

# Event-based Gaze Control System for Accurate Real-time Spin Estimation in Professional Ball Games

Yunpu Hu<sup>1\*</sup> , Fabian Schilling<sup>2\*</sup> , Valentina Cavinato<sup>3\*</sup> ,  
Asude Aydin<sup>2</sup> , Agis Politis<sup>2</sup>, Ricardo Tapiador Morales<sup>2</sup> ,  
Kirk Y.W. Scheper<sup>3</sup> , Peter Dürr<sup>2</sup> , and Naoya Takahashi<sup>2</sup>  

<sup>1</sup> Sony AI, Tokyo, Japan

<sup>2</sup> Sony AI, Zürich, Switzerland

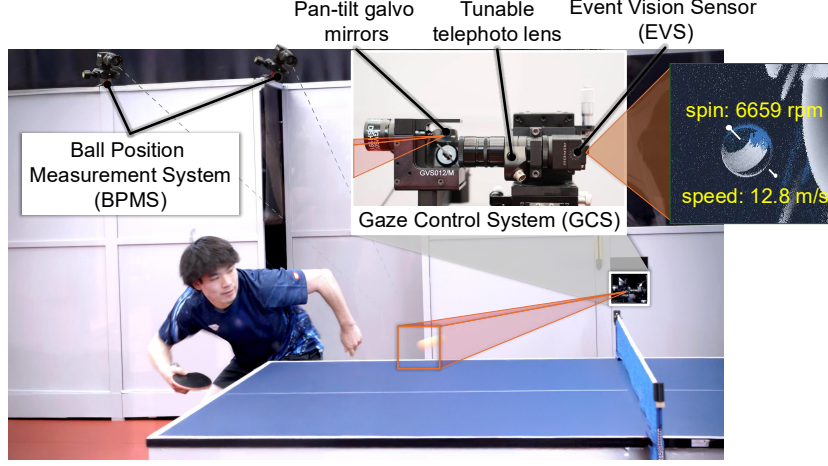
<sup>3</sup> Sony Advanced Visual Sensing, Sony Europe Ltd., Zurich, Switzerland.

**Abstract.** Spin plays a crucial role in many ball sports due to its effect on the trajectory of the ball. Vision-based estimation of the ball’s spin during a game with conventional cameras is challenging due to the ball’s small size, high speed, and fast rotation. To address these challenges, we propose an event-based active vision system that can track unmodified balls and measure their spin in real time. The system consists of an event camera for its high temporal resolution and minimal motion blur, high-speed pan/tilt galvanometer mirrors to keep the ball in the field of view, and a low-latency focus-tunable telephoto lens to increase the spatial resolution on the ball and keep it in focus. To track the ball, we use a hybrid approach that combines 2D event-based detection for centering and 3D positions from a ball localization system for re-initialization. For high-accuracy spin estimation, we propose an offline method that performs contrast maximization on the sphere (s-CMax). This method achieves state-of-the-art accuracy on static balls across multiple sports (table tennis, baseball, tennis, and golf), with mean magnitude and axis errors of 1.2% and 1.5 degrees, respectively. We then develop a low-latency online method for table tennis as a case study in real-time applications. This method uses an uncertainty-aware convolutional neural network trained on pseudo-ground-truth spin labels from the offline approach, combined with a GPU-accelerated batch implementation of contrast maximization for refinement. We demonstrate reliable tracking and spin estimation with a three-view setup during professional table tennis matches, with high accuracy (8.8% magnitude and 6.4 degrees axis mismatch w.r.t. the offline method), 3 ms latency, and 750 Hz throughput.

**Keywords:** Event-based cameras · Object pose estimation and tracking  
· Computer vision for robotics

---

\*These authors contributed equally.



**Fig. 1:** The Gaze Control System (GCS) actively tracks the ball and measures its spin.

## 1 Introduction

Spin plays a key role in many ball sports due to its effect on the ball’s flight and contact dynamics. Athletes routinely apply spin to control the ball’s flight path, deceive opponents, or gain a tactical advantage. Accurate spin estimation is thus of vital importance for performance analysis, sports statistics, physics modeling [6], trajectory prediction [3], and competitive robotic gameplay [7]. For example, in table tennis, rallies can feature balls spinning at magnitudes of up to 9 krpm [38]. In baseball, curve balls can reach spin rates up to 2.5 krpm [18], tennis serves can impart spins up to 5 krpm [22], and golf wedge shots can exceed 10 krpm [40].

Vision-based ball spin estimation is challenging mainly for two reasons. First, balls are typically small, spanning only a few pixels in the field of view (FoV) of a wide-angle camera and making surface features hard to observe; a narrow-angle camera, in contrast, suffers from shallow depth-of-field and limited coverage. Second, balls’ motion combines high linear velocities with high spins. For a frame-based camera, the high image-plane velocity can introduce motion blur even at short exposures, and the spin often exceeds the Nyquist limit set by the frame rate, causing magnitude underestimation through temporal aliasing. While an event-based vision sensor (EVS) [10] can alleviate blur and aliasing, the high linear velocity typically pollutes the events and obscures the spin.

We address the above challenges with the gaze control system (GCS), an event-based active vision system that tracks unmodified balls and measures their spin with high accuracy in real time. We employ an EVS for its high temporal resolution, which avoids aliasing and reduces motion blur. We equip the EVS with a telephoto lens for higher spatial resolution on the ball surface. The focal power of the lens is electrically tuned with 3 ms response time to keep the ball in sharp focus despite the shallow depth of field resulting from the long focal length.

The system utilizes two precisely controlled pan/tilt galvanometer mirrors that actively control the camera’s gaze direction, to keep the fast-moving ball in its FoV, and to ensure that the generated events primarily encode the ball spin, by compensating for the ball’s translational motion with the mirror motion.

Despite its advantages, operating the GCS in a highly dynamic setting poses unique challenges. The ball’s high speed and the telephoto lens’s narrow FoV make robust tracking difficult, especially when the ball abruptly changes direction upon contact (e.g., with a racket, table, bat, or club). We address this with a hybrid tracking strategy: a ball position measurement system (BPMS) controls the mirrors and tunable lens, while the 2D ball position from the active EVS view provides secondary feedback, compensating for latency and noise in the BPMS, keeping the ball centered, and preventing it from exiting the FoV.

To estimate the spin from these events, we build on contrast maximization [11,12], a model-based framework that recovers motion parameters by maximizing the sharpness of an image of warped events (IWE) [13]. While recent works [16,26,31] have explored event-based spin estimation, they employ static, wide-angle cameras, which limits spatial resolution on the ball and conflates translational and rotational motion in the captured events. Since the GCS compensates for the ball’s translational motion via active mirror tracking, the events captured by the EVS primarily encode the ball’s rotation, which addresses the conflation problem. Nevertheless, prior methods apply contrast maximization on a planar image projection, which cannot faithfully represent the geometry of a spinning sphere: events generated across the curved ball surface get projected onto a flat plane, causing depth ambiguities and concentrating contrast near the ball’s silhouette rather than distributing it uniformly over the surface. We address this with a spherical formulation of contrast maximization that models events directly on the ball’s surface, and develop two complementary estimation modes: an offline method for high-accuracy spin extraction, and a real-time pipeline that combines an uncertainty-aware convolutional neural network (CNN) with GPU-accelerated contrast maximization for refinement.

Our main contributions can be summarized as follows:

- An event-based active vision system with a focus-tunable telephoto lens and actuated pan/tilt mirrors, capable of smoothly tracking a ball at a mirror angular velocity of 142.5 rad/s while focusing on its surface during a game.
- A high-accuracy offline spin estimation method based on spherical contrast maximization (s-CMax) that generalizes across multiple sports (table tennis, baseball, tennis, and golf), with mean magnitude and axis errors of 1.2 % and 1.5° on spinner-mounted balls, outperforming prior event-based methods [16, 26]. We further validate the method on in-flight balls actively tracked by the GCS, achieving 2.1 % magnitude and 5.4° axis error.
- A low-latency online spin estimation method featuring an uncertainty-aware convolutional neural network trained on pseudo-ground-truth spin labels from the offline approach, combined with a GPU-accelerated batch implementation of s-CMax for refinement.

- A method for simultaneous tracking and spin estimation of unmodified balls from multiple actively controlled views, demonstrating reliable tracking and spin estimation during professional table tennis games with 750 Hz throughput, 3 ms latency, and 8.8 % magnitude and  $6.4^\circ$  axis mismatch w.r.t. the offline method.

## 2 Related work

*Active gaze control for high-speed tracking.* Capturing fast-moving objects with high spatial resolution requires active vision systems that steer the camera’s gaze in real time. The Saccade Mirror line of work [21, 24, 29, 30] introduced galvanometer-based optical gaze controllers that achieve high-speed pan/tilt tracking without physically moving the camera. Sueishi *et al.* [34] coupled a similar galvanometer system with multi-exposure imaging for spin measurement of dotted table tennis balls. Notably, all of these systems rely on frame-based cameras, and only [34] addresses spin estimation, albeit requiring modified balls with custom dot patterns. Our work extends this line of research by pairing galvanometer mirrors with an event camera and a focus-tunable telephoto lens, enabling simultaneous active tracking and spin estimation of unmodified balls.

*Ball spin estimation with frame-based cameras.* Spin estimation methods can be broadly grouped by how they observe the ball. Trajectory-based approaches [1, 5, 20, 33, 39] infer spin indirectly from the Magnus effect on the ball’s trajectory, but require highly accurate position measurements, depend on calibrated aerodynamic models (Magnus and drag coefficients vary with velocity and air conditions), and offer limited observability of the full spin axis. Image-based methods instead observe the ball surface directly. Some rely on custom markers [9, 17, 34, 36, 37, 41], limiting practical applicability. Methods for unmodified balls typically track surface texture across frames [15, 35, 39, 44, 45], or recover 3D spin from motion blur in a single frame [4], though the latter requires carefully controlled exposure. Closest to our approach, Zhang *et al.* [44, 45] paired a wide-angle multi-camera system for 3D localization with pan/tilt telephoto cameras for spin observation, estimating spin by segmenting the logo across successive frames. However, their evaluation reaches only 2000 rpm and relies on indirect, trajectory-based error metrics. More broadly, all frame-based methods face fundamental limits at high spin: even short exposures introduce motion blur, and the discrete frame rate imposes a Nyquist limit beyond which temporal aliasing underestimates the spin.

*Event-based spin estimation.* Event cameras [10] respond asynchronously to per-pixel brightness changes with microsecond resolution, producing substantially reduced motion blur. These properties are well suited for observing fast-spinning balls. Nakabayashi *et al.* [26, 31] were the first to apply contrast maximization (CMax) [11, 12] to ball spin estimation. Their work demonstrated results on simulated data, although with manually provided ball detections and without reporting runtime or real-world quantitative evaluation. Gossard *et al.* [16] proposed an optic-flow-based approach for event-based spin estimation that leverages the periodicity in the event stream to set velocity constraints.

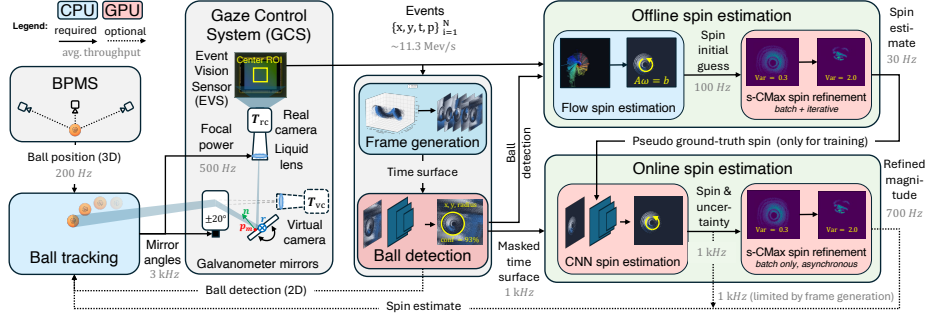


Fig. 2: Overview of the information flow in the GCS.

However, all existing event-based methods employ static, wide-angle cameras, which severely limits the spatial resolution on the ball and conflates translational motion with spin, a fundamental challenge for in-flight estimation. Our work addresses both limitations through active tracking with a telephoto lens, proposes a spherical formulation of CMax for more accurate rotation estimation, and develops a real-time pipeline combining an uncertainty-aware CNN with GPU-accelerated CMax refinement.

### 3 Methods

#### 3.1 Overview

The present system is designed to actively track a ball and estimate its spin, supporting both an offline mode for high-accuracy estimates and a real-time mode for time-sensitive applications (Fig. 2). Both modes share the same active tracking pipeline. We estimate the state of the ball using 3D ball positions obtained continuously from a BPMS (Sec. 3.1.1) to control the GCS (Sec. 3.1.2). We detect the ball asynchronously within the EVS event stream to track it and to filter events for spin estimation (Sec. 3.1.3). The hybrid tracking framework uses the 3D ball positions and 2D ball detections to provide smooth and centered tracking (Sec. 3.2). Using this active tracking framework, we run s-CMax offline to extract high-accuracy pseudo-ground-truth spin labels (Sec. 3.3), on which we train a low-latency uncertainty-aware CNN for real-time operation (Sec. 3.4); the resulting estimates are fed back to improve the tracking (Sec. 3.2.1).

##### 3.1.1 Ball position measurement system (3D)

The BPMS provides a continuous stream of 3D ball position measurements with appropriate coverage, accuracy, latency, and update rate for the active tracking system. In theory, any system providing 3D position measurements can be used, e.g., a multi-camera system, depth sensors, or radar-based systems. In this work, the real table tennis game recordings used a nine-camera active pixel sensor (APS) BPMS [7], where each camera detects the 2D ball position and the 3D position is estimated through triangulation. The spinner dataset (Sec. 4.2) was recorded using a monocular camera tracking an AprilTag.

### 3.1.2 Gaze control system (GCS)

The GCS comprises three main hardware components: 1) an EVS, 2) a focus-tunable telephoto lens, and 3) pan/tilt galvanometer mirrors (Fig. 1). We choose an event camera for its high temporal resolution and minimal motion blur, which are critical for resolving the fine surface texture of a fast-spinning ball without aliasing. The event camera is a Prophesee EVK4 featuring a Sony/Prophesee IMX 636 sensor [8]. We use a centered hardware ROI of  $320 \times 320$  to avoid event loss caused by bandwidth limitation in high event rate scenes. The telephoto lens is used to increase the spatial resolution on the ball; however, its narrow field of view and shallow depth of field demand active focus control as the ball distance changes rapidly. We use an Optotune EL-10-30 focus-tunable liquid lens because it offers a fast step response on the order of milliseconds. The lens is embedded in a custom optical design with 50 mm focal length and an effective field of view of  $\sim 1.8^\circ$ . To steer the camera’s gaze, we use galvanometer mirrors rather than physically rotating the camera. The galvanometer is a Thorlabs GVS012/M which provides a maximal mechanical scan angle of  $\pm 20^\circ$  for each pan and tilt mirror. Since each mirror rotates only a small, lightweight optical element, its rotational inertia is extremely low, enabling a small-angle step response of  $400 \mu\text{s}$  and a sustained mirror speed of  $142.5 \text{ rad/s}$  (roughly  $855 \text{ m/s}$  target lateral motion at 3 m distance), well exceeding the speeds in professional ball sports. The system is calibrated in four stages, all referenced to a common frame via AprilTags [43]: camera intrinsics (APS and reconstruction-based EVS [25]), mirror extrinsics [32], lens distance-to-focal-power map, and BPMS-to-GCS transform (Suppl. Sec. A.3). Spin estimation runs in the virtual camera frame and is largely insensitive to calibration error: s-CMax and the CNN operate in 2D image space, so intrinsics do not propagate, while mirror and extrinsic errors are bounded by the narrow FoV ( $\pm 0.9^\circ$ , well below the prior state-of-the-art axis error of  $5.0^\circ$  (Tab. 1))—in practice we observe  $\sim 1 \text{ px}$  reprojection error ( $5.6 \times 10^{-3}^\circ$ ). Any residual error is shared across methods and thus cancels in our comparisons.

*Multi-system fusion.* In practice, the spin-relevant pattern may be visible from only from a limited subset of viewpoints (e.g., the table tennis ball; Fig. 3), so it can be necessary to deploy multiple GCS placed to maximize the visible ball-surface area. In our table-tennis game setup [7], we fuse per-view estimates by selecting the prediction with the lowest uncertainty, improving robustness to occlusions and expanding the observable workspace (Suppl. Sec. A.7).

### 3.1.3 Event-based frame generation and ball detection (2D)

We detect the ball in the EVS frame 1) to isolate it from the background for spin estimation and 2) to provide a direct feedback signal for active ball tracking. Since the appearance of the ball within the event stream can change drastically based on the mirror motion, we opt for a deep-learning-based detector. We convert the event stream to a polarity-separated time surface representation [2] since it preserves motion information and has a small memory footprint for inference, while allowing convenient truncation during online operation. We use a YOLO object detection model [42] to predict a single bounding circle ( $x, y$  pixel loca-

tion, radius, and confidence) from the time surface with an event accumulation time of 5 ms. For details on the dataset and training methodology, please refer to the supplementary material (Suppl. Sec. A.5).

### 3.2 State estimation and ball tracking

The hybrid ball tracking method combines complementary open-loop position-based and closed-loop image-based visual servoing techniques that are based on a real-time state estimate. The position-based method takes 3D ball positions obtained from the BPMS (Sec. 3.1.1) whereas the image-based method takes 2D ball detections from the EVS (Sec. 3.1.3). The tracking method controls the mirror angles of the galvo and the focal power of the liquid lens to keep the ball within the frame and in sharp focus, respectively.

#### 3.2.1 State estimation

We employ an improved Nakashima model [6, 28] to estimate the ball states, accounting for factors such as air drag, gravity, and the Magnus effect. The model parameters are fitted using datasets collected from real table tennis games. For each free-flight trajectory, the filter initializes its states, including ball position and velocity, using the first few ball position inputs. The filter allows for smooth tracking through latency compensation by predicting future ball positions and by controlling the mirrors incrementally in the absence of a ball position input.

*Contact handling in tracking.* Most contacts in ball games can be regarded as the start of a new free-flight trajectory and trigger a tracking re-initialization. However, for some contacts, such as table contacts in table tennis, we predict the post-contact trajectory from the pre-contact ball state since the normal direction of the contact surface is known [27, 28]. Note that this model requires the pre-contact spin as the input, for which we are feeding the real-time spin estimates.

*Uncertainty-aware spin filter.* To temporally smooth the spin estimates, we aggregate spin predictions from online spin estimation by maintaining a sliding history buffer of up to 45 estimates along with their uncertainties. Each prediction is weighted by the inverse of its uncertainty value, effectively prioritizing more confident estimates in the aggregation. Assuming the spin is constant between contacts, the filter rejects outliers based on 1) median absolute deviation thresholding over spin, and 2) uncertainty thresholding. The filter state resets only upon a detected contact, ensuring stable and consistent estimates even when the logo is temporarily occluded. The final estimate is the inverse-uncertainty-weighted mean of the inlier set.

#### 3.2.2 Open-loop position-based tracking

The GCS receives 3D ball state estimates, which are continuously updated by the BPMS. Both systems are synchronized using a centralized hardware trigger generator, which allows estimation of the average sensing latency from exposure time to mirror control time. For each 3D ball position input  $\mathbf{p}_t$ , the mirror angles  $\phi_0$  and  $\phi_1$  are computed by solving the optimization problem

$$\phi_0, \phi_1 = \arg \min_{\phi_0, \phi_1} \left\| z_t^{-1} K_c T_{vc} \mathbf{p}_t - [c_x, c_y, 1]^\top \right\| \quad (1)$$

where  $K_c$  is the intrinsic matrix,  $z_t$  is the z-coordinate of the ball position, and  $(c_x, c_y)$  are the center pixel coordinates of the event camera. This optimization problem (Eq. 1) can be solved using gradient descent. The lens focal power is simultaneously adjusted based on the target distance to the virtual camera.

### 3.2.3 Closed-loop image-based tracking

In practice, position noise and temporal instability in the BPMS may result in tracking failures, which motivates the use of 2D ball detection in the EVS frame. We 1) reinitialize the 3D ball state estimator using new BPMS inputs if the ball is not detected in the EVS frame, and 2) adjust the 3D ball position estimate based on the detection, ensuring that the ball remains centered in the frame. For 2), we compute  $\Delta \mathbf{p}_t = z_t R_{vc}^{-1} K_c^{-1} [\Delta u, \Delta v, 1]^\top$ , where  $(\Delta u, \Delta v)$  represent the 2D coordinate differences from the image center, and  $\Delta \mathbf{p}_t$  denotes the difference between the expected and detected 3D ball positions. When computing, we assume that the depth  $z_t$  has minimal difference from the filter ball state.  $\Delta \mathbf{p}_t$  is then added to the filter state to lock the ball to the image center. This hybrid tracker is explicitly designed to tolerate BPMS noise, latency, and brief dropouts. We have verified that a simple two-camera stereo rig (30 cm baseline) suffices for tracking at 3 m to 20 m without noticeable degradation.

## 3.3 Offline spin estimation

The offline estimator runs in two stages: optical flow first initializes spin from local event motion on the detected ball; this initialization accelerates convergence and reduces the risk that the subsequent non-convex s-CMax optimization settles in an incorrect local optimum. s-CMax then refines this estimate by maximizing event contrast on the sphere.

### 3.3.1 Optical flow spin estimation initialization

For this initialization, we compute 2D flow vectors on masked ball events using plane fitting [2], normalize them by the ball radius, and lift them to 3D tangential velocities  $\mathbf{v}_k$  using the spherical geometry. Since the tangential velocity at position  $\mathbf{x}_k$  on the sphere relates to spin via  $\mathbf{v}_k = \boldsymbol{\omega} \times \mathbf{x}_k = -[\mathbf{x}_k]_\times \boldsymbol{\omega}$  (where  $[\mathbf{x}_k]_\times$  is the skew-symmetric matrix of  $\mathbf{x}_k$ ), we construct an overdetermined system of linear equations. Since each optical flow measurement provides only a scalar constraint along the observed flow direction (normal flow constraint [46]), we project each observation model using  $P_k = \mathbf{v}_k \mathbf{v}_k^T / (\mathbf{v}_k^T \mathbf{v}_k)$  before stacking. Stacking  $N$  flow measurements yields the system  $\mathbf{b} = A \boldsymbol{\omega}$  where  $\mathbf{b} \in \mathbb{R}^{3N}$  contains the projected velocities and  $A \in \mathbb{R}^{3N \times 3}$  contains the projected observation models. Least squares then gives the coarse spin vector that seeds s-CMax: brute-force magnitude search (Suppl. Fig. 7), then axis-and-magnitude refinement.

### 3.3.2 s-CMax spin estimation refinement

We adopt the contrast maximization framework [12] to estimate the ball’s angular velocity  $\boldsymbol{\omega} \in \mathbb{R}^3$  from a set of  $N_e$  events  $\{e_k\}_{k=1}^{N_e}$ . However, unlike [26], we use a spherical formulation of this framework specifically for angular motion estimation which we call s-CMax to differentiate it from the formulation used



for angular *and* linear motion estimation. Each event is represented by a tuple  $e_k = (\mathbf{u}_k, t_k, p_k)$ , where  $\mathbf{u}_k = [u_k, v_k]^\top \in \mathbb{N}_0^2$  is the 2D pixel location,  $t_k \in \mathbb{R}$  the timestamp, and  $p_k \in \{0, 1\}$  the polarity (either OFF or ON, respectively).

We use the ball detection result obtained at time  $t_{\text{ref}}$  (Sec. 3.1.3) to discard background events and lift the 2D pixel coordinates to 3D points  $\mathbf{x}_k = [x_k, y_k, z_k]^\top$  on the unit sphere. Assuming an orthographic projection, this lifting is computed as

$$x_k = (u_k - c_x)/r, \quad y_k = (v_k - c_y)/r, \quad z_k = -\sqrt{1 - (x_k^2 + y_k^2)} \quad (2)$$

where  $(c_x, c_y)$  is the detected ball center and  $r$  is its radius in pixels.

Each 3D event point  $\mathbf{x}_k$  is warped to  $\mathbf{x}'_k = f_{\text{warp}}(\mathbf{x}_k, t_k, \boldsymbol{\omega}) = R(-\boldsymbol{\omega} \cdot \Delta t_k) \mathbf{x}_k$  by rotating it according to the candidate spin, where  $\Delta t_k = t_k - t_{\text{ref}}$  and  $R(\cdot) \in \text{SO}(3)$  is the rotation matrix constructed from the axis-angle representation.

We convert the warped 3D points to unit spherical coordinates  $\mathbf{s}'_k = [\theta'_k, \phi'_k]$  where  $\theta'_k = \arcsin(z'_k)$  and  $\phi'_k = \arctan2(y'_k, x'_k)$  denote the polar and azimuth angles, respectively. The IWE  $I$  is obtained by accumulating events into a 2D histogram over discrete spherical coordinates

$$I[\mathbf{s}] = \sum_{k=1}^{N_e} \delta(\mathbf{s} - \mathbf{s}'_k) \quad (3)$$

where  $\delta(\mathbf{x})$  is the Kronecker delta (1 if  $|\mathbf{x}| = 0$ , else 0). The spin estimate is found by maximizing the variance of the IWE

$$\boldsymbol{\omega}^* = \arg \max_{\boldsymbol{\omega}} \text{Var}(I(\boldsymbol{\omega})). \quad (4)$$

We solve this optimization problem in two steps: 1) we brute force search the spin magnitude and then 2) further refine it using the Nelder-Mead simplex algorithm [14] as we find it yields better solutions than gradient-based methods in fewer iterations. We use bilinear interpolation when accumulating events into the IWE and smooth it with a  $5 \times 5$  Gaussian kernel and unit standard deviation to facilitate convergence. It is worth highlighting that the use of a spherical instead of planar projection of the IWE [26] enables us to warp events to their true location on the ball without ambiguities. We found that a non-unique planar projection of the IWE leads to alignment of events from disparate locations on the ball surface, as well as the tendency for events to concentrate on the ball edge, which leads to high-variance configurations that do not reflect the underlying motion and thus bias the spin estimates.

### 3.4 Online spin estimation

#### 3.4.1 Online CNN for spin estimation

While s-CMax yields accurate spin estimates, its iterative optimization precludes real-time operation. Optical flow is faster but degrades under limited texture visibility and exhibits event-rate-dependent latency, and neither method provides

an interpretable uncertainty estimate. To address these limitations, we train a custom CNN based on ResNet-18 [19] to predict a 3D spin vector and a 3D variance vector that captures prediction uncertainty [23]. We regress the angular velocity components, expressed in the virtual camera frame. During training, we normalize the target by the maximum spin magnitude observed in the dataset and train the network to predict  $\hat{\tilde{\omega}}$ .

For balls with a spatially concentrated texture, like a logo (e.g., table tennis or golf), the texture may not be observable by the GCS within the input time window. Through the heteroscedastic loss, the network learns to represent uncertainty arising from limited texture visibility, translational motion, occlusions, and sensor noise without requiring explicit uncertainty labels, which would be cumbersome and ambiguous to produce.

The CNN takes as input a normalized polarity-separated event time surface accumulated over 15 ms, which is chosen such that, even at low spin rates, a sufficient fraction of a revolution is observed. This representation is masked using the predicted bounding box from the ball detection network (Fig. 2). In addition to the spin estimate  $\hat{\tilde{\omega}}$ , the network predicts a per-component log-variance vector  $\mathbf{s} = (s_x, s_y, s_z)$ , where  $s_i = \log \sigma_i^2$ . Predicting log-variances improves numerical stability and ensures strictly positive variances. We train the model using a heteroscedastic Gaussian negative log-likelihood with diagonal covariance:

$$\mathcal{L}(\hat{\tilde{\omega}}, \mathbf{s}) = \frac{1}{2} \sum_{i \in x, y, z} \left( \exp(-s_i) \left( \tilde{\omega}_i - \hat{\tilde{\omega}}_i \right)^2 + s_i \right).$$

This formulation encourages the network to increase predicted uncertainty when the input is ambiguous (e.g., due to occlusions or missing logo evidence), while remaining confident when the spin is well observed. For training, we label a dataset of free-flight shots obtained from a ball thrower and real, elite-level table tennis games, using offline s-CMax. We use a total of 3969 event recordings, each corresponding to a single free-flight shot (2233 for training, 686 for validation, and 1050 for testing; roughly 4 h of data in total), with magnitudes ranging from 0–9 krpm (see Suppl. Tab. 3 for the distribution of the throws across magnitude ranges). To improve robustness, we apply augmentations during training, including random noise injection, event dropping, temporal jitter, and spatial translation. We additionally apply random in-plane rotations to the event representation and rotate the ground-truth spin axis to improve axis coverage. Note that, although the spin vector is constant during free-flight, the spin axis expressed in the (virtual) camera frame changes over time due to the mirror motion of the GCS. Adapting the CNN to a new ball requires sport-specific training but no manual labels: one records event data and runs offline s-CMax to obtain pseudo-ground-truth.

### 3.4.2 Online s-CMax for batch spin refinement

While the CNN provides robust spin axis predictions that align well with the pseudo-ground-truth, we observe that its magnitude estimates tend to be conservative in high-spin scenes, likely due to the limited availability of training

data in this regime (Suppl. Tab. 3). To leverage the complementary strengths of learning-based and model-based approaches, we develop an online batch version of s-CMax to asynchronously refine the spin magnitude, with the CNN output as a reliable directional prior. Since we want to avoid time-consuming memory allocations in the decoder thread of the EVS, we convert the time surface representation back to events which we found to be fast and effective. However, due to latency constraints, rather than iteratively optimizing the spin vector as in the offline version (Sec. 3.3), we opt to explore all reasonable magnitudes for a given axis in parallel on the GPU (see Suppl. Sec. A.6 for details). For each batch, we evaluate the best GPU-searched magnitude and the original CNN magnitude with the same IWE-variance objective. If the searched candidate yields a higher variance, we keep the CNN axis and replace only the magnitude.

## 4 Experiments

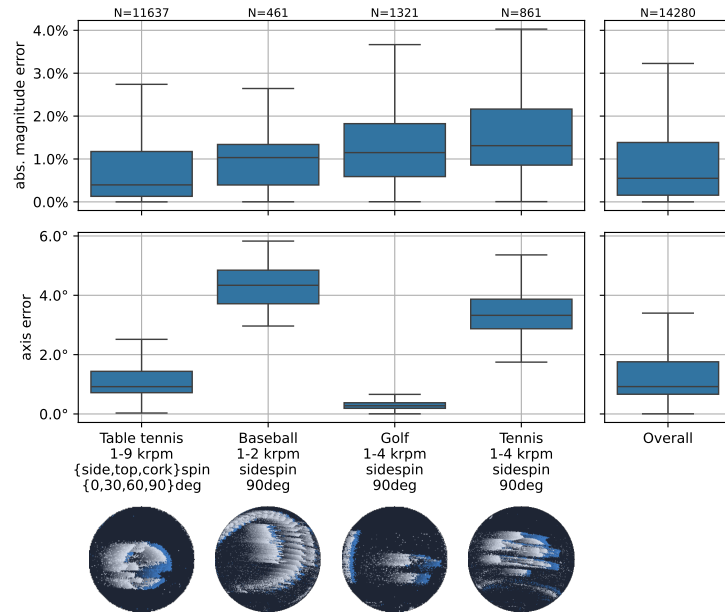
We evaluate real-world spin estimation accuracy using two primary metrics: 1) spin magnitude error, expressed as a percentage (%) or in revolutions per minute (rpm), and 2) spin axis error in degrees ( $^{\circ}$ ). A fundamental challenge in evaluating spin estimation methods is the difficulty of obtaining reliable ground truth: attaching sensors or markers to a ball in free flight alters its dynamics, and frame-based vision references degrade with distance, motion blur, and temporal aliasing at high spin rates. We therefore design three experiments that progressively trade controlled ground truth for realism: a static ball spinner with encoder-based ground truth (Sec. 4.2), a ball thrower with vision-based pseudo-ground-truth labels (Sec. 4.3), and elite-level table tennis rallies where ground truth is only known via the methods presented in this work (Sec. 4.4).

### 4.1 Experimental setup

We run the GCS on a dedicated computer with a multi-core CPU and two GPUs: the frame generation, spin filter, and ball state estimator run asynchronously on the CPU, while the ball detection, CNN, and s-CMax modules use separate GPUs to decrease latency and increase throughput. The online components are written in C++ and CUDA, using TensorRT for neural network acceleration (see Suppl. Sec. A.1 for details). For the ball thrower (Sec. 4.3) and table tennis match (Sec. 4.4) experiments, we use a BPMS that covers a  $14\text{ m} \times 7\text{ m} \times 5\text{ m}$  volume with about 3.0 mm accuracy, streaming ball positions via ROS2 at 200 Hz with 7.5 ms latency. All hardware is off-the-shelf (Prophesee EVK4, Thorlabs GVS012/M galvanometers, and an Optotune EL-10-30 lens) except for two 3D-printed mounts; we will publicly release our event datasets, pseudo-ground-truth labels, pseudocode, and a bill of materials with CAD files.

### 4.2 Offline spin estimation experiments: ball spinner

We use motorized ball spinners to collect real-world spin estimation data with ground-truth axis and magnitude labels. The spin magnitude is obtained from the motor encoders, whereas the spin axis is obtained by detecting an AprilTag



**Fig. 3:** The effect of different ball types on the s-CMax magnitude and axis error. The bottom row shows a time surface representation of the different balls when their pattern is visible, spinning at 1 krpm.

[43] that is aligned with the spin axis of the motor (see Suppl. Sec. B.4 for details).

We recorded 10 raw event files in a sidespin configuration across three ball types: 4 files each for the golf and tennis ball from 1–4 krpm, and 2 files for the baseball from 1–2 krpm, all in 1 krpm increments. We additionally recorded 108 table tennis-specific raw event files covering three spin types (topspin, sidespin, and corksidespin), four logo orientations (0–90° in 30° increments), and nine spin magnitudes from 1–9 krpm in 1 krpm increments.

Unless otherwise noted, we compute spin estimates with an event accumulation time of 10 ms. In order to exclude samples in which no events are generated by the pattern on the ball, e.g., when the pattern is rotated away from the observer, we set a 0.1 minimum variance threshold.

#### 4.2.1 s-CMax spin estimation

We conduct a ball spinner experiment to show the generalization of s-CMax to different visual appearances (Fig. 3). We obtain similar magnitude and axis errors for all the ball types we tested in our experiments. Some variations, e.g., the slightly higher axis error of around 4° for the baseball and tennis ball, can be attributed to setup inaccuracies: the heavier balls must be spun at lower magnitudes for safety and their AprilTag-based axis alignment on the spinner is less precise (see Suppl. Sec. B.4 for details).

#### 4.2.2 Comparison to other spin estimation methods

We compare our spin estimation methods against a flow-based [16] and a contrast-maximization-based baseline [26]. To ensure a fair comparison between methods, we make some modifications to the baselines (see Suppl. Sec. B.7 for details). The flow-based method [16] depends on estimating the spin magnitude via peak finding in the event rate to set event accumulation time and minimum/maximum optic flow speed constraints. The CMax-based method [26] depends on random initialization of the spin vector, which we replace by the deterministic result from our optic flow method (Sec. 3.3.1). We further adapt its parameters to match s-CMax where they yield better results on our dataset: rotation-only estimation, a larger  $5 \times 5$  Gaussian kernel for blurring the IWE, and the same convergence criterion.

The results show clear improvements from each of our contributions (Tab. 1). Our flow-based method, which serves as the initialization for the CMax-based methods, already halves the magnitude error and reduces the axis error by  $4\times$  compared to Gossard *et al.* [16], providing a stronger starting point for the subsequent optimization. Building on this initialization, s-CMax achieves the best accuracy across both metrics (1.2% magnitude and  $1.5^\circ$  axis error) while requiring only about one ninth of the runtime of CMax [26], striking a favorable balance between accuracy and computational cost.

**Table 1:** Comparison of spin estimation methods on the spinner dataset. Magnitude and axis errors are reported as mean  $\pm$  standard deviation.

| Method        | Mag. error [%] $\downarrow$     | Axis error [ $^\circ$ ] $\downarrow$ | Runtime [ms] $\downarrow$        |
|---------------|---------------------------------|--------------------------------------|----------------------------------|
| Flow [16]     | $35.2 \pm 19.5$                 | $37.1 \pm 24.2$                      | $7.7 \pm 14.1$                   |
| Flow (ours)   | $17.3 \pm 22.6$                 | $9.0 \pm 17.6$                       | <b><math>7.3 \pm 13.3</math></b> |
| CMax [26]     | $8.2 \pm 14.3$                  | $5.0 \pm 10.4$                       | $317.4 \pm 425.3$                |
| s-CMax (ours) | <b><math>1.2 \pm 2.3</math></b> | <b><math>1.5 \pm 2.2</math></b>      | $36.7 \pm 19.5$                  |

#### 4.3 Offline spin estimation experiments: ball thrower

We place a GCS next to a static EVS with a wide-angle lens—a common configuration in related works [16, 26]—both 2 m from the ball thrower, so that both observe the same 30 yellow dotted shots from the same perspective (see Suppl. Sec. B). This isolates the two key benefits of the GCS over a static wide-angle camera: a magnified, higher-resolution view of the ball surface, and compensation of linear motion via active tracking.

The results show that it is possible to obtain reliable spin estimates with the GCS in this setting, whereas the static EVS cannot provide satisfactory results (Tab. 2). There are two main reasons for this discrepancy: 1) linear motion that triggers events and 2) limited resolution on the ball surface. For the flow-based methods, the optic flow estimates follow the ball motion rather than the spin

**Table 2:** Comparison of spin estimation methods on the dotted table tennis ball on the ball thrower dataset. Magnitude and axis errors are reported as mean  $\pm$  standard deviation. A large axis error (**red numbers**) indicates complete failure, even when the magnitude error is low. Pseudo-ground-truth labels were obtained using SpinDOE [17].

| Sensor setup Method |               | Mag. error [%] $\downarrow$     | Axis error [ $^\circ$ ] $\downarrow$ |
|---------------------|---------------|---------------------------------|--------------------------------------|
| Static EVS          | Flow [16]     | $18.8 \pm 12.9$                 | <b><math>88.7 \pm 8.1</math></b>     |
| Static EVS          | Flow (ours)   | $14.1 \pm 8.7$                  | <b><math>59.9 \pm 5.0</math></b>     |
| Static EVS          | CMax [26]     | n/a                             | n/a                                  |
| Static EVS          | s-CMax (ours) | n/a                             | n/a                                  |
| GCS (ours)          | Flow [16]     | $14.7 \pm 12.6$                 | <b><math>75.6 \pm 18.5</math></b>    |
| GCS (ours)          | Flow (ours)   | $18.0 \pm 5.4$                  | $11.8 \pm 29.4$                      |
| GCS (ours)          | CMax [26]     | $10.0 \pm 20.0$                 | $12.3 \pm 29.3$                      |
| GCS (ours)          | s-CMax (ours) | <b><math>2.1 \pm 2.5</math></b> | <b><math>5.4 \pm 3.7</math></b>      |

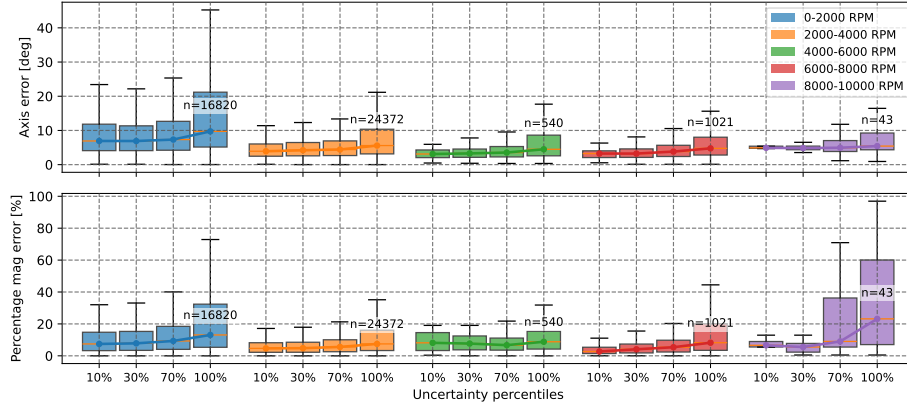
and closely match the image velocity found by computing finite differences from subsequent ball detections, indicating that only the linear ball motion can be extracted despite considerable spin. For the CMax-based methods, we did not obtain results beyond the initial guess provided by the flow method due to an insufficient number of events on the ball, suggesting that the resolution is too limited for these methods. The seemingly reasonable errors on the flow methods for the static EVS are coincidental: the ball thrower couples speed and spin so translational motion produces magnitudes numerically similar to that of the true spin – but with the axis wrong, the magnitude is meaningless. This is precisely the translation-rotation conflation the GCS resolves.

#### 4.4 Online spin estimation experiments: real table tennis games

We evaluate the CNN on the held-out test split (1050 recordings) to assess spin prediction accuracy and the quality of the uncertainty estimates. Fig. 4 analyzes error as a function of predicted uncertainty reported by spin magnitude range.

Across all ranges, the error decreases as we restrict evaluation to more confident predictions, indicating that the predicted uncertainty is informative for identifying reliable spin estimates. Both the median error and the spread increase for higher spin magnitudes, consistent with reduced training coverage in these regimes. In total, the mean error on the 50% uncertainty percentile across all ranges is  $8.8 \pm 12.4\%$  for the magnitude and  $6.4 \pm 6.7$  degrees for the axis.

The dominant failure mode is logo invisibility: across 2642 free-flight segments from 7 elite table tennis matches, 2.2% had the logo hidden for more than half of the duration, and the predicted uncertainty enables graceful degradation through multi-view fusion (Suppl. Sec. A.7). A qualitative visualization (Suppl. Sec. B) confirms that clearly visible logo traces yield accurate, low-uncertainty predictions, whereas limited logo evidence increases both error and predicted uncertainty.



**Fig. 4:** CNN magnitude and axis errors on the held-out test set for different uncertainty percentiles and magnitude ranges.

## 5 Conclusion

We presented an event-based active vision system for real-time ball tracking and spin estimation, combining an event camera with a focus-tunable telephoto lens and fast galvanometer mirrors to obtain a magnified, motion-compensated view of unmodified balls in flight. The hybrid tracking framework fuses 3D ball localization with 2D event-based detection for smooth, robust active tracking at professional ball-sport speeds. The offline spin estimation method, based on a spherical contrast maximization formulation, achieves state-of-the-art accuracy on a variety of officially approved balls, outperforming existing baselines. For real-time operation, we introduced an uncertainty-aware CNN trained on pseudo-ground-truth labels from the offline method, combined with a GPU-accelerated batch contrast maximization refinement step and a multi-view fusion strategy, achieving state-of-the-art latency and throughput during professional table tennis matches. We believe the system’s ability to track and resolve fine surface details at high temporal resolution opens promising avenues for sports broadcasting, high-speed object inspection, and competitive robotic play.

## Acknowledgments

We thank Dario Brescianini for initiating the project, Markus Kamm for the custom lens optics, Robin Frauenfelder and Carter Fang for the initial ball tracking and detector prototype, Alexander Sigrist, Stefan Heusser and Nobuhiko Mukai for building the camera synchronization hardware, ball spinners, and ball throwers, Christian Conti Fujiwara for the ball aerodynamic model and the contact physics models, Christian Conti Fujiwara and Hamdi Sahloul for the SpinDOE implementation, Valentin Monferrato and Adriano Patane for data labeling and cleaning, and Aleksandar Stoimenov for debugging and fixing technical issues.

## References

1. Aakanksha, Kumar, A., Rajagopalan, A.: Ball trajectory and spin analysis from asynchronous videos. *IEEE Sensors Letters* **9**(3), 1–4 (2025). <https://doi.org/10.1109/SENSOR.2025.3537116>, art. no. 3500704
2. Benosman, R., Clercq, C., Lagorce, X., Sio-Hoi Ieng, Bartolozzi, C.: Event-Based Visual Flow. *IEEE Transactions on Neural Networks and Learning Systems* **25**(2), 407–417 (2014). <https://doi.org/10.1109/TNNLS.2013.2273537>, <http://ieeexplore.ieee.org/document/6589170/>
3. Bi, Y., Conti, C., Yang, B., Sigrist, A., Dürr, P., Takahashi, N.: Bridging the sim2real gap in the table tennis robot with a transformer-based ball states predictor (2026), <https://arxiv.org/abs/2606.11464>
4. Boracchi, G., Caglioti, V., Giusti, A.: Estimation of 3D Instantaneous Motion of a Ball from a Single Motion-Blurred Image. In: Ranchordas, A., Araújo, H.J., Pereira, J.M., Braz, J. (eds.) *Computer Vision and Computer Graphics. Theory and Applications*. pp. 225–237. Springer, Berlin, Heidelberg (2009). [https://doi.org/10.1007/978-3-642-10226-4\\_18](https://doi.org/10.1007/978-3-642-10226-4_18)
5. Chen, X., Tian, Y., Huang, Q., Zhang, W., Yu, Z.: Dynamic model based ball trajectory prediction for a robot ping-pong player. In: 2010 IEEE International Conference on Robotics and Biomimetics. pp. 603–608 (Dec 2010). <https://doi.org/10.1109/ROBIO.2010.5723394>
6. Conti, C., Yang, B., Sigrist, A., Miele, L., Saraiji, Y., Dürr, P., Takahashi, N.: Physics models for sim-to-real transfer in professional-level robot table tennis (2026), <https://arxiv.org/abs/2606.28805>
7. Dürr, P., El Gheche, M., Maeda, G.J., Mukai, N., Takahashi, N., Heusser, S., Sahloul, H., Saraiji, Y., Adodin, P., Bi, Y., Blakeman, S., Conti, C., Fuentes Hitos, D., Hu, Y., Khadivar, F., Kreiser, R., Martinez, L., Schilling, F., Tapiador Morales, R., Torrente, G., Ynocente Castro, M., Abecassis, L., Giammarino, A., Huang, Y.T., Nagel, Y., Scotti, A., Sigrist, A., Silva, T., Walther, E., Wong, J., Yang, B., Aydin, A., Grover, D., Saha, A., Cavinato, V., Kakinuma, T., Kunori, T., Monferrato, V., Richter, S., Charalambous, S., Guist, S., Kuhlmann-Jorgensen, M.A., Miele, L., Politis, A., Scardecchia, M., Kitano, H., Wurman, P.R., Stone, P., Spranger, M.: Outplaying elite table tennis players with an autonomous robot. *Nature* **652**(8111), 886–891 (Apr 2026). <https://doi.org/10.1038/s41586-026-10338-5>, <https://doi.org/10.1038/s41586-026-10338-5>
8. Finateu, T., Niwa, A., Matolin, D., Tsuchimoto, K., Mascheroni, A., Reynaud, E., Mostafalu, P., Brady, F., Chotard, L., LeGoff, F., Takahashi, H., Wakabayashi, H., Oike, Y., Posch, C.: A 1280×720 Back-Illuminated Stacked Temporal Contrast Event-Based Vision Sensor with 4.86µm Pixels, 1.066GEPS Readout, Programmable Event-Rate Controller and Compressive Data-Formatting Pipeline. In: 2020 IEEE International Solid-State Circuits Conference - (ISSCC). pp. 112–114 (Feb 2020). <https://doi.org/10.1109/ISSCC19947.2020.9063149>
9. Furuno, S., Kobayashi, K., Okubo, T., Kurihara, Y.: A study on spin-rate measurement using a uniquely marked moving ball. In: 2009 ICCAS-SICE. pp. 3439–3442 (Aug 2009)
10. Gallego, G., Delbrück, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A.J., Conradt, J., Daniilidis, K., Scaramuzza, D.: Event-Based Vision: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**(1), 154–180 (Jan 2022). <https://doi.org/10.1109/TPAMI.2020.3008413>



11. Gallego, G., Gehrig, M., Scaramuzza, D.: Focus Is All You Need: Loss Functions for Event-Based Vision. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 12280–12289 (2019)
12. Gallego, G., Rebecq, H., Scaramuzza, D.: A Unifying Contrast Maximization Framework for Event Cameras, with Applications to Motion, Depth, and Optical Flow Estimation. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3867–3876 (Jun 2018). <https://doi.org/10.1109/CVPR.2018.00407>
13. Gallego, G., Scaramuzza, D.: Accurate Angular Velocity Estimation With an Event Camera. *IEEE Robot. Autom. Lett.* **2**(2), 632–639 (Apr 2017). <https://doi.org/10.1109/LRA.2016.2647639>
14. Gao, F., Han, L.: Implementing the Nelder-Mead simplex algorithm with adaptive parameters. *Computational Optimization and Applications* **51**(1), 259–277 (2012). <https://doi.org/10.1007/s10589-010-9329-3>, <https://doi.org/10.1007/s10589-010-9329-3>
15. Glover, J., Kaelbling, L.P.: Tracking the spin on a ping pong ball with the quaternion Bingham filter. In: IEEE Int. Conf. Robot. Autom. pp. 4133–4140. IEEE, Hong Kong, China (May 2014). <https://doi.org/10.1109/ICRA.2014.6907460>
16. Gossard, T., Krismer, J., Ziegler, A., Tebbe, J., Zell, A.: Table Tennis Ball Spin Estimation with an Event Camera. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 3347–3356 (2024)
17. Gossard, T., Tebbe, J., Ziegler, A., Zell, A.: SpinDOE: A ball spin estimation method for table tennis robot (Mar 2023). <https://doi.org/10.48550/arXiv.2303.03879>
18. Hashimoto, Y., Nagami, T., Yoshitake, S., Nakata, H.: The relationship between pitching parameters and release points of different pitch types in major league baseball players. *Frontiers in Sports and Active Living* **5**, 1113069 (2023). <https://doi.org/10.3389/fspor.2023.1113069>, <https://pmc.ncbi.nlm.nih.gov/articles/PMC10164925/>
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778. IEEE (2016). <https://doi.org/10.1109/CVPR.2016.90>, <https://ieeexplore.ieee.org/document/7780459/>
20. Huang, Y., Xu, D., Tan, M., Su, H.: Trajectory prediction of spinning ball for ping-pong player robot. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 3434–3439 (Sep 2011). <https://doi.org/10.1109/IRoS.2011.6095044>
21. Iida, K., Oku, H.: Saccade Mirror 3: High-speed gaze controller with ultra wide gaze control range using triple rotational mirrors. In: IEEE Int. Conf. Robot. Autom. pp. 624–629 (May 2016). <https://doi.org/10.1109/ICRA.2016.7487186>
22. Kashiwagi, R., Okamura, S., Iwanaga, S., Murakami, S., Numata, K., Takahashi, H.: The differences in the ball speed and the spin rate depending on the results of a tennis serve. *Malaysian Journal of Movement, Health & Exercise* **10**(1), 48 (2021). <https://doi.org/10.4103/2231-9409.328217>
23. Kendall, A., Gal, Y.: What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In: *Advances in Neural Information Processing Systems*. vol. 30, pp. 5574–5584. Curran Associates, Inc. (2017)
24. Miyashita, L., Ishikawa, M.: Saccade Argos: Hierarchical Robust Tracking System for High Spatio-temporal Resolution Vision. In: 2025 IEEE/SICE International Symposium on System Integration (SII). pp. 811–816. IEEE, Munich, Germany (Jan 2025). <https://doi.org/10.1109/SII59315.2025.10871046>

25. Muglikar, M., Gehrig, M., Gehrig, D., Scaramuzza, D.: How to Calibrate Your Event Camera. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1403–1409. IEEE, Nashville, TN, USA (Jun 2021). <https://doi.org/10.1109/CVPRW53098.2021.00155>
26. Nakabayashi, T., Higa, K., Yamaguchi, M., Fujiwara, R., Saito, H.: Event-based Ball Spin Estimation in Sports. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 3367–3375 (Jun 2024)
27. Nakashima, A., Ogawa, Y., Kobayashi, Y., Hayakawa, Y.: Modeling of rebound phenomenon of a rigid ball with friction and elastic effects. In: Proceedings of the 2010 American Control Conference. pp. 1410–1415 (2010). <https://doi.org/10.1109/ACC.2010.5530520>, <https://ieeexplore.ieee.org/document/5530520>
28. Nakashima, A., Ogawa, Y., Liu, C., Hayakawa, Y.: Robotic table tennis based on physical models of aerodynamics and rebounds. In: 2011 IEEE International Conference on Robotics and Biomimetics. pp. 2348–2354 (2011). <https://doi.org/10.1109/ROBIO.2011.6181649>
29. Okumura, K., Oku, H., Ishikawa, M.: High-speed gaze controller for millisecond-order pan/tilt camera. In: 2011 IEEE International Conference on Robotics and Automation. pp. 6186–6191 (2011). <https://doi.org/10.1109/ICRA.2011.5980080>
30. Okumura, K., Yokoyama, K., Oku, H., Ishikawa, M.: 1 ms Auto Pan-Tilt – video shooting technology for objects in motion based on Saccade Mirror with background subtraction. *Advanced Robotics* **29**(7), 457–468 (Apr 2015). <https://doi.org/10.1080/01691864.2015.1011299>
31. Sato, K., Nakabayashi, T., Yamaguchi, M., Higa, K., Fujiwara, R., Saito, H.: Time-consistent Ball Tracking and Spin Estimation with Event Camera. In: Proceedings of the 7th ACM International Workshop on Multimedia Content Analysis in Sports. pp. 59–64. ACM, Melbourne VIC Australia (Oct 2024). <https://doi.org/10.1145/3689061.3689067>
32. Sturm, P., Bonfort, T.: How to Compute the Pose of an Object Without a Direct View? In: Narayanan, P.J., Nayar, S.K., Shum, H.Y. (eds.) *Computer Vision – ACCV 2006*. pp. 21–31. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2006). [https://doi.org/10.1007/11612704\\_3](https://doi.org/10.1007/11612704_3)
33. Su, H., Fang, Z., Xu, D., Tan, M.: Trajectory Prediction of Spinning Ball Based on Fuzzy Filtering and Local Modeling for Robotic Ping-Pong Player. *IEEE Transactions on Instrumentation and Measurement* **62**(11), 2890–2900 (Nov 2013). <https://doi.org/10.1109/TIM.2013.2263672>
34. Sueishi, T., Tochioka, H., Ishikawa, M.: High-speed spin measurement system for dotted table tennis ball using single-frame M-sequence multi-exposures. *SICE Journal of Control, Measurement, and System Integration* **18**(1), 2466881 (Dec 2025). <https://doi.org/10.1080/18824889.2025.2466881>
35. Tamaki, S., Yamagata, S., Hashizume, S.: Spin measurement system for table tennis balls based on asynchronous non-high-speed cameras. *International Journal of Computer Science in Sport* **23**(1), 37–53 (Feb 2024). <https://doi.org/10.2478/ijcss-2024-0003>
36. Tamaki, T., Sugino, T., Yamamoto, M.: Measuring Ball Spin by Image Registration. *The 10th Korea-Japan Joint Workshop on Frontiers of Computer Vision* pp. 269–274 (2004)
37. Tamaki, T., Wang, H., Raytchev, B., Kaneda, K., Ushiyama, Y.: Estimating the spin of a table tennis ball using Inverse Compositional Image Alignment. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing

- (ICASSP). pp. 1457–1460 (Mar 2012). <https://doi.org/10.1109/ICASSP.2012.6288166>
38. Tang, H.p., Mizoguchi, M., Toyoshima, S.: Speed and spin characteristics of the 40mm table tennis ball. *Table Tennis Sciences* **4**, 278–284 (2002)
  39. Tebbe, J., Klamt, L., Gao, Y., Zell, A.: Spin Detection in Robotic Table Tennis. In: *IEEE Int. Conf. Robot. Autom.* pp. 9694–9700 (May 2020). <https://doi.org/10.1109/ICRA40945.2020.9196536>
  40. The R&A, United States Golf Association: Second Report on the Study of Spin Generation. Tech. rep. (Jan 2007), [https://www.jga.or.jp/jga/html/rules/image/info\\_groove\\_second\\_report\\_spin\\_generation\\_en.pdf](https://www.jga.or.jp/jga/html/rules/image/info_groove_second_report_spin_generation_en.pdf)
  41. Theobalt, C., Albrecht, I., Haber, J., Magnor, M., Seidel, H.P.: Pitching a baseball: Tracking high-speed motion with multi-exposure images. In: *ACM SIGGRAPH 2004 Papers*. pp. 540–547. SIGGRAPH '04, Association for Computing Machinery, New York, NY, USA (Aug 2004). <https://doi.org/10.1145/1186562.1015758>
  42. Wang, C.Y., Yeh, I.H., Liao, H.Y.M.: YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information (Feb 2024). <https://doi.org/10.48550/arXiv.2402.13616>
  43. Wang, J., Olson, E.: AprilTag 2: Efficient and robust fiducial detection. In: *IEEE/RSJ Int. Conf. Intell. Robots Syst.* pp. 4193–4198 (Oct 2016). <https://doi.org/10.1109/IRoS.2016.7759617>
  44. Zhang, Y., Xiong, R., Zhao, Y., Wang, J.: Real-Time Spin Estimation of Ping-Pong Ball Using Its Natural Brand. *IEEE Transactions on Instrumentation and Measurement* **64**(8), 2280–2290 (Aug 2015). <https://doi.org/10.1109/TIM.2014.2385173>
  45. Zhang, Y., Zhao, Y., Xiong, R., Wang, Y., Wang, J., Chu, J.: Spin observation and trajectory prediction of a ping-pong ball. In: *IEEE Int. Conf. Robot. Autom.* pp. 4108–4114 (May 2014). <https://doi.org/10.1109/ICRA.2014.6907456>
  46. Zhu, Q., Triesch, J., Shi, B.E.: An Event-by-Event Approach for Velocity Estimation and Object Tracking With an Active Event Camera. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* **10**(4), 557–566 (2020). <https://doi.org/10.1109/JETCAS.2020.3040329>