# From Corners to Fiducial Tags: Revisiting Checkerboard Calibration for Event Cameras

Supplementary Material

## Overview

In this supplementary material, we provide additional studies and results as follows:

- **Full description of our analysis:** We provide a full description of our analysis, which is about ER on checkerboard corner and edge described in Sec. 3.2. Description contains the details of previous works and our analysis.
- **Additional studies for checkerboard:** We explain which patches are filtered out during the corner initialization step and how they are removed. Then, we provide additional quantitative and qualitative results.
- **Additional studies for AprilTag:** We present results for binarization of various tag IDs and their corresponding corner indexing. In addition, we provide extra qualitative results for AprilTag detection as well as a calibration result using AprilTag.

## A. Full Description of the Analysis in Sec. 3.2

### A.1. Intensity Model on Checkerboard Corner

The ideal intensity distribution of checkerboard edge is usually defined with step function $\rho(\cdot)$. Then, checkerboard corner can be represented as the intersection of two edges [7]:

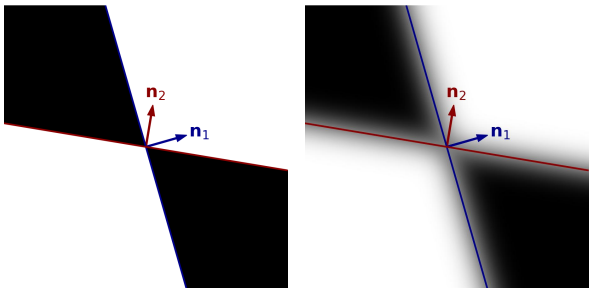
$$\hat{I}(\mathbf{x}) = \rho(\mathbf{n}_1^\top \mathbf{x})\rho(\mathbf{n}_2^\top \mathbf{x}), \quad \text{where } \rho(x) = \begin{cases} +1 & \text{if } x > 0, \\ -1 & \text{otherwise,} \end{cases} \tag{A1}$$

where $\mathbf{n}_i$ denotes unit normals to checkerboard edges (see Fig. A1 (a)) and, in this model, the ideal checkerboard corner is located $\mathbf{x} = \mathbf{0}$.

However, when frame-based camera captures ideal checkerboard $\hat{I}(\mathbf{x})$, there is a blurring effect by the lens (see Fig. A1 (b)). To define this blurring effect, Krüger *et al*. [7] consider the Point Spread Function (PSF) to be an ideal circular Gaussian filter $g$ characterized by the parameter $\tilde{\sigma}$ [11]:

$$\tilde{I}(\mathbf{x}) = g(||\mathbf{x}||_2, \tilde{\sigma}) * \hat{I}(\mathbf{x}), \quad \text{where } g(x, \tilde{\sigma}) = \frac{1}{\sqrt{2\pi}\tilde{\sigma}} \exp\left(\frac{x^2}{2\tilde{\sigma}^2}\right). \tag{A2}$$

Because the Gaussian kernel $g$ is separable, convolving a 1D step function $\rho(\cdot)$ with $g(\cdot, \tilde{\sigma})$ reduces to integrating the Gaussian along the edge normal direction. Since the integral of a Gaussian yields the error function, this step response becomes




**(a) Ideal** **(b) Blurred**

Figure A1. **Illustration of checkerboard intensity.**

$$g(x, \tilde{\sigma}) * \rho(x) = \text{erf}\left(\frac{x}{\sqrt{2}\tilde{\sigma}}\right), \tag{A3}$$

where $\text{erf}(x) = (2/\sqrt{\pi}) \int_0^x e^{-t^2}\, dt$. Hence, $\tilde{I}$ can be represented as:

$$\tilde{I}(\mathbf{x}) = G(\mathbf{n}_1^\top \mathbf{x}, \tilde{\sigma})\, G(\mathbf{n}_2^\top \mathbf{x}, \tilde{\sigma}), \quad \text{where } G(x, \tilde{\sigma}) = \text{erf}\left(\frac{x}{\sqrt{2}\tilde{\sigma}}\right). \tag{A4}$$

Finally, Krüger *et al*. [7] define the intensity model of the checkerboard corner $\tilde{I}$ by approximating $G(x, \tilde{\sigma})$ with a centered sigmoid $S(\cdot, \sigma)$, corresponding to Eq. (2):

$$\tilde{I}(\mathbf{x}) \approx S(\mathbf{n}_1^\top \mathbf{x}, \sigma) S(\mathbf{n}_2^\top \mathbf{x}, \sigma), \quad \text{where } S(x, \sigma) = \frac{2}{1 + e^{-x/\sigma}} - 1 \tag{A5}$$

where $\sigma = \tilde{\sigma}\sqrt{\pi/8}$ produces an equivalent slope of $G(x, \tilde{\sigma})$ and $S(x, \sigma)$ at $x = 0$. Note that since construction $S(x, \sigma) \in (-1, 1)$, $\tilde{I}(\mathbf{x}) \in [-1, 1]$ represents the idealized contrast pattern of the checkerboard, rather than the actual brightness observed by the camera.

### A.2. Event Generation Rate

Event $e_k$ is triggered based on log intensity change with constant threshold [5]:

$$\Delta L(\mathbf{x_k}, t_k) := L(\mathbf{x}_k, t_k) - L(\mathbf{x}_k, t_k - \Delta t_k) = p_k C, \tag{A6}$$

where $L(\mathbf{x}, t) = \log I(\mathbf{x}, t)$ denotes the logarithmic image intensity, $C > 0$ is the constant threshold, and $\Delta t_k$ represents the duration since the pixel last generated an event.

To analyze Event Rate (ER), it is necessary to first characterize how events accumulate over a given time interval.

Since the rate is defined in terms of the number of events generated within a temporal window, it is necessary to relate the accumulated events to the underlying log-intensity signal that drives event generation. Using the triggering condition in Eq. (A6), an event $e_k = (\mathbf{x}_k, t_k, p_k)$ can be written in distributional form as [9]:

$$e_k(\mathbf{x}, t) = p_k C\, \delta(t - t_k)\, \delta(\mathbf{x} - \mathbf{x}_k), \tag{A7}$$

where $\delta(\cdot)$ indicates the Dirac delta function. Over a time interval of length $t$, the events accumulated at pixel $\mathbf{x}$ can be written as:

$$\int_0^t \sum_k e_k(\mathbf{x}, \tau)\, d\tau = L(\mathbf{x}, t) - L(\mathbf{x}, 0) + \int_0^t \eta(\mathbf{x}, \tau)\, d\tau, \tag{A8}$$

where $\eta(\mathbf{x}, t)$ denotes sensor noise [5].

From the event models described in Eq. (A7) and Eq. (A8), Lin *et al*. [8] establish a relationship between the Event Rate (ER) and the image gradient. To establish relationship, they first define ER as follows:

$$R_e(\mathbf{x}, t, \Delta t) = \frac{\int_{t-\Delta t/2}^{t+\Delta t/2} \sum_k \delta(t - t_k)\delta(\mathbf{x} - \mathbf{x}_k) d\tau}{\Delta t}, \tag{A9}$$

where $R_e(\mathbf{x}, t, \Delta t)$ is ER within time interval $[t{-}\Delta t/2,\ t{+}\Delta t/2]$, which indicates the average number of events accumulated within a time interval of duration $\Delta t$. In addition, they formulate the intensity changing rate within time interval $[t{-}\Delta t/2,\ t{+}\Delta t/2]$, omitting noise tern in Eq. (A8):

$$\begin{aligned} \frac{\Delta L(\mathbf{x}, t)}{\Delta t} &= \frac{L(\mathbf{x}, t + \Delta t/2) - L(\mathbf{x}, t - \Delta t/2)}{\Delta t} \\ &= \frac{\int_{t-\Delta t/2}^{t+\Delta t/2} \sum_k e_k(\mathbf{x}, t) d\tau}{\Delta t}. \end{aligned} \tag{A10}$$

By taking the magnitude of Eq. (A10) and substituting the event model $e_k(\mathbf{x}, t)$ using Eq. (A7), the resulting intensity changing rate magnitude can be approximated by the event rate scaled by the constant $C$:

$$\begin{aligned} \left| \frac{\Delta L(\mathbf{x}, t)}{\Delta t} \right| &\approx \frac{C \int_{t-\Delta t/2}^{t+\Delta t/2} \sum_k \delta(t - t_k)) \delta(\mathbf{x} - \mathbf{x}_k) d\tau}{\Delta t} \\ &= C \cdot R_e(\mathbf{x}, t, \Delta t), \end{aligned} \tag{A11}$$

where the polarity $p_k$ vanishes since the magnitude operator eliminates its sign.

Using the brightness constancy assumption, the intensity changing rate also can be approximated as [6]:

$$\frac{\Delta L(\mathbf{x}, t)}{\Delta t} \approx -\nabla L(\mathbf{x}, t) \cdot \mathbf{v}, \tag{A12}$$

where $\mathbf{v}$ is the constant motion vector on image plane [6]. Finally, combining Eq. (A11) and Eq. (A12), the event rate and gradient value are related as follows, corresponding to Eq. (3):

$$R_e(\mathbf{x}, t, \Delta t) \approx \frac{1}{C} |\nabla L(\mathbf{x}, t) \cdot \mathbf{v}|. \tag{A13}$$

For simplicity, we omit the temporal variable $t$ in the following analysis, using $L(\mathbf{x})$ and $R_e(\mathbf{x})$ instead of $L(\mathbf{x}, t)$ and $R_e(\mathbf{x}, t, \Delta t)$ respectively.

### A.3. Event Rate on Checkerboard Corner

To analyze Event Rate on checkerboard corner, we combine Eq. (A5) and Eq. (A13). Before combining them, we first reformulate centered sigmoid $S(x, \sigma)$, noting that $S(x, \sigma)$ can be expressed as a hyperbolic tangent function,

$$\begin{aligned} S(x, \sigma) &= \frac{2}{1 + e^{-x/\sigma}} - 1 \\ &= \frac{1 - e^{-x/\sigma}}{1 + e^{-x/\sigma}} \\ &= \frac{e^{x/2\sigma} - e^{-x/2\sigma}}{e^{x/2\sigma} + e^{-x/2\sigma}} \\ &= \tanh\left(\frac{x}{2\sigma}\right), \end{aligned} \tag{A14}$$

corresponding to Eq. (4). Then, we recall that the hyperbolic tangent function has the following Taylor expansion:

$$\tanh(x) = x - \frac{1}{3}x^3 + \frac{2}{15}x^5 - \cdots, \text{ when } |x| < \frac{\pi}{2}. \tag{A15}$$

Since we are primarily interested in the neighborhood of the ideal checkerboard corner, where its region is in $|x| \ll 1$, the Taylor expansion of $\tanh(x)$ is valid, and the higher-order terms can be neglected as

$$S(x, \sigma) = \tanh\left(\frac{x}{2\sigma}\right) \approx \frac{x}{2\sigma} + \mathcal{O}(x^3) \approx \frac{x}{2\sigma}, \tag{A16}$$

where $\mathcal{O}(x^3)$ denotes the third-order remainder in the Taylor expansion.

Based on this convention, we reformulate Eq. (A5) using the approximation in Eq. (A16), deriving its local analytical form around the checkerboard corner:

$$\tilde{I}(\mathbf{x}) \approx \frac{1}{4\sigma^2}(\mathbf{n}_1^\top \mathbf{x})(\mathbf{n}_2^\top \mathbf{x}), \quad \nabla \tilde{I}(\mathbf{x}) \approx G\,\mathbf{x}, \tag{A17}$$

where $G = \frac{1}{4\sigma^2}(\mathbf{n}_1\mathbf{n}_2^\top + \mathbf{n}_2\mathbf{n}_1^\top)$ is symmetric. Since $\tilde{I}(\mathbf{x})$ represents an idealized contrast pattern rather than an actual brightness value, we obtain the image intensity via a linear camera response applied to $\tilde{I}(\mathbf{x})$ [7]:

$$I(\mathbf{x}) = I_0 + \alpha\, \tilde{I}(\mathbf{x}). \tag{A18}$$

where $\alpha$ and $I_0$ correspond to the camera gain and offset, respectively. Through this linear response, $\tilde{I}(\mathbf{x}) \in [-1, 1]$ is mapped to the image intensity $I(\mathbf{x})$ within the dynamic range of camera, typically $[0, 255]$.

Having established the local intensity model of the checkerboard corner, we now analyze the event rate on checkerboard corner by substituting Eq. (A18) into the event rate formulation in Eq. (A13). Since $L(\mathbf{x}) = \log I(\mathbf{x})$, we have

$$\begin{aligned} \nabla L(\mathbf{x}) &= \frac{\nabla I(\mathbf{x})}{I(\mathbf{x})} \\ &= \frac{\alpha}{I_0 + \alpha\,\tilde{I}(\mathbf{x})} \nabla \tilde{I}(\mathbf{x}) \\ &\approx \frac{\alpha}{I_0 + \alpha\,\tilde{I}(\mathbf{x})} G\mathbf{x}, \end{aligned} \tag{A19}$$

corresponding to Eq. (9).

Since $\tilde{I}(\mathbf{x}) \approx 0$ around the corner, the term $\alpha/(I_0 + \alpha\,\tilde{I}(\mathbf{x}))$ can be approximated by $\alpha/I_0$ and treated as constant. Therefore, Eq. (A13) can be rewritten as

$$\begin{aligned} R_e(\mathbf{x}) &\approx \frac{1}{C} \left|\nabla L(\mathbf{x}) \cdot \mathbf{v}\right| \\ &\approx \gamma \left|(G\mathbf{x}) \cdot \mathbf{v}\right| = \gamma \left|(G\mathbf{v})^\top \mathbf{x}\right|, \end{aligned} \tag{A20}$$

where $\gamma = \alpha/CI_0$ is a constant factor and Eq. (A20) corresponds Eq. (10).

To interpret the meaning of Eq. (A20), we first investigate whether $G$ can be zero. Note that, in any valid checkerboard structure, the two edge normals cannot be parallel and are therefore linearly independent. With this geometric constraint, we examine whether $G$ can vanish by multiplying $G$ by $\mathbf{n}_1$ (or $\mathbf{n}_2$):

$$\begin{aligned} G\mathbf{n}_1 &= \frac{1}{4\sigma^2}(\mathbf{n}_1\mathbf{n}_2^\top + \mathbf{n}_2\mathbf{n}_1^\top)\mathbf{n}_1 \\ &= \frac{1}{4\sigma^2}\Big[(\mathbf{n}_1\mathbf{n}_2^\top)\mathbf{n}_1 + (\mathbf{n}_2\mathbf{n}_1^\top)\mathbf{n}_1\Big] \\ &= \frac{1}{4\sigma^2}\Big[(\mathbf{n}_2^\top\mathbf{n}_1)\mathbf{n}_1 + \mathbf{n}_2\Big]. \end{aligned} \tag{A21}$$

If $G$ were zero, then $G\mathbf{n}_1 = \mathbf{0}$ must hold, which means that the following equation holds:

$$(\mathbf{n}_2^\top\mathbf{n}_1)\mathbf{n}_1 + \mathbf{n}_2 = \mathbf{0}. \tag{A22}$$

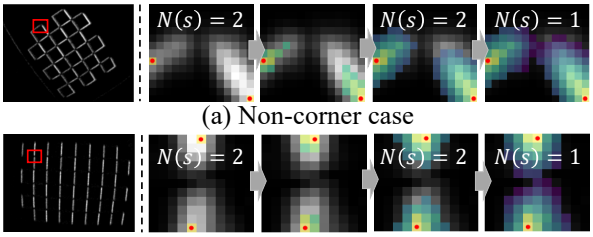
Then we can obtain following relationship between $\mathbf{n}_1$ and $\mathbf{n}_2$:

$$\begin{aligned} \mathbf{n}_2 &= -(\mathbf{n}_2^\top\mathbf{n}_1)\mathbf{n}_1 \\ &= \lambda\mathbf{n}_1, \quad \text{for some scalar } \lambda. \end{aligned} \tag{A23}$$

It is logically equivalent to $\mathbf{n}_2 \in \mathrm{span}\{\mathbf{n}_1\}$, which directly contradicts the valid checkerboard structure that requires the two edge normals to be linearly independent. Hence, $G$ cannot be zero for any valid checkerboard structure.

Since $G$ is nonzero for a valid checkerboard corner, the event rate at the exact corner satisfies $R_e = 0$ for any motion vector $\mathbf{v}$. Thus, from Eq. (A20), we find that

$$\mathbf{x} = \mathbf{0} \implies R_e(\mathbf{0}) = 0. \tag{A24}$$



(a) Non-corner case

(b) Single-edge case

Figure B2. **Filtered cases in pixel propagation.**

This result holds locally near a checkerboard corner, where the image gradients are modeled as a pair of intersecting edges. In contrast, the intensity model of the checkerboard edge can be modeled as

$$\tilde{I}(\mathbf{x}) \approx \frac{1}{2\sigma}(\mathbf{n}^\top\mathbf{x}), \quad \nabla\tilde{I}(\mathbf{x}) \approx \frac{\mathbf{n}}{2\sigma}. \tag{A25}$$

Here, the checkerboard edge corresponds to the set of points $\mathbf{x}$ that satisfy $\mathbf{n}^\top\mathbf{x} = \mathbf{0}$. Then we can calculate $\nabla L(\mathbf{x})$ in edge case as follows:

$$\nabla L(\mathbf{x}) \approx \frac{\alpha}{I_0 + \alpha\,\tilde{I}(\mathbf{x})} \frac{\mathbf{n}}{2\sigma}. \tag{A26}$$

As in the checkerboard corner case, since $\tilde{I}(\mathbf{x}) \approx 0$ near the edge, we can formulate ER on the checkerboard edge as follows:

$$\begin{aligned} R_e(\mathbf{x}) &\approx \frac{1}{C} \left|\nabla L(\mathbf{x}) \cdot \mathbf{v}\right| \\ &\approx \kappa\left|\mathbf{n} \cdot \mathbf{v}\right|, \quad \text{where } \kappa = \gamma/2\sigma, \end{aligned} \tag{A27}$$

and it corresponds to Eq. (13). This expression indicates that events are reliably generated along the edge as long as the motion vector is not parallel to the edge direction. Conversely, when the motion is orthogonal to the edge normal, no events are produced. In other words, events may be triggered only on one side of the checkerboard edge under such motion [14].

## B. Additional Studies for Checkerboard

### B.1. Filtered Cases in Pixel Propagation

As described in Sec. 3.4, in corner initialization process, we use the quantity $N(s)$ to filter out invalid patches. This section explains which cases are filtered out and why.

We broadly categorize discarded patches into two representative types: non-corner case and single-edge case (see Fig. B2).

Patches with $N(s) \neq 4$ are filtered out as invalid patches. (a) non-corner case and (b) single-edge case are common cases. During seed propagation, $N(s)$ does not increase and instead only decreases, so these patches are not selected as valid ones. (a) non-corner case corresponds to regions that

Table B1. **Intrinsics results ($\mu \pm \sigma$) on synthetic data with GT.**

| Method | $f_x$ | $f_y$ | $c_x$ | $c_y$ |
|---|---|---|---|---|
| GT | 250 | 250 | 150 | 100 |
| E-Calib* | **250.16**±7.64 | 249.04±8.38 | 147.62±1.64 | 98.56±1.45 |
| eKalibr* | 251.20±8.06 | 248.81±4.69 | 151.01±0.91 | 97.95±2.75 |
| E2VID | 249.51±3.02 | 249.59±3.28 | 149.36±0.80 | 99.40±1.38 |
| HyperE2VID | 248.27±8.44 | 248.15±8.63 | 149.18±2.40 | 100.10±1.66 |
| Ours | 250.38±**2.60** | **250.00**±**2.36** | **149.39**±**0.44** | **99.93**±**0.69** |

*: circular-based method; E-Calib shows the $f_x$ mean closest to GT but high $\sigma$.

do not contain a true corner, yielding $N(s) = 2$, whereas (b) single-edge case corresponds to camera motion parallel to the checkerboard such that no corner is observed, also resulting in $N(s) = 2$.

## B.2. Quantitative results

In this section, we present additional quantitative results on synthetic and real-world data, including comparisons with circular patterns, detection ratio analysis, and runtime measurements. Since E2Calib [10] adopts a reconstruction-based pipeline in which event-to-image conversion is followed by standard calibration (e.g., Kalibr [4] or OpenCV [1]), replacing E2VID [13] with a more recent reconstruction network does not change the framework. Hence, we additionally evaluate the pipeline using HyperE2VID [3] to ensure a fair comparison with recent reconstruction models.

To compare our method with circle-grid calibration methods [2, 14], we synthesize data in Blender using fixed calibration targets and randomly generated camera trajectories, varying only the calibration pattern. Event streams are simulated with ESIM [12]. As shown in Tab. B1, Our method estimates parameters close to GT with lower standard deviations averaged over 10 sequences.

For real-world experiments, we extend evaluation in Sec. 4.2 by incorporating HyperE2VID and report only the standard deviation. Using the same $7 \times 10$ checkerboard dataset in the paper, we set the sampling number $B$ to 15 due to fewer corresponding images, which increases the standard deviations compared to the paper; nevertheless, our method still achieves the lowest values (see Tab. B2).

We further compare the detection ratio and runtime with E2VID. Although E2VID detects more images, calibration accuracy is governed by corner quality and pose diversity, and our fewer but higher-quality detections yield better results (see Tab. B3). Treating grayscale reconstruction in E2VID and IWE generation as the same event-processing step, we measure runtime of each step. Our method remains substantially more efficient (see Tab. B4).

## B.3. Qualitative results

We provide additional qualitative results for our corner detection method in this section (see Fig. B3). These examples illustrate its robustness across diverse motion trajectories and checkerboard poses.

As a qualitative complement to Tab. 2, we further provide qualitative results on checkerboard corner location in Fig. B4. Our method produces corner locations that are nearly coincident with the pseudo ground truth (*i.e.,* frame-

Table B2. **Intrinsics results ($\pm\sigma$) on real data.**

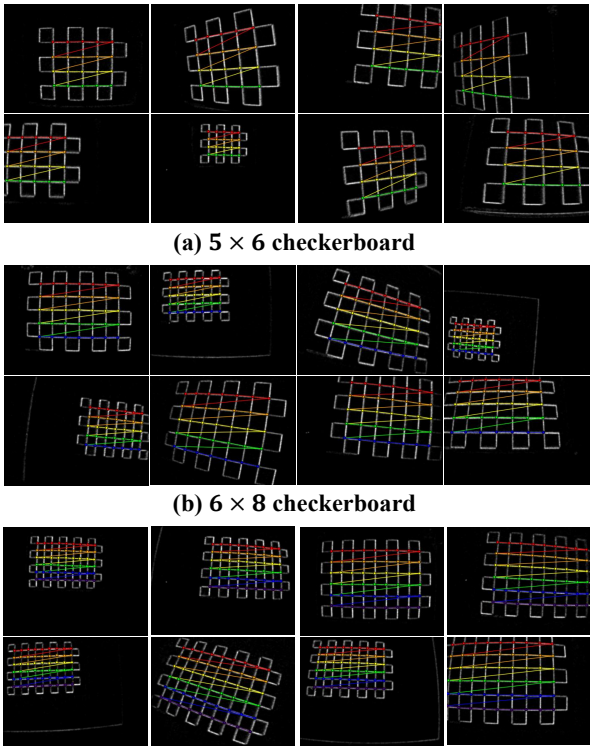
| | $\sigma f_x$ | $\sigma f_y$ | $\sigma c_x$ | $\sigma c_y$ | $\sigma k_1$ |
|---|---|---|---|---|---|
| E2VID | 8.51 | 8.47 | 1.59 | 2.79 | 0.018 |
| HyperE2VID | 9.45 | 9.26 | 1.46 | 4.38 | 0.020 |
| Ours | **3.79** | **3.80** | **0.73** | **1.31** | **0.009** |

Table B3. **Detection ratio and intrinsic results.**

| Method | #img | $f_x$ | $f_y$ | $c_x$ | $c_y$ |
|---|---|---|---|---|---|
| GT | 300 | 250 | 250 | 150 | 100 |
| E2VID | **187** | 244.12 | 243.37 | 148.35 | 103.21 |
| Ours | 99 | **247.38** | **248.08** | **150.31** | **99.39** |

Table B4. **Comparison of runtime (sec).**

| Method | Event processing | Corner detection | Total |
|---|---|---|---|
| E2VID | 182.96 | **59.86** | 242.82 |
| Ours | **35.15** | 70.58 | **105.73** |



**(a) $5 \times 6$ checkerboard**

**(b) $6 \times 8$ checkerboard**

**(c) $7 \times 10$ checkerboard**

Figure B3. **Qualitative results for corner detection**

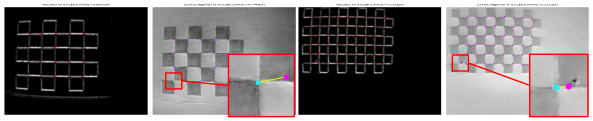



Figure B4. **Comparison with E2VID** Cyan denotes pseudo ground-truth corners obtained from frame-based detection; pink corresponds to E2VID, and red to our method.

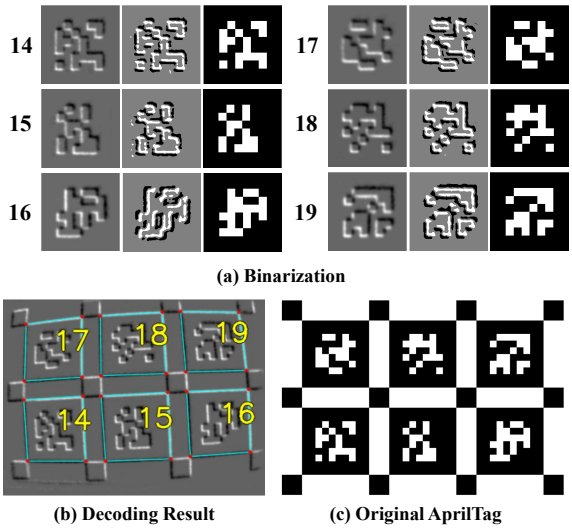



Figure C5. **Qualitative results of binarization.**

based), whereas E2VID exhibits systematic offsets and less consistent alignment, particularly around blurred or low-contrast regions.

### B.4. AprilTag Detection

We present various matching results for AprilTag detection in various tag IDs. As shown in Fig. C5, our binarization of the quad regions extracted from the pIWE remains reliable across different tag IDs despite variations in their visual patterns. Although a few grid cells may be misclassified during binarization, the resulting patterns remain within the error tolerance margin inherently built into the AprilTag design, which is intended to accommodate such small perturbations. Additional qualitative results are illustrated in Fig. C6, where all visible AprilTags are successfully detected even when the board is rotated.

## C. Additional Studies for AprilTag

### C.1. Corner Indexing from AprilTag

The advantage of AprilTag in camera calibration lies in its ability to create diverse observation sets, even for partial observations, by assigning a unique number to each corner. This correspondence eliminates the ambiguity that typically arises in calibration using checkerboard or circle-grid,

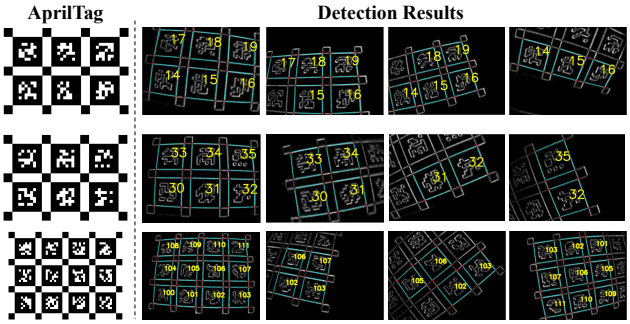



Figure C6. **Qualitative results for AprilTag detection.** The rightmost example in the last row illustrates a 180° rotation.


Figure C7. **AprilTag detection and corner indexing results for various rotations.** Each quad and detected tag ID is colored cyan and yellow, respectively. The unique number for the corner is colored the same as its corner color.

Table C5. **Calibration results using AprilTag.**

| Parameters | Values |
|---|---|
| Intrinsics $(f_x, f_y, c_x, c_y)$ | 332.25, 332.31, 179.10, 137.02 |
| Distortion $(k_1, k_2, p_1, p_2)$ | -0.3265, 0.1596, -0.0042, 0.00138 |
| Reprojection error | 0.516 |

where corner ordering must be inferred from geometry.

As shown in Fig. C7, our detection algorithm robustly identifies each corner even under rotation, occlusion, and other extreme conditions. This property enables stable calibration input without requiring an additional corner ordering process.

### C.2. Calibration using AprilTag

Using a robust corner indexing process, we perform event camera calibration with AprilTag. As shown in Tab. C5, our pipeline yields reliable parameter estimates when applied to AprilTag.

## References

[1] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, 2008.

[2] Shuolong Chen, Xingxing Li, Liu Yuan, and Ziao Liu. ekalibr: Dynamic intrinsic calibration for event cameras

from first principles of events. *IEEE RA-L*, 10(7):7094–7101, 2025.

[3] Burak Ercan, Onur Eker, Canberk Saglam, Aykut Erdem, and Erkut Erdem. HyperE2VID: Improving event-based video reconstruction via hypernetworks. *IEEE TIP*, 33: 1826–1837, 2024.

[4] Paul Furgale, Joern Rehder, and Roland Siegwart. Unified temporal and spatial calibration for multi-sensor systems. In *IEEE IROS*, 2013.

[5] Guillermo Gallego et al. Event-based vision: A survey. *IEEE TPAMI*, 44(1):154–180, 2020.

[6] Daniel Gehrig, Henri Rebecq, Guillermo Gallego, and Davide Scaramuzza. EKLT: Asynchronous, photometric feature tracking using events and frames. *IJCV*, 128(3):601–618, 2020.

[7] Lars Krüger and Christian Wöhler. Accurate chequerboard corner localisation for camera calibration. *Pattern Recognition Letters*, 32(10):1428–1435, 2011.

[8] Shijie Lin, Yinqiang Zhang, Lei Yu, Bin Zhou, Xiaowei Luo, and Jia Pan. Autofocus for event cameras. In *CVPR*, 2022.

[9] Elias Mueggler, Henri Rebecq, Guillermo Gallego, Tobi Delbruck, and Davide Scaramuzza. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *IJRR*, 36(2):142–149, 2017.

[10] Manasi Muglikar, Mathias Gehrig, Daniel Gehrig, and Davide Scaramuzza. How to calibrate your event camera. In *CVPRW*, 2021.

[11] Alex Paul Pentland. A new sense for depth of field. *IEEE TPAMI*, PAMI-9(4):523–531, 1987.

[12] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. ESIM: an open event camera simulator. In *CoRL*, 2018.

[13] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE TPAMI*, 43(6):1964–1980, 2021.

[14] Mohammed Salah, Abdulla Ayyad, Muhammad Humais, Daniel Gehrig, Abdelqader Abusafieh, Lakmal Seneviratne, Davide Scaramuzza, and Yahya Zweiri. E-calib: A fast, robust, and accurate calibration toolbox for event cameras. *IEEE TIP*, 33:3977–3990, 2024.