

# Generative Event Pretraining with Foundation Model Alignment

Jianwen Cao      Jiayu Xing      Nico Messikommer      Davide Scaramuzza

Robotics and Perception Group, University of Zurich

## Abstract

Event cameras provide robust visual signals under fast motion and challenging illumination thanks to their microsecond latency and high dynamic range. However, their unique sensing characteristics and limited labeled data make it challenging to train event-based visual foundation models (VFMs), which are crucial for learning visual features transferable across tasks. To tackle this problem, we propose GEP (Generative Event Pretraining), a two-stage framework that transfers semantic knowledge learned from internet-scale image datasets to event data while learning event-specific temporal dynamics. First, an event encoder is aligned to a frozen VFM through a joint regression-contrastive objective, grounding event features in image semantics. Second, a transformer backbone is autoregressively pretrained on mixed event-image sequences to capture the temporal structure unique to events. Our approach outperforms state-of-the-art event pretraining methods on a diverse range of downstream tasks, including object recognition, segmentation, and depth estimation. Together, VFM-guided alignment and generative sequence modeling yield a semantically rich, temporally aware event model that generalizes robustly across domains. Code: [uzh-rpg/generative\\_event\\_pretraining](https://github.com/uzh-rpg/generative_event_pretraining)

## 1. Introduction

Event cameras [29] measure per-pixel brightness changes asynchronously with microsecond latency and high dynamic range. Unlike conventional RGB frames, event streams are sparse and provide high temporal resolution, allowing robust perception in challenging lighting and fast-motion scenarios [15, 35]. However, these advantages also introduce challenges: events contain limited texture, differ fundamentally from images, and lack access to large-scale training datasets [15, 47]. These challenges make it difficult to transfer semantic knowledge from images to events, and hinder the robustness of models that rely on large-scale, texture-rich supervision.

Consequently, many existing approaches rely on lim-

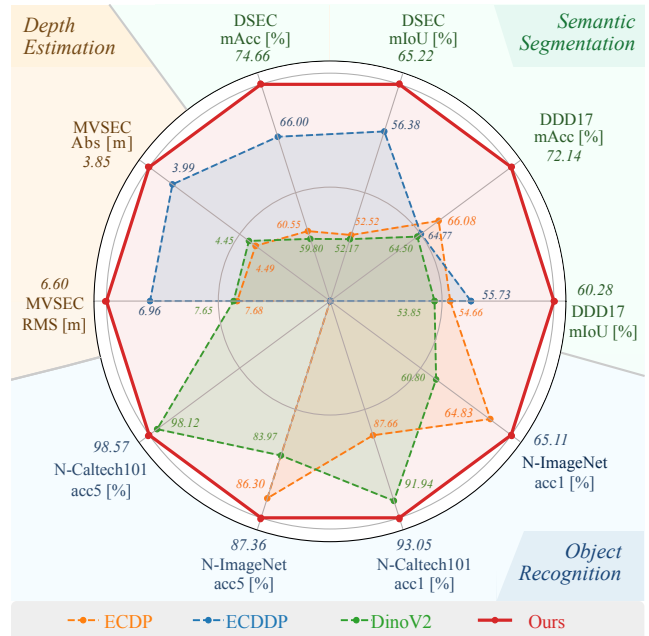


Figure 1. Overall comparison across ten metrics on various datasets. With the same backbone model, our method demonstrates superior and consistent performance across all tasks.

ited event-only pretraining, small task-specific datasets, or transfer directly from image-pretrained models that are not optimized for the sparse and asynchronous nature of event data [26, 33, 52, 58, 59]. These strategies either miss long-range temporal dynamics or fail to import rich semantic priors from RGB due to suboptimal cross-modality adaptation.

While recent advances in vision foundation models (VFMs) [11, 13, 39, 45] demonstrate that large-scale pretraining on diverse image corpora produces semantically rich and task-transferable representations, these benefits have not yet been fully exploited in the event domain. We address this gap with a unified pretraining paradigm that transfers VFM-level semantics to event data while explicitly modeling temporal dynamics in event streams. By aligning event features with foundation-level image representations, the event encoder inherits broad semantic priors and

narrows the discrepancy between event and image modalities. The subsequent generative pretraining stage further equips the model with long-horizon predictive capabilities, enabling scalable, task-agnostic learning on large datasets featuring events and images.

To this end, we introduce *Generative Event Pretraining (GEP)*, a two-stage training framework. In the first stage, an event encoder is aligned to a frozen image encoder (DI-NOv2 [39]) using a combination of MSE and InfoNCE [38] objectives, transferring semantic structure from the image to the event domain. Such cross-modal alignment can also be viewed as a teacher-student training paradigm [34], where a pretrained teacher provides rich supervision for a student operating under different observation modalities. In the second stage, we perform autoregressive pretraining on aligned event and image sequences using a causal transformer. Unlike masked autoencoding [22], our model predicts future token sequences over randomly masked temporal gaps, encouraging long-horizon temporal reasoning and providing dense generative supervision. We pretrain on the large-scale Event-1.8M corpus with 1.8M event-image pairs, including EventScape [17], N-ImageNet [25], and DSEC [18], covering diverse spatial and temporal patterns.

To evaluate generalization, we conduct experiments on object recognition (N-ImageNet [25], N-Caltech101 [40]), semantic segmentation (DDD17 [3], DSEC [18]), and depth estimation (MVSEC [63]). Among them, N-Caltech101, DDD17, and MVSEC are not included during pretraining and thus serve as out-of-distribution benchmarks.

As summarized in Fig. 1, our method consistently achieves superior performance across all 10 metrics, surpassing both event-specific and image-pretrained baselines while requiring only 24 pretraining epochs. These results demonstrate that aligning event and image features, coupled with autoregressive generative pretraining, enables effective cross-modal transfer and robust task generalization

The main contributions of our work are as follows:

1. *VFM-guided semantic alignment.* We align an event encoder with a pretrained VFM encoder to transfer semantic knowledge learned from large-scale image datasets, effectively grounding event features in rich visual priors.
2. *Autoregressive event pretraining.* We propose an autoregressive pretraining strategy on unlabeled event sequences paired with aligned images, which enables long-horizon temporal reasoning and captures the distinct temporal dynamics of event streams.
3. *Cross-domain generalization.* With about only 15% training epochs compared to previous methods, our model outperforms state-of-the-art methods on recognition (N-ImageNet, N-Caltech101), segmentation (DDD17, DSEC), and depth estimation (MVSEC) benchmarks. Our pretrained backbone demonstrates robust generalization across diverse domains, providing a

strong foundation for future event-based vision research.

## 2. Related Work

### 2.1. Transfer Learning Across Modalities

In contrast to the conventional pretrain–finetune paradigm in the event domain, transferring knowledge across modalities involves aligning representations between images and events. Prior work in this direction can be broadly divided into two groups: Unsupervised domain adaptation and feature-level distillation.

**Unsupervised domain adaptation.** UDA bridges the gap between labeled images and unlabeled events by enforcing consistency in features or predictions [33, 52, 57]. ESS [52] aligns embeddings through reconstruction and prediction losses to transfer labels from Cityscapes to unpaired event data. CMES [57] adds attention-guided soft alignment and joint decoder constraints for dense prediction.

**Feature level distillation.** Another line distills intermediate features from image models into event encoders using aligned image-event pairs. Hu et al. [24] use grafting by replacing early image layers with an event front end and regressing internal features for label free adaptation. Depth AnyEvent [2] distills multi scale representations into event-based depth estimators and shows that semantic priors help reconstruction and estimation.

Across UDA and distillation, existing methods are designed for a single task and trained on limited data. In contrast, we align events to foundation-level image features in a task-agnostic way and couple the alignment with generative autoregressive pretraining, which captures temporal structure and enables transfer to both recognition and dense prediction tasks.

### 2.2. Pretraining for Event Vision

Event pretraining targets the learning of general feature representations that are subsequently finetuned for downstream tasks using task labels.

**Masked and self reconstruction pretraining.** Masked Event Modeling adapts masked autoencoding and frame reconstruction to event streams. MEM [26] follows MAE [22] by masking patches and reconstructing from context. These methods capture local statistics but remain reconstruction-oriented and lack explicit temporal reasoning.

**Contrastive and multimodal alignment.** Recent methods align events with images or with joint vision–language embedding spaces using contrastive objectives. ECDP [58] aligns event and RGB patch embeddings through momentum-based contrastive learning, leveraging an image encoder pretrained with MoCov3 [7] for supervision but without fully exploiting its rich semantic priors. ECDDP [59] extends this concept to spatial feature maps for pixel level tasks. EventCLIP [55] projects event features

into CLIP space for zero-shot and few-shot recognition. EventBind [62] applies multi-stage contrastive finetuning for stronger event image text alignment. In contrast, our framework adds feature regression toward foundation-level image representations and couples alignment with autoregressive pretraining, which yields semantically grounded and temporally aware event features.

**Autoregressive and predictive pretraining.** Temporal modeling learns motion dynamics directly from events. Event Transformer [50] introduces a sparse-aware transformer with patch-based representations and latent memory tokens for online recognition. Recent models replace recurrent blocks with structured state space modules S4 and S5 [19, 64], which provide learnable time scales and adapt linear attention language modeling, such as RWKV [41], for asynchronous encoding with multi-step or next representation prediction [21]. These approaches confirm the strong temporal structure of events but are usually single modality, task-specific, and are often trained from scratch on limited data. We pretrain in a task-agnostic way under guidance from vision foundation models to unify semantic alignment with autoregressive temporal reasoning.

### 2.3. Event VLMs with LLM backbones

A recent popular direction toward generalizable event understanding is to extend large multimodal language models to event data for open-ended reasoning and description. EventGPT [30] aggregates spatiotemporal tokens before feeding a language model to support captioning and question answering. Event-VL [27] builds a generative event-based multimodal model with dynamic semantic alignment. EP VLM [43] uses event prior guided sparsification to prune redundant tokens and can adapt large backbones such as Qwen2-VL [54]. Many event language and video language systems compress visual tokens by about an order of magnitude for efficiency, for example, EventGPT, Event-VL, VideoLLaMA3 [61], and ARVideo [48]. We observe that compression improves reasoning efficiency but harms dense prediction that requires fine spatial detail, see Sec. 4.6. Similar findings have been reported for segmentation and other dense tasks [31, 53]. Large LLM backbones of billions of parameters also hinder real-time inference at high event rates. Our approach avoids heavy language backbones, preserves spatial detail, and transfers to recognition and dense prediction.

## 3. Method

### 3.1. Overview

Our method consists of two main stages: *alignment* and *pre-training*. An overview of the entire framework is illustrated in Fig. 2. Following the event accumulation as detailed in Sec 3.2, each temporal window yields a normalized event

frame  $X_e$  paired with a synchronized image  $X_i$ , as illustrated in Fig. 2 (a). Given a paired event–image sample  $(X_e, X_i)$ , the event encoder  $E_e$  and the image encoder  $E_i$  extract modality-specific embeddings  $(Z_e, Z_i)$  used for the modality alignment. For the pretraining, we alternately accumulate events and sample images over time, feeding them into the corresponding encoders to obtain a multimodal feature sequence. All encoded embeddings are concatenated into a long sequence  $S$ , corresponding to the token sequence fed to the causal transformer in Fig. 2 (b). For each slice  $S_{s,s+w}$ , a causal transformer  $T$  autoregressively predicts the next slice  $S_{s+1,s+1+w}$ :

$$\hat{S}_{s+1,s+1+w} = T(S_{s,s+w}), \quad (1)$$

where  $s$  is a randomly sampled starting index and  $w$  is a fixed window length.

### 3.2. Event Accumulation

Event streams are composed of asynchronous brightness changes represented as tuples  $(x, y, t, p)$ , where  $(x, y)$  denotes pixel coordinates,  $t$  the timestamp, and  $p \in \{+1, -1\}$  the polarity indicating whether the brightness increased or decreased. To construct a dense tensor representation suitable for transformer-based modeling, we accumulate events within a fixed temporal window  $\Delta t$  into a pseudo-frame  $X_e \in \mathbb{R}^{H \times W \times 3}$ .

For each event, we locate its corresponding pixel cell  $(x, y)$  and increment the first channel count  $M_r(x, y)$  if the polarity is positive ( $p = +1$ ), or the third channel count  $M_b(x, y)$  if it is negative ( $p = -1$ ). For the second channel, we assign full intensity whenever any event occurs at  $(x, y)$  within the temporal window, serving as an activity mask that marks all active pixels regardless of polarity. After processing all events in the window, the accumulated counts are normalized to suppress local activity spikes and maintain a balanced dynamic range.

Let  $M_c(x, y)$  denote the raw accumulated count in channel  $c \in \{r, b\}$ . We compute a normalization scale  $\alpha_n$  as the  $n$ -th percentile of all pixel values across both channels (by default  $n = 99$ ). Each channel is then clipped and normalized as

$$X_e(x, y, c) = \frac{\min(M_c(x, y), \alpha_n)}{\alpha_n}, \quad c \in \{r, b\}. \quad (2)$$

This percentile-based clipping and normalization ensure robust scaling and prevent a few high-activity pixels from dominating the distribution, resulting in a stable and balanced input representation for subsequent encoding.

At the end of this stage, each event window produces a normalized event frame  $X_e$  that can be paired with a synchronized image frame  $X_i$ . These paired samples  $(X_e, X_i)$  serve as inputs to the following alignment and pretraining stages.

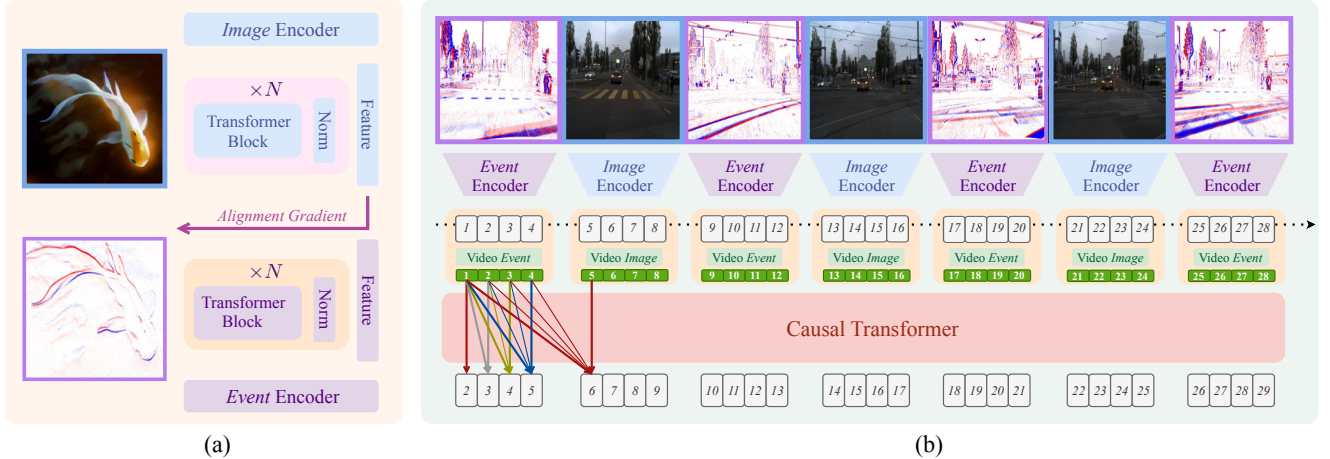


Figure 2. The overall two-stage framework. (a) Alignment stage: Event frames and synchronized images are encoded by an event encoder and a frozen VFM encoder. The event encoder is optimized with regression, contrastive, and preservation terms to match the semantic structure of image features. (b) Autoregressive pretraining stage: Aligned event and image embeddings are interleaved into a single sequence and processed by a causal transformer, which predicts future slices from partial windows, learning long-range temporal structure and cross-modal consistency. Arrows indicate causal dependencies; only a subset is shown for visual clarity.

### 3.3. Alignment

The weights of both encoders are initialized from DINOv2 [39]. We freeze the image branch  $E_i$  (parameters  $\theta_i$ ) and update only the event branch  $E_e$  (parameters  $\theta_e$ ). Since vision foundation models already capture strong and well-structured semantics from large-scale RGB corpora, their features provide a reliable reference. Freezing  $E_i$  therefore treats the image encoder as a fixed semantic teacher, allowing the event encoder to learn meaningful alignment instead of co-adapting to unstable event features. The alignment objective combines a cosine similarity and an InfoNCE term, while a preservation loss constrains the behavior of  $E_e$  on image inputs. Unless otherwise noted, we use DINOv2 [39] as the image foundation model.

**Event-image alignment.** Given a paired input  $(X_e, X_i)$ , we extract features

$$Z_e = E_e(X_e), \quad Z_i = \text{sg}(E_i(X_i)),$$

where  $\text{sg}$  stops gradients through the frozen image branch. We use the following alignment loss

$$\mathcal{L}_a = \lambda_{\cos} (1 - \cos(Z_e, Z_i)) + \lambda_{\text{ncc}} \mathcal{L}_{\text{ncc}}(Z_e, Z_i), \quad (3)$$

where  $\cos(\cdot, \cdot)$  denotes cosine similarity. Following standard in-batch contrastive learning, the InfoNCE loss is defined as

$$\mathcal{L}_{\text{ncc}} = -\frac{1}{N} \sum_{n=1}^N \log \frac{\exp(\cos(\tilde{z}_e^n, \tilde{z}_i^n)/\tau)}{\sum_{m=1}^N \exp(\cos(\tilde{z}_e^n, \tilde{z}_i^m)/\tau)}, \quad (4)$$

where  $\tilde{z} = z/\|z\|_2$  denotes  $\ell_2$ -normalized features,  $\tau$  is the temperature, and  $N$  is the batch size. Each positive pair

$(\tilde{z}_e^n, \tilde{z}_i^n)$  is contrasted against all other image embeddings in the batch, which act as negative anchors.

The cosine term enforces directional consistency between event and image embeddings, ensuring that both modalities lie in a shared feature subspace regardless of scale, while the InfoNCE term encourages global separation from unrelated image embeddings, leading to semantically consistent yet discriminative representations. In practice, we apply a lightweight projection head  $g(\cdot)$  before (4) following SimCLR [6]. This allows the contrastive branch to adapt its embedding distribution without disturbing the main cosine alignment space, stabilizing optimization and improving cross-modal transfer.

**Capability preservation on images.** To avoid drift when  $E_e$  processes RGB inputs, we pass  $X_i$  through  $E_e$  and match the frozen image features

$$Z_e^{(I)} = E_e(X_i), \quad Z_i = \text{sg}(E_i(X_i)),$$

with the preservation loss

$$\mathcal{L}_p = \mu (1 - \cos(Z_e^{(I)}, Z_i)). \quad (5)$$

Equation (5) regularizes  $E_e$  by preventing its feature space from collapsing or drifting away from the semantic manifold learned by  $E_i$ . By enforcing consistent orientation when both branches process image inputs, this term discourages degenerate alignment and maintains semantic grounding.

**Total objective.** The alignment stage optimizes

$$\mathcal{L} = \mathcal{L}_a + \mathcal{L}_p, \quad \lambda_{\cos} > 0, \lambda_{\text{ncc}} > 0, \mu > 0. \quad (6)$$

To balance the loss terms, we use separate weights for cosine alignment ( $\lambda_{\text{cos}}$ ), contrastive separation ( $\lambda_{\text{ncc}}$ ), and preservation ( $\mu$ ). Batch composition and temperature  $\tau$  follow standard practice for in-batch contrastive learning.

### 3.4. Pre-training

After alignment, both encoders produce semantically aligned feature embeddings. We concatenate them into a multimodal sequence  $S = [S_1, S_2, \dots, S_K]$  that mixes samples from image datasets and paired event-video datasets, as depicted in Fig. 2 (b). Each token is projected into a shared embedding space and augmented with positional and modality encodings before entering the causal transformer.

**Token representation.** Given an encoded token  $S_k$ , its input representation is

$$X_k = S_k + P_k + M_k, \quad (7)$$

where  $P_k$  is the positional embedding and  $M_k$  is the modality encoding, indicating whether the token originates from an event frame, image frame, or video sequence. The modality encoding allows the transformer to distinguish heterogeneous sources within the same batch, while positional encoding preserves temporal order.

**Dense autoregressive training.** We slice a long multimodal sequence into overlapping windows of length  $w$  with a stride 1 to form a dense training target. In other words, within one training sample, the model performs

$$S_1 \rightarrow S_2, \quad (S_1, S_2) \rightarrow S_3, \quad \dots, \quad S_{1:w-1} \rightarrow S_w,$$

yielding a dense sequence of autoregressive targets:

$$\mathcal{L}_{\text{pre}} = \frac{1}{w} \sum_{j=1}^w \|\hat{S}_{s+j} - S_{s+1+j}\|_2^2, \quad (8)$$

encouraging temporally coherent and modality-consistent predictions.

## 4. Experiments

### 4.1. Datasets and Baselines

We evaluate on established event-vision benchmarks that cover both recognition, segmentation, and depth estimation tasks. For object recognition, we use N-ImageNet [25] and N-Caltech101 [40], the neuromorphic counterparts of ImageNet [10] and Caltech101 [14]. N-ImageNet contains large-scale event recordings of 1K ImageNet categories captured with a moving DAVIS sensor [4], while N-Caltech101 provides 101 object classes with simpler backgrounds and fewer samples per class. For semantic segmentation, we use DDD17 [3] and DSEC [18, 52], two large-scale driving datasets with synchronized events and

Table 1. Object recognition on N-ImageNet and N-Caltech101. We report top-1 (acc1) and top-5 (acc5) accuracies. The two best-performing methods for each evaluation metric are highlighted in green and orange.

Method	Backbone	Dataset	Ep.	N-ImageNet acc1↑	N-ImageNet acc5↑	N-Caltech101 acc1↑	N-Caltech101 acc5↑
<i>Training from scratch</i>							
ViT[11]	ViT-S/16	N-ImageNet	300	46.70	69.89	55.63	–
<i>Specific trained</i>							
EST[16]	–	–	–	48.93	–	–	–
<i>Self-supervised pretraining</i>							
BeiT[1]	ViT-B/16	ImageNet-1K	800	47.15	69.27	53.10	–
MAE[22]	ViT-B/16	ImageNet-1K	800	51.25	72.64	67.68	–
MoCo-v3[7]	ViT-S/16	ImageNet-1K	300	45.77	68.89	76.59	–
DINOv2[39]	ViT-S/16	LVD-142M	–	60.80	83.97	91.94	98.12
<i>Event-specific pretraining</i>							
MEM[26]	ViT-S/16	N-ImageNet	75	57.89	–	–	–
ECDP[58]	ViT-S/16	N-ImageNet	300	64.83	86.30	87.66	–
EventBind[59]	ViT-B/16	N-ImageNet	–	51.40	–	94.08	–
STP [28]	Swin-T	N-ImageNet	–	68.87	89.65	–	–
Ours	ViT-S/16	Event-1.8M	24	65.11	87.36	93.05	98.57
Ours (Sup. Align)	ViT-S/16	N-ImageNet	6	69.41	90.20	–	–
Ours	ViT-B/16	Event-1.8M	24	75.20	92.90	96.47	99.56

grayscale frames. DDD17 offers 12 hours of urban driving sequences with 6 semantic classes derived from intensity frames, while DSEC provides 8,082 higher-resolution stereo event-image-annotation pairs for training. For depth estimation, we use the MVSEC dataset [63], which provides synchronized stereo DAVIS346 event cameras with grayscale frames and accurate depth maps. MVSEC contains indoor and outdoor sequences captured from handheld, hexacopter, car, and motorcycle platforms, enabling depth evaluation under diverse motions and illumination conditions.

For pretraining, we adopt Event-1.8M, a large-scale event corpus that integrates EventScape [17], N-ImageNet, and DSEC. This mixture covers diverse spatial patterns and motion statistics, serving as a task-agnostic dataset for representation learning. DDD17, MVSEC, and N-Caltech101 are excluded from pretraining and used only for out-of-distribution evaluation. More training details are included in the supplementary material.

We compare our model against four groups of baselines to ensure a fair and comprehensive evaluation. The first group trains a ViT [11] from scratch on event data without any pretraining. The second group is specifically designed for each task. The third group includes self-supervised pretraining on images, represented by MAE [22], BeiT [1], and DINOv2 [39]. The last group consists of event-specific pretraining methods, including MEM [26], ECDP [58], ECDDP [59], and EventBind [62].

Table 2. Semantic segmentation on DDD17 and DSEC. We report mean IoU (mIoU) and mean ACC (mAcc). The two best-performing methods for each evaluation metric are highlighted in green and orange.

Method	Backbone	Dataset	Ep.	DDD17		DSEC	
				mIoU $\uparrow$	mAcc $\uparrow$	mIoU $\uparrow$	mAcc $\uparrow$
<i>Training from scratch</i>							
ViT[11]	ViT-S/16	-	-	36.65	46.21	32.66	40.84
<i>Specific trained</i>							
ESS[52]	-	Cityscape	50	61.37	70.87	53.30	62.94
<i>Self-supervised pretraining</i>							
BeiT[1]	ViT-B/16	ImageNet1K	800	52.39	61.95	51.90	59.66
MAE[22]	ViT-B/16	ImageNet1K	800	53.76	64.78	51.96	59.84
DINOv2[39]	ViT-S/14	LVD-142M	-	53.85	64.50	52.17	59.80
<i>Event-specific pretraining</i>							
MEM[26]	ViT-S/16	N-ImageNet	75	-	-	44.62	51.39
ECDDP[58]	ViT-S/16	N-ImageNet	300	54.66	66.08	52.52	60.55
ECDDP[59]	ViT-S/16	E-TartanAir	300	55.73	64.77	56.38	66.00
TESPEC[37]	Swin-T	1Mpx [42]	-	-	-	62.77	70.61
Ours	ViT-S/14	N-ImageNet	24	58.42	69.59	60.43	68.76
Ours	ViT-S/14	Event-1.8M	24	60.28	72.14	65.22	74.66
Ours	ViT-B/14	Event-1.8M	24	61.90	72.39	67.37	76.27

## 4.2. Object Recognition

As can be observed by the top-1 and top-5 accuracy reported in Table 1, our model consistently surpasses event-specific pretraining baselines. Compared with ECDDP, we improve top-1 from 64.83% to 65.11% on N-ImageNet and from 87.66% to 93.05% on N-Caltech101. Although the event encoder is aligned to DINOv2, our model achieves higher accuracy on event streams than its VFM teacher, trained on 142M RGB images and applied to events. This shows the benefit of the alignment followed by autoregressive modeling to transfer semantics while adapting the features to the sparsity and asynchronicity of events. With only 24 epochs of training (including alignment), our approach outperforms DINOv2 on the tested event benchmarks, demonstrating strong data efficiency and generalization. Remarkably, our method outperforms STP [28] under fair settings, while using only 25% of the pre-training schedule (Ours Sup. Align). As shown in Table 1, STP’s parameter efficiency relies on privileged supervised pre-training with explicit labels.

## 4.3. Semantic Segmentation

For DDD17, we use sequence1 for training and the remaining sequences for evaluation, as done in ESS [52]. Following DINOv2, a simple linear segmentation head is attached on top of the encoder. Thus, the performance mainly reflect the quality of the backbone representations. We use cross-entropy and Dice losses [36] on labeled events.

Note that both ECDDP and ECDDP adopt a stronger UperNet [56] decoder. ECDDP further incorporates a 3D expanded patch embedding [60], while ECDDP additionally applies test-time augmentation with horizontal flipping

Table 3. Depth estimation on MVSEC. We report absolute error (Abs $\downarrow$ ) and root mean squared error (RMS $\downarrow$ ). The two best-performing methods for each evaluation metric are highlighted in green and orange.

Method	Backbone	Dataset	Ep.	Abs $\downarrow$	RMS $\downarrow$
<i>Specific trained</i>					
HMNet[20]	-	-	-	4.61	8.60
<i>Self-supervised pretraining</i>					
BeiT[1]	ViT-B/16	ImageNet-1K	800	4.40	7.56
MAE[22]	ViT-B/16	ImageNet-1K	800	4.45	7.60
DINOv2[39]	ViT-S/16	LVD-142M	-	4.45	7.65
<i>Event-specific pretraining</i>					
ECDDP[58]	ViT-S/16	N-ImageNet	300	4.49	7.68
ECDDP[59]	ViT-S/16	N-ImageNet	300	3.99	6.96
Ours	ViT-S/16	Event-1.8M	24	3.85	6.60

and multi-scale features. Despite our lighter linear head and the absence of test time augmentation, our method achieves higher segmentation accuracy, indicating that the gains come from the learned representations rather than from a heavy decoder or inference heuristics.

Table 2 summarizes the semantic segmentation performance reported in mIoU and mAcc. Our model attains the best performance on both datasets, outperforming ECDDP by +4.0 mIoU and +5.0 mAcc on DSEC, indicating that purely static pretraining struggles to capture event dynamics, while our alignment plus autoregressive pretraining bridges this gap.

## 4.4. Depth estimation

We follow the protocol of ECDDP [59] on MVSEC. Training is conducted on the ‘outdoor\_day2’ split, and validation is performed on ‘outdoor\_day1’, ‘outdoor\_night1’, ‘outdoor\_night2’, and ‘outdoor\_night3’. Following ECDDP, the

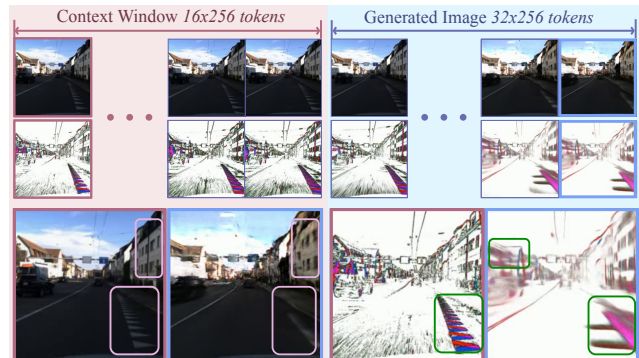


Figure 3. Visualization of a 16-frame context (blue) and a 32-frame autoregressively generated future (red) on validation interleaved event and image streams. Green boxes highlight consistent motion. Insets enlarge the first context and last generated frames for clarity.

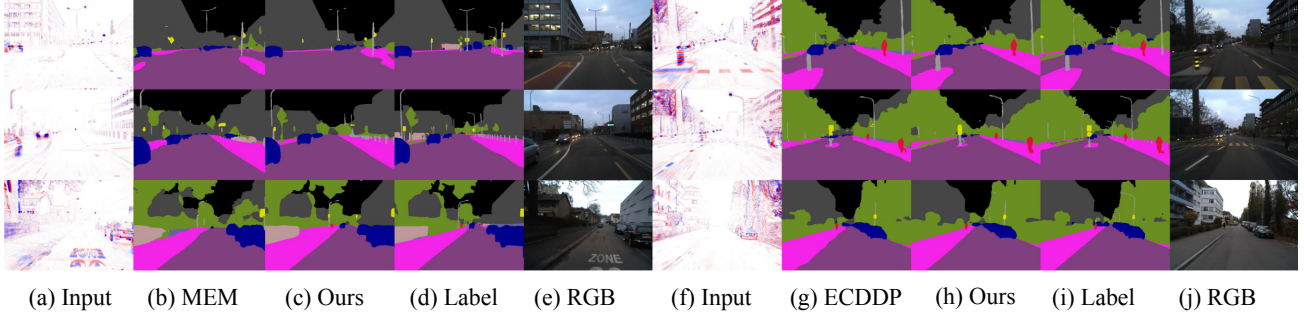


Figure 4. Qualitative results on the DSEC validation set. The RGB images are shown only for visual reference and are not used by the model.

network is trained to predict normalized log depth. The loss combines a scale-invariant term and a multi-scale gradient matching term. At test time, predictions are converted back to metric depth, and we report Absolute error and Root Mean Squared error in meters without limiting the maximum depth during evaluation, consistent with prior work [17]. Table 3 summarizes the results. Our method achieves lower Abs and RMS errors than ECDDP and other baselines, demonstrating that the proposed pretraining remains highly effective for depth estimation tasks that require strong 3D spatial understanding.

#### 4.5. Qualitative results

**Autoregressive long-horizon generation.** We visualize sequence rollouts using a frozen, lightweight decoder attached to the encoder, which is used only for qualitative inspection of latent predictions. Each rollout is conditioned on a multimodal context of  $256 \times 8 \times 2 = 4096$  tokens and generates  $256 \times 16 \times 2 = 8192$  future tokens. Here, 256 denotes spatial tokens per frame, 8 and 16 are the context and predicted steps, and 2 corresponds to interleaved event and image modalities. A sliding causal window is applied when the target horizon exceeds the pretraining window length. Figure 3 indicates that the model extrapolates ego motion, facade parallax, lane-marking drift, and object displacement even when such cues are weak in the last context frame. Despite moderate sensor noise and the minimal decoder, rollouts remain stable and geometrically consistent, which suggests the transformer internalizes long-range temporal structure.

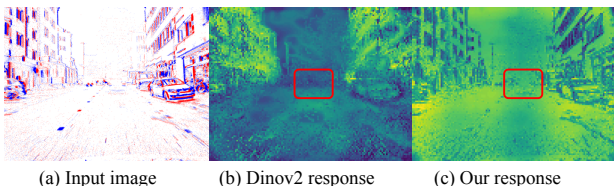


Figure 5. Attention map response on event camera data.

Table 4. Clustering metrics on encoder features (20 classes, 50 samples/class). Higher is better for Silhouette and Calinski–Harabasz; lower is better for Davies–Bouldin.

Metric	ViT-Small		ViT-Base	
	Dinov2	Aligned	Dinov2	Aligned
Silhouette[49] $\uparrow$	-0.03	<b>0.19</b>	0.03	<b>0.27</b>
Davies–Bouldin[9] $\downarrow$	5.24	<b>2.21</b>	3.75	<b>2.04</b>
Calinski–Harabasz[5] $\uparrow$	9.39	<b>33.21</b>	14.97	<b>48.56</b>

**Segmentation results.** Figure 4 compares our model with MEM [26] and ECDDP[59]. Our predictions exhibit cleaner boundaries and more coherent region semantics, especially in low-texture or overexposed scenes where event measurements are sparse. In the last row, the roadside pedestrian is barely discernible in the event input due to missing texture, which makes fine structures difficult to detect. With VFM-guided alignment and generative pretraining, the network recovers semantically consistent contours and preserves layout in spite of weak appearance cues.

We refer to the supplementary material for more qualitative task results

**Attention responses.** Figure 5 analyzes attention on event inputs. The red box marks a distant car that is hard to perceive in the raw events. DINOv2 with registers [8] usually produces meaningful attention on RGB data, yet on events, its response is weak and spatially inconsistent with spurious activations on the road surface. Our aligned and generatively pretrained encoder yields concentrated and structured attention that focuses on the car and other salient elements, indicating effective cross-modal alignment and stronger semantic grounding.

**Alignment Clustering.** Figure 6 shows a t-SNE plot for event features before and after alignment. Although the alignment pretraining is class-agnostic, it still exhibits pronounced class-wise separation, consistent with near-linear separability in the learned feature space. Table 4 quantifies the effect, which indicates reduced intra-class variance and stronger inter-class separation, supporting the view that

Table 5. Feature alignment results on N-ImageNet using 30,000 training steps. Each cell shows Original→Aligned with relative change (%).

Metric	Distance Reduction	ImageNet Acc. (%)
MSE	2.22→0.78 (-64.7%)	31.18→61.45
InfoNCE	2.28→0.57 (-75.0%)	31.18→60.99
Cosine	0.70→0.27 (-61.8%)	31.18→61.80
KL Diver.	0.48→0.15 (-67.5%)	31.18→59.20
Attention	38.76→19.42 (-49.9%)	31.18→56.72
Cosine + InfoNCE	1.40→0.36 (-74.3%)	31.18→ <b>62.73</b>
All	0.96→0.37 (-61.1%)	31.18→61.40

alignment organizes event features into semantically meaningful manifolds that aid downstream tasks.

#### 4.6. Ablations

We report linear probing with a frozen backbone to isolate representation quality in order to ablate alignment, the second-stage autoregressive modeling, masked reconstruction, and token aggregation.

**Feature alignment analysis.** Table 5 systematically evaluates different alignment objectives on N-ImageNet. Among the individual objectives, *Cosine + InfoNCE* achieves the best trade-off between compactness and transferability, yielding a +31.6% N-ImageNet accuracy improvement. For the attention-map alignment loss, we match the final-layer attention distributions of the two encoders using an MSE objective. Combining all objectives slightly reduces transfer, suggesting that overly mixed signals may hinder convergence.

Overall, these results confirm that our multi-objective alignment design effectively improves semantic consistency between event and image modalities.

**Training paradigm analysis.** Removing image-to-event

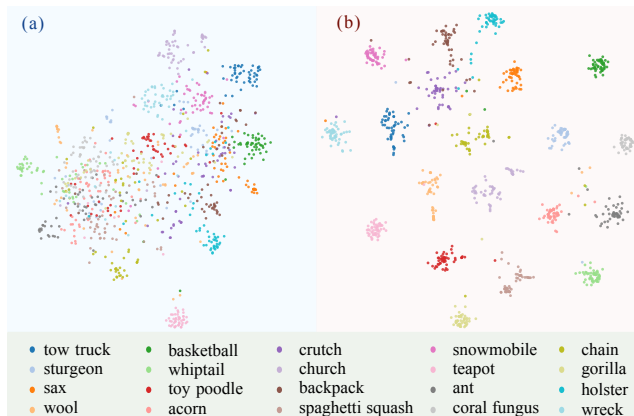


Figure 6. t-SNE visualization of ViT-Base encoder features (20 classes, 50 samples/class from N-ImageNet validation set). Left (a): *Before Align*; right (b): *After Align*. Each color denotes a semantic class.

Table 6. Ablation on alignment and training paradigm (frozen backbone).

Variant	Align	Parad.	DSEC (mIoU / mAcc)
<b>Ours (2 stages)</b>	✓	AR	63.27 / <b>73.13</b>
Event-only AR (Stage 2)	✓	AR	<b>63.51</b> / 72.11
Ours - pretrain	✓	None	61.97 / 71.83
Ours - alignment	✗	AR	52.38 / 60.97
Ours - alignment + finetune	✗	AR	58.24 / 66.82
Ours + aggregator [30]	✓	AR	61.24 / 71.53
Masked Modeling + alignment	✓	MAE	62.33 / 71.65

alignment means pretraining on the original DINOv2 feature sequence, which leads to a performance drop on DSEC from 63.27 to 52.38 mIoU, highlighting the importance of alignment. Moreover, the two-stage design outperforms the single-stage variant by +1.30 mIoU, showing that autoregressive pretraining introduces temporal consistency beyond static alignment. In the same setting of our framework, autoregressive (AR) pretraining surpasses masked modeling (MAE) by +0.94 mIoU and +1.48 mAcc, indicating stronger sequence-level reasoning. Even without alignment (*Ours - alignment + finetune*), our autoregressive variant still outperforms state-of-the-art approaches, yielding +5.72 mIoU over ECDP and +1.86 mIoU over ECDDP.

Furthermore, we ablate the downsampling of recent autoregressive video learners and multimodal LLMs, which often cluster spatio-temporal tokens for efficiency [48, 61]. Our controlled pretraining study (*Ours + aggregator*) follows EventGPT [30], which employs a spatio-temporal aggregator that performs independent pooling along the temporal and spatial dimensions and concatenates the results into a compact token sequence. The lower performance achieved by this aggregation demonstrates that aggressive token compression degrades dense prediction, which relies on high spatial fidelity.

## 5. Conclusion

We presented GEP (Generative Event Pretraining), a unified framework that bridges large-scale VFMs and event-based representation learning. By aligning event features with pretrained VFM representations and performing autoregressive sequence modeling on mixed event-image data, our method effectively transfers semantic knowledge from internet-scale image corpora to the event domain while preserving temporal sensitivity.

## Acknowledgment

This work was supported by the European Union’s Horizon Europe Research and Innovation Programme under grant agreement No. 101120732 (AUTOASSESS) and the European Research Council (ERC) under grant agreement No. 864042 (AGILEFLIGHT).

# Supplementary Material for Generative Event Pretraining with Foundation Model Alignment

## 6. Overview

This supplementary document provides additional implementation details and qualitative results that complement the main paper “Generative Event Pretraining with Foundation Model Alignment”. Specifically, Sec. 7 presents extended method details, including training hyperparameters and task-specific settings for semantic segmentation and depth estimation, while Sec. 9 reports additional qualitative results for depth prediction and attention visualization.

## 7. Extended Method Details

### 7.1. Alignment hyperparameters

During the alignment stage, we fine-tune the pretrained backbone with a lightweight optimization setup. We use the AdamW [32] optimizer with a batch size of 32 and an input resolution of  $224 \times 224$ . The base learning rate is set to  $3 \times 10^{-5}$  without weight decay, and we employ a linear warmup for the first 100 steps followed by a cosine decay schedule down to a minimum learning rate of 0. Training is performed for  $2.5 \times 10^5$  steps.

To improve robustness, we apply moderate data augmentation, including polarity swap and horizontal flipping, each with probability 0.5. In addition, we use random resized cropping with scale range [0.25, 1.0] and a random upscale operation (scale factor 2) with probability 0.1. All alignment hyperparameters are summarized in Table 7.

### 7.2. Pretraining hyperparameters

For the pretraining stage, we adopt a similar optimization strategy but with settings tailored to longer context and larger-scale training. We use AdamW with a batch size of 8 and a context window size of 4096 (Following GPT2 [44]). The base learning rate is  $5 \times 10^{-4}$  with weight decay of  $1 \times 10^{-5}$ . We apply linear warmup for the first 100 steps and then use a cosine decay schedule towards an asymptote at 0, training for  $2.5 \times 10^5$  steps in total. The full set of pretraining hyperparameters is listed in Table 8.

### 7.3. Semantic Segmentation Settings

For semantic segmentation, we adopt a lightweight decoding head that converts the sequence of visual tokens produced by the backbone into a full-resolution prediction map. The design consists of two main components: a linear

Table 7. Hyperparameters for the alignment stage.

Setting	Value
optimizer	AdamW
Batch size	32
Learning rate	$3 \times 10^{-5}$
Weight decay	0
Minimum learning rate	0
Warmup steps	100
Training steps	$2.5 \times 10^5$
Input resolution	$224 \times 224$
Learning rate schedule	linear warmup cosine decay
<b>Data augmentation</b>	
Polarity swap [58]	prob = 0.5
Horizontal flip	prob = 0.5
Random resized crop	scale = [0.25, 1.00]
Random upscale	prob = 0.1, scale = 2

Table 8. Hyperparameters for the pretraining stage.

Setting	Value
optimizer	AdamW
Batch size	8
Window size	4096
Learning rate	$5 \times 10^{-4}$
Weight decay	$1 \times 10^{-5}$
Minimum learning rate	0
Warmup steps	100
Training steps	$2.5 \times 10^5$
Learning rate schedule	linear warmup cosine decay

patch-wise decoding stage and a shallow convolutional refinement stage. We use cross-entropy loss and dice loss [51] with equal weights.

**Linear Patch-wise Decoding.** Vision Transformers (ViTs) operate on a grid of non-overlapping image patches. Given an input image with spatial resolution  $H \times W$  and a

patch size  $P$ , the encoder produces a sequence of

$$L = \frac{H}{P} \times \frac{W}{P} \quad (9)$$

tokens, each with feature dimension  $D$ . In contrast, dense prediction tasks provide supervision at the pixel level, i.e., one label per location in the original  $H \times W$  grid.

To fully exploit this pixel-wise supervision, we adopt a simple *linear patch-wise decoder*. For each token, we apply a shared linear projection that maps the  $D$ -dimensional feature into  $CP^2$  channels, where  $C$  is the number of classes. By reshaping and rearranging the outputs of all  $L$  tokens, we recover a dense prediction map of shape  $C \times H \times W$ .

Formally, let  $F \in \mathbb{R}^{L \times D}$  denote the token features from the ViT encoder. Our decoder learns a weight matrix

$$W_{\text{dec}} \in \mathbb{R}^{D \times (CP^2)}, \quad (10)$$

and computes

$$\hat{Y} = \text{reshape}(FW_{\text{dec}}) \in \mathbb{R}^{C \times H \times W}, \quad (11)$$

where the reshape operation rearranges the per-token  $CP^2$  outputs into their corresponding  $P \times P$  spatial locations in the original image grid. This linear patch-wise decoding is computationally lightweight and allows us to leverage all pixel-level supervision without resorting to a heavy decoder.

**Convolutional Refinement.** To enhance spatial coherence and refine local details, the reconstructed feature map is further processed by a lightweight convolutional refinement module. This stage applies several  $3 \times 3$  convolutions with nonlinear activations while preserving the original spatial resolution. The module introduces only 0.2% (ViT-Small) and 0.07% (ViT-Base) additional parameters, resulting in negligible impact on overall performance and computational cost, and is mainly employed to improve the visual quality of the predictions.

#### 7.4. Depth Estimation Settings

Following prior work on monocular depth estimation [46, 59], we supervise training with the scale-invariant loss of [12] and the multi-scale scale-invariant gradient matching loss of [23], using fixed loss weights of 1 and 0.25, respectively. The network is optimized to predict normalized log-depth values during training, and these predictions are converted back to metric depth at inference time.

**Decoder backbone.** Given a batch of encoder tokens  $T \in \mathbb{R}^{B \times L \times C}$ , we treat  $B$  as the batch size,  $L$  as the number of tokens per image, and  $C$  as the channel dimension of each token. These tokens lie on a patch grid of size  $H_p \times W_p$ ,

obtained from an input image of spatial resolution  $H \times W$  using a patch size  $p$ , so that  $H_p = H/p$  and  $W_p = W/p$ .

We first apply a linear projection to each token and reshape the result into a low-resolution feature map  $F_0 \in \mathbb{R}^{B \times C_0 \times H_p \times W_p}$ , where  $C_0$  is the number of feature channels used in the decoder. On top of  $F_0$ , we build a convolutional decoder with  $S$  stages, indexed by  $s = 1, \dots, S$ . Each stage is implemented as a residual-like block, denoted  $\text{DecBlock}_s$ , that consists of two  $3 \times 3$  convolutions with GELU activations and an identity skip connection. The input to stage  $s$  is  $F_{s-1}$  and the output is  $F_s$ .

Each decoder stage has a target spatial resolution  $(H_s, W_s)$ . These resolutions gradually increase from the patch resolution  $(H_p, W_p)$  at the first stage up to the full image resolution  $(H, W)$  at the last stage. If the output  $F_s$  of stage  $s$  does not match  $(H_s, W_s)$ , we use a bilinear interpolation operator, denoted  $\text{Interp}(\cdot; H_s, W_s)$ , to upsample it to the target size and obtain a feature map  $\tilde{F}_s$  at resolution  $(H_s, W_s)$ .

To aggregate information across scales, all stage outputs are first brought to full resolution  $(H, W)$  using the same interpolation operator and then concatenated along the channel dimension. This yields a fused feature map  $F_{\text{fuse}} \in \mathbb{R}^{B \times C_{\text{fuse}} \times H \times W}$ , where  $C_{\text{fuse}}$  is the total number of channels after concatenation.

Finally, a small convolutional prediction head is applied to  $F_{\text{fuse}}$ . This head consists of two  $3 \times 3$  convolutions with GELU activations, followed by a  $1 \times 1$  convolution and a sigmoid function. The output is a normalized log-depth map  $Y_{\text{norm}} \in [0, 1]^{B \times 1 \times H \times W}$ , where the single channel corresponds to normalized log-depth for each pixel.

**Log-depth parameterization.** We convert the normalized log-depth  $Y_{\text{norm}}$  into metric depth using a fixed minimum and maximum depth range. Let  $d_{\text{min}} > 0$  and  $d_{\text{max}} > d_{\text{min}}$  denote, respectively, the minimum and maximum metric depth considered during training. We define their logarithms  $\ell_{\text{min}} = \log d_{\text{min}}$  and  $\ell_{\text{max}} = \log d_{\text{max}}$ , and the log-range  $\Delta\ell = \ell_{\text{max}} - \ell_{\text{min}}$ . For each pixel, the normalized log-depth prediction  $Y_{\text{norm}}$  is first mapped to the corresponding log-depth value  $\ell = Y_{\text{norm}} \Delta\ell + \ell_{\text{min}}$ , and the final metric depth prediction is obtained as

$$D = \exp(\ell) \in \mathbb{R}^{B \times 1 \times H \times W} \quad (12)$$

Thus,  $D$  contains the predicted depth (in meters) at each pixel. This log-depth parameterization improves numerical stability over large depth ranges and makes the predictions more robust to global scale shifts, since uniform shifts in log-depth correspond to multiplicative changes in metric depth.

**Multi-scale log-depth gradient loss.** To further regularize the spatial structure of the prediction, we match log-depth gradients at multiple resolutions. We consider a

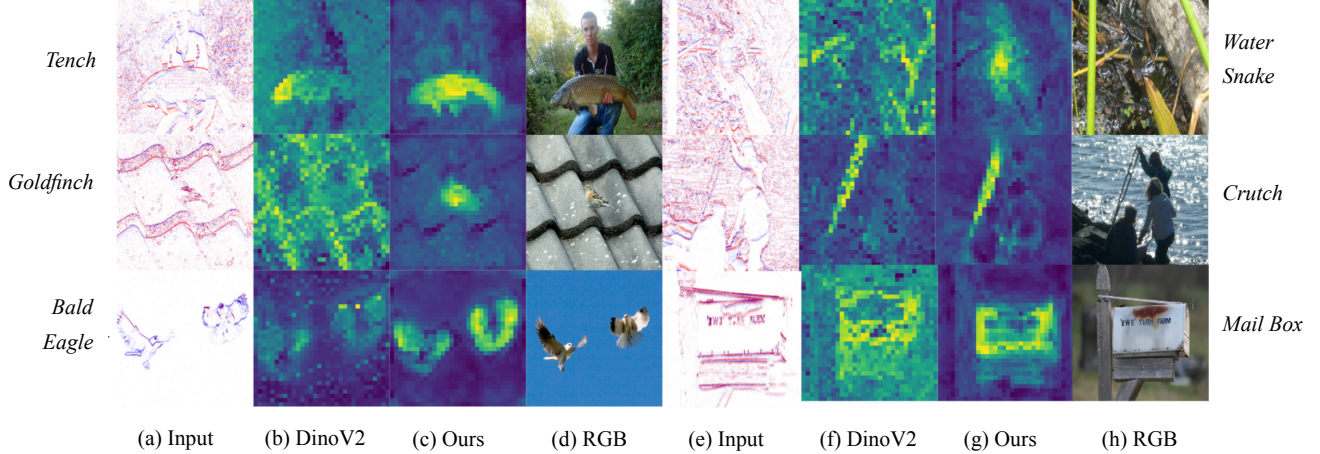


Figure 7. Visual comparison of attention maps on N-ImageNet [25] validation event inputs. We compare the attention responses of DINOv2 [39] (b, f) and our method (c, g) given input events (a, e), alongside reference RGB images (d, h). Our approach demonstrates more precise localization of semantic structures compared to the diffuse activations of DINOv2.

set of scales  $\mathcal{S}$  (e.g., different downsampling factors). For each scale  $s \in \mathcal{S}$ , we downsample the predicted depth, the ground-truth depth, and the validity mask with a fixed operator  $\text{Down}_s(\cdot)$ :

$$D^{(s)} = \text{Down}_s(D), \quad (13)$$

$$(D^*)^{(s)} = \text{Down}_s(D^*), \quad (14)$$

$$M^{(s)} = \mathbf{1}\{\text{Down}_s(M) > 0.5\}. \quad (15)$$

**Training loss.** Supervision is provided by LiDAR depth measurements  $D^* \in \mathbb{R}^{B \times 1 \times H \times W}$  and a binary validity mask  $M \in \{0, 1\}^{B \times 1 \times H \times W}$ , where  $M_i = 1$  indicates that a valid ground-truth depth value is available at pixel  $i$  and  $M_i = 0$  otherwise. All losses are computed only on pixels with  $M_i = 1$ .

To concisely express masked averages, we define a masked mean operator. Given any quantity  $X$  defined per pixel (and per batch element), and a corresponding binary mask  $M$ , we write

$$\langle X \rangle_M = \frac{\sum_i M_i X_i}{\sum_i M_i + \varepsilon}, \quad (16)$$

where  $i$  indexes all spatial positions and batch elements, and  $\varepsilon$  is a small constant used to avoid division by zero when no valid pixels are present.

**Scale-invariant log RMSE.** Following Eigen et al [12], we measure the discrepancy between predicted and ground-truth depths in log space. For each valid pixel  $i$ , we define the log-depth residual

$$y_i = \log D_i - \log D_i^*,$$

where  $D_i$  and  $D_i^*$  denote the predicted and ground-truth depth, respectively. Using the masked mean operator in Eq. 16, the scale-invariant log-depth loss is

$$L_{\text{silog}} = \sqrt{\langle y_i^2 \rangle_M - \lambda \langle y_i \rangle_M^2}. \quad (17)$$

Here,  $\lambda \in [0, 1]$  is a scalar hyperparameter that controls the degree of scale invariance; in our experiments we use  $\lambda = 0.85$ . The first term encourages small squared log-errors, while the second term subtracts a global bias term, making the loss invariant (up to  $\lambda$ ) to additive shifts in log-depth, i.e., multiplicative changes in depth scale.

At scale  $s$ , we compute horizontal and vertical log-depth gradient residuals using finite differences:

$$G_x^{(s)} = \nabla_x \log D^{(s)} - \nabla_x \log (D^*)^{(s)}, \quad (18)$$

$$G_y^{(s)} = \nabla_y \log D^{(s)} - \nabla_y \log (D^*)^{(s)}. \quad (19)$$

The corresponding masked mean absolute errors are

$$E_x^{(s)} = \langle |G_x^{(s)}| \rangle_{M^{(s)}}, \quad (20)$$

$$E_y^{(s)} = \langle |G_y^{(s)}| \rangle_{M^{(s)}}, \quad (21)$$

where  $\langle \cdot \rangle_{M^{(s)}}$  denotes the masked mean defined analogously to Eq. (16), but using the mask  $M^{(s)}$ .

The gradient loss at scale  $s$  and the final multi-scale gradient loss are

$$L_{\text{grad}}^{(s)} = E_x^{(s)} + E_y^{(s)}, \quad (22)$$

$$L_{\text{ms-grad}} = \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} L_{\text{grad}}^{(s)}, \quad (23)$$

where  $|\mathcal{S}|$  is the number of scales in  $\mathcal{S}$ .

**Final objective.** The total depth training objective is a weighted sum of the scale-invariant log RMSE and the multi-scale gradient loss:

$$L_{\text{total}} = w_{\text{silog}} L_{\text{silog}} + w_{\text{ms.grad}} L_{\text{ms.grad}}, \quad (24)$$

where  $w_{\text{silog}}$  and  $w_{\text{ms.grad}}$  are non-negative scalar hyperparameters that balance the two terms. Gradients are back-propagated only through pixels with  $M_i = 1$  (and their downsampled counterparts), so invalid or missing depth values do not influence learning.

## 8. Computational Efficiency and Temporal Analysis

### 8.1. Computational Efficiency

As quantified in Tabs. 9 and 10, GEP achieves  $10\times$  and  $\sim 65\times$  efficiency gains in pretraining and finetuning respectively with superior performance. The GPU hours are computed with a single RTX8000.

Table 9. **Pre-training Efficiency & Data.** Comparison of backbone architectures, pre-training paradigm, resolution, compute cost, data scale, and peak memory usage.

Method	Backbone	Pretrain Mode	Pretrain Res.	Pretrain $EFLOPs$ (h)	Pretrain Data Size	Peak GPU Mem.
TESPEC [37]	Swin-T	Unsup.	360×640	31.7 (540h)	~3.5 TB	-
STP [28]	Swin-T	Sup.	224×224	-	<b>0.57 TB</b>	-
<b>GEP (Ours)</b>	ViT-S	Unsup.	224×224	<b>3.2 (54h)</b>	0.94 TB	<b>17G</b>

Table 10. Segmentation Performance & Cost.

Method	Head	Trainable Param.	Finetune (h)	DSEC Seg. mIoU	DSEC Seg. mAcc
TESPEC [37]	UperNet	45M	2.7 (46h)	62.77	70.61
STP [28]	UperNet	42M	-	62.05	-
<b>GEP (ours)</b>	Linear	<b>22M</b>	<b>0.04 (0.7h)</b>	<b>65.22</b>	<b>74.66</b>

### 8.2. Temporal Resolution Analysis

Our approach unlocks VFM semantics *without* compromising temporal fidelity. As empirically proven in Tab. 11, our model supports *flexible time sampling*: it generalizes robustly to unseen high-temporal resolutions ( $N = 10$  bins), demonstrating the capability to flexibly adapt the integration window  $\Delta t$  to extreme motion speeds on the fly. Furthermore, it is essentially representation-agnostic and can seamlessly adopt inputs like Voxel Grids.

## 9. More Qualitative Results

### 9.1. Depth Estimation

Figure 8 presents qualitative depth estimation results on several outdoor driving scenes. Compared to the ECDDP baseline, our method recovers a smoother road surfaces and

Table 11. **Temporal Resolution Ablation.** All experiments are transferred based on 10% schedule pretrained pilot runs.

Pre-training Strategy (Bins per RGB interval)	Eval. Resolution (Bins per RGB interval)	DSEC Seg.	
		mIoU	mAcc
Fixed ( $N = 1$ ) (Ours)	$N = 1$ (Ours)	42.98	53.09
	$N = 10$ (Unseen)	42.25	51.39
Dynamic ( $N \in \{1, 3, 5\}$ )	$N = 1$ (Ours)	42.67	52.02
	$N = 10$ (Random)	21.31	28.58

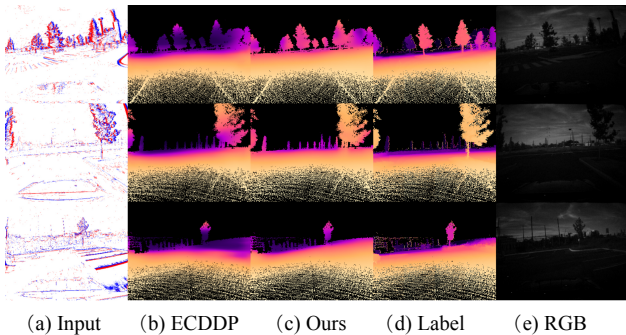


Figure 8. Qualitative comparison of depth estimation on MVSEC [63]. From left to right: (a) event-based input, (b) ECDDP baseline, (c) our method, (d) ground-truth depth, and (e) corresponding reference RGB images (not used). Our model produces depth maps that more closely match the ground truth, with sharper object boundaries and fewer artifacts on thin structures such as trees and poles.

sharper discontinuities at object boundaries, while preserving thin structures such as trees and poles. The predicted depth maps align better with the LiDAR ground truth, illustrating the benefit of our representation for event-based depth estimation.

### 9.2. Attention Map

In addition to depth estimation, we further analyze the discriminative power of our learned features by visualizing attention maps in Figure 7. For each sample, we show the input event frame, the attention responses produced by DINOv2 [39] (our teacher VFM during alignment) and our method, and the corresponding RGB image. As observed in the figure, our approach focuses more accurately on semantically meaningful structures such as object contours, fine edges, and distinctive regions. In contrast, DINOv2 exhibits more diffuse or noisy activations and often fails to localize these structures precisely. These qualitative results indicate that our representation extracts more discriminative features from event data and aligns better with the underlying scene semantics.

## References

- [1] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021. 5, 6
- [2] Luca Bartolomei, Enrico Mannocci, Fabio Tosi, Matteo Poggi, and Stefano Mattoccia. Depth anyevent: A cross-modal distillation paradigm for event-based monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19669–19678, 2025. 2
- [3] Jonathan Binas, Daniel Neil, Shih-Chii Liu, and Tobi Delbruck. Ddd17: End-to-end davis driving dataset. *arXiv preprint arXiv:1711.01458*, 2017. 2, 5
- [4] Christian Brandli, Raphael Berner, Minhao Yang, Shih-Chii Liu, and Tobi Delbruck. A  $240 \times 180 \times 130$  db  $3 \mu\text{s}$  latency global shutter spatiotemporal vision sensor. *IEEE Journal of Solid-State Circuits*, 49(10):2333–2341, 2014. 5
- [5] Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974. 7
- [6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PmlR, 2020. 4
- [7] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9640–9649, 2021. 2, 5
- [8] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. *arXiv preprint arXiv:2309.16588*, 2023. 7
- [9] David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, pages 224–227, 2009. 7
- [10] 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 conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5
- [11] Alexey Dosovitskiy. An image is worth  $16 \times 16$  words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 5, 6
- [12] David Eigen, Christian Puhusch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014. 10, 11
- [13] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19358–19369, 2023. 1
- [14] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004. 5
- [15] Guillermo Gallego, Tobi Delbrück, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J Davison, Jörg Conradt, Kostas Daniilidis, et al. Event-based vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(1):154–180, 2020. 1
- [16] Daniel Gehrig, Antonio Loquercio, Konstantinos G Derpanis, and Davide Scaramuzza. End-to-end learning of representations for asynchronous event-based data. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5633–5643, 2019. 5
- [17] Daniel Gehrig, Michelle Rüegg, Mathias Gehrig, Javier Hidalgo-Carrió, and Davide Scaramuzza. Combining events and frames using recurrent asynchronous multimodal networks for monocular depth prediction. *IEEE Robotics and Automation Letters*, 6(2):2822–2829, 2021. 2, 5, 7
- [18] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. Dsec: A stereo event camera dataset for driving scenarios. *IEEE Robotics and Automation Letters*, 6(3):4947–4954, 2021. 2, 5
- [19] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021. 3
- [20] Ryuhei Hamaguchi, Yasutaka Furukawa, Masaki Onishi, and Ken Sakurada. Hierarchical neural memory network for low latency event processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22867–22876, 2023. 6
- [21] Haiqing Hao, Nikola Zubić, Weihua He, Zhipeng Sui, Davide Scaramuzza, and Wenhui Wang. Maximizing asynchronicity in event-based neural networks. *arXiv preprint arXiv:2505.11165*, 2025. 3
- [22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 2, 5, 6
- [23] Javier Hidalgo-Carrió, Daniel Gehrig, and Davide Scaramuzza. Learning monocular dense depth from events. In *2020 International Conference on 3D Vision (3DV)*, pages 534–542. IEEE, 2020. 10
- [24] Yuhuang Hu, Tobi Delbruck, and Shih-Chii Liu. Learning to exploit multiple vision modalities by using grafted networks. In *European Conference on Computer Vision*, pages 85–101. Springer, 2020. 2
- [25] 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*, pages 2146–2156, 2021. 2, 5, 11
- [26] Simon Klenk, David Bonello, Lukas Koestler, Nikita Araslanov, and Daniel Cremers. Masked event modeling: Self-supervised pretraining for event cameras. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2378–2388, 2024. 1, 2, 5, 6, 7
- [27] Pengteng Li, Yunfan Lu, Pinghao Song, Wuyang Li, Huizai Yao, and Hui Xiong. Eventvl: Understand event streams via multimodal large language model. *arXiv preprint arXiv:2501.13707*, 2025. 3

- [28] Quanmin Liang, Qiang Li, Shuai Liu, Xinzi Cao, Jinyi Lu, Feidiao Yang, Wei Zhang, Kai Huang, and Yonghong Tian. Efficient event camera data pretraining with adaptive prompt fusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8656–8667, 2025. 5, 6, 12
- [29] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128×128 120 db 15 us latency asynchronous temporal contrast vision sensor. *IEEE Journal of Solid-State Circuits*, 43(2):566–576, 2008. 1
- [30] Shaoyu Liu, Jianing Li, Guanghui Zhao, Yunjian Zhang, Xin Meng, Fei Richard Yu, Xiangyang Ji, and Ming Li. Eventgpt: Event stream understanding with multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29139–29149, 2025. 3, 8
- [31] Yifei Liu, Mathias Gehrig, Nico Messikommer, Marco Cannici, and Davide Scaramuzza. Revisiting token pruning for object detection and instance segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2658–2668, 2024. 3
- [32] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 9
- [33] Nico Messikommer, Daniel Gehrig, Mathias Gehrig, and Davide Scaramuzza. Bridging the gap between events and frames through unsupervised domain adaptation. *IEEE Robotics and Automation Letters*, 7(2):3515–3522, 2022. 1, 2
- [34] Nico Messikommer, Jiaxu Xing, Elie Aljalbout, and Davide Scaramuzza. Student-informed teacher training. *International Conference on Learning Representations*, 2025. 2
- [35] Nico Messikommer, Jiaxu Xing, Leonard Bauersfeld, Marco Cannici, Elie Aljalbout, and Davide Scaramuzza. Approximate imitation learning for event-based quadrotor flight in cluttered environments. *arXiv preprint arXiv:2603.07578*, 2026. 1
- [36] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)*. Ieee, 2016. 6
- [37] Mohammad Mohammadi, Ziyi Wu, and Igor Gilitschenski. Tespec: Temporally-enhanced self-supervised pretraining for event cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7782–7793, 2025. 6, 12
- [38] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 2
- [39] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1, 2, 4, 5, 6, 11, 12
- [40] Garrick Orchard, Ajinkya Jayawant, Gregory K Cohen, and Nitish Thakor. Converting static image datasets to spiking neuromorphic datasets using saccades. *Frontiers in neuroscience*, 9:437, 2015. 2, 5
- [41] Bo Peng, Daniel Goldstein, Quentin Anthony, Alon Albalak, Eric Alcaide, Stella Biderman, Eugene Cheah, Xingjian Du, Teddy Ferdinan, Haowen Hou, et al. Eagle and finch: Rwkv with matrix-valued states and dynamic recurrence. *arXiv preprint arXiv:2404.05892*, 2024. 3
- [42] Etienne Perot, Pierre De Tournemire, Davide Nitti, Jonathan Masci, and Amos Sironi. Learning to detect objects with a 1 megapixel event camera. *Advances in Neural Information Processing Systems*, 33:16639–16652, 2020. 6
- [43] Haotong Qin, Cheng Hu, and Michele Magno. Event-prior-based vision-language model for efficient visual understanding. In *International Joint Conference on Artificial Intelligence*, pages 16–30. Springer, 2025. 3
- [44] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 9
- [45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 1
- [46] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021. 10
- [47] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE transactions on pattern analysis and machine intelligence*, 43(6):1964–1980, 2019. 1
- [48] Sucheng Ren, Hongru Zhu, Chen Wei, Yijiang Li, Alan Yuille, and Cihang Xie. Arvideo: Autoregressive pretraining for self-supervised video representation learning. *arXiv preprint arXiv:2405.15160*, 2024. 3, 8
- [49] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987. 7
- [50] Alberto Sabater, Luis Montesano, and Ana C Murillo. Event transformer. a sparse-aware solution for efficient event data processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2677–2686, 2022. 3
- [51] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *International Workshop on Deep Learning in Medical Image Analysis*, pages 240–248. Springer, 2017. 9
- [52] Zhaoning Sun, Nico Messikommer, Daniel Gehrig, and Davide Scaramuzza. Ess: Learning event-based semantic segmentation from still images. In *European Conference on Computer Vision*, pages 341–357. Springer, 2022. 1, 2, 5, 6
- [53] Quan Tang, Bowen Zhang, Jiajun Liu, Fagui Liu, and Yifan Liu. Dynamic token pruning in plain vision transformers for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 777–786, 2023. 3
- [54] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin

- Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. [3](#)
- [55] Ziyi Wu, Xudong Liu, and Igor Gilitschenski. Eventclip: Adapting clip for event-based object recognition. *arXiv preprint arXiv:2306.06354*, 2023. [2](#)
- [56] Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434, 2018. [6](#)
- [57] Chuyun Xie, Wei Gao, and Ren Guo. Cross-modal learning for event-based semantic segmentation via attention soft alignment. *IEEE Robotics and Automation Letters*, 9(3): 2359–2366, 2024. [2](#)
- [58] Yan Yang, Liyuan Pan, and Liu Liu. Event camera data pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2023. [1](#), [2](#), [5](#), [6](#), [9](#)
- [59] Yan Yang, Liyuan Pan, and Liu Liu. Event camera data dense pre-training. In *European Conference on Computer Vision*, pages 292–310. Springer, 2024. [1](#), [2](#), [5](#), [6](#), [7](#), [10](#)
- [60] Xiaoyu Yue, Shuyang Sun, Zhanghui Kuang, Meng Wei, Philip HS Torr, Wayne Zhang, and Dahua Lin. Vision transformer with progressive sampling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 387–396, 2021. [6](#)
- [61] Boqiang Zhang, Kehan Li, Zesen Cheng, Zhiqiang Hu, Yuqian Yuan, Guanzheng Chen, Sicong Leng, Yuming Jiang, Hang Zhang, Xin Li, et al. Videollama 3: Frontier multi-modal foundation models for image and video understanding. *arXiv preprint arXiv:2501.13106*, 2025. [3](#), [8](#)
- [62] Jiazhou Zhou, Xu Zheng, Yuanhuiyi Lyu, and Lin Wang. Eventbind: Learning a unified representation to bind them all for event-based open-world understanding. In *European Conference on Computer Vision*, pages 477–494. Springer, 2024. [3](#), [5](#)
- [63] 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. *IEEE Robotics and Automation Letters*, 3(3):2032–2039, 2018. [2](#), [5](#), [12](#)
- [64] Nikola Zubic, Mathias Gehrig, and Davide Scaramuzza. State space models for event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5819–5828, 2024. [3](#)