BlurBall: a ball detector with blur estimation

Supplementary Material

Dataset

The dataset was generated from publicly available online videos under a Creative Commons license to ensure compliance with copyright regulations. We deliberately covered a wide range of scenarios, including different playing conditions, camera angles, and lighting variations, to improve model robustness. Orange table tennis balls, although officially approved for competition, are not included. This omission is unlikely to affect performance, as white balls are overwhelmingly preferred and almost exclusively used in professional matches.

The dataset is publicly available at: https://cogsys-tuebingen.github.io/blurball/. Each video is accompanied by a CSV file with the following structure:

Frame	Visibility	X	Y	θ	l
000049	1	581.62	295.26	-152.5	2.8
000050	1	572.98	292.86	171.8	2.1

Table 4. CSV description. Each row contains the ball position (X, Y), blur orientation (θ), and half-length (l).

Angles are given in degrees and follow the convention illustrated in Figure 3.

Camera calibration

We provide camera calibration for each table tennis match in the dataset. Specifically, we include the focal length f and the camera extrinsics: rotation vector $\boldsymbol{r}$ and translation vector $\boldsymbol{T}$. The world frame is defined by the table geometry, with the table's length aligned to the Y-axis and the surface normal aligned to the Z-axis, as shown in Figure 10.

This choice is motivated by the precise, known geometry of the table, which makes it a reliable calibration target. We manually annotate keypoints such as the four table corners, the midline–backline intersection, and the intersection between the net and the table's side edge (Figure 10). The camera pose is then estimated by minimizing the reprojection error via a standard PnP optimization.

Due to the limited and near-coplanar set of keypoints, distortion coefficients d_d and the optical center cannot be reliably estimated. We therefore assume an ideal pinhole model, with intrinsic parameters reduced to the focal length f.

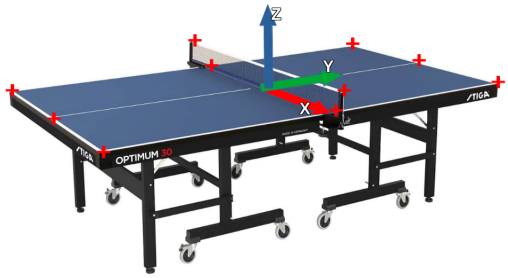


Figure 10. Annotated table keypoints used for camera calibration and definition of the world frame.

Model training

We followed the training setup of WASB [34] for consistency across models. Each model was trained for 30 epochs with an appropriate loss function and optimizer:

- **DeepBall** [16] and **DeepBall-large**: BCE loss, Adam optimizer, learning rate 3×10^{-4}.
- **BallSeg** [44]: focal loss ($\gamma = 2$), Adam optimizer, learning rate 3×10^{-4}.
- **TrackNetV2** [32] and **ResTrackNetV2**: focal loss, AdaDelta optimizer, learning rate 1.0.
- **MonoTrack** [22]: Combo loss, AdaDelta optimizer, learning rate 1.0.
- **WASB** [34]: quality focal loss, Adam optimizer, learning rate 3×10^{-4}.

This ensures fair comparison, with each model optimized using strategies suited to its architecture.

BlurBall: further insights

Impact of the threshold value

To evaluate the influence of the detection threshold in our BlurBall model, we plot the Precision-Recall (PR) curve in Figure 11. Overall, the 1-step variant achieves a better balance than the 3-step variant. However, for identical threshold values, the 1-step detector consistently yields higher recall but lower precision, indicating that its middle-frame predictions tend to be overconfident. This trade-off motivates the choice of a threshold value of $\delta = 0.7$, which we adopt for subsequent tracking experiments in Section 4.3.

Blur prediction

Figure 12 shows the relationship between blur length and angle estimation error. As expected, longer blur streaks yield more accurate angle estimation. For $l > 3$, the angular

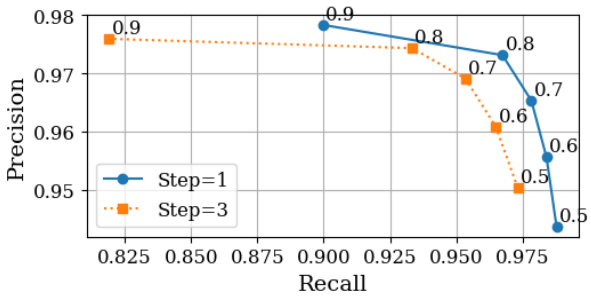


Figure 11. PR curves for BlurBall at different confidence thresholds δ.

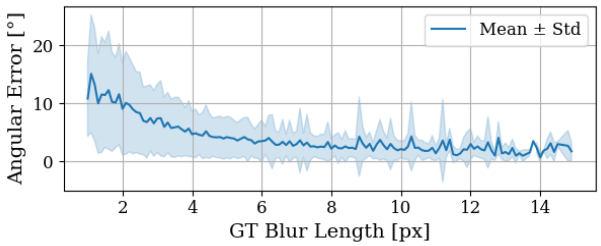
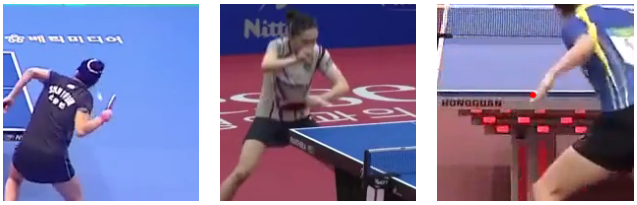


Figure 12. Angular estimation error as a function of ground-truth blur length.



(a) FN1 (b) FN2 (c) FP1

Figure 13. Examples of BlurBall failures. FN1 and FN2: missed detections. FP1: false positive on a hand.

error remains below 10°, validating its use in downstream tasks such as trajectory prediction (Section 4.5).

Failure cases

Figure 13 shows typical failure cases. In FN1, the ball briefly appears between two occluded frames, leading to a missed detection. In FN2, the ball overlaps with a moving player's body, a scenario difficult even for human observers. In FP1, a hand is incorrectly detected as the ball.

Trajectory prediction benchmark

We benchmark trajectory prediction using position-only (`Pos`) versus position+blur (`Pos+Blur`) fitting. Figure 14 and Figure 15 report full and zoomed-in box plots.

As shown in Figure 15, `Pos+Blur` achieves a lower median error (19.9 px vs. 28.4 px) and a narrower interquartile range, indicating greater consistency. The full distribution (Figure 14) also reveals fewer extreme outliers and a shorter upper whisker (140.2 px vs. 224.6 px), confirming improved robustness in difficult cases. Overall, incorporating blur leads to more accurate and stable trajectory predictions, especially under sparse or noisy observations.

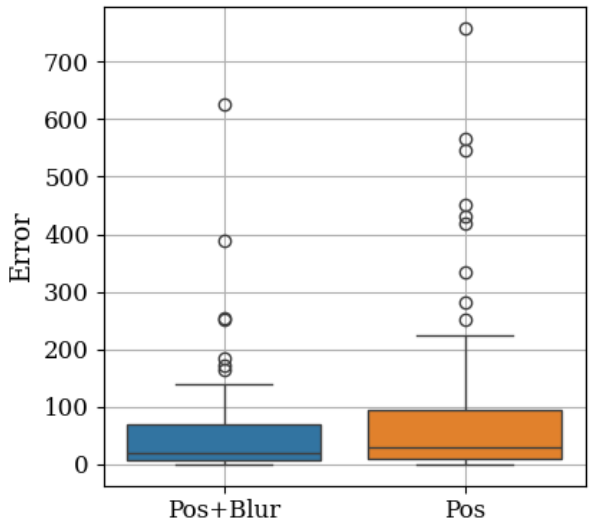


Figure 14. Full box plot of trajectory prediction errors for `Pos` and `Pos+Blur`, including outliers.

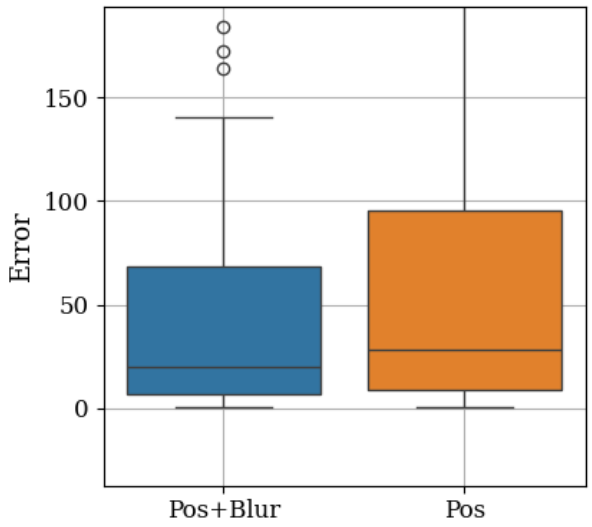


Figure 15. Zoomed-in box plot of trajectory prediction errors.