EVENT-BASED BATTING IMPACT ESTIMATION

Supplementary Material

Contents

1. OVERVIEW OF THE SUPPLEMENTARY MATERIAL

This supplementary material provides additional details that could not be included in the main paper due to space constraints. In Section 2, we visualize the experimental setup used for data collection and validate our ground truth strategy. Section 3 presents a detailed quantitative evaluation, including per-scene performance breakdowns and ablation studies. Section 4 describes the implementation details, including hyperparameters and hardware specifications. Finally, Section 5 provides an extended discussion on related work, a visual validation of baseline limitations, and a failure analysis of our method.

2. EXPERIMENTAL SETUP AND GT VALIDATION

2.1. Physical Configuration

The physical configuration of our data collection environment is illustrated in Fig. S1. We utilized a standard pitching machine to deliver balls to a batter within an indoor facility.

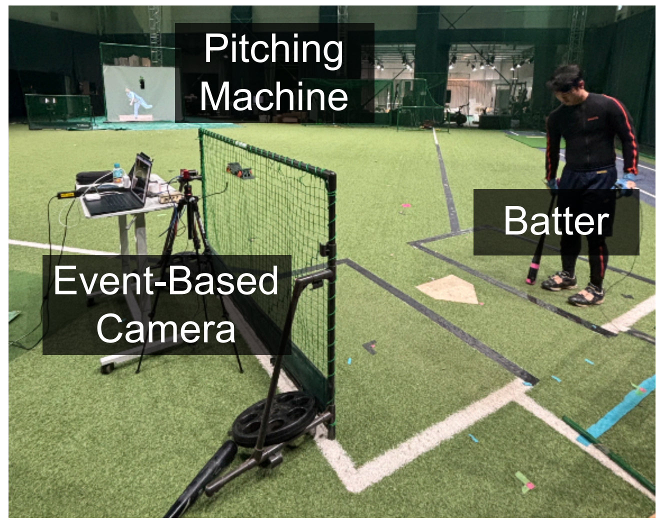

Fig. S1: Experimental setup used for data collection. A pitching machine delivers balls to a batter, while an event camera records the side view to capture high-speed motion.

An event camera was positioned to capture the side view of the trajectory to record high-temporal-resolution data.

2.2. Validation of Ground Truth Strategy

To justify the necessity of manual annotation over sensor-based automation, we conducted a preliminary validation using an IMU attached to the bat knob. The IMU signal was sampled at 1000 Hz, while the event camera recorded at 10,000 fps. The sensors were hardware-synchronized via an LED trigger.

IMU Impact Detection. We defined the impact timing from the IMU signal based on the magnitude of the acceleration vector. Specifically, the impact frame t_{IMU} was identified as the moment where the squared norm of the acceleration $(x^2 + y^2 + z^2)$ exceeded twice the value of the preceding frame. This thresholding method was chosen to detect the sharp initial shock of the impact.

Latency Analysis. We compared t_{IMU} against the manual ground truth t_{GT} obtained from the high-speed event stream. Across 28 valid trials, the time lag $(t_{\text{IMU}} - t_{\text{GT}})$ exhibited a significant, variable latency ranging from 0.40 ms to 7.07 ms (Mean: 3.32 ± 2.12 ms). This stochastic delay is likely due to mechanical propagation within the bat and internal sensor buffering. Since our method requires sub-millisecond accuracy, this jitter renders the IMU signal insufficient as a ground truth (Fig. S2). Consequently, we adopted manual annotation on 10,000 fps frames to ensure the highest temporal precision.

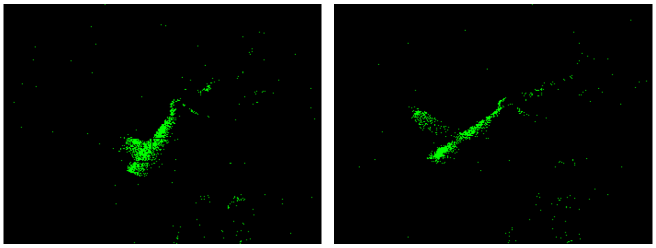

Fig. S2: Visual comparison at the maximum IMU latency (+7.07 ms). **Left:** The true impact moment annotated manually (t_{GT}). **Right:** The frame when the IMU triggered (t_{IMU}). The ball has already moved significantly away from the bat, demonstrating that the IMU signal is too slow for sub-millisecond evaluation.

3. ADDITIONAL QUANTITATIVE RESULTS

3.1. Per-Scene Evaluation

While the main paper reports the average performance across all datasets, Table S1 presents the detailed Mean Absolute Error (MAE) for each of the seven scenes individually. The results indicate that our method achieves consistent performance across varying ball speeds and trajectories. In particular, while baseline methods such as E2E-Spot and SAM 3 struggle significantly in complex scenarios (e.g., Scene 4 and Scene 5), our approach maintains a low error rate, demonstrating its robustness.

3.2. Ablation Studies

Component Analysis. Table S2 analyzes the contribution of each component in our proposed pipeline. The baseline with only forward refinement shows limited accuracy. Incorporating bidirectional refinement (Refine bwd) significantly reduces the MAE, proving the effectiveness of leveraging temporal context from both directions. Furthermore, the addition of the geometric circle loss ($\mathcal{L}_{\text{circ}}$) and smoothness loss ($\mathcal{L}_{\text{smooth}}$) further refines the trajectory, leading to the best performance.

Impact of Class Balancing Weights. Given the small size of the ball target, appropriate weighting for the Cross-Entropy loss is crucial. Table S3 investigates the impact of varying the ball class weight λ_{ball}. Standard weighting ($\lambda_{\text{ball}} = 1.0$) results in high MAE due to the dominance of the background class. Increasing λ_{ball} improves the Success Rate (SR). We empirically found that $\lambda_{\text{ball}} = 13.0$ yields the optimal trade-off, minimizing MAE while maintaining a consistently high SR. Excessive weighting (e.g., 20.0) degrades performance, likely due to an increase in false positives.

Parameter Sensitivity. We also evaluated the network's sensitivity to the event accumulation parameter. When testing different accumulation values of 5, 10, and 20, the resulting MAEs were 2.93, 0.77, and 1.65, respectively. This confirms that our chosen value of 10 is optimal for capturing the necessary temporal information while maintaining precision.

4. IMPLEMENTATION DETAILS

Our method was implemented using PyTorch. The input frames were resized to a resolution of 256×320 pixels. The network was trained using the Adam [1] optimizer with an initial learning rate of 5×10^{-4}, a batch size of 8, and a total of 50 epochs. The balancing weights for the total loss function were set to $\lambda_{\text{dice}} = 1.0$, $\lambda_{\text{ce}} = 0.5$, $\lambda_{\text{smooth}} = 0.1$, and $\lambda_{\text{circle}} = 0.05$.

To address the severe class imbalance inherent in the dataset where the ball occupies a very small portion of the frame compared to the background, we applied weighted Cross-Entropy loss. The class weights were assigned as 0.5, 1.0, and 13.0 for the background, bat, and ball classes, respectively. The selection process for these weights is discussed in Section 3. All experiments were conducted on a workstation equipped with an NVIDIA GeForce RTX 4070 GPU. Under this environment, the inference time for our proposed module is exceptionally fast at just 3 ms per frame. However, the overall speed depends on the preprocessing step required to generate the raw mask via the segmentation module. For instance, using SAM3 takes approximately 150 ms per frame. Regarding the data preparation, the manual annotation burden is relatively light. It takes approximately 30 sec to create a ground truth label for a single frame.

5. EXTENDED RELATED WORK AND DISCUSSION

This section provides a detailed comparison with existing literature, focusing on technical distinctions and domain-specific applications.

5.1. Comparison with Existing Event-based Methods

Regarding the novelty of our event representation and bidirectional reasoning, we provide further clarification against recently suggested works:

- **Event Representation:** Unlike motion-compensated tensors [3] or static 6-channel statistics [4], our dense event frames use sliding window accumulation. This overlapping approach maintains the high temporal resolution critical for capturing high-speed dynamics.

Table S1: Detailed quantitative evaluation per scene (MAE in ms). Our method outperforms baselines in most scenarios and maintains stability in challenging scenes (e.g., scene 5).

	scene1	**scene2**	**scene3**	**scene4**	**scene5**	**scene6**	**scene7**	**Avg.**
E2E-Spot	6.907	10.708	2.811	21.856	25.511	30.419	18.413	17.204
SAM 3	0.594	5.568	1.915	2.03	7.95	1.706	2.448	3.087
YOLO 11	0.574	**0.333**	2.915	0.946	6.1	1.866	**1.206**	2.056
Ours	**0.366**	0.433	**0.338**	**0.841**	**1.117**	**0.666**	1.5	**0.767**

Table S2: Ablation study on method components. “Refine (fwd/bwd)” denotes the refinement modules, and $\mathcal{L}_{\text{circ/smooth}}$ represents the auxiliary loss terms.

+ Refine (fwd)	**+ Refine (bwd)**	**+ $\mathcal{L}_{\text{circ}}$**	**+ $\mathcal{L}_{\text{smooth}}$**	**scene1**	**scene2**	**scene3**	**scene4**	**scene5**	**scene6**	**scene7**	**Avg.**
				0.594	5.568	1.915	2.03	7.95	1.706	<u>2.448</u>	3.087
✓				0.414	4.768	2.962	4.212	1.15	1.834	2.74	2.665
✓	✓			**0.354**	<u>1.608</u>	0.522	0.919	1.16	**0.57**	8.412	1.787
✓	✓	✓		0.474	1.617	**0.198**	**0.801**	**1.06**	0.966	7.88	<u>1.696</u>
✓	✓	✓	✓	<u>0.366</u>	**0.433**	<u>0.338</u>	<u>0.841</u>	<u>1.117</u>	<u>0.666</u>	**1.5**	**0.767**

Table S3: Ablation study on the loss weights for background, bat, and ball classes. We investigate the impact of varying λ_{ball} while fixing the others. The most balanced performance is archived when $\lambda_{\text{ball}} = 13$.

Weights			**Metrics**		
λ_{bg}	λ_{bat}	λ_{ball}	**MAE** (ms) $\downarrow$	**SR** ($< 1\sigma$) (%) $\uparrow$	**SR** ($< 2\sigma$) (%) $\uparrow$
1.0	1.0	1.0	2.523	40.0	75.0
0.5	1.0	1.0	2.139	42.5	72.5
0.5	1.0	10.0	1.54	**57.5**	**77.5**
0.5	1.0	13.0	**0.767**	52.5	**77.5**
0.5	1.0	20.0	2.192	47.5	72.5

- **Bidirectional Reasoning:** While some automotive segmentation models [3, 5] leverage temporal context, they primarily focus on feature propagation for mask consistency. Our novelty lies in using bidirectional tracking specifically for precise high-speed object time estimation.

- **Asynchronous Learning and Tracking:** Other approaches like EvDistill [6] use cross-modal distillation, and TETO [7] focuses on motion estimation. While relevant, our framework prioritizes a lightweight refinement-based approach for impact detection.

5.2. Domain-Specific Applications and Hardware Considerations

Several studies have explored event cameras in sports, particularly for ball spin estimation [8, 9]. While these works share the domain, our focus remains on the precise timing of impact events rather than rotational dynamics. Furthermore, while global shutter cameras offer advantages in certain scenarios, the inherent limitations compared to event cameras—such as motion blur and lower temporal resolution—justify our event-based approach for high-speed analysis, as discussed in the experimental comparisons by Holešovský et al. [10].

5.3. Visual Validation of Baseline Limitations

As a qualitative supplement to the quantitative analysis, we further examine the limitations of existing adaptations like EventSAM [2]. Fig. S3 provides a visual breakdown of its failure modes in high-speed scenarios.

As evident in the figure, since EventSAM processes each frame independently, the predicted masks exhibit significant temporal jitter. The bat mask fluctuates violently in shape or disappears entirely between consecutive frames, failing to track the object reliably. This lack of temporal context exacerbates susceptibility to environmental noise, confirming that frame-wise segmentation without domain-specific temporal adaptation is insufficient for robust microsecond-level analysis.

5.4. Failure Analysis of the Proposed Method

While our bidirectional refinement framework is generally robust against environmental noise, it can still fail when the initial segmentation masks are heavily degraded. Fig. S4 illustrates a typical failure case. If the raw masks generated by the baseline contain severe artifacts in both the forward and backward tracking directions, our refinement module may not

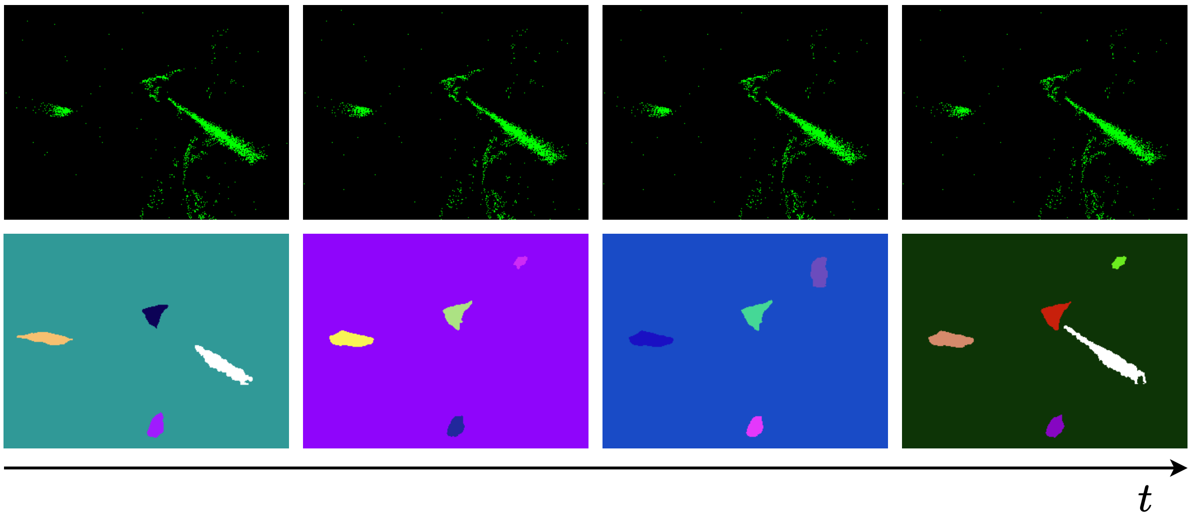

Fig. S3: Visual verification of the limitations of EventSAM [2]. **Top Row:** Input densified event frames. **Bottom Row:** Corresponding segmentation masks from EventSAM. The baseline exhibits temporal instability and susceptibility to environmental noise, confirming the necessity of our proposed bidirectional framework.

Fig. S4: A typical failure case of our refinement module. **Left / Center:** Raw bat masks from forward and backward tracking (using SAM 3) exhibiting severe noise. **Right:** The resulting refined mask. When significant artifacts exist simultaneously in both directional priors, the residual noise propagates to the final mask, decreasing the accuracy of the impact timing estimation.

fully eliminate this simultaneous noise. Consequently, the residual noise in the finalized mask can distort the object's spatial representation, leading to a shifted estimation of the impact timing.

6. REFERENCES

[1] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization.," in *ICLR*, 2015.

[2] Zhiwen Chen, Zhiyu Zhu, Yifan Zhang, Junhui Hou, Guangming Shi, and Jinjian Wu, "Segment any event streams via weighted adaptation of pivotal tokens," in *CVPR*, 2024.

[3] Zhen Yao, Xiaowen Ying, and Mooi Choo Chuah, "Rethinking rgb-event semantic segmentation with a novel bidirectional motion-enhanced event representation," 2025.

[4] Inigo Alonso and Ana C. Murillo, "Ev-segnet: Semantic segmentation for event-based cameras," in *CVPRW*, June 2019.

[5] Lakshmi Annamalai, Vignesh Ramanathan, and Chetan Singh Thakur, "Eventmask: A frame-free rapid human instance segmentation with event camera through constrained mask propagation," *IEEE Robotics and Automation Letters*, vol. 9, no. 4, pp. 3948–3955, 2024.

[6] Lin Wang, Yujeong Chae, Sung-Hoon Yoon, Tae-Kyun Kim, and Kuk-Jin Yoon, "Evdistill: Asynchronous events to end-task learning via bidirectional reconstruction-guided cross-modal knowledge distillation," in *CVPR*, 2021, pp. 608–619.

[7] Jini Yang, Eunbeen Hong, Soowon Son, Hyunkoo Lee, Sunghwan Hong, Sunok Kim, and Seungryong Kim, "Teto: Tracking events with teacher observation for motion estimation and frame interpolation," 2026.

[8] Takuya Nakabayashi, Kyota Higa, Masahiro Yamaguchi, Ryo Fujiwara, and Hideo Saito, "Event-based ball spin estimation in sports," in *CVPRW*, 2024, pp. 3367–3375.

[9] Thomas Gossard, Julian Krismer, Andreas Ziegler, Jonas Tebbe, and Andreas Zell, "Table tennis ball spin estimation with an event camera," in *CVPRW*, 2024, pp. 3347–3356.

[10] Ondřej Holešovský, Radoslav Škoviera, Václav Hlaváč, and Roman Vítek, "Experimental comparison between event and global shutter cameras," *Sensors*, vol. 21, no. 4, 2021.