M^{2}E-UAV: A Benchmark and Analysis for Onboard Motion-on-Motion Event-Based Tiny UAV Detection

Weiqi Yan[1,2,3]* **Lixin Chen**[1,2,3]* **Xiangrui Hou**[1,2,3] **Zhipeng Cai**[4]†
Youbiao Wang[1,2,3] **Yangyang Shi**[4] **Yu Zang**[1,2,3]† **Cheng Wang**[1,2,3]

[1]Fujian Key Laboratory of Urban Intelligent Sensing and Computing, Xiamen University, Xiamen, China
[2]Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, Xiamen, China
[3]The Belt and Road Research Institute, Xiamen University, Xiamen, China
[4]Meta, Menlo Park, USA

yanweiqi199888@gmail.com zangyu7@126.com

Abstract

Tiny UAV detection from an onboard event camera is difficult when the observer and target move at the same time. In this motion-on-motion regime, ego-motion activates background edges across buildings, vegetation, and horizon structures, while the UAV may appear as a sparse event cluster. We present M^{2}E-UAV, a dataset and benchmark for studying this practical problem. It provides 1280×720 event streams paired with inertial measurements across four illumination–background conditions, with event-level foreground supervision derived from temporally propagated and reviewed bounding-box annotations. The single-target inventory contains 108,325 packets, with temporally separated training, validation, and test partitions. We evaluate frame-, tensor-, and point-based reference methods against shared detection targets defined from complete event packets. Our analysis shows that full-packet spatial context substantially improves the fixed-budget point reference. A staged proposal-recovery analysis improves temporal-test $AP_{50:95}$ from 0.2111 to 0.4921. Simple global IMU conditioning does not improve the matched three-seed baseline. These findings establish M^{2}E-UAV as an empirical foundation for studying sparse target evidence, temporal recovery, and cross-recording generalization under coupled observer and target motion. Project page: https://github.com/Wickyan/M2E-UAV.

1 Introduction

Tiny UAV detection is important for safety, inspection, and collision avoidance, but it remains difficult for onboard perception. Conventional RGB or frame cameras can be affected by fast motion, motion blur, illumination changes, and the limited spatial footprint of distant UAVs. Event cameras are attractive for this regime because they respond asynchronously to brightness changes and capture high-temporal-resolution motion cues [5]. However, onboard motion-on-motion detection (Figure 1) is harder than static-observer detection: the sensing platform and target UAV both move, and ego-motion activates cluttered background edges across buildings, vegetation, and horizon structures.

Related event-based aerial work has considered onboard obstacle sensing and aerial-robot perception [2, 4], but detecting an external tiny UAV from a moving UAV-mounted observer remains a distinct task.

*Equal contribution.
†Corresponding authors.
Meta was not involved in conducting the data collection or experiments discussed in this paper.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

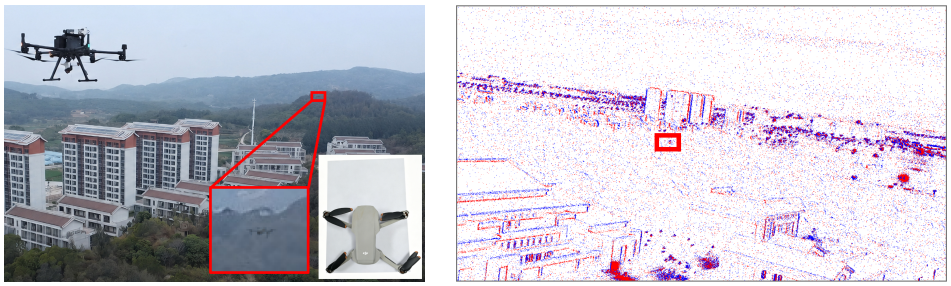

Figure 1: **Onboard motion-on-motion detection.** The left panel shows the carrier UAV observing a small target UAV. The right panel shows an onboard event-camera packet in which observer motion produces dense background activity around sparse target evidence.

We introduce M^{2}E-UAV as a dataset and benchmark for onboard motion-on-motion event-based tiny UAV detection. M^{2}E-UAV provides 1280×720 event data and synchronized event-IMU packets across sunny/sunset and building–forest/farm–village conditions. Bounding boxes initialized at 10 Hz are temporally propagated and manually reviewed to derive event-level foreground supervision. A temporally separated Train/Val/Test protocol supports evaluation on unseen segments, while a recording-held-out protocol examines transfer across acquisition conditions.

Beyond dataset construction, this paper evaluates representative baselines across event representations, including event-frame, event-tensor, and event-point methods. We provide M^{2}E-Point as a point-based event baseline and M^{2}E-Point + IMU as its IMU-conditioned variant. Fixed-budget point sampling can discard the already sparse target evidence before it reaches a model; our analysis examines sampling, proposal recovery, and variation across recordings.

Our contributions are:

- We introduce M^{2}E-UAV, a 1280×720 event–IMU dataset for external tiny-UAV detection from a moving UAV-mounted observer, with packet-level target supervision derived from temporally propagated and manually reviewed annotations.
- We establish a benchmark protocol that specifies the single-target inventory, strict temporal split, event–IMU timing calibration, annotation semantics, and treatment of target-present weak-response packets.
- We provide reference results across event-frame, event-tensor, and event-point approaches, together with a staged point-based analysis that quantifies proposal-recovery gains across temporal and recording-held-out evaluations.

2 Related Work

Event-based object detection. Event-based object detection has been studied through several input representations. A common direction converts events into frame-like or voxel-like tensors and then applies image-style or video-style detection backbones. Recent event-native detectors use recurrent or transformer-based architectures, such as RVT [7] and SAST [16], while LEOD [22] studies label-efficient detection through pseudo-labeling. M^{2}E-UAV studies the setting in which the observer itself is airborne and moving while the target is a distant external UAV.

Aerial event sensing and drone datasets. EED studies independent-motion perception with a quadrotor as a moving object [15]. ODA combines onboard event sensing and inertial measurements for micro-aerial-vehicle obstacle avoidance [4]. M3ED provides aerial-robot recordings with stereo event cameras, an IMU, aggressive motion, and multiple independently moving objects [2]. Other datasets address fast UAV detection, RGB–event drone detection, event-based tiny-object detection, flying-object recognition, and drone perception [3, 6, 11–14]. M^{2}E-UAV complements them with a benchmark specifically defined around external tiny-UAV detection from a moving UAV-mounted event camera, paired event–IMU measurements, packet-level target supervision, and explicit treatment of target-present weak-response packets.

Point-based event modeling and inertial conditioning. Point-set neural networks provide a natural tool for sparse geometric data. DGCNN introduces EdgeConv to capture local geometric relationships in point clouds [21], while KPConv [18] and RandLA-Net [8] provide representative point-cloud segmentation architectures. COSeg [1] is also included as a point-set baseline in our benchmark protocol. FiLM offers a lightweight feature-conditioning mechanism for incorporating auxiliary measurements [17]. These methods provide useful modeling primitives for event packets represented as $[x, y, t, p]$ point sets.

3 Benchmark Setup

3.1 Data acquisition platform and synchronization

Figure 2 summarizes data collection with a DJI Matrice 300 RTK carrier UAV observing a DJI Mini 2 target UAV. A SilkyEvCam HD event camera with an IMX636 sensor and an 8 mm lens, together with an FDI Systems DETA100D4G IMU, is mounted beneath the carrier platform on a vibration-damping plate. The acquisition system includes an STM32-based trigger-capable synchronization board. The recorded sensor clocks exhibit a small event-sensor/host rate mismatch of approximately 23–25 ppm. We therefore align the event and inertial streams using a per-recording affine clock calibration that corrects both offset and relative clock rate. Across the four recordings, a fixed-offset conversion alone would accumulate 10.33–18.19 ms of end-of-recording error, comparable to the approximately 10 ms packet duration.

The corrected mapping is applied per recording before querying the raw IMU stream at the event-window center. Interpolation uses observed inertial support; missing measurements remain explicit in the integrity metadata. The 99th-percentile absolute alignment residuals are approximately 2.35–3.97 ms across recordings after calibration. The calibration values and missing-observation details are given in Supplementary Section A.

3.2 Event and IMU packet construction

Each processed benchmark sample contains an event packet and its temporally aligned IMU vector. The event packet is represented as

$$\mathcal{E} = \{(x_i, y_i, t_i, p_i)\}_{i=1}^{N}, \tag{1}$$

where (x_i, y_i) is the event location, t_i is the timestamp inside the packet, and p_i is polarity. Events remain asynchronous event data before being converted into the representation required by each baseline, such as accumulated event frames, native event tensors, or event point sets. The matched IMU input is a 10-D vector containing four quaternion entries, three angular-velocity entries, and three linear-acceleration entries. The IMU samples are queried from the raw inertial stream on the affine-corrected event timeline. Missing observations are recorded in the dataset metadata.

3.3 Annotation pipeline

UAV boxes are manually initialized at 10 Hz and propagated to the packet-aligned sequence. The annotator reviews and corrects the propagated boxes, and the authors conduct two additional review-and-correction passes over the full dataset. For each event, the time-interpolated reviewed box assigns a positive foreground label when the event coordinate lies inside it. The box-level detection target is the spatial envelope of *all* positive events in the complete packet.

Foreground labels are generated on the complete packet, and the detection target remains the same for frame, tensor, and point representations. At training time, the selected events inherit their packet-aligned foreground labels for point supervision. The dense annotation export and lifting procedure were checked on 80 samples from each of the four recording tasks, with agreement on all 320 checked samples.

3.4 Dataset statistics and temporal split

The processed inventory contains 108,618 approximately 10 ms, non-overlapping packets. Removing a manually identified 293-packet multi-object interval leaves 108,325 single-target packets. Within

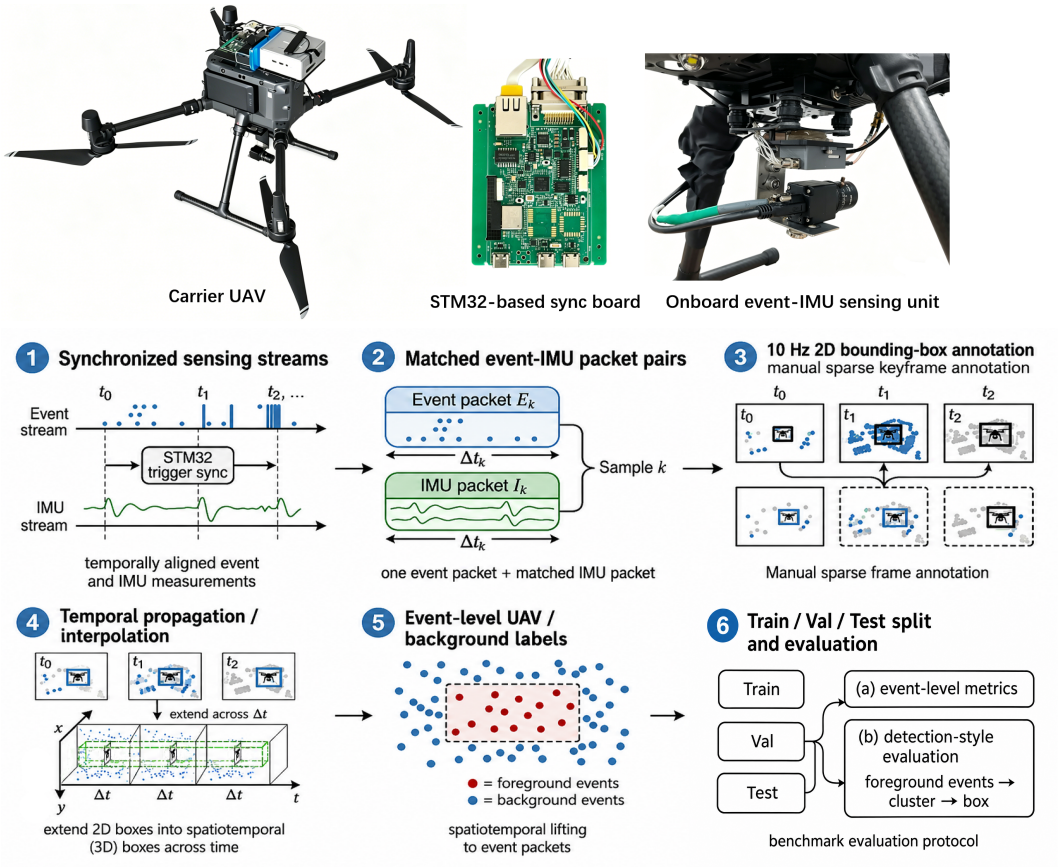

Figure 2: **M^{2}E-UAV data collection and benchmark construction.** The top row shows the carrier UAV and synchronized event–IMU acquisition hardware. The current temporal partition, 5 s exclusion gap, annotation lifting, and weak-response masking are specified in Sections 3.4 and 3.3. The lower panel shows the Train/Val/Test partition and the event-level and detection-style paths. Event labels supervise foreground prediction; reported F1 and AP are box-level detection metrics.

each recording, samples are ordered by raw packet index; the 80th and 90th percentiles define the Train/Val and Val/Test boundaries. A band of 500 raw packet IDs, centered on each boundary, is discarded to provide a nominal 5 s temporal gap. The gap reduces overlap in adjacent observations and temporal context across partitions. Packets in this band are excluded from all three partitions.

Packets with at most seven positive events are marked *weak-response*: the target is present, but its event support is insufficient for direct box supervision. Within Train, Val, and Test, these packets remain as temporal context, with box losses and detection metrics masked. Packets in the temporal exclusion gaps are excluded from both the partitions and their temporal contexts. The valid-box Train/Val/Test populations are 82,038/10,247/10,381 after filtering for weak response and zero-area event support. Supplementary Section A gives the complete inventory and filtering counts.

Four recordings cover the sunny/sunset and building–forest/farm–village combinations (one per combination; Figure 3). Table 1 reports valid-box packets for each recording. The temporal split tests new segments of the same recordings; recording-held-out evaluation excludes an entire recording from training and validation to assess transfer across illumination, background, observer motion, and target trajectory.

3.5 Dataset comparison

Table 2 compares M^{2}E-UAV with representative event-based UAV, flying-object, and tracking datasets. The summarized properties are compiled from the benchmark and task definitions reported in the

Table 1: **Valid-box packets by recording and temporal split.** Counts exclude weak-response and invalid-support packets.

Recording / scene family	Train	Val	Test
A: Sunny building-forest	23,377	2,875	2,903
B: Sunny farm-village	26,063	3,387	3,414
C: Sunset building-forest	13,735	1,663	1,713
D: Sunset farm-village	18,863	2,322	2,351
Total	82,038	10,247	10,381

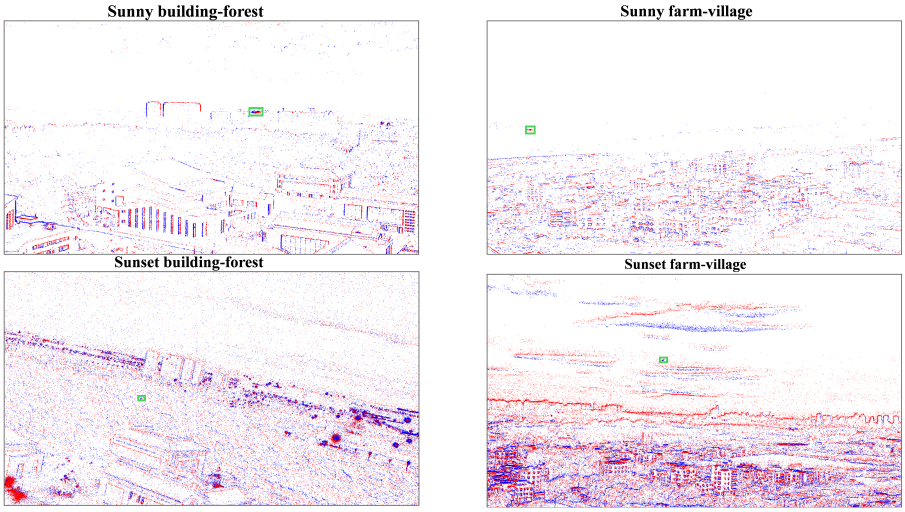

Figure 3: **Scene-family showcase.** The montage covers sunny/sunset illumination and building-forest/farm-village backgrounds.

corresponding papers. The comparison separates sensor resolution from the detection setting: observer motion specifies whether ego-motion-induced background activity accompanies target motion, while tiny-target focus and annotation granularity specify the evidence and supervision available to a detector. Paired inertial measurements additionally support studying how observer motion can inform detection. M^{2}E-UAV combines a moving airborne observer, an external tiny UAV, paired event–IMU streams, and box-derived event supervision, with temporal and recording-held-out evaluation of this setting.

Table 2: **Related event-vision datasets.** Resolution refers to the event sensor. Observer motion and IMU indicate documented camera motion and paired inertial releases, respectively. Tiny target denotes a tiny-UAV task focus. F-UAV-D refers to the detector study's recordings; event labels in EV-UAV and ours are box-derived.

Dataset	Resolution	UAV-centric	Tiny target	Observer motion	IMU	Annotation granularity
EED [15]	240 × 180	×	×	✓	×	2D boxes
VisEvent [19]	346 × 260	×	×	✓	×	Tracking boxes
EventVOT [20]	1280 × 720	×	×	✓	×	Tracking boxes
F-UAV-D [14]	1280 × 720	✓	×	×	×	Frame boxes
EV-UAV [3]	346 × 260	✓	✓	×	×	Box-derived events
EV-Flying [11]	1280 × 720	×	×	×	×	Boxes + event-class tracks
NeRDD [12]	1280 × 720	✓	×	×	×	2D boxes
CRSOT [23]	1280 × 800	×	×	✓	×	Tracking boxes
FRED [13]	1280 × 720	✓	×	×	×	Frame boxes + track IDs
Ours	1280 × 720	✓	✓	✓	✓	Box-derived events

4 Method

4.1 Overview

Figure 4 shows M^{2}E-Point, the main point-based baseline for M^{2}E-UAV. It keeps each event packet as a sampled $[x, y, t, p]$ point set, extracts local event geometry with EdgeConv, predicts event-level UAV foreground scores, and converts the predicted foreground cluster into a detection-style box using DBSCAN. M^{2}E-Point + IMU is an optional analysis variant that uses synchronized IMU measurements to modulate the global event feature through lightweight FiLM conditioning.

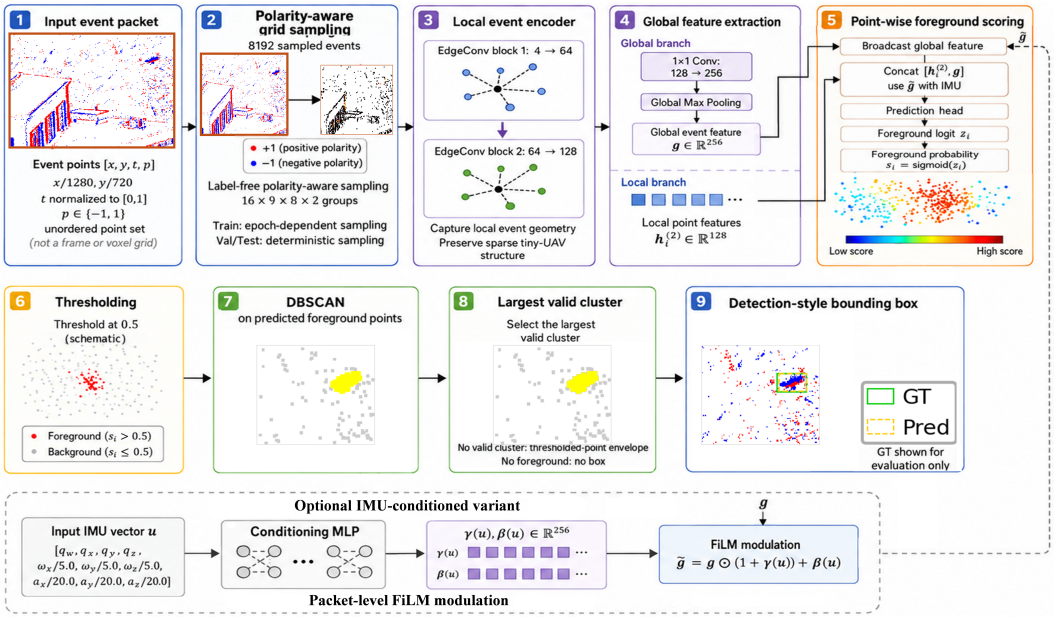

Figure 4: **M^{2}E-Point with optional IMU conditioning.** The event branch samples $[x, y, t, p]$ points, extracts local event features with EdgeConv, and aggregates them into a global packet feature. The optional IMU-conditioned variant modulates the global feature via FiLM. Local features and the packet feature are fused for event-level foreground scoring, and DBSCAN converts predicted foreground clusters into a detection-style box.

4.2 Event representation and sampling

An event packet is represented as an unordered point set. For event i, M^{2}E-Point uses the normalized input feature

$$\mathbf{e}_i = [\bar{x}_i, \bar{y}_i, \bar{t}_i, \bar{p}_i], \qquad \bar{x}_i = \frac{x_i}{1280}, \quad \bar{y}_i = \frac{y_i}{720}. \tag{2}$$

Timestamps are normalized using the minimum and maximum timestamp of the complete packet before subsampling,

$$\bar{t}_i = \frac{t_i - t_{\min}}{t_{\max} - t_{\min}}, \tag{3}$$

and polarity is mapped to $\bar{p}_i \in \{-1, 1\}$. This representation preserves spatial location, within-packet temporal position, and polarity without collapsing the packet into a dense image.

The reference model uses a fixed budget of 8,192 event points per packet. A polarity-aware spatiotemporal grid sampler uses 16×9 spatial cells, eight temporal bins, and separate polarity groups to allocate samples across occupied regions using only event coordinates, timestamps, and polarity. Sampling is repeated during training and made deterministic for evaluation. If a packet contains fewer events than the requested budget, events are repeated to maintain the fixed input size.

4.3 Event encoding and box prediction

The event branch encodes local event geometry with two EdgeConv blocks [21]. Each block recomputes a 16-nearest-neighbor graph in its input feature space. Given event features, EdgeConv constructs local graph features from an event and its neighbors, applies a shared MLP, and aggregates neighborhood responses by max pooling. The first EdgeConv block maps the 4-D event input to 64 channels, and the second maps 64 channels to 128 channels:

$$\{\mathbf{e}_i\}_{i=1}^{8192} \rightarrow \{\mathbf{h}_i^{(1)} \in \mathbb{R}^{64}\}_{i=1}^{8192} \rightarrow \{\mathbf{h}_i^{(2)} \in \mathbb{R}^{128}\}_{i=1}^{8192}. \tag{4}$$

A 1×1 convolution maps the 128-D local feature to 256 channels, and global max pooling produces a packet-level event feature $\mathbf{g} \in \mathbb{R}^{256}$. The local EdgeConv features are preserved for point-wise foreground scoring.

For M^{2}E-Point, the packet feature $\mathbf{g}$ is broadcast to every sampled event and concatenated with the 128-D local EdgeConv feature. The prediction head outputs one UAV foreground logit z_i for each sampled event, and the foreground probability is $s_i = \sigma(z_i)$. For detection-style evaluation, event foreground probabilities are thresholded at 0.5. The selected foreground events are clustered in image coordinates with DBSCAN using $\epsilon = 15.0$ pixels and $\mathrm{min_samples} = 3$. The largest valid non-noise cluster is converted to an axis-aligned bounding box. If no valid cluster is found, the thresholded-point envelope is used; if no foreground point is selected, no box is emitted. The box confidence is the maximum foreground probability among the points used to form the box. The resulting boxes are evaluated against the packet-level benchmark annotations.

4.4 Training objective

M^{2}E-Point is optimized from event-level binary foreground labels using a focal-plus-Tversky objective. For sampled labels $y_i \in \{0, 1\}$ and foreground logits z_i,

$$\mathcal{L}_{\mathrm{seg}} = \mathcal{L}_{\mathrm{focal}}(z, y; \alpha = 0.75, \gamma = 2) + \mathcal{L}_{\mathrm{Tversky}}(z, y; \alpha_T = 0.7, \beta_T = 0.3). \tag{5}$$

Focal emphasizes hard foreground/background decisions; Tversky balances soft false positives and false negatives. Training uses AdamW for 100 epochs, with validation-based checkpoint selection. The optional IMU branch applies FiLM conditioning to the global event feature while retaining the same event encoder, sampler, and objective. Supplementary Section B.1 gives the expanded loss definitions (Eqs. 6 and 8), optimization settings, IMU features, and conditioning equations.

5 Experiments

5.1 Evaluation protocol

Detection targets. The benchmark evaluates single-target box detection from an event packet and, where applicable, its aligned IMU measurement. Each model may emit one bounding box or no box. The reference target is the complete-packet positive-event envelope derived from reviewed temporal annotations (Section 3.3), computed before model-specific input sampling. Only packets with non-degenerate target support are scored. Target-present weak-response packets are retained as ignored observations rather than negatives. The current box-detection benchmark does not include a verified target-absent false-alarm test population.

Two evaluation settings. The main benchmark uses the 8:1:1 temporal construction in Section 3.4: all four recordings contribute disjoint Train/Val/Test segments. The four-fold leave-one-recording-out (LORO) experiment instead tests on an entire unseen recording. For each fold, the other three recordings are independently split 9:1 into Train/Val, with the same centered 500-packet exclusion band at each source-recording boundary. All learned components are trained for that fold; validation and parameter selection use only its three source recordings. Temporal histories stay within the relevant partition and recording. Table S7 gives the four fold populations.

Metrics and model selection. Direct detectors retain their highest-confidence box; point references use the cluster-to-box rule in Section 4. A packet with no emitted box contributes a false negative. We report detection F1 at IoU≥ 0.50, AP_{50}, and $\mathrm{AP}_{50:95}$ (the mean over ten thresholds from 0.50 to 0.95 in steps of 0.05). AP integrates the all-points precision–recall envelope. Detection

F1 is $2\mathrm{TP}/(2\mathrm{TP} + \mathrm{FP} + \mathrm{FN})$; an incorrect emitted box contributes both an FP and a missed GT. Checkpoints retain their full-validation selections. Supplementary Section B.3 specifies the stored-output scoring implementations and selection criteria.

Compared methods. YOLO11s uses accumulated event frames; RVT and SAST use event tensors; EV-SpSegNet, KPConv, RandLA-Net, COSeg, and M^{2}E-Point use event-derived point representations. The four external point methods are adapted to the packet-level task. COSeg uses a fixed labeled support example from Train for evaluation queries. M^{2}E-Point and its IMU variant use 8,192 points and eight temporal sampling bins. The matched multi-seed experiment uses fixed evaluation sampling and is reported in the Supplementary Material. The proposal-recovery pipelines add a learned scorer, dual-scale context, causal recovery, and box refinement; they are analyzed separately in Table 4.

5.2 Temporal benchmark comparison

Table 3 shows RVT leading in Det-F1$_{50}$ and AP$_{50}$, with SAST leading in AP$_{50:95}$. RVT's recurrent aggregation can combine intermittent target evidence; SAST's adaptive window and token selection can emphasize informative event regions [7, 16]. SAST's higher AP$_{50:95}$ despite lower AP$_{50}$ indicates stronger aggregate performance under stricter overlap requirements. Both detectors learn boxes from spatially organized event tensors. M^{2}E-Point instead derives boxes from thresholded foreground points, making localization sensitive to missing target support and clustering.

Table 3: **Detection comparison on the temporal Test set (10,381 valid-box packets).** Bold and underline indicate the best and second-best scores, respectively. Evaluation details are provided in Supplementary Section B.3.

Method	Input	Det-F1$_{50}$ ↑	AP$_{50}$ ↑	AP$_{50:95}$ ↑
YOLO11s [10]	Frame	0.7787	0.7590	0.3939
RVT [7]	Tensor	**0.8781**	**0.8390**	<u>0.4198</u>
SAST [16]	Tensor	<u>0.8447</u>	<u>0.8230</u>	**0.4468**
EV-SpSegNet [3]	Points	0.3528	0.2566	0.1172
KPConv [18]	Points	0.4257	0.3066	0.1307
RandLA-Net [8]	Points	0.2980	0.2343	0.1030
COSeg [1]	Points	0.4143	0.3263	0.1575
M^{2}E-Point	Points	0.5209	0.3815	0.2109
M^{2}E-Point + IMU	Points + IMU	0.4867	0.3677	0.2051

5.3 Sampling under a fixed point budget

M^{2}E-Point processes 8,192 sampled events per packet. The polarity-aware grid sampler distributes this budget across non-empty spatial, temporal, and polarity groups, then selects events within each group. When target events occupy only a small fraction of a group, some may be omitted during sampling, leaving fewer target points for foreground prediction and box construction.

The detection target is computed from all foreground events in the complete packet. The sampled target points may therefore cover only part of the region that the detector must recover. After foreground thresholding, these points must form a spatially coherent cluster; incomplete or fragmented support can produce a missed or inaccurate box (Figure 5). The following analysis examines full-packet spatial context, temporal proposal recovery, and proposal-edge refinement as ways to improve the available evidence and resulting boxes.

5.4 Progressive analysis of the event-point reference

Table 4 reports a cumulative progression from the event-only M^{2}E-Point Base through dual-scale spatial context (DSSC), temporal proposal recovery (TPR), and proposal-edge refinement (PER). DSSC adds full-packet spatial context to the sampled points; TPR generates candidates from current and preceding proposals and selects among them with a learned quality regressor; PER refines proposal coordinates and predicts an IoU-based localization-quality score. Training and implementation

details are provided in Supplementary Section B. Stage-wise AP increments are computed from unrounded scores.

Table 4: **Progressive event-only point-reference results on the strict temporal test set.** All stages use the same 10,381 valid-box Test packets. The Base uses the event-only predictions from Table 3; the common evaluation used here is specified in Supplementary Section B.3.

Stage	Det-F1$_{50}$ ↑	AP$_{50}$ ↑	AP$_{50:95}$ ↑
Base	0.5209	0.3815	0.2111
DSSC	0.7816	0.7068	0.4049
DSSC+TPR	0.7920	0.7456	0.4454
DSSC+TPR+PER	0.8233	0.7844	0.4921

DSSC supplements the sampled points with full-packet spatial evidence. Motivated by learned importance scoring in SAST [16], a foreground-trained CNN extracts 2-pixel and 4-pixel histogram features, which are queried at each sampled point's coordinates and fused with its local point features through residual projections. Each retained point thus receives context from surrounding events omitted from the point set. These maps use ten temporal bins and two polarities (20 channels); the point sampler retains eight temporal bins and unchanged indices. AP$_{50:95}$ rises from 0.2111 to 0.4049 (+0.1938), the largest stage increment, supporting the value of spatial evidence beyond the sampled point set.

TPR addresses fluctuations in single-packet proposals through temporal candidate generation and quality selection. It considers current and scale-adjusted boxes, the preceding box, a current–previous combination, and linear prediction from preceding boxes. A quality regressor fitted on training data predicts candidate IoU, selects among these candidates, and supplies the confidence of the selected box. Det-F1$_{50}$ increases from 0.7816 to 0.7920, AP$_{50}$ from 0.7068 to 0.7456, and AP$_{50:95}$ from 0.4049 to 0.4454 (+0.0405). Temporal candidates supply alternative geometry when current foreground support is fragmented, while predicted quality prioritizes candidates expected to overlap the target.

PER targets proposal boundaries and localization confidence. It pools point features inside the box, along four boundary bands, in the surrounding region, and globally, then predicts four coordinate residuals and an IoU-supervised quality score. The boundary features support spatial corrections, while the quality head models how well a box localizes the target, related to localization-confidence modeling in IoU-Net [9]. AP$_{50:95}$ increases from 0.4454 to 0.4921 (+0.0466). Boundary-band features guide side-specific coordinate corrections, while the quality head ranks refined proposals by their expected overlap with the target.

5.5 Cross-recording generalization and IMU conditioning

Table 5 shows that model rankings depend on the held-out recording. SAST leads the reference models on average; the full event-only point-based progression performs best on A and B, while C remains difficult across representations.

The progression raises macro AP$_{50:95}$ from 0.1727 to 0.3938, with roughly 70% of the gain at DSSC. Querying full-packet neighborhoods at unchanged sampled locations supplies spatial context omitted from the point subset, helping foreground grouping under the fixed point budget. Box construction still depends on the retained target points. Temporal evaluation learns from disjoint training segments of every recording; LORO transfers the learned foreground and proposal-quality models to a recording excluded from training and validation.

C is harder than both A, which shares its background category, and D, which shares sunset illumination. On C, TPR improves AP$_{50}$ with little change in F1 (Supplementary Table S8). Causal proposal history within the held-out recording provides alternative geometry when the current packet yields an unstable box, while quality prediction selects and ranks these candidates. PER then raises F1 from 0.1056 to 0.1181 but lowers AP$_{50}$ from 0.0527 to 0.0228. This pattern suggests a mismatch between refined geometry and predicted quality on C: useful coordinate corrections can coexist with high-ranked inaccurate boxes or low-ranked correct boxes.

Table 5: **Recording-held-out (LORO) results.** Values are $AP_{50:95}$; the macro average weights the four recordings equally. M^{2}E-Point is the event-only Base; successive rows cumulatively add DSSC, TPR, and PER.

Method	Held-out A	Held-out B	Held-out C	Held-out D	Macro
YOLO11s [10]	0.4862	0.3776	0.0999	0.1927	0.2891
RVT [7]	0.4740	0.3906	0.0765	0.2718	0.3032
SAST [16]	0.5688	0.4639	0.0957	0.3051	0.3584
EV-SpSegNet [3]	0.3297	0.1883	1.17×10^{-5}	0.0314	0.1373
KPConv [18]	0.3079	0.2817	3.44×10^{-5}	0.0185	0.1520
RandLA-Net [8]	0.2190	0.2506	0.0000	0.0154	0.1213
COSeg [1]	0.2746	0.3391	0.0004	0.0593	0.1683
M^{2}E-Point	0.2815	0.3696	7.87×10^{-7}	0.0399	0.1727
DSSC	0.6119	0.5346	0.0054	0.1574	0.3273
DSSC+TPR	0.6519	0.5904	0.0173	0.2366	0.3741
DSSC+TPR+PER	0.6974	0.5960	0.0082	0.2736	0.3938

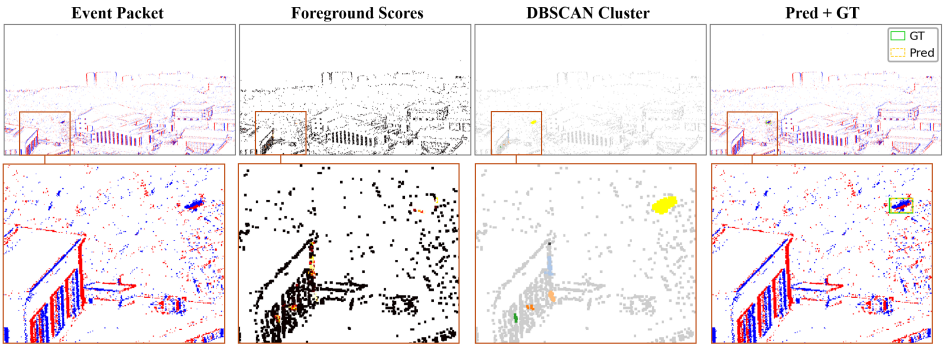

Figure 5: **Point-level prediction and clustering.** The panels show the input event packet, predicted foreground scores, the DBSCAN cluster, and the predicted and ground-truth bounding boxes. The lower row zooms in on the tiny-UAV region to show sparse target evidence and nearby clutter.

The matched three-seed comparison gives $AP_{50:95}$ of 0.2098 ± 0.0010 for event-only input and 0.2052 ± 0.0006 with IMU conditioning (Supplementary Section C.1). The IMU branch applies a shared global modulation to every sampled event. The local event-neighborhood pathway is unchanged by IMU conditioning.

5.6 Limitations

The present benchmark focuses on single-target detection in a defined set of outdoor motion-on-motion scenarios. Extending the evaluation to additional target types, carrier platforms, and operating environments is a natural direction for future work.

6 Conclusion

We presented M^{2}E-UAV, a 1280×720 synchronized event-IMU dataset and benchmark for onboard motion-on-motion event-based tiny UAV detection, with box-derived event-level UAV supervision across four scene families. Its strict temporal evaluation protocol explicitly handles weak-response observations. Reference models across input representations and a staged point-based analysis reveal difficulties involving sparse target evidence, candidate recovery, and recording variation. Together, the dataset, evaluation protocols, and reference results establish a benchmark for studying sparse target evidence, proposal recovery, and cross-recording performance under coupled observer and target motion.

Acknowledgments and Disclosure of Funding

This work was supported in part by the National Natural Science Foundation of China (Nos. 62471415 and 62401225), the Natural Science Foundation of Fujian Province of China (Nos. 2023J01004, 2025J0141, and 2024J01115), and the Natural Science Foundation of Xiamen of China (No. 3502Z202472018). The authors declare no competing interests.

References

[1] Zhaochong An, Guolei Sun, Yun Liu, Fayao Liu, Zongwei Wu, Dan Wang, Luc Van Gool, and Serge Belongie. Rethinking few-shot 3D point cloud semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[2] Kenneth Chaney, Fernando Cladera, Ziyun Wang, Anthony Bisulco, M. Ani Hsieh, Christopher Korpela, Vijay Kumar, Camillo J. Taylor, and Kostas Daniilidis. M3ED: Multi-robot, multi-sensor, multi-environment event dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 4016–4023, 2023.

[3] Nuo Chen, Chao Xiao, Yimian Dai, Shiman He, Miao Li, and Wei An. Event-based tiny object detection: A benchmark dataset and baseline. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[4] Julien Dupeyroux, Raoul Dinaux, Nikhil Wessendorp, and Guido De Croon. A novel obstacle detection and avoidance dataset for drones. In *Proceedings of the 2022 Workshop on System Engineering for Constrained Embedded Systems (DroneSE) and Rapid Simulation and Performance Evaluation (RAPIDO)*, pages 8–13, 2022. doi: 10.1145/3522784.3522786.

[5] Guillermo Gallego, Tobi Delbruck, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J. Davison, Jorg Conradt, Kostas Daniilidis, and Davide Scaramuzza. Event-based vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):154–180, 2022.

[6] Eike Gebauer and Benjamin Risse. Event based drone detection and segmentation (EDDS). Zenodo, 2026. doi: 10.5281/zenodo.18629643.

[7] Mathias Gehrig and Davide Scaramuzza. Recurrent vision transformers for object detection with event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[8] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. RandLA-Net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[9] Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tete Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 784–799, 2018.

[10] Glenn Jocher and Jing Qiu. Ultralytics YOLO11. `https://docs.ultralytics.com/models/yolo11/`, 2024. Ultralytics software.

[11] Gabriele Magrini, Federico Becattini, Giovanni Colombo, and Pietro Pala. EV-Flying: An event-based dataset for in-the-wild recognition of flying objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 4947–4955, 2025. doi: 10.1109/CVPRW67362.2025.00487.

[12] Gabriele Magrini, Federico Becattini, Pietro Pala, Alberto Del Bimbo, and Antonio Porta. Neuromorphic drone detection: An event-RGB multimodal approach. In *Computer Vision – ECCV 2024 Workshops*, volume 15646 of *Lecture Notes in Computer Science*, pages 259–275. Springer, 2025. doi: 10.1007/978-3-031-92460-6_16.

[13] Gabriele Magrini, Niccolò Marini, Federico Becattini, Lorenzo Berlincioni, Niccolò Biondi, Pietro Pala, and Alberto Del Bimbo. FRED: The Florence RGB-event drone dataset. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 13170–13176, 2025. doi: 10.1145/3746027.3758271.

[14] Jakub Mandula, Jonas Kühne, Luca Pascarella, and Michele Magno. Towards real-time fast unmanned aerial vehicle detection using dynamic vision sensors. In *Proceedings of the 2024 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, pages 1–6, 2024. doi: 10.1109/I2MTC60896.2024.10561168.

[15] Anton Mitrokhin, Cornelia Fermuller, Chethan M. Parameshwara, and Yiannis Aloimonos. Event-based moving object detection and tracking. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018. doi: 10.1109/IROS.2018.8593805.

[16] Yansong Peng, Hebei Li, Yueyi Zhang, Xiaoyan Sun, and Feng Wu. Scene adaptive sparse transformer for event-based object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[17] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[18] Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, and Leonidas J. Guibas. KPConv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[19] Xiao Wang, Jianing Li, Lin Zhu, Zhipeng Zhang, Zhe Chen, Xin Li, Yaowei Wang, Yonghong Tian, and Feng Wu. VisEvent: Reliable object tracking via collaboration of frame and event flows. *IEEE Transactions on Cybernetics*, 54(3):1997–2010, 2024. doi: 10.1109/TCYB.2023.3318601.

[20] Xiao Wang, Shiao Wang, Chuanming Tang, Lin Zhu, Bo Jiang, Yonghong Tian, and Jin Tang. Event stream-based visual object tracking: A high-resolution benchmark dataset and a novel baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19248–19257, 2024.

[21] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph CNN for learning on point clouds. *ACM Transactions on Graphics*, 38(5):146:1–146:12, 2019.

[22] Ziyi Wu, Mathias Gehrig, Qing Lyu, Xudong Liu, and Igor Gilitschenski. LEOD: Label-efficient object detection for event cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[23] Yabin Zhu, Xiao Wang, Chenglong Li, Bo Jiang, Lin Zhu, Zhixiang Huang, Yonghong Tian, and Jin Tang. CRSOT: Cross-resolution object tracking using unaligned frame and event cameras. *IEEE Transactions on Multimedia*, 27:6529–6542, 2025. doi: 10.1109/TMM.2025.3586135.

Supplementary Material

A Dataset and Acquisition Details

Dataset statistics. The processed inventory contains 108,618 packets. Removing the manually identified 293-packet multi-object interval leaves 108,325 single-target packets. Of these, 2,033 weak-response packets and one training packet with a zero-area event-support envelope are excluded from the valid-box projection, leaving 106,291 valid-box packets.

Before valid-box filtering, the temporal split assigns 83,765 / 10,442 / 10,440 packets to Train / Val / Test, with 3,678 packets in the exclusion gaps. The weak-response counts are 1,726 / 195 / 59 / 53 across Train / Val / Test / gap. The 1,980 weak-response packets assigned to Train, Val, or Test remain as temporal context within their respective partitions; the 53 weak-response packets in the temporal exclusion gaps enter no partition or temporal context. After filtering, the valid-box Train / Val / Test populations are 82,038 / 10,247 / 10,381 packets at 1280×720.

Annotation consistency. Initial 10-Hz box keyframes are propagated to the approximately 100-Hz packet-aligned sequence and reviewed as described in Section 3.3. The dense annotation exports and event-label generation agree on all 320 checked samples across the four recordings.

Table S1 lists the carrier, target, and sensor configuration used across the four recordings.

Table S1: Hardware configuration of the M^{2}E-UAV data-collection platform.

Component	Specification
Carrier UAV	DJI Matrice 300 RTK
Target UAV	DJI Mini 2, $159 \times 203 \times 56$ mm
Event camera	SilkyEvCam HD, IMX636 CMOS sensor, 8 mm lens
IMU	FDI Systems DETA100D4G
Synchronization hardware	STM32-based trigger-capable board

Each event sample stores coordinates, sensor timestamps, polarity, and lifted foreground labels; IMU records contain orientation, angular velocity, and linear acceleration. The dataset metadata records the per-recording affine event-clock calibration used for event–IMU alignment.

Table S2: **Event-sensor clock calibration.** Rate mismatch is relative to the host/IMU timeline; the last column is the error accumulated by fixed-offset conversion at the end of each recording.

Recording	Host ns / sensor ns	Mismatch	Fixed-offset end error
Sunset farm	1.0000233376	23.3 ppm	16.54 ms
Sunset building	1.0000250468	25.0 ppm	10.33 ms
Sunny farm	1.0000248943	24.9 ppm	18.19 ms
Sunny building	1.0000247712	24.8 ppm	16.94 ms

Table S2 shows that fixed-offset timing accumulates 10.33–18.19 ms of error, motivating affine clock correction. The robust affine fits have typical millisecond-scale residuals (99th-percentile absolute residuals approximately 2.35–3.97 ms across the four recordings). The integrity checks also identify six missing IMU samples and 63 missing EventPacket messages. Two training samples intersect a missing-IMU interval, whereas validation and test samples do not; no labeled sample overlaps an identified event-packet-loss interval, including a 5 ms caution margin. Missing observations are recorded as integrity metadata.

The temporal boundaries are the 80th and 90th percentiles within each recording, with a centered 500-packet exclusion band at each boundary (Section 3.4).

B Implementation and Evaluation Protocol

B.1 Model implementation and training

With $p_i = \sigma(z_i)$, $p_{t,i} = y_i p_i + (1 - y_i)(1 - p_i)$, and $\alpha_{t,i} = 0.75 y_i + 0.25(1 - y_i)$, the focal term is

$$\mathcal{L}_{\text{focal}} = -\frac{1}{N_s} \sum_i \alpha_{t,i} (1 - p_{t,i})^2 \log p_{t,i}. \tag{6}$$

Here N_s counts all sampled events in the training mini-batch. The Tversky loss uses soft counts over these same events:

$$\begin{aligned} \text{TP}_{\text{soft}} &= \sum_i p_i y_i, \\ \text{FP}_{\text{soft}} &= \sum_i p_i (1 - y_i), \\ \text{FN}_{\text{soft}} &= \sum_i (1 - p_i) y_i. \end{aligned} \tag{7}$$

The Tversky term is

$$\mathcal{L}_{\text{Tversky}} = 1 - \frac{\text{TP}_{\text{soft}} + \epsilon}{\text{TP}_{\text{soft}} + 0.7\,\text{FP}_{\text{soft}} + 0.3\,\text{FN}_{\text{soft}} + \epsilon}, \tag{8}$$

with $\epsilon = 10^{-5}$. This objective addresses the imbalance between sparse target-foreground events and the larger ego-motion background population. The focal class weights favor foreground events, while $(1 - p_{t,i})^2$ downweights correctly classified, high-confidence events. The Tversky denominator assigns soft false positives a larger coefficient than soft false negatives (0.7 versus 0.3). Both loss terms enter the total objective with coefficient one. The point-based reference uses 8,192 sampled events per packet and an eight-bin temporal representation. Training uses AdamW with a learning rate of 10^{-3}, weight decay of 10^{-4}, cosine annealing, physical batch size 8, gradient accumulation 2, and 100 epochs. The three-seed baseline comparison uses independent seeds 3407, 3408, and 3409. Validation and test sampling are deterministic across seeds to keep the evaluated inputs matched. For the IMU-conditioned baseline, the event representation, point-sampling policy, and optimization recipe are matched to the event-only baseline. The additional input is the 10-D inertial vector, aligned to the affine-corrected event timeline and fused with the global event feature through FiLM conditioning.

Table S3 summarizes the baseline configuration and the separately trained components used for progressive proposal recovery. The learned scorer supplies full-packet histogram features to the dual-scale spatial context (DSSC) adapter; temporal proposal recovery (TPR) is fitted using training data only; proposal-edge refinement (PER) is trained separately on the recovered proposals. The scorer is trained for four epochs. DSSC then trains its residual context projections and upper point-prediction layers for 20 epochs while freezing the EdgeConv encoder and scorer. PER is trained for 30 epochs.

IMU conditioning. The supplementary IMU-conditioned reference uses the same event encoder and adds a packet-level FiLM branch. The 10-D inertial vector is queried at the center of the affine-corrected event window from the complete raw IMU stream, using interpolation only within observed support. It contains orientation, angular velocity, and linear acceleration:

$$\mathbf{u} = [q_w, q_x, q_y, q_z, \omega_x/5.0, \omega_y/5.0, \omega_z/5.0, a_x/20.0, a_y/20.0, a_z/20.0]. \tag{9}$$

The quaternion entries are used directly, while angular velocity and linear acceleration are scaled by 5.0 and 20.0, respectively. The conditioning MLP maps $\mathbf{u}$ to FiLM parameters [17]:

$$[\gamma(\mathbf{u}), \beta(\mathbf{u})] = \text{MLP}(\mathbf{u}), \qquad \gamma(\mathbf{u}), \beta(\mathbf{u}) \in \mathbb{R}^{256}. \tag{10}$$

The global event feature is modulated as

$$\tilde{\mathbf{g}} = \mathbf{g} \odot (1 + \gamma(\mathbf{u})) + \beta(\mathbf{u}). \tag{11}$$

The IMU branch modulates the global packet representation while leaving the local event-neighborhood pathway unchanged.

Table S3: **Implementation details of the point-based reference and progressive analysis.**

Item	Setting
Event representation	Normalized $[x, y, t, p]$; time normalized by complete-packet $t_{\min}, t_{\max}$
Point budget	8,192 events per packet
Spatiotemporal sampler	$16 \times 9 \times 8 \times 2$ $(x, y, t, \text{polarity})$ groups; label-free sampling
IMU representation	Quaternion, angular velocity / 5.0, linear acceleration / 20.0
Optimizer / schedule	AdamW, 10^{-3} learning rate, 10^{-4} weight decay, cosine annealing
Baseline training	Batch 8, gradient accumulation 2, 100 epochs
DSSC dense context	Full-packet 2px/4px histograms; 10 time bins $\times$ 2 polarities
Recovery-stage training	Scorer: 4 epochs; DSSC: 20 epochs; TPR: Train-only fitting; PER: 30 epochs
Loss	Focal ($\alpha = 0.75, \gamma = 2$) + Tversky ($\alpha_T = 0.7, \beta_T = 0.3$)
Checkpoint selection	Validation-only selection using the stage-specific criterion
Foreground threshold	0.5
Box derivation	DBSCAN, $\epsilon = 15$ pixels, $\text{min_samples} = 3$; largest valid cluster; thresholded-point envelope if no valid cluster; no box if no foreground is selected

B.2 Detection metrics

The following definitions specify the common exact-grid evaluation of the progressive stage results and recording-held-out comparisons. Temporal reference evaluation implementations and checkpoint-selection procedures are distinguished in Section B.3. Each scored packet has one ground-truth box and at most one prediction; let $N_g > 0$ be the number of ground-truth boxes.

Bounding-box IoU. For continuous XYXY coordinates $b = (x_1, y_1, x_2, y_2)$, the area is $|b| = (x_2 - x_1)(y_2 - y_1)$. Intersection widths and heights are truncated at zero, giving

$$\text{IoU}(b, g) = \begin{cases} \dfrac{|b \cap g|}{|b \cup g|}, & |b| > 0 \text{ and } |g| > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Widths and heights use coordinate differences without an inclusive $+1$ pixel term. Finite zero-area predictions remain in the scored population and have IoU zero; non-finite or inverted boxes are rejected during preflight.

Detection F1. At IoU threshold τ, an emitted prediction is a true positive only when it matches the GT of its own packet with $\text{IoU} \geq \tau$. With N_p emitted predictions, $\text{FP}_\tau = N_p - \text{TP}_\tau$ and $\text{FN}_\tau = N_g - \text{TP}_\tau$. An incorrect prediction contributes both an FP and an FN; a missing prediction contributes only an FN. Detection F1 is

$$\text{DetF1}_{50} = \frac{2\text{TP}_{50}}{2\text{TP}_{50} + \text{FP}_{50} + \text{FN}_{50}}. \tag{13}$$

A zero denominator is assigned zero. All reported F1 scores are computed from box-level detections.

Average precision. For a fixed τ, first sort predictions by ascending sample ID, then stably sort by descending confidence. Let K be the number of emitted predictions and $u_j(\tau)$ indicate whether prediction j is a true positive. At rank k, the cumulative count is $T_k(\tau) = \sum_{j=1}^{k} u_j(\tau)$, with precision and recall

$$P_k = \frac{T_k(\tau)}{k}, \qquad R_k = \frac{T_k(\tau)}{N_g}. \tag{14}$$

Using $R_0 = 0$, the all-points precision envelope and recall-step integral are

$$\widetilde{P}_k = \max_{k \leq j \leq K} P_j, \qquad \text{AP}_\tau = \sum_{k=1}^{K} (R_k - R_{k-1}) \widetilde{P}_k. \tag{15}$$

If $K = 0$, AP is zero. Equal-confidence predictions remain ordered by sample ID; tied scores are not pooled into groups. We report $\mathrm{AP}_{50} = \mathrm{AP}_{0.50}$ and

$$\mathrm{AP}_{50:95} = \frac{1}{10} \sum_{\tau \in \{0.50, 0.55, \ldots, 0.95\}} \mathrm{AP}_{\tau}. \tag{16}$$

The ten thresholds are individually constructed from exact decimal labels before conversion to numerical values.

B.3 Evaluation implementations and checkpoint selection

The baseline comparison in Table 3 retains the original single-box AP implementation, with a floating-point grid of nominal IoU thresholds from 0.50 to 0.95 in steps of 0.05. The progressive analysis uses the definitions in Section B.2. Floating-point threshold construction can affect boundary matches: the same event-only Base predictions yield $\mathrm{AP}_{50:95}$ 0.2109 with the original implementation and 0.2111 with the exact threshold grid, as displayed in the baseline and progressive tables. The predictions and checkpoint are unchanged.

The geometry source is fixed by stage: Base uses its original stored coordinates, DSSC uses the retained float32 candidate boxes, TPR uses the retained float32 proposal boxes, and PER uses its final stored boxes. Float32 coordinates are promoted to float64 for arithmetic only, retaining their serialized values. The ground-truth projection remains fixed within each chain, and no checkpoint is selected or changed using Test results.

The evaluator uses the stored box coordinates without post-processing.

For the progressive analysis, DSSC checkpoints are selected by full-validation $\mathrm{AP}_{50:95}$, PER checkpoints by full-validation $\mathrm{AP}_{50:95}$ using predicted localization-quality scores as box confidence, and TPR is fitted on training proposals only. The event-only temporal progression begins with the run included in the matched baseline comparison. The IMU-conditioned progression begins with a separately trained baseline (0.2092 $\mathrm{AP}_{50:95}$), rather than the matched IMU baseline (0.2051 under its original evaluation). The baseline runs use independent seeds 3407, 3408, and 3409; progressive experiments use seed 3407.

C Additional Experimental Results

C.1 Matched three-seed comparison

To examine global IMU conditioning across training initializations, we compare the event-only and IMU-conditioned references over three independent seeds, with eight temporal bins and fixed validation/test sampling. Table S4 reports their means and sample standard deviations.

Table S4: **Matched M^{2}E-Point comparison.** Three independent seeds; mean±sample SD.

Configuration	Det-F1$_{50}$	AP_{50}	$\mathrm{AP}_{50:95}$
Event-only	0.5157 ± 0.0049	0.3819 ± 0.0005	0.2098 ± 0.0010
IMU-conditioned	0.4894 ± 0.0023	0.3690 ± 0.0014	0.2052 ± 0.0006

Global IMU conditioning. In the matched three-seed experiment, event-only $\mathrm{AP}_{50:95}$ is 0.2098 ± 0.0010, compared with 0.2052 ± 0.0006 with global IMU FiLM, a difference of -0.0045 (computed from unrounded means). Detection F1 likewise decreases from 0.5157 ± 0.0049 to 0.4894 ± 0.0023. The tested global conditioning branch does not improve aggregate detection.

C.2 Temporal progressive results

The event-only and independently trained IMU-conditioned progressions are reported separately on the same temporal Test population.

Table S5: **Temporal Test, M²E-Point.**

Stage	Det-F1$_{50}$	AP$_{50}$	AP$_{50:95}$
Base	0.5209	0.3815	0.2111
DSSC	0.7816	0.7068	0.4049
DSSC+TPR	0.7920	0.7456	0.4454
DSSC+TPR+PER	0.8233	0.7844	0.4921

Table S5 shows temporal event-only AP$_{50:95}$ increasing from 0.2111 to 0.4921, with the largest increment at DSSC.

Table S6: **Temporal Test, independently trained M²E-Point + IMU progression.**

Stage	Det-F1$_{50}$	AP$_{50}$	AP$_{50:95}$
Base	0.4888	0.3725	0.2092
DSSC	0.7816	0.6954	0.3953
DSSC+TPR	0.7924	0.7414	0.4429
DSSC+TPR+PER	0.8298	0.7864	0.4948

In Table S6, the IMU progression reaches AP$_{50:95}$ 0.4948 from an independently trained IMU Base of 0.2092.

C.3 Recording-held-out evaluation

C.3.1 Protocol and partitions

Recordings A, B, C, and D correspond to sunny building–forest, sunny farm–village, sunset building–forest, and sunset farm–village conditions, respectively. Each leave-one-recording-out fold holds out one complete recording for Test. Each of the three source recordings is split at its 90th-percentile annotated-sample boundary into Train and Val. A centered band of 500 raw packet IDs is removed at each source Train/Val boundary, providing a nominal 5 s separation. The measured retained-packet separations are 5.0023, 5.3795, 5.0004, and 4.9995 s for A/B/C/D; sparse annotation intervals can make them longer than the exclusion band. No exclusion band is cut inside the held-out Test recording.

For example, ABC→D trains and validates on A, B, and C, and tests on all valid-box packets in D. All learned components are trained for the fold, and model selection uses its source recordings. Source-gap packets are excluded from the source partitions and their contexts. Weak-response packets within an assigned partition remain temporal context and are masked from direct box supervision and detection metrics. Temporal processing is reset at recording and split boundaries. The four disjoint held-out Test sets cover 106,291 valid-box packets, whereas the temporal Test split contains 10,381.

Table S7: **Partitions for recording-held-out evaluation.** Train, Val, and Test count valid-box packets. Excluded source packets count all packets in the source Train/Val exclusion bands, including weak-response packets.

Fold	Train	Val	Test	Excluded source packets
ABC→D	72,235	8,185	24,511	1,379
ABD→C	78,014	8,822	18,084	1,373
ACD→B	63,903	7,117	33,797	1,492
BCD→A	67,404	7,634	29,899	1,372

Four-fold macro average. For recording-held-out evaluation, the four-fold macro average is

$$\mathrm{MacroAP}_{50:95} = \frac{1}{4} \sum_{r \in \{A,B,C,D\}} \mathrm{AP}^{(r)}_{50:95}. \tag{17}$$

The four recording-level AP values receive equal weight in the macro-average.

C.3.2 Reference-model comparison

Main-text Table 5 reports the reference configurations and event-only progressive stages across all four held-out recordings using the exact-grid evaluator in Section B.2. All results are obtained from the existing validation-selected checkpoints with seed 3407.

Among the reference models, SAST obtains the highest macro-average $AP_{50:95}$ and the highest scores on A, B, and D; YOLO11s obtains the highest score on C. Among the external point references, COSeg obtains the highest macro-average $AP_{50:95}$ (0.1683), with scores ranging from 0.0004 on C to 0.3391 on B.

C.3.3 Event-only progressive results

The progressive analysis follows Base → DSSC → TPR → PER for each held-out recording. Base is the event-only M^{2}E-Point row in Table 5; all subsequent stages also use event input only, without IMU conditioning.

Macro-average $AP_{50:95}$ increases from 0.1727 at Base to 0.3938 after the full progression, with the largest increment at DSSC. TPR increases AP on all four recordings. PER further improves A, B, and D, while TPR retains the highest AP on C.

Table S8: **Detailed recording-held-out results of the event-only progression.** Each block evaluates one complete held-out recording.

Held-out	Stage	Det-$F1_{50}$	AP_{50}	$AP_{50:95}$
A	Base	0.6063	0.4869	0.2815
	DSSC	0.9365	0.9036	0.6119
	DSSC+TPR	0.9425	0.9317	0.6519
	DSSC+TPR+PER	0.9508	0.9445	0.6974
B	Base	0.6848	0.5828	0.3696
	DSSC	0.8586	0.7818	0.5346
	DSSC+TPR	0.8554	0.8383	0.5904
	DSSC+TPR+PER	0.8574	0.8395	0.5960
C	Base	0.0002	7.09×10^{-6}	7.87×10^{-7}
	DSSC	0.1050	0.0162	0.0054
	DSSC+TPR	0.1056	0.0527	0.0173
	DSSC+TPR+PER	0.1181	0.0228	0.0082
D	Base	0.2214	0.0976	0.0399
	DSSC	0.5220	0.3430	0.1574
	DSSC+TPR	0.5322	0.4783	0.2366
	DSSC+TPR+PER	0.5637	0.5177	0.2736

The detailed metrics in Table S8 show different stage-wise trends. On A and D, all three metrics improve at every stage. On B, detection F1 decreases slightly at TPR while AP increases. On C, PER increases detection F1 from 0.1056 to 0.1181, while AP_{50} falls from 0.0527 to 0.0228 and $AP_{50:95}$ from 0.0173 to 0.0082. Detection F1 counts matches among the retained outputs, whereas AP also depends on their confidence ordering.

D Broader Impact

M^{2}E-UAV supports research on aerial perception for inspection, collision avoidance, and autonomous flight. Aerial target detection can also be used for unwanted monitoring or tracking. Deployment decisions should account for privacy, operational context, and the consequences of detection errors.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes].

   Justification: The abstract and introduction identify the dataset and benchmark as the paper's contributions and present the point-based model as a reference method. The main experiments compare reference detectors and analyze the cumulative point-reference progression; the Supplementary Material distinguishes temporal progressions, matched three-seed comparisons, and recording-held-out results.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes].

   Justification: The discussion identifies the benchmark's single-target scope and the need to extend evaluation to additional target types, carrier platforms, and operating environments.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A].

   Justification: The paper does not present formal theoretical results or proofs.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes].

   Justification: The manuscript documents the benchmark populations and splits, annotation and masking rules, point-reference architecture, sampling, losses, training settings, seeds, validation-only checkpoint selection, progressive recovery stages, and detection evaluation protocol for the main M2E-UAV experiments.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes].

   Justification: The project repository (`https://github.com/Wickyan/M2E-UAV`) publicly provides primary-method source code, installation instructions, a synthetic smoke test, and a dataset download link with its extraction code.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes].

   Justification: The manuscript specifies the sensor configuration, annotation and filtering rules, temporal split, point budget, sampler, optimizer, learning-rate schedule, loss, training duration, validation-based checkpoint selection, and evaluation thresholds for the reported point-reference experiments.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes].

   Justification: The supplementary matched baseline comparison reports mean and sample standard deviation over three independent seeds. The progressive stage results are presented as individual benchmark evaluations.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No].

   Justification: The paper reports the principal training configuration. Per-method compute hardware, runtime, and memory usage are not fully documented.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes].

   Justification: The authors have confirmed compliance with the NeurIPS Code of Ethics, including data-collection permissions, privacy handling, third-party asset use, and release arrangements. The work does not involve human-subject experiments.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes].

    Justification: Supplementary Section D discusses aerial-perception research benefits, unwanted monitoring or tracking, privacy, and consequences of detection errors.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A].

    Justification: The release consists of an outdoor UAV event dataset and reference-method source code, with no high-risk pretrained model release. Potential misuse and privacy considerations are discussed in Supplementary Section D.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes].

    Justification: Prior datasets and methods are cited, and the released primary-method code is licensed under Apache-2.0. A complete third-party software license inventory is not included in the manuscript.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes].

    Justification: Dataset construction, annotation semantics, split definitions, and evaluation are documented in Section 3 and the Supplementary Material. The project repository (`https://github.com/Wickyan/M2E-UAV`) provides dataset access instructions, source-module descriptions, installation and dependency instructions, and a synthetic smoke test; the primary-method code is licensed under Apache-2.0.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A].

    Justification: The paper does not involve crowdsourcing or human-subject experiments.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A].

    Justification: The paper does not involve human-subject research.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A].

    Justification: The core method does not use LLMs as a model component.