

EVEE: Event-Based Online Adaptation for Matching on Unknown Targets

Zejing Zhao^{1*}, Cheng Ju¹, Yanwen Zhang¹, and Akio Namiki¹

Robotics Lab.(Namiki Lab.), Chiba University, Japan
<https://www.em.eng.chiba-u.jp/~namiki/>
zejingzh@gmail.com

Abstract. Offline-trained detectors and matchers often degrade in deployment when encountering previously unknown targets, since offline training cannot anticipate the appearance and structural variations of unknown targets. We propose EVEE, a feature adaptation framework that leverages temporally accumulated event evidence as test-time proxy supervision for adapting detection and matching to previously unseen targets. Rather than performing fully asynchronous event-level matching or using events merely as auxiliary inputs, EVEE constructs event-derived proxy supervision to drive label-free online updates. EVEE contains three components. (i) Training Warehouse (TraW) accelerates convergence within a limited observation window via multi-threaded updates and best-weight broadcasting. (ii) Event Voting Bank (EVB) accumulates bidirectional positive/negative votes to separate foreground from background, thereby reducing the impact of background noise on the model. (iii) ReWeight improves correspondence reliability by rescoring candidates using appearance similarity, geometric consistency, and EVB priors. Experiments on public datasets and a real RGB–event platform, show that EVEE achieves 1.9% higher average pose ACC/AUC on unknown targets than strong baselines. Meanwhile, it runs at 51.3 FPS for matching and 14.52 FPS for online adaptation. The code is publicly available at <https://github.com/ZeJZhao/EVEE>.

Keywords: Event-camera · Robotics-vision · Computer-vision · Online proxy supervision

1 Introduction

Local feature detection and matching are core components of many vision systems, including face recognition and autonomous driving [33, 34]. On the detection side, SuperPoint [7] has become a widely used backbone; on the matching side, LightGlue [15] significantly improves correspondence quality with attention-based reasoning. Despite this progress, state-of-the-art local feature pipelines still struggle with two practical issues: (i) limited generalization to objects unseen during training, and (ii) substantial offline training cost. This is particularly

* Corresponding author.

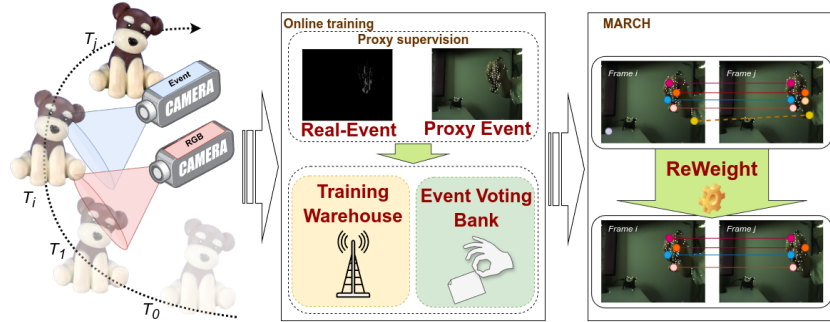


Fig. 1: Overview of EVEC. EVEC takes synchronized RGB and event-camera observations as input. Event-derived proxy supervision guides rapid target-specific adaptation of the RGB keypoint detector with Training Warehouse and Event Voting Bank, producing candidate correspondences that are refined by ReWeight and dual-softmax filtering into high-quality matches for targets unseen during pre-training.

problematic for robotics applications such as 3D reconstruction and 6-DoF pose tracking, where targets may change frequently and cannot be fully covered by offline datasets. Recent large-scale approaches therefore often rely on heavy offline training on multi-GPU clusters (e.g., Zero-1-to-3 series [5, 16, 17, 29] and OmniGlue [13]), which hinders scalability and leaves little room for fast test-time adaptation. Moreover, most methods keep feature extractors frozen at inference, while event-camera pipelines typically treat events as an auxiliary modality rather than a supervisory signal.

To address these limitations, we propose EVEC (Fig. 1), an event-driven online framework for class-agnostic local feature detection and geometric matching under unknown-target settings. Instead of relying solely on expensive offline training, EVEC turns short-window event activity into online proxy supervision and uses it to adapt a lightweight RGB keypoint head in real time, enabling label-free, target-specific feature learning at deployment. Thus, events are used as supervision for updating the RGB detector rather than as a simple RGB-event feature concatenation or a semantic open-vocabulary prompt.

Deploying unknown-target matching under event streams introduces three practical constraints: rapid convergence within a short observation window, sparse/noisy events, and correspondence outliers under weak texture and fast motion. EVEC addresses them with a three-stage pipeline: offline pre-training learns an RGB-to-event-corner proxy branch without using the deployment target; online adaptation uses Training Warehouse (TraW) for multi-threaded updates and Event Voting Bank (EVB) for temporally decayed foreground and background priors; and online matching applies ReWeight as a lightweight final gate that attenuates candidate pairs in logit space using shared-feature descriptors, geometric cues, and EVB priors. Together, these components convert event-driven supervision into reliable correspondences for pose estimation on previously unseen targets.

Contributions. (1) We propose EVEE, an event-driven online framework for local feature detection and matching under unknown-target settings, which constructs temporally accumulated event-derived proxy supervision to enable label-free, target-specific online adaptation.

(2) We introduce TraW and EVB to support efficient online learning with long-term memory, combining multi-threaded training, weight broadcasting, and bidirectional foreground/background voting with noise suppression.

(3) We use ReWeight as a lightweight final re-scoring step in the matcher. A tiny fixed MLP takes the base match logit, a soft geometric cue, and EVB priors, and outputs a bounded weight in $(0, 1]$ that only downweights unreliable candidate pairs before dual-softmax mutual filtering.

2 Related work

Local feature detection and matching. Classical local features such as SIFT [18], SURF [1], and ORB [25] use hand-crafted detectors and descriptors to encode local structure, and remain standard components in SfM and SLAM pipelines [4, 27]. Recently, learned methods have become dominant: SuperPoint [7] jointly learns keypoint detection and description and is widely adopted as a backbone; follow-ups such as DISK [31] and ALIKED [41], as well as hybrids that combine classical detectors with learned descriptors (e.g., SIFT with HardNet [18, 21]), further improve robustness. On the matching side, SuperGlue [26] formulates feature matching as an attention-based graph neural network on top of keypoints, inspiring many subsequent context-aware matchers.

Event-based vision and feature learning. Event cameras output asynchronous per-pixel brightness changes with microsecond latency and high dynamic range [2, 37], and have been widely used in low-level vision and 6-DoF tracking [24]. Most existing learning-based methods either design hand-crafted operators in the event domain or fuse event representations with frame-based features [8, 9], effectively treating events as an additional input modality.

Beyond fusion, a recent trend is to use events as supervision at scale: dense pre-training directly on event streams improves downstream dense prediction and robustness [37]. Differentiable event-generation/simulation further enables self-supervised training—ranging from Video-to-Events (V2E) and v2e to the 2024 EvINR approach that optimizes an implicit intensity representation via the event model [11, 35]. For geometry and radiance, event-conditioned NeRF [20]/3D Gaussian-splatting [14] pipelines fit rendered brightness changes to observed events for high-speed motion and HDR scenes [6, 36, 39, 40].

EVEE follows the events-as-supervision framework but operates in a different regime: it uses short-window event activity to supervise online adaptation of an RGB keypoint detector for class-agnostic geometric matching, rather than learning a semantic or prompt-based open-vocabulary recognizer. See Section 3 for details.

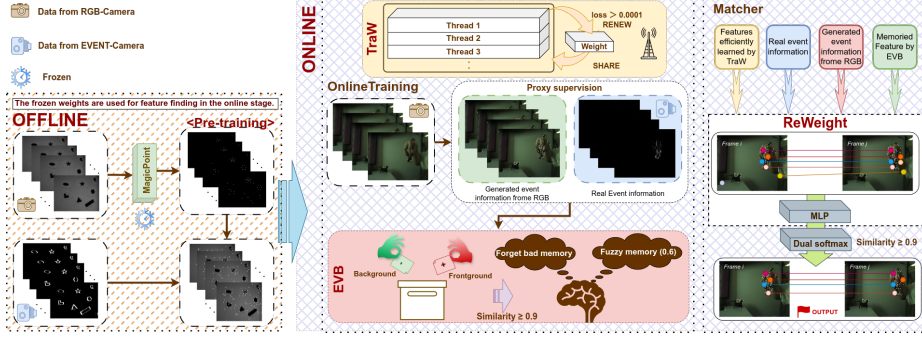


Fig. 2: Structure of EVEC. The pipeline consists of offline event-corner pre-training (left), online adaptation with TraW and EVB (middle), and online matching with EVB priors and ReWeight (right). See Section 3 for details.

3 EVEC

3.1 Model Overview

Fig. 2 overviews EVEC, which treats events as real-time supervision rather than as an auxiliary input for unknown-target matching. Motivated by the lack of target-specific priors at deployment, EVEC constructs event-derived proxy supervision for label-free online adaptation of an RGB keypoint detector.

Offline pre-training. We pre-train a keypoint network to predict event-like corner responses from RGB using event-camera sequences and aligned corner cues (e.g., MagicPoint [7]), enabling event-derived responses from RGB.

Online training. EVEC accumulates RGB-synchronized events within a short window and fuses them with the RGB-derived response to form proxy supervision, then adapts a lightweight keypoint head in real time on top of a frozen backbone. TraW accelerates convergence via multi-threaded updates and best-weight broadcasting, while EVB separately aggregates temporally decayed votes to build a long-term foreground/background confidence map for suppressing background noise.

Online matching. EVEC forms candidate correspondences by combining appearance, geometry, and EVB priors; the TraW-selected best online head θ_H^* provides target-aware responses. ReWeight acts as a tiny standalone 2-layer MLP gate that outputs $w_{ij} \in (0, 1]$; applied as a non-positive logit penalty before dual-softmax mutual selection, it attenuates selected candidate pairs and makes down-weighted matches less competitive.

Pseudocode for online training and matching is provided in the Supplementary Material.

3.2 EVEC Details

Since the shape of unknown objects is not available in advance, we approximate them with basic geometric primitives at the contour level and seek sparse, geo-

metrically stable keypoints rather than densely sampling all edges. Prior work shows that corner-like locations with high repeatability are crucial for accurate matching and pose estimation [19, 28, 30], while points sampled only along long straight edges are often ambiguous and suffer from a local aperture problem [32, 38]. EVEE therefore encourages responses at corners together with their nearby contour structures, instead of producing dense activations along edges. Event cameras provide microsecond temporal resolution and high dynamic range, making them robust to motion blur and illumination changes and thus well suited to capture contours and textures of fast-moving, unknown targets. EVEE is nevertheless not designed as an asynchronous event-by-event matcher: the adapted detector, EVB, and pose solver operate on RGB-synchronized spatial maps from a calibrated RGB-event rig. We therefore use a single normalized event-count/magnitude confidence map for proxy supervision and voting, rather than preserving polarity as two separate channels. However, accumulated event images merely show where brightness changes occur and tend to spread responses along edges and sensor noise, which conflicts with the above sparsity and stability requirements if used directly as keypoint supervision. Conversely, discarding real events and using only proxy event maps predicted from RGB removes these physical advantages and effectively degenerates the supervision into conventional gradient or edge maps that are less robust to blur and lighting changes.

Event-Corner Features (Pre-train) We pre-train the feature detector on the `shapes_6dof` sequence from the Event-Camera Dataset (DAVIS) [22]. As shown in the pre-training module in Fig. 2, we extract corners from RGB images using MagicPoint and associate their responses with aligned event frames to construct “event corners”. We then learn a mapping from RGB appearance to these event corners so that, at inference time, the network can predict from RGB alone a proxy event map that is consistent with real event activity.

Feature detection module. The resulting event corners, together with the RGB frames, are fed into the Feature Finding module (red cuboid in Fig. 2) to compute keypoint features. This module is implemented as a lightweight CNN-based keypoint head. We first apply a Haar-wavelet-based contour extractor, which produces event-like high-frequency responses that emphasize corners and thin contours while suppressing large texture regions, thereby yielding a spatial structure that closely matches event count images at negligible computational cost. The resulting contour features are then processed by a $3 \times 3 \times 3$ Conv3D with d_{model} input and output channels for short-term temporal fusion; this lightweight temporal aggregation over a few neighboring frames stabilizes keypoint responses under fast motion and sensor noise, without resorting to heavy recurrent or Transformer-style modules. Finally, a simple detector consisting of a 3×3 convolution ($d_{\text{model}} \rightarrow 128$), a ReLU activation, a 1×1 convolution ($128 \rightarrow 1$), and a Sigmoid layer produces a single-channel keypoint probability map in $[0, 1]$.

Training Warehouse (TraW) Leveraging the RGB-to-event feature-corner mapping learned offline, we reuse the Feature Detection Module for real-time

Table 1: Notation used in the Event Voting Bank.

Symbol	Description
(H, W)	Height and width of the original RGB/event frame
(H_f, W_f)	Height and width of the feature map used by EVB
(i, j)	Pixel coordinates in the original frame
(p, q)	Spatial coordinates on the $H_f \times W_f$ feature map
$E(i, j)$	Event count / accumulated magnitude at pixel (i, j)
Ω_{pq}	Receptive field in the image for feature location (p, q)
$\tilde{E}(p, q)$	Downsampled event map at feature resolution
$w(p, q)$	Event-confidence map used for EVB voting
$F(p, q) \in \mathbb{R}^C$	Fused RGB feature at location (p, q)
μ^+, m^+	Foreground prototype and its total vote in the current frame
μ^-, m^-	Background prototype and its total vote in the current frame
ρ	Temporal decay factor for votes in the bank

adaptation and initialize its weights from the offline checkpoint. In the online stage, the network predicts an event-derived response from the current RGB frame, fuses it with real event measurements, and treats the fused map as target-related event-derived online proxy supervision to drive online updates. This allows the RGB branch to gradually adapt to the target appearance while retaining the physical robustness of real events to blur and illumination changes.

This online stage is not target-specific pre-use fine-tuning: the target is unknown until deployment, and the updates are performed from the short observation window available at test time. Since online learning must converge within this window, we construct TraW with a multi-threaded parallel training pipeline. In this work, 23 training threads are employed, and each thread sequentially receives video segments composed of RGB frames and their corresponding event-derived online proxy supervision (with a maximum length of 64 frames per thread). To focus updates on informative samples and reduce overfitting to noisy supervision, we introduce a minimum loss threshold $\tau = 0.0001$: only samples whose event-derived online proxy supervised loss remains above τ are used for further updates, while near-converged frames are skipped. The detailed online loss $\mathcal{L} = \mathcal{L}_{\text{det}} + \mathcal{L}_{\text{evt}}$ is provided in Supplementary (Section 1.2). We periodically compare the average loss across threads and treat the lowest-loss checkpoint as the current best model, whose weights are broadcast to all threads for subsequent optimization. Through this “parallel trial-and-error + best-weight broadcasting + loss-thresholding” scheme, a target-specific best checkpoint can be obtained within a limited number of frames. In our implementation, TraW performs about 250 lightweight updates within 20 seconds, compared with about 50 updates using a single training thread. During inference, the Feature Detection Module is frozen at this best checkpoint to produce target-aware responses.

Event Voting Bank (EVB) As discussed in Section 3.2, event cameras provide strong motion cues but also substantial background noise. EVB summarizes these signals as temporally decayed positive/negative prototypes, so that patterns receiving consistent votes across frames are retained while short-lived noise is forgotten. The notation used below is summarized in Table 1.

Event confidence. EVB operates at feature resolution (H_f, W_f) while receiving supervision from the event stream. Let $E(i, j)$ be the event count (or accumulated event magnitude) at pixel (i, j) in the latest event frame, and let $\Omega_{pq} \subset \{1, \dots, H\} \times \{1, \dots, W\}$ be the receptive field of feature location (p, q) . A downsampled event map is obtained by average pooling:

$$\bar{E}(p, q) = \frac{1}{|\Omega_{pq}|} \sum_{(i, j) \in \Omega_{pq}} E(i, j), \quad (1)$$

for $p = 1, \dots, H_f$ and $q = 1, \dots, W_f$. We convert $\bar{E}$ into an event-confidence map

$$w(p, q) = \text{Norm}(\bar{E}(p, q)) \in [0, 1], \quad (2)$$

where $\text{Norm}(\cdot)$ rescales values within each frame. Locations with stronger real-event responses thus receive higher confidence; these real events are later used together with online proxy events.

Positive/negative prototypes. Let the fused RGB feature map at the EVB level be

$$F(p, q) \in \mathbb{R}^C, \quad p = 1, \dots, H_f, \quad q = 1, \dots, W_f. \quad (3)$$

Given $w(p, q)$, EVB aggregates foreground evidence via a confidence-weighted average:

$$\mu^+ = \frac{\sum_{p, q} w(p, q) F(p, q)}{\sum_{p, q} w(p, q)}, \quad m^+ = \sum_{p, q} w(p, q), \quad (4)$$

where μ^+ is the foreground prototype and m^+ the total positive vote; confidence-weighted averaging suppresses abundant low-confidence background pixels, and μ^+ is ℓ_2 -normalized before being inserted into the positive bank to remove event-count scale bias.

Background information is summarized analogously using $1 - w(p, q)$:

$$\begin{aligned} \mu^- &= \frac{\sum_{p, q} (1 - w(p, q)) F(p, q)}{\sum_{p, q} (1 - w(p, q))}, \\ m^- &= \sum_{p, q} (1 - w(p, q)). \end{aligned} \quad (5)$$

Before each update, all votes in the bank are multiplied by a decay factor $\rho = 0.6$, so that only patterns that persist in high-confidence (for μ^+) or low-confidence (for μ^-) regions are retained, while noisy, non-recurring events and hot pixels are gradually forgotten. Here the positive/negative votes denote foreground/background evidence, not event polarity. To further preserve sharpness, we reuse the deterministic Haar-wavelet contour extractor from Section 3.2 when computing features at the EVB level. When applying morphological filtering to the EVB confidence map, OPEN removes isolated activations and hot pixels, while CLOSE reconnects fragmented contours before matching.

Matcher On the right of Fig. 2, the Matcher takes (i) the frozen offline backbone to produce a shared feature map, (ii) the TraW-selected best online head θ_H^* to produce target-aware keypoint responses, and (iii) EVB priors as long-term foreground/background memory. Concretely, let the backbone output $F \in \mathbb{R}^{H_f \times W_f \times C}$ and let the online head output the online keypoint map; both are reused during matching.

We form keypoint descriptors by sampling from F at keypoint locations and optionally augmenting them with EVB priors. We build a local KNN graph over keypoints and apply intra-image self-attention and inter-image cross-attention to propagate context and favor geometrically consistent matches. EVB down-weights locations repeatedly voted as background. We then apply a two-stage geometric filter: a loose pixel-displacement gate with stride-level Top- K pre-filtering, followed by epipolar pruning when a reliable fundamental matrix is available.

ReWeight. We apply a lightweight correspondence re-scoring gate implemented as a standalone 2-layer MLP reweighter. Given keypoints i (frame 1) and j (frame 2) at $x_i, x_j \in \mathbb{R}^2$, we compute a base logit $S_{ij}^{(0)}$ from appearance similarity (dot-product of descriptors sampled from the frozen shared feature map at the selected keypoint locations), optionally incorporating EVB priors and a geometric bias, under a loose geometric gate (full formulation in Supplementary 2.5). In the unified pre-reweight logit, the optional epipolar bias is $g_{ij}^{\text{epi}} = -d_{\text{epi}}^2 / (2\sigma_{\text{epi}}^2)$ when the fundamental matrix is reliable, and $g_{ij}^{\text{epi}} = 0$ otherwise. Separately, g_{ij}^{geo} denotes a soft Gaussian pixel-distance cue.

The reweighter is a tiny 2-layer MLP with parameters ϕ , which is fixed during both online adaptation and inference. It is used as a bounded candidate-attenuation heuristic rather than as a learned semantic reliability predictor. For each candidate pair (i, j) , it combines the base logit and auxiliary cues to compute a bounded multiplicative attenuation weight:

$$w_{ij} = \text{MLP}_{\phi}(S_{ij}^{(0)}, g_{ij}^{\text{geo}}, b_i, b_j) \in (0, 1], \quad (6)$$

where $b_i, b_j \in [0, 1]$ are EVB foreground-reliability priors queried at x_i, x_j . The sigmoid output is clamped to $[10^{-3}, 1]$, and the final logit is obtained in logit space:

$$S_{ij} = S_{ij}^{(0)} + \log(w_{ij} + \epsilon), \quad \epsilon = 10^{-6}. \quad (7)$$

Since $w_{ij} \leq 1$, the added term is non-positive up to the numerical ϵ (at most $\log(1 + 10^{-6})$); thus ReWeight only attenuates selected candidate pairs rather than boosting low-scoring matches. This attenuation makes down-weighted correspondences less competitive under the subsequent dual-softmax mutual selection.

Dual-softmax with dustbin and final selection. We compute a symmetric match probability P_{ij} using a dual-softmax scheme with an explicit dustbin option, and then keep only mutual matches with confidence and dustbin-dominance checks. This yields the final set of high-confidence correspondences used for pose estimation.

4 Experiments

We conduct quantitative experiments on public sequential datasets with ground-truth camera poses, including NAVI-MultiView, TUD-L, and the public Event-Camera Dataset. Following the procedure described in the previous section, we compute ACC and pose AUC under three angular thresholds, 5° , 10° , and 20° . Our goal is to verify that, while maintaining faster inference speed, event-driven online learning can achieve comparable or even superior accuracy to strong baselines for feature extraction and matching on previously unseen targets. Our comparisons include representative state-of-the-art methods that exploit event cues and/or RGB information. The evaluated datasets cover diverse challenges, including cross-frame variations, illumination changes, large-scale camera motion (panoramic movement), and real-world public event-camera recordings. In addition, Section 4.4 presents an ablation study in which we re-evaluate EVEE without providing event information to its matcher, and compare it again against competing methods. Finally, we use our in-house multi-camera hardware platform only for qualitative deployment validation.

4.1 Datasets and protocol

NAVI-MultiView [12]. We use the NAVI-MultiView dataset, which is used for feature detection and matching, and construct $\sim 25\text{K}$ image pairs. Since NAVI contains RGB views but no real event stream, we generate RGB-synchronized events using Prophesee’s event-stream simulator [23]. This preserves a common RGB-baseline protocol but also introduces simulator and interpolation artifacts, which we treat as a dataset limitation.

TUD-L [10]. TUD-L is an RGB-D household-object dataset with $\sim 24\text{K}$ frames, suitable for evaluating detection and matching under severe illumination changes in sequential frames. Since TUD-L also does not provide event streams, we generate RGB-synchronized events using the same Prophesee event-stream simulator [23] as NAVI.

Event-camera Dataset [22]. We evaluate EVEE on this public real event-sensor dataset using ACC/AUC metrics; the quantitative results are reported in Table 4.

Real-world multi-camera rig (qualitative). An RGB–event multi-camera rig is further built to demonstrate feasibility and stable real-hardware operation. The setup uses a beam splitter to obtain a common field of view, an event resolution of 456×360 , and RGB-synchronized event windows at 60 FPS. These rig results are used for qualitative deployment validation.

4.2 Task and metrics

Correspondence estimation. We use matched keypoints for geometric verification and pose estimation. For reprojection error, we follow OmniGlue and use two thresholds: errors below $3px$ are correct, while those above $5px$ are incorrect.

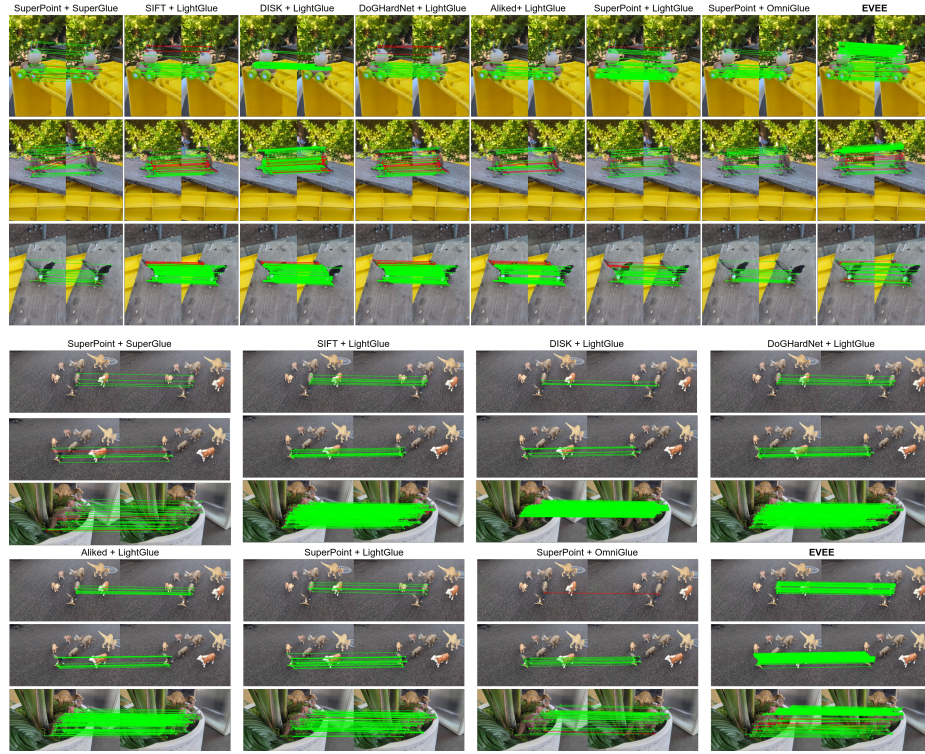


Fig. 3: Comparison of EVEE and baseline methods on the NAVI dataset.

Camera pose estimation. Camera pose estimation is evaluated following the protocol of OmniGlue. ACC denotes the percentage of image pairs whose pose error falls below 5° , 10° , and 20° , while AUC denotes the area under the pose-error curve at the same thresholds. The predicted rotation is converted to an axis-angle representation using Rodrigues’ formula and compared with the ground truth rotation to compute the relative rotation error.

Comparison Baselines We compare against strong baselines for keypoint detection and matching, including SuperGlue and the LightGlue series. We further include SuperEvent (ICCV 2025) [3], a strong baseline that takes event information as input. We report the released SuperEvent+SuperGlue pipeline rather than an unreported SuperEvent+LightGlue variant, because the released SuperEvent pipeline is built from SP+SG pseudo-labels. For all baselines, we use official implementations and default settings. For same-input comparisons, we additionally report Table 3.

Experiment Environment We conduct all experiments on one NVIDIA RTX 3090 GPU and an Intel i9-12900K CPU, using CUDA 12.1 and PyTorch 2.1.2 + cu121.

Table 2: Quantitative comparison with recent feature detection and matching methods on NAVI-MultiView and TUD-L. We report pose ACC and pose AUC at 5°, 10°, and 20°, as well as the average end-to-end runtime from detection to matching.

Method	NAVI						TUD-L						Ave. Result	FPS
	ACC			AUC			ACC			AUC				
	5°	10°	20°	5°	10°	20°	5°	10°	20°	5°	10°	20°		
SP(CVPR'18)+SuperGlue (CVPR'20)	54.3	79.3	87.2	20.9	45.9	65.0	72.7	75.4	78.8	69.9	72.0	74.6	66.3	42.5
SIFT + LG (ICCV'23)	52.9	76.4	87.0	21.8	45.0	63.8	70.7	75.7	82.9	67.2	70.2	74.9	65.7	4.2
DISK (NeurIPS'20) + LG (ICCV'23)	46.1	70.4	81.0	18.1	39.9	58.3	71.8	75.0	79.5	68.5	71.0	74.2	62.8	4.1
DoGHardNet + LG (ICCV'23)	53.2	76.5	87.2	22.0	45.2	64.0	70.3	75.5	82.7	66.8	69.9	74.7	65.7	3.5
ALIKED (CVPR'23) + LG (ICCV'23)	50.3	73.8	84.2	20.8	43.1	61.6	71.5	76.0	82.5	67.9	70.9	75.2	64.8	7.1
SP(CVPR'18) + LG (ICCV'23)	55.1	78.4	88.3	22.5	46.4	65.3	72.0	76.1	81.7	68.7	71.4	75.2	66.8	6.9
OmniGlue (CVPR'24)	49.2	73.8	81.7	18.6	42.1	60.4	74.5	77.0	80.2	71.5	73.7	76.2	64.9	9.1
EVEE	54.6	81.4	89.5	20.3	46.4	66.3	77.0	78.9	81.0	74.8	76.4	78.3	68.7	51.3

Table 3: Accuracy: with and without event input.

Method	Event	ACC			AUC		
		5°	10°	20°	5°	10°	20°
SP+LG (ICCV'23)	×	55.1	78.4	88.3	22.5	46.4	65.3
EVEE	×	51.7	79.9	88.3	18.4	44.4	64.7
SP+LG (ICCV'23)	✓	54.0	77.8	87.5	21.7	45.6	64.6
SE+SG (ICCV'25)	✓	38.4	71.4	85.3	12.2	35.6	57.9
EVEE	✓	54.6	81.4	89.5	20.3	46.4	66.3

4.3 Results

Feature detection and matching performance comparison In feature detection and matching tasks, we quantitatively evaluate EVEE on the widely used NAVI-MultiView and TUD-L benchmarks. Qualitative results are shown in Fig. 3, and quantitative comparisons are reported in Table 2. To isolate the contribution of event cues, we further conduct an input ablation by removing event information from EVEE and by adding the same event-derived masks to SuperPoint+LightGlue (SP+LG). We compare against strong recent RGB baselines and an event-input baseline (SuperEvent [3]+SuperGlue). The results are summarized in Table 3.

Since NAVI-MultiView provides annotated keyframes only every few frames, the viewpoint change between two consecutive annotated frames can be relatively large. Under the strictest metric ACC/AUC@5°, our method is slightly lower than SuperGlue, LightGlue, and OmniGlue in the overall comparison; however, within the 5°–20° range it consistently outperforms prior methods, delivering an ~2.1% gain over the strongest baseline on ACC@10°. Under the same-input comparison (Table 3), removing event measurements causes a notable accuracy drop, because the online update supervision and the long-term foreground/background memory in EVB depend on real event activity to suppress background noise and stabilize target-specific features. Without events, the supervision signal becomes weaker and more prone to drift, which degrades robustness for cross-frame matching and pose estimation.

Interestingly, when event information is provided, SP+LG performs worse than its original RGB-only version, suggesting that simply adding events does

Table 4: Comparison with existing methods on Event-Camera Datasets. Since shapes-6DoF is used for event-corner pre-training, its results are reported as a reference seen-sequence setting; the unseen-sequence comparison is mainly supported by boxes-6DoF and poster-6DoF.

Method	shapes-6DoF						boxes-6DoF						poster-6DoF						AVE. ACC&AUC
	ACC			AUC			ACC			AUC			ACC			AUC			
	5°	10°	20°	5°	10°	20°	5°	10°	20°	5°	10°	20°	5°	10°	20°	5°	10°	20°	
SP+SuperGlue	54.8	62.4	68.1	40.3	49.6	57.7	66.7	75.2	76.9	46.7	59.4	67.9	51.1	58.6	65.0	36.6	46.2	54.1	57.6
SIFT+LG	55.7	60.8	64.8	41.1	50.0	56.6	62.7	71.5	72.5	44.0	56.1	64.2	54.5	62.2	68.3	39.1	48.9	57.0	57.2
DISK+LG	63.4	70.2	74.0	50.7	59.0	65.8	70.5	78.2	79.8	51.2	63.2	71.3	53.7	60.4	66.0	37.7	47.7	55.4	62.1
DoGHardNet+LG	54.1	60.3	64.7	40.1	48.8	55.9	65.0	72.3	74.0	45.7	57.6	65.5	53.1	60.4	66.7	38.5	48.0	55.8	57.0
ALIKED+LG	41.8	47.1	49.6	30.4	37.7	43.2	63.0	70.0	71.5	44.2	55.7	63.4	52.6	58.9	64.8	37.4	46.8	54.6	51.8
SP+LG	56.1	61.1	65.2	41.6	50.3	57.0	71.1	77.6	78.9	51.4	63.3	71.0	53.4	60.3	66.6	38.4	47.8	55.7	59.3
Omniglue	55.5	63.3	68.5	39.7	49.7	58.1	65.4	72.7	75.0	45.7	57.9	66.1	53.1	60.0	65.5	38.0	47.6	55.3	57.6
EVEE	65.5	73.5	77.1	47.2	58.9	67.3	71.1	80.6	85.4	49.7	63.3	73.5	62.4	75.6	79.9	42.0	56.2	67.3	66.5

not necessarily improve accuracy. Raw events can be noisy, sparse, and imperfectly aligned with frame descriptors; EVEE structures them through TraW supervision, EVB memory, and conservative matching instead of using them as direct feature concatenation. Moreover, SE+SG yields the lowest accuracy: its ACC@10° is even 10% lower than EVEE. This gap is closely related to their offline pretraining strategies. EVEE follows MagicPoint-style training to learn event-corner-like features from RGB, which improves robustness for cross-frame evaluation. In contrast, SuperEvent explicitly relies on consecutive frames during training: its pseudo-label generation discards pairs with insufficient matches and repeatedly uses $i+1$ as the new reference frame, which biases the training distribution toward high-overlap consecutive pairs and leads to poorer generalization when the frame gap increases and overlap decreases.

As observed in Fig. 3, EVEE tends to produce more complete and coherent feature coverage on target surfaces. For instance, in *bunny_racer*, EVEE yields substantially more reliable matches around the head and ear regions. EVEE is also more robust on small objects: it produces more valid correspondences within limited visible areas and benefits from the contour robustness of event information. Consequently, thin structures with low color contrast (e.g., the triceratops’ horns) are still detected and matched by EVEE, whereas other methods yield few or no valid responses in these regions. This trend is further confirmed on TUD-L (see Supplementary Fig. 1): in the Small-object subset with a nearly pure-white kettle, most baselines fail to extract and match enough reliable keypoints for downstream geometric evaluation.

Beyond accuracy, EVEE is efficient. On NAVI, excluding I/O and visualization overhead, EVEE runs at 14.52 FPS for online learning and 51.31 FPS for online matching.

Quantitative comparison on public real event-camera sequences We further conduct experiments on the Event-Camera Dataset [22] to verify the accuracy gain brought by event cameras in high-speed motion scenarios. We select test sequences with the “increasing speed” and “6DoF” tags, which cover both simple geometric patterns and complex textured backgrounds.

Table 4 reports a quantitative comparison between EVEE and several existing methods on the Event-Camera Dataset. To ensure a same-input comparison, we also use the event stream to construct foreground masks when evaluating the baselines, so that they can suppress background noise as much as possible. As shown in the table, EVEE exhibits consistently strong performance in most settings. The reported average over all three sequences shows a 4.4% gain over the second-best method. Because shapes-6DoF is also used during pre-training, we treat it as a reference seen-sequence result and additionally verify the conclusion on the two unseen sequences, boxes-6DoF and poster-6DoF, where EVEE obtains an average ACC/AUC of 67.2 and remains ahead of the strongest baseline at 61.3, corresponding to an approximately 9.6% relative gain. This demonstrates that EVEE maintains superior high-precision pose accuracy. These observations further corroborate the main results summarized in Table 2.

Moreover, the Event-Camera Dataset is captured with a moving camera, where the scene contains dense event activity and substantial noise. Despite this, EVEE still achieves higher accuracy than competing methods in our experiments, demonstrating its effectiveness under camera motion in background-noisy environments.

Qualitative on-device rig validation The qualitative study in this subsection refers only to our in-house RGB–event rig; the public Event-Camera Dataset is evaluated quantitatively in Table 4. Fig. 4 and 5 in the Supplementary Material visualize the rig results of keypoint extraction and feature matching, respectively. These visualizations indicate that our method is not only effective in dataset-based evaluations, but can also operate stably on a calibrated RGB–event hardware setup. We regard this result as qualitative evidence of deployability rather than a separate quantitative benchmark.

4.4 Ablation

Table 5 first isolates the four main components of EVEE. Removing real events, proxy events, EVB, or online learning consistently reduces ACC/AUC and pair rate, showing that the gains do not come from a single component. Real events localize the active target and narrow the search space, while proxy events provide denser online supervision for the RGB detector; their combination improves matching reliability at the cost of a modest speed drop from 54.49 FPS to 51.31 FPS. EVB and online learning mainly improve the 10° and 20° metrics, indicating that long-term voting memory and target-specific feature updates are especially useful when pose changes are larger.

The Non-Event variant is also informative: even without event input, it remains competitive with strong offline matchers and runs fastest, which confirms that the learned RGB detector and matcher are effective by themselves. Nevertheless, the full model clearly improves all ACC/AUC metrics and achieves the highest pair rate, so event supervision is not merely an auxiliary cue but the source of the final robustness gain. Supplementary Fig. 2 further shows that

Table 5: Ablation of the four main components.

Variant	ACC			AUC			FPS	Pair rate
	5°	10°	20°	5°	10°	20°		
non_EVB	53.2	79.8	87.5	19.4	45.3	64.9	52.33	97.14%
non_Learning	53.9	80.8	89.0	20.0	46.0	65.9	51.86	97.68%
non_Event	51.7	79.9	88.3	18.4	44.4	64.7	54.49	96.83%
non_proxy event	53.1	80.1	87.6	19.5	45.4	65.0	51.39	96.70%
EVVE(full)	54.6	81.4	89.5	20.3	46.4	66.3	51.31	97.88%

Table 6: Fine-grained module ablations on NAVI.

Module	Variant	Setting	ACC (NAVI)			AUC (NAVI)		
			5°	10°	20°	5°	10°	20°
Pre-trained	Non-wavelet	–	53.7	80.6	88.1	19.8	45.8	65.4
	Temporal Fusion	–	49.2	74.9	82.3	17.6	42.2	60.7
TraW	Single-thread	–	53.9	80.4	88.8	19.9	45.8	65.5
	Single-thread	$\tau = 0.0001$	54.4	80.7	88.8	20.1	46.0	65.8
	Multi-threads	–	53.6	80.2	88.9	19.7	45.6	65.5
EVB	Non-wavelet	–	53.6	80.1	88.6	19.9	45.8	65.4
		OPEN×/CLOSE×	54.2	80.4	88.7	20.0	45.9	65.7
	Morphological	OPEN×/CLOSE✓	54.3	81.0	89.1	20.1	46.2	66.1
		OPEN✓/CLOSE×	54.1	80.5	88.8	20.1	45.8	65.6
Matcher	Non-ReWeight	–	53.6	81.2	89.1	20.0	46.1	66.0
	Non-EVB	–	54.1	80.9	89.2	20.0	46.0	66.0
EVVE (full)		–	54.6	81.4	89.5	20.3	46.4	66.3

TraW discovers loss-reducing features more frequently than a single-thread baseline, and Supplementary Fig. 3 reports a non-linear thread-count trend with the best accuracy at 23 threads.

Fine-grained module analysis. Table 6 decomposes the above trend by module. Pair rate denotes the percentage of the 8,636 consecutive NAVI-MultiView frame pairs that pass geometric verification under our pose and inlier thresholds. The full configuration achieves the best ACC/AUC, while each ablation degrades at least one accuracy threshold, confirming that the design choices in the method section are complementary.

For pre-training, removing the Haar-wavelet contour extractor or temporal fusion weakens the event-like corner response; the temporal-fusion removal causes the largest drop, showing that short-term aggregation is important for stable supervision. In TraW, the loss threshold and multi-threaded trial-and-broadcast design preserve or slightly improve accuracy relative to single-thread updates while supporting the runtime reported earlier, which verifies its practicality for fast online convergence.

For EVB, ablating wavelet sharpening or the open–close geometric filtering reduces the gains, especially at 10° and 20° . This supports the role of structure-aware voting in suppressing background noise while preserving foreground contours. Finally, removing ReWeight or EVB priors from the matcher lowers both ACC and AUC, indicating that appearance, geometry, and EVB history should be fused into a compact confidence score for robust match re-scoring.

5 Conclusion

We present EVEE, an event-driven online matcher that leverages event streams as online proxy supervision to adapt local feature detection and matching to previously unseen targets. The framework couples an offline event-corner pretraining stage with an online adaptation pipeline comprising three components: TraW, a multi-threaded update-and-broadcast scheme for rapid convergence; EVB, a temporally accumulated foreground/background memory; and ReWeight, a lightweight matcher that refines correspondences by combining appearance cues, geometric consistency, and EVB priors. Experiments on NAVI-MultiView and TUD-L show robust performance under severe illumination changes, small targets, and weakly textured surfaces, with competitive accuracy against strong RGB and event-based baselines. On Event-Camera datasets, we further demonstrate reliable operation in real robotic environments. Qualitative results on our in-house RGB–event platform additionally suggest that the method is deployable in practical settings. Ablations validate the contribution of each module and consistently confirm that event cues provide an indispensable and positive benefit.

Limitation. Although the approach outperforms SuperEvent in cross-frame evaluation on NAVI-MultiView, it remains challenged by pairs with large view-point changes, where dedicated matchers such as LightGlue and related approaches are still stronger.

Acknowledgements

This work was supported by Japan Science and Technology Agency Support for Pioneering Research Initiated by the Next Generation (JST SPRING) under Grant Number JPMJSP2109.

References

1. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: European conference on computer vision. pp. 404–417. Springer (2006)
2. Bugueno-Cordova, I., Ruiz-del Solar, J., Verschae, R.: Human-robot navigation using event-based cameras and reinforcement learning. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5004–5012 (2025)

3. Burkhhardt, Y., Schaefer, S., Leutenegger, S.: Superevent: Cross-modal learning of event-based keypoint detection for slam. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 8918–8928 (2025)
4. Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., Leonard, J.J.: Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics* **32**(6), 1309–1332 (2017)
5. Chen, Y., Fang, J., Huang, Y., Yi, T., Zhang, X., Xie, L., Wang, X., Dai, W., Xiong, H., Tian, Q.: Cascade-zero123: One image to highly consistent 3d with self-prompted nearby views. In: *European Conference on Computer Vision*. pp. 311–330. Springer (2024)
6. Chen, Z., Lu, Z., Ma, D., Tang, H., Jiang, X., Zheng, Q., Pan, G.: Evhdr-gs: Event-guided hdr video reconstruction with 3d gaussian splatting. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 39, pp. 2367–2375 (2025)
7. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. pp. 224–236 (2018)
8. Devulapally, A., Khan, M.F.F., Advani, S., Narayanan, V.: Multi-modal fusion of event and rgb for monocular depth estimation using a unified transformer-based architecture. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 2081–2089 (2024)
9. Gehrig, D., Rüegg, M., Gehrig, M., Hidalgo-Carrió, J., Scaramuzza, D.: Combining events and frames using recurrent asynchronous multimodal networks for monocular depth prediction. *IEEE Robotics and Automation Letters* **6**(2), 2822–2829 (2021)
10. Hodaň, T., Michel, F., Brachmann, E., Kehl, W., Glent Buch, A., Kraft, D., Drost, B., Vidal, J., Ihrke, S., Zabulis, X., Sahin, C., Manhardt, F., Tombari, F., Kim, T.K., Matas, J., Rother, C.: BOP: Benchmark for 6D object pose estimation. *European Conference on Computer Vision (ECCV)* (2018)
11. Hu, Y., Liu, S.C., Delbruck, T.: v2e: From video frames to realistic dvs events. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 1312–1321 (2021)
12. Jampani, V., Maninis, K.K., Engelhardt, A., Karpur, A., Truong, K., Sargent, K., Popov, S., Araujo, A., Martin Brualla, R., Patel, K., et al.: Navi: Category-agnostic image collections with high-quality 3d shape and pose annotations. *Advances in Neural Information Processing Systems* **36**, 76061–76084 (2023)
13. Jiang, H., Karpur, A., Cao, B., Huang, Q., Araujo, A.: Omniglue: Generalizable feature matching with foundation model guidance. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 19865–19875 (2024)
14. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.* **42**(4), 139–1 (2023)
15. Lindenberger, P., Sarlin, P.E., Pollefeys, M.: Lightglue: Local feature matching at light speed. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 17627–17638 (2023)
16. Liu, M., Shi, R., Chen, L., Zhang, Z., Xu, C., Wei, X., Chen, H., Zeng, C., Gu, J., Su, H.: One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 10072–10083 (2024)
17. Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., Vondrick, C.: Zero-1-to-3: Zero-shot one image to 3d object. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 9298–9309 (2023)

18. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *International journal of computer vision* **60**(2), 91–110 (2004)
19. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. *International journal of computer vision* **60**(1), 63–86 (2004)
20. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM* **65**(1), 99–106 (2021)
21. Mishchuk, A., Mishkin, D., Radenovic, F., Matas, J.: Working hard to know your neighbor’s margins: Local descriptor learning loss. *Advances in neural information processing systems* **30** (2017)
22. Mueggler, E., Rebecq, H., Gallego, G., Delbruck, T., Scaramuzza, D.: The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *The International journal of robotics research* **36**(2), 142–149 (2017)
23. Prophesee: Metavision sdk: Event-based vision software (2025)
24. Ren, H., Zhu, J., Zhou, Y., Fu, H., Huang, Y., Cheng, B.: A simple and effective point-based network for event camera 6-dofs pose relocation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 18112–18121 (2024)
25. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: An efficient alternative to sift or surf. In: *2011 International conference on computer vision*. pp. 2564–2571. Ieee (2011)
26. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 4938–4947 (2020)
27. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 4104–4113 (2016)
28. Shi, J., et al.: Good features to track. In: *1994 Proceedings of IEEE conference on computer vision and pattern recognition*. pp. 593–600. IEEE (1994)
29. Shi, R., Chen, H., Zhang, Z., Liu, M., Xu, C., Wei, X., Chen, L., Zeng, C., Su, H.: Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110* (2023)
30. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: Loftr: Detector-free local feature matching with transformers. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 8922–8931 (2021)
31. Tyszkiewicz, M.J., Fua, P., Trulls, E.: DISK: Learning local features with policy gradient. In: *Advances in Neural Information Processing Systems*. vol. 33, pp. 14254–14265 (2020)
32. Wang, B., Chen, C., Cui, Z., Qin, J., Lu, C.X., Yu, Z., Zhao, P., Dong, Z., Zhu, F., Trigoni, N., et al.: P2-net: Joint description and detection of local features for pixel and point matching. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 16004–16013 (2021)
33. Wang, Y., He, X., Peng, S., Tan, D., Zhou, X.: Efficient loftr: Semi-dense local feature matching with sparse-like speed. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 21666–21675 (2024)
34. Wang, Y., Wei, W.: Local and global feature attention fusion network for face recognition. *Pattern Recognition* **161**, 111227 (2025)
35. Wang, Z., Lu, Y., Wang, L.: Revisit event generation model: Self-supervised learning of event-to-video reconstruction with implicit neural representations. In: *European Conference on Computer Vision*. pp. 321–339. Springer (2024)

36. Xiong, T., Wu, J., He, B., Fermuller, C., Aloimonos, Y., Huang, H., Metzler, C.A.: Event3dgs: Event-based 3d gaussian splatting for high-speed robot egomotion. arXiv preprint arXiv:2406.02972 (2024)
37. Yang, Y., Pan, L., Liu, L.: Event camera data dense pre-training. In: European Conference on Computer Vision. pp. 292–310. Springer (2024)
38. Yuan, D., Burner, L., Wu, J., Liu, M., Chen, J., Aloimonos, Y., Fermüller, C.: Learning normal flow directly from event neighborhoods. arXiv preprint arXiv:2412.11284 (2024)
39. Yura, T., Mirzaei, A., Gilitschenski, I.: Eventsplat: 3d gaussian splatting from moving event cameras for real-time rendering. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26876–26886 (2025)
40. Zahid, S., Rudnev, V., Ilg, E., Golyanik, V.: E-3dgs: Event-based novel view rendering of large-scale scenes using 3d gaussian splatting. arXiv preprint arXiv:2502.10827 (2025)
41. Zhao, X., Wu, X., Chen, W., Chen, P.C., Xu, Q., Li, Z.: Aliked: A lighter keypoint and descriptor extraction network via deformable transformation. *IEEE Transactions on Instrumentation and Measurement* **72**, 1–16 (2023)