our task network for classification follows the architecture of Resnet18 [17]. In particular, we use the first layers up to the third residual block of Resnet18 without the first max-pooling layer for both sensor encoders. The second of these residual blocks is shared between the event and image encoder. The task network consists of the remaining Resnet18 layers. All the Resnet18 layers were initialized with weights pre-trained on ImageNet [35]. The architectures proposed in Drit++ [23] were adopted for the sensor-specific encoder, decoder, shared latent discriminator, and event frame discriminator. The event histogram [25] is used as event representation to facilitate the grayscale to event frame translation. We augment the pseudo-flow with random translation fields (Section 3.3) as N-Caltech101 only contains translational motions.

To find the best model during training, we deviate from the pure UDA settings as we use a small set of labeled target data, which acts as a validation set. The labeled target data is solely used for validation and testing purposes. This is commonly done in the UDA literature as the validation without labels is itself an open research question. Similar to supervised learning, we split the target data into training, validation, and test data. Thus, no testing sample was seen during training/validation, neither in the image nor event domain.

Results The current state-of-the-art methods performing transfer learning with unpaired grayscale and event data are VID2E [13] and E2VID [33]. In addition to those two methods applicable to the UDA setting, we also include supervised methods, which have access to the event labels during training. The classification accuracies are reported in Tab. 1.

Our approach outperforms the state-the-art method E2VID by 2.7% in terms of accuracy. Moreover, our inference network is a simple Resnet18, which is computationally much more lightweight than E2VID. Compared to the second baseline VID2E, our approach achieves a 4.1% higher accuracy. There are two possible reasons for the increased performance. First, our method focuses specifically on task transfer and thus exploits the image and event domain to learn task-relevant features. This multimodal learning helps to extract more informative features, which was confirmed in recent work [36] as well. Second, VID2E generates events based solely on model assumptions, whereas our approach uses the generative event model combined with a data-driven network to approximate target events. Thus, our network can adjust better to the specific target event data, which is influenced by the event camera model and parameter settings. As can be observed in Fig. 5, our approach generates realistic-looking event frames based on single grayscale images.

Our approach even outperforms the supervised methods HATS [39] and EST [14], as can be seen in Table 1. One advantage of our method compared to HATS and EST is the increased size of the training dataset. As our approach can use single images without corresponding events, we can easily extend the training dataset with additional class samples, as it was done for VID2E. This also explains the higher classification accuracy of our transfer learning approach compared to the supervised setting with the same architecture. Since EvDistill [44] was trained on a different training split for Caltech101 in the image domain, it is hard to compare against this approach. Nevertheless, we report the performance of our approach trained on the complete Caltech101 dataset and the performance reported in [44].

We additionally report the performance of a simple cycle translation UDA framework without the embedding split into shared and event-specific features. The significantly lower performance shows that the feature space split is crucial for the task transfer between images and events. In the case of classification on N-Caltech101, the flow prediction does not provide any improvement compared to a model, which directly predicts an event representation. This can be explained due to the simple planar motion distribution present in the event samples of N-Caltech101. For a more complex motion distribution, i.e., in driving car sequences, the flow prediction almost doubles the object detection performance, reported in Tab. 2. In the same Tab. 2, we show that the flow augmentation, explained in Section 3.3, improves the detection accuracy significantly. This validates that the flow augmentation helps with the split into shared and event-specific features, increasing the task performance.

Method	UDA	Accuracy ↑
E2VID [33]	✔	0.821
VID2E [13]	✔	0.807
Simple Cycle	✔	0.577
Ours w/o Flow	✔	**0.848**
Ours	✔	**0.848**
E2VID [33]	✗	0.866
VID2E [13]	✗	0.906
EST [14]	✗	0.817
HATS [39]	✗	0.642
Ours supervised	✗	0.839
EvDistill* [44]	✔	0.902
Ours*	✔	0.938

Table 1. Classification accuracies on the N-Caltech101 dataset. The upper part of the table shows the methods applicable in the UDA setting, i.e., they do not have access to the event labels during training. We have also listed methods that use the ground truth labels during training and are thus not applicable for UDA. To stay consistent with the evaluation of EvDistill, we report the performance achieved by our model trained on the whole Caltech101 dataset(*).

Method	Unpaired	mAP ↑
ESIM [31]	✔	0.02
E2VID [33]	✔	0.28
Ours w/o flow	✔	0.26
Ours w/o augm	✔	0.48
Ours	✔	**0.54**
EventGAN [51]	✗	0.30
YOLOv3-GN* [20]	✗	0.70

Table 2. Mean average precision for the task of object detection on the MVSEC dataset. *Different test labels and trained on the same sequence

4.2. Object Detection

Experimental Setup In the case of object detection, we evaluate on the Multi-Vehicle Stereo Event Camera Dataset (MVSEC) [50]. The authors of EventGAN [51] provided us with car bounding box labels for the outdoor_day_2 sequence, on which they evaluated EventGAN. For training, we use the two outdoor sequences from MVSEC and add the DDD17 [1] dataset, which contains unlabeled event sequences captured with the same event camera. As a labeled source dataset in the image domain, we use Waymo Open Dataset [41].

Except for the task branch, we use the same network layers as for the classification task. Similar to EventGAN and the network grafting approach YOLOv3-GN [20], the task branch for object detection consists of YOLOv3 [34] layers.

Results We compare against the event simulator ESIM [31] as well as E2VID on the task of object detection. Additionally, YOLOv3-GN and EventGAN are included as paired baselines, i.e., they were trained with event sequences and the corresponding frames. For the performance of YOLOv3-GN, we report the value published in their paper, which was computed on the same outdoor_day_2 sequence, but with different bounding boxes generated by a frame-based object detector applied to the grayscale images. The object detection performances on the outdoor_day_2 sequence of MVSEC are reported in Table 2 as mean Average Precision (mAP) [6]. When comparing approaches that train on unpaired data, our approach achieves the highest performance, outperforming the next best method, [32] by 26% in terms of mAP. We credit this result to the fact that, while E2VID is only concerned with event-to-image translation, our method explicitly optimizes for the task objective, thus generating more task optimized representations, which lead to better task performance. The low performance of the object detector trained using ESIM can be explained by the large domain gap between the generated and real events. As reported in [51], the synthetic data from ESIM was generated with a uniform planar motion, which differs greatly from driving sequence motions.

We also compare our method against paired approaches [51, 20] even though the training setting is different. Both methods use the grayscale images and the corresponding events of the outdoor_day_2 sequence during training. Nevertheless, our method outperforms EventGAN by a significant margin (0.24 mAP). Compared to EventGAN, which generates labeled events from two frames, we focus on the specific task transfer from images to events. By splitting the embedding space into shared and event-specific motion features, we can leverage the labeled images to extract task-specific knowledge in the shared space. Therefore, the task network can focus on task-specific features. Moreover, the significant improvement can be attributed to the combination of our novel flow module (an improvement from 0.26 to 0.48) and our flow augmentation approach (an improvement from 0.48 ”Ours w/o augm” to 0.54 ”Ours”). Without these modules, our method achieves similar performance to EventGAN. As our method combines prior sensor knowledge with adversarial training, we can generate a more realistic event distribution and thus have an advantage at transferring the object detection task from images to events. In the case of YOLOv3GN, a pre-trained Yolov3 network was adapted directly to frames from the outdoor_day_2 with the corresponding events to align the embedding space with paired data. The same sequence is then used as a test set. Thus, a fair comparison is difficult since the reported test score is likely to overestimate the true test score. By contrast, while our method has a 16% lower performance, it is important to note that our method does not have access to labels in the target domain and thus solely relies on unpaired images and labels from a vastly different domain. The method in [20] on the other hand, is limited in its task transfer capability since it assumes paired (*i.e.* per pixel aligned and synchronized data) to work.

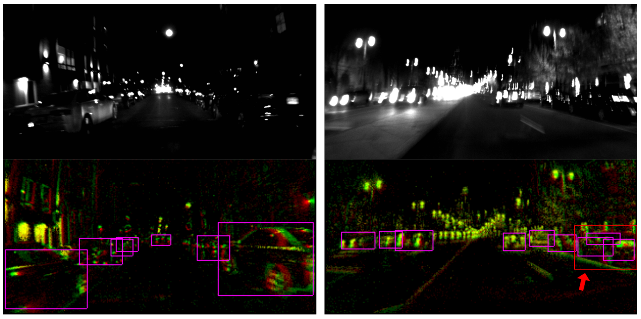

Figure 6. Our task transfer framework enables to transfer from daylight images to events recorded during the night. The images show two scenes from the MVSEC night sequence recorded on a motorbike. The top row shows the VI-sensor images, which are underexposed (left) and suffer from motion blur (right). In contrast, the event histograms (bottom row) include much more details than the images. The prediction of our transferred object detection network is depicted in magenta. The red arrow on the bottom right event histogram indicates the only three false predictions, which lay on top of just one car.

Daylight Images to Night-Time Events With our transfer framework, we can fully exploit the benefits of event cameras as we are no longer dependent on high-quality images in the same scenario for labeling. To demonstrate this ability, we use the relatively high-quality images from the Waymo Open Dataset to transfer the task of car detection to events recorded during the night, where standard cameras are underexposed. We visualize our detector in this scenario on MVSEC [52] (Fig. 6). Standard frames recorded with a VI-Sensor [29] (top row) are underexposed and blurry, while event data has a higher dynamic range and does not suffer from motion blur. Our method detects all cars present in the event stream (orange boxes), only making a single mistake by misidentifying a cluster of cars (red arrow). Crucially, this method was entirely trained with labeled images in daylight scenarios, without ever seeing a label in the dark. This example highlights the enormous potential of the method for transferring task knowledge to challenging night-time scenarios. The robustness of our approach can also be verified in the supplementary video.

5. Conclusion

Learning for event-based vision has been held back by the scarcity of training data. In contrast, image or video-based methods have tremendously improved in performance due to the availability of large-scale datasets. This work proposes a framework to address this problem by leveraging large-scale image datasets with unsupervised domain adaptation. To achieve this, our method transfers task-specific knowledge from frame-based datasets to the event-based domain without the need for paired sensor data. Therefore, this framework allows models to be trained directly with labeled images and *unlabeled* event data. This unlocks the potential to use any frame-based dataset to train an event-based network. By large-scale datasets like the extended N-Caltech101 dataset [13] or the Waymo Open Dataset [41], we outperform state-of-the-art method for classification by 2.7% and object detection by 26% mAP, in the UDA setting.

6. Acknowledgements

The authors thank Prof. Kostas Daniilidis and his group at the University of Pennsylvania for providing the bounding boxes for MVSEC, and Yuhuang Hu, Prof. Toby Delbruck and Prof. Shih-Chii Liu, University of Zurich and ETH Zurich, for sharing the code of the Network Grafting Algorithm. This work was supported by the National Centre of Competence in Research (NCCR) Robotics through the Swiss National Science Foundation (SNSF).

Supplementary

A1 Losses

In the following, we give a detailed explanation of the losses applied in our proposed framework.

A1.1 Adversarial Losses

Latent Space Loss

To ensure that the task branch T can seamlessly transfer between events and frames, we enforce its input, i.e., both latent representation $\mathbf{z}_{\text{img}}$ and $\mathbf{z}_{\text{event}}$, to lay on one manifold. This is done by applying adversarial training [16] with a discriminator network Dis_{lat}. The adversarial training aligns the distribution of $\mathbf{z}_{\text{img}} = E_{\text{img}}(\mathbf{y}_{\text{img}})$ and $\mathbf{z}_{\text{event}} = E_{\text{event}}(\mathbf{y}_{\text{event}})$. Similar to [51], a hinge adversarial loss [43] is adopted.

$$\begin{aligned}\mathcal{L}_{\text{lat.,disc.}} =& \mathbb{E}_{\mathbf{y}_{\text{img}}}[\max(0, 1 - Dis_{\text{lat}}(\mathbf{z}_{\text{img}}))] \\ &+ \mathbb{E}_{\mathbf{y}_{\text{event}}}[\max(0, 1 + Dis_{\text{lat}}(\mathbf{z}_{\text{event}}))] \\ \mathcal{L}_{\text{lat.,gen.}} =& \mathbb{E}_{\mathbf{z}_{\text{img}}}[Dis_{\text{lat}}(\mathbf{z}_{\text{img}})] - \mathbb{E}_{\mathbf{y}_{\text{event}}}[Dis_{\text{lat}}(\mathbf{z}_{\text{event}})].\end{aligned} \tag{9}$$

Image-to-event Translation Loss

As an additional constraint, we force the latent representations $\zeta_{\text{event}}, \mathbf{z}_{\text{img}}$ to carry sufficient information, such that they can be decoded into an artificial event-frame. To do this, we combine the event-specific features ζ_{event} from a random event representation $\mathbf{y}_{\text{event}}$ and the content features $\mathbf{z}_{\text{img}}$ from an image $\mathbf{y}_{\text{img}}$ to generate artificial events $\hat{\mathbf{y}}_{\text{event}} = R_{\text{ref}}(D_{\text{flow}}(\mathbf{z}_{\text{img}}, \zeta_{\text{event}}))$ using a pseudo-flow decoder D_{flow} and a refinement net R_{ref}. These events share the image content but contain the event-specific features, e.g., motion distribution, of the reference events $\mathbf{y}_{\text{event}}$. The following adversarial loss is applied on the image-to-event reconstruction $\hat{\mathbf{y}}_{\text{event}}$. Similar to the embedding space alignment, a PatchGAN [22] discriminator and the hinge loss [43] is adopted for the sensor translation.

$$\begin{aligned}\mathcal{L}_{\text{recons.,disc.}} =& \mathbb{E}_{\hat{\mathbf{y}}_{\text{event}}}[\max(0, 1 - Dis_{\text{event}}(\hat{\mathbf{y}}_{\text{event}}))] \\ & \mathbb{E}_{\mathbf{y}_{\text{event}}}[\max(0, 1 + Dis_{\text{event}}(\mathbf{y}_{\text{event}}))] \\ \mathcal{L}_{\text{recons.,gen.}} =& \mathbb{E}_{\hat{\mathbf{y}}_{\text{event}}}[Dis_{\text{event}}(\hat{\mathbf{y}}_{\text{event}})]\end{aligned} \tag{10}$$

A1.2 Translation Consistency

By translating from single images to events, we can formulate consistency losses on the latent variables $\mathbf{z}_{\text{img}}$ and ζ_{event} as summarized below:

$$\begin{aligned}\mathcal{L}_{\text{cycle}} =& |\mathbf{z}_{\text{img}} - E_{\text{event}}(R_{\text{ref}}(D_{\text{flow}}(\mathbf{z}_{\text{img}}, \zeta_{\text{event}})))|_1 \\ &+ |\zeta_{\text{event}} - E_{\text{event, attr}}(R_{\text{ref}}(D_{\text{flow}}(\mathbf{z}_{\text{img}}, \zeta_{\text{event}})))|_1\end{aligned} \tag{11}$$

A1.3 Flow Augmentation Loss

As shown in Figure 4 in the manuscript, we augment the flow by combining the augmented, event-specific features $\zeta^{\text{aug}}_{\text{event}}$ with the shared features $z^{\text{ref}}_{\text{img}}$ from the reference frame $y^{\text{ref}}_{\text{img}}$. The resulting event representation should be identical to the events generated by the augmented flow. By enforcing this identity, the network can only store the augmented flow information in the event-specific features $\zeta^{\text{aug}}_{\text{event}}$ since the shared features are constant.

$$\begin{aligned}\mathcal{L}_{\text{augm}} =& |E_{\text{event, attr}}(y^{\text{aug}}_{\text{event}}) \\ &- E_{\text{event, attr}}(R_{\text{ref}}(D_{\text{flow}}(z^{\text{ref}}_{\text{img}}, \zeta^{\text{aug}}_{\text{event}})))|_1\end{aligned} \tag{12}$$

A1.4 Flow Smoothness

As common in the literature, we add a smoothness loss $\mathcal{L}_{\text{smooth}}$ to further constrain the pseudo-flow prediction. The smoothness loss consists of the Charbonnier loss function [40] applied to the difference of a flow vector with its eight neighboring flow vectors (including diagonal neighbors):

$$\mathcal{L}_{\text{smooth}} = \sum_{\mathbf{x}} \sum_{\mathbf{y} \in \mathcal{N}(\mathbf{x})} \rho(\mathbf{v}_{\mathbf{x}} - \mathbf{v}_{\mathbf{y}}) \tag{13}$$

Where $\mathbf{x} = (x, y)$ are image pixels, $\mathcal{N}(\mathbf{x})$ denotes the 8 neighbors of $\mathbf{x}$ and $\rho(x) = (\epsilon^{\alpha} + x^{\alpha})^{\frac{1}{\alpha}}$. We use $\alpha = 0.45$ and $\epsilon = 0.001$.

A1.5 Image Gradient Loss

To prevent mode collapse of the image-to-event generation, a gradient loss $\mathcal{L}_{\text{grad}}$ is applied on the event reconstructions. This loss penalizes areas that have few events and high image gradient as events are generated mainly by image gradients. We enforce this constraint by introducing the following loss:

$$\mathcal{L}_{\text{grad}} = \sum_{(x,y)} \max(0, 0.7 - N_{(x,y)}) |\nabla I_{(x,y)}| \tag{14}$$

Where we sum over all pixels that have an image gradient magnitude $|\nabla I_{(x,y)}| > 0.7$. Here $N_{(x,y)} \in [0, 1]$ represents the normalized number of events per pixel, as predicted by the event generation module. Without this loss, the event generation focuses too strongly on the noisy and low-quality events present in the real event domain. This noise prediction helps the generator fool the discriminator since it can just predict noisy event frames that do not contain any structural information. Due to these noisy event frames, the task transfer performance degenerates. Therefore, we mitigate this effect by introducing the proposed image gradient loss.

A1.6 Task Loss

For the task loss $\mathcal{L}_{\text{task}}$, we use the standard cross-entropy loss and the loss defined in Yolov3 [34] for classification and object detection, respectively. As illustrated with $\mathcal{L}_{\text{task}}$ in Fig.3 in the manuscript, the task loss is applied twice, once in the image domain and once in the translated event domain.

A1.7 Training Procedure

The final loss is presented in Equation 15. As common in adversarial training, we train the generator and discriminator in separate steps. The network parameters are updated alternatingly by minimizing the following losses, whereby we train the generator for one step after two discriminator training steps.

$$\begin{aligned} L_{Gen} =& \mathcal{L}_{\text{lat,gen.}} + \mathcal{L}_{\text{recons.,gen.}} + \mathcal{L}_{\text{cycle}} \\ &+ 2\mathcal{L}_{\text{augm}} + \mathcal{L}_{grad} + \mathcal{L}_{task} \\ &+ \mathcal{L}_{\text{smooth}} \\ L_{Dis} =& \mathcal{L}_{\text{lat,disc.}} + \mathcal{L}_{\text{recons.,disc.}} \end{aligned} \tag{15}$$

A2 Ablation

We ablated our proposed design choices with multiple experiments for the task of classification on N-Caltech101 [30] as well as for the task of object detection on MVSEC [50].

A2.1 Classification

The experiments conducted on N-Caltech101 underline the improved performance of our proposed transfer framework compared to standard *Unsupervised Domain Adaptation (UDA)* methods, as reported in Table 3.

In a first experiment, we have adapted our network to a simple cycle GAN framework [53], in which we predict events and grayscale images directly from a shared feature space (Simple cycle). The significantly lower performance of 0.577 compared to 0.848 of our final approach verifies the need of considering prior sensor knowledge and the feature space split into a shared embedding space and an event-specific space for motion information.

Method	UDA	Accuracy ↑
Simple cycle	✔	0.577
Ours transl	✔	0.832
Ours w/o ref	✔	0.592
Ours w/o flow	✔	**0.848**
Ours	✔	**0.848**
Ours supervised	✗	0.839

Table 3. Ablation for the classification task on the N-Caltech101 dataset.

Method	Unpaired	mAP ↑
Ours w/o flow	✔	0.26
Ours w/o split	✔	0.41
Ours w/o augm	✔	0.48
Ours	✔	**0.54**

Table 4. Ablation for the object detecion task on the MVSEC dataset.

In a second experiment, we use our final network to solely translate from grayscale images to events (Ours transl). This way, we generate a labeled event dataset on which a task network can be trained. The performance of 0.832 verifies the accurate single image-to-event translation of our final network. However, the lower performance compared to our final task transfer network confirms that the task network benefits from simultaneously learning on images and events to detect the most relevant task features. A similar conclusion was also drawn in [36], where the authors achieved an increased task performance by learning with paired images and optical flow frames.

For the validation of our refinement network, we report the classification accuracy achieved without adding the residual event representation to the events constructed based on the flow and image gradients (Ours w/o ref). As expected, the performance suffers a substantial drop compared to the final network, which shows the importance of the refinement network. Due to the simple planar motion distribution, the event generation based on pseudo flow achieves the same performance as a direct prediction of the event representation (Ours w/o flow).

The benefits of the event generation based on the generative event model for more complex motions are shown in the experiments conducted on MVSEC. Finally, as reported in the manuscript in Section M4.1, our transferred network achieves higher performance than the same architecture trained with ground truth labels. This can be explained mainly for two reasons. First, our UDA method allows us to include additional labeled image data to help train the event classification network. Second, the simultaneous learning in the event and image representation increases the overall task performance.

A2.2 Object Detection

The ablation experiments for the task of objection detection on MVSEC strongly confirm our proposed network modules. The first experiment (Ours w/o flow) shows that the image-to-event translation without the generative event model decreases the task transfer by a large margin. The direct event representation prediction without the pseudo-flow estimation is not able to capture the complex motion distribution of a driving car sequence. The second experiment (Ours w/o split) verifies the benefits of splitting the feature space into sensor shared and event-specific features. Without the embedding space split, the performance suffers a significant mAP drop from 0.54 to 0.41 In the third experiment (Ours w/o augm), we validate the introduced flow augmentations, which help to split the embedding space into shared and event-specific features. The flow augmentations improve the detection score by 0.06 mAP. In conclusion, each of our proposed network modules substantially improves the object detection performance.

References

[1] Jonathan Binas, Daniel Neil, Shih-Chii Liu, and Tobi Delbruck. DDD17: End-to-end DAVIS driving dataset. In *ICML Workshop on Machine Learning for Autonomous Vehicles*, 2017. 7

[2] Yun-Chun Chen, Yen-Yu Lin, Ming-Hsuan Yang, and Jia-Bin Huang. Crdoco: Pixel-level domain transfer with cross-domain consistency. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019. 3

[3] Stanley Coren, Lawrence M. Ward, and James T. Enns. *Sensation and Perception*. Wiley Online Library, 2003. 1

[4] Tobi Delbruck, Yuhuang Hu, and Zhe He. V2E: From video frames to realistic DVS event camera streams. *arXiv e-prints*, 2020. 2, 3

[5] Qi Dou, Cheng Ouyang, Cheng Chen, Hao Chen, and Pheng-Ann Heng. Unsupervised cross-modality domain adaptation of convnets for biomedical image segmentations with adversarial loss. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 691–697, 2018. 2

[6] Mark Everingham, Luc Van Gool, Christopher Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, 06 2010. 8

[7] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(4):594–611, 2006. 6

[8] Robert Fisher. Cvonline: Image databases. https://homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.htm, 2021. 2

[9] Guillermo Gallego, Tobi Delbruck, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew Davison, Jörg Conradt, Kostas Daniilidis, and Davide Scaramuzza. Event-based vision: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020. 1

[10] Guillermo Gallego, Christian Forster, Elias Mueggler, and Davide Scaramuzza. Event-based camera pose tracking using a generative event model. arXiv:1510.01972, 2015. 3, 5

[11] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 1180–1189, 2015. 2

[12] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, Jan. 2016. 2

[13] Daniel Gehrig, Mathias Gehrig, Javier Hidalgo-Carrió, and Davide Scaramuzza. Video to Events: Recycling video datasets for event cameras. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2020. 2, 3, 6, 7, 9

[14] Daniel Gehrig, Antonio Loquercio, Konstantinos G. Derpanis, and Davide Scaramuzza. End-to-end learning of representations for asynchronous event-based data. In *Int. Conf. Comput. Vis. (ICCV)*, 2019. 7

[15] B. Gholami, P. Sahu, M. Kim, and V. Pavlovic. Task-discriminative domain alignment for unsupervised domain adaptation. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 1327–1336, 2019. 2

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Conf. Neural Inf. Process. Syst. (NIPS)*, pages 2672–2680, 2014. 4, 9

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 770–778, 2016. 6

[18] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A. Efros, and Trevor Darrell. Cycada: Cycle consistent adversarial domain adaptation. In *International Conference on Machine Learning (ICML)*, 2018. 3

[19] W. Hong, Z. Wang, M. Yang, and J. Yuan. Conditional generative adversarial network for structured domain adaptation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 1335–1344, 2018. 2

[20] Yuhuang Hu, Tobi Delbruck, and Shih-Chii Liu. Learning to exploit multiple vision modalitiesby using grafted networks. In *Eur. Conf. Comput. Vis. (ECCV)*, 2020. 2, 3, 7, 8

[21] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018. 3

[22] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 5967–5976, 2017. 4, 9

[23] Hsin-Ying Lee, Hung-Yu Tseng, Qi Mao, Jia-Bin Huang, Yu-Ding Lu, Maneesh Kumar Singh, and Ming-Hsuan Yang. Drit++: Diverse image-to-image translation viadisentangled representations. *International Journal of Computer Vision*, pages 1–16, 2020. 2, 3, 6

[24] Peilun Li, Xiaodan Liang, Daoyuan Jia, and Eric P. Xing. Semantic-aware grad-gan forvirtual-to-real urban

scene adaption. In *British Mach. Vis. Conf. (BMVC)*, 2018. 2

[25] 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 *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 5419–5427, 2018. 6

[26] S.M. Mostafavi I., Lin Wang, Yo-Sung Ho, and Kuk-Jin Yoon Yoon. Event-based high dynamic range image and very high frame rate video generation using conditional generative adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019. 3

[27] S. Mohammad Mostafavi I., Jonghyun Choi, and Kuk-Jin Yoon. Learning to super resolve intensity images from events. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 2768–2786, June 2020. 3

[28] Z. Murez, S. Kolouri, D. Kriegman, R. Ramamoorthi, and K. Kim. Image to image translation for domain adaptation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4500–4509, 2018. 3

[29] J. Nikolic, J. Rehder, M. Burri, P. Gohl, S. Leutenegger, P. Furgale, and R. Siegwart. A synchronized visual-inertial sensor system with FPGA pre-processing for accurate real-time SLAM. In *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2014. 8

[30] Garrick Orchard, Ajinkya Jayawant, Gregory K. Cohen, and Nitish Thakor. Converting static image datasets to spiking neuromorphic datasets using saccades. *Front. Neurosci.*, 9:437, 2015. 2, 6, 10

[31] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. ESIM: an open event camera simulator. In *Conf. on Robotics Learning (CoRL)*, 2018. 2, 3, 7

[32] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. Events-to-video: Bringing modern computer vision to event cameras. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019. 3, 8

[33] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019. 2, 3, 6, 7

[34] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv e-prints*, abs/1804.02767, 2018. 7, 10

[35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Fei-Fei Li. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, Apr. 2015. 6

[36] N. Sayed, Biagio Brattoli, and Björn Ommer. Cross and learn: Cross-modal self-supervision. In *German Conference on Pattern Recognition (GCPR) (Oral)*, Stuttgart, Germany, 2018. 7, 10

[37] Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4058–4065. AAAI Press, 2018. 2

[38] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. Learning from simulated and unsupervised images through adversarial training. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 2242–2251, 2017. 2

[39] Amos Sironi, Manuele Brambilla, Nicolas Bourdis, Xavier Lagorce, and Ryad Benosman. HATS: Histograms of averaged time surfaces for robust event-based object classification. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 1731–1740, 2018. 7

[40] Deqing Sun, Stefan Roth, and Michael J. Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. *Int. J. Comput. Vis.*, 2014. 9

[41] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2446–2454, 2020. 7, 9

[42] Ryuhei Takahashi, Atsushi Hashimoto, Motoharu Sonogashira, and Masaaki Iiyama. Partially-shared variational auto-encoders for unsupervised domain adaptation with target shift. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 1–17. Springer International Publishing, 2020. 2, 3

[43] Dustin Tran, Rajesh Ranganath, and David M. Blei. Hierarchical implicit models and likelihood-free variational inference. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5529–5539, Red Hook, NY, USA, 2017. Curran Associates Inc. 4, 9

[44] Lin Wang, Yujeong Chae, Sung-Hoon Yoon, Tae-Kyun Kim, and Kuk-Jin Yoon. Evdistill: Asynchronous events to end-task learning via bidirectional reconstruction-guided cross-modal knowledge distillation. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2021. 3, 7

[45] L. Wang, Tae-Kyun Kim, and Kuk-Jin Yoon. Eventsr: From asynchronous events to image reconstruction, restoration, and super-resolution via end-to-end adversarial learning. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 8312–8322, 2020. 3

[46] Garrett Wilson and Diane J. Cook. A survey of unsupervised deep domain adaptation. *ACM Trans. Intell. Syst. Technol.*, 11(5), July 2020. 2

[47] Junlin Yang, Nicha C. Dvornek, Fan Zhang, Julius Chapiro, MingDe Lin, and James S. Duncan. Unsupervised domain adaptation via disentangled representations: Application to cross-modality liver segmentation. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, pages 255–263, 2019. 2, 3

[48] Song Zhang, Yu Zhang, Zhe Jiang, Dongqing Zou, Jimmy Ren, and Bin Zhou. Learning to see in the dark with events. In *Eur. Conf. Comput. Vis. (ECCV)*, 2020. 3

[49] Ziqiang Zheng, Yang Wu, Xinran Han, and Jianbo Shi. Forkgan: Seeing into the rainy night. In *The IEEE European Conference on Computer Vision (ECCV)*, August 2020. 2, 3

[50] Alex Zihao Zhu, Dinesh Thakur, Tolga Ozaslan, Bernd Pfrommer, Vijay Kumar, and Kostas Daniilidis. The multivehicle stereo event camera dataset: An event camera dataset for 3D perception. *IEEE Robot. Autom. Lett.*, 3(3):2032–2039, July 2018. 2, 7, 10

[51] Alex Zihao Zhu, Ziyun Wang, Kaung Khant, and Kostas Daniilidis. Eventgan: Leveraging large scale image datasets for event cameras. *arXiv preprint arXiv:1912.01584*, 2019. 2, 3, 4, 7, 8, 9

[52] Alex Zihao Zhu, Liangzhe Yuan, Kenneth Chaney, and Kostas Daniilidis. EV-FlowNet: Self-supervised optical flow estimation for event-based cameras. In *Robotics: Science and Systems (RSS)*, 2018. 8

[53] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 2242–2251, 2017. 2, 3, 10