A multi-modal table tennis robot system

Andreas Ziegler*, Thomas Gossard*, Karl Vetter, Jonas Tebbe, Andreas Zell
Coginitive Systems Group
Dept. Informatics
University of Tübingen

Abstract: In recent years, robotic table tennis has become a popular research challenge for perception and robot control. Here, we present an improved table tennis robot system with high accuracy vision detection and fast robot reaction. Based on previous work, our system contains a KUKA robot arm with 6 DOF, with four frame-based cameras and two additional event-based cameras. We developed a novel calibration approach to calibrate this multimodal perception system. For table tennis, spin estimation is crucial. Therefore, we introduced a novel, and more accurate spin estimation approach. Finally, we show how combining the output of an event-based camera and a Spiking Neural Network (SNN) can be used for accurate ball detection.

Keywords: CoRL, RoboLetics, Table tennis robot

1 Introduction

Table tennis is a fast-paced and exhilarating sport that demands agility, precision, and lightning-fast reflexes. It is a sport enjoyed by millions of enthusiasts worldwide, ranging from casual players to professional athletes. In recent years, the fusion of technology and sports has led to the development of various training aids and innovations aimed at enhancing the skills of players and fostering their competitive edge. Among these technological advancements, table tennis robots have also emerged. While not yet able to compete with professional players, table tennis robots are an interesting research environment to bring perception and control algorithms towards their limits. Thus, it is not surprising, that more and more research groups use table tennis robots as a test bed for their algorithms [1][2][3][4].

2 Related Work

Ever since Billingsley initiated a robot table tennis competition in 1983 [5], robotic table tennis has been a popular tool for research in computer vision and robot control. Various types of manipulators have been used. Huang et al. [6] used a 5-DOF robot with three linear axes plus a pan-tilt unit. Xiong et al. [7] developed two human-like robots Wu & Kong. Both robots have 30 DOF in total with two 7-DOF arms, two 6-DOF legs, and 4-DOF for head and hip. Omron frequently showcases its Delta robot at trade shows, with a table tennis racket attached after two additional swivel joints. Buchler et al. [8] have designed a completely new pneumatic robot arm able to attain very high-end-effector speeds. Particularly popular are industrial 6 or 7-axis articulated arm robots in which all joints are rotational and which are relatively similar to the human arm [9][10][11]. Our system also employs this type of robot, the Agilus KR6 R900 sixx made by KUKA [12].

For the perception of the table tennis ball, either conventional image processing techniques [13] or Convolutional Neural Network (CNN) based approaches [3][4] are used. While using a CNN for

*Equal contribution Correspondence to {andreas.ziegler, thomas.gossard}@uni-tuebingen.de
This research is funded by Sony AI.

the ball detection results in a slightly higher accuracy [3][12] these approaches are more cumbersome to integrate and debug compared to their conventional counterparts [12]. Therefore, we use a conventional image processing approach to detect the balls, introduced in Tebbe et al. [1] and later improved in Tebbe [12].

Since the spin of a table tennis ball is crucial in table tennis, we also aim to detect the spin with our perception system. There are multiple ways to estimate the spin of a table tennis ball. Model-based approaches use a physical model of the ball and estimate the spin by using the effect of the magnus force [14]. As further research has shown, estimating the spin with a visual perception system leads to better results. In most cases, the logo is used to determine the rotation of the ball and thus the spin [14].

3 Table Tennis Robot System

Our research is based on the table tennis robot system presented in Tebbe et al. [1]. Which was further improved in Tebbe et al. [14] and Tebbe [12]. This system consists of four PointGrey Chameleon3 frame-based cameras (140 fps, 1280x1024 pixels), one PointGrey Grasshoper3 frame-based camera (350 fps, 1920x1200 pixels) and a 6-DOF KUKA Agilus robot.

We added accurate and reliable spin estimation using the high framerate camera, attached to the ceiling. The algorithm used is described in section 3.1. We also extended the setup with two Prophesee EVK4 event-based cameras (1280x720 pixels), as event-cameras offer a lot of potential with the lower latency. In section 3.2 we explain how we calibrated our setup to be able to use both types of camera simulteneously. Ball positions estimated with frame cameras can then be used for learning-based method for event cameras as ground truth. This use of event cameras for table tennis is studied in section 3.3 where we describe a Spiking Neural Network (SNN) approach, using event-based data for ball detection.

3.1 Spin Detection

For table tennis, spin estimation is crucial. Several methods have been developed to solve this problem. We focus on direct spin observation methods from ball images. They can be separated into two categories: logo-based [15] [16][17][18][19] or pattern-based [20][21][14]. The logo-based methods can give out accurate spin estimations, but they fail when the logo is not visible, which makes them unreliable. Previous pattern-based methods relied on registration, which requires high frame rate and high resolution cameras. To overcome these problems, we implemented SpinDOE[22]. SpinDOE uses a dot pattern drawn onto the ball with a 3D printed stencil for accurate and repeatable dot positions.

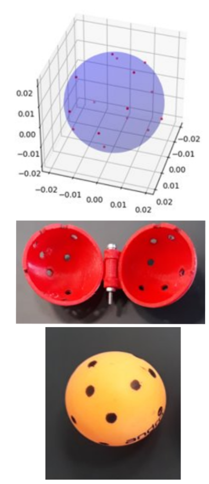

Figure 1: Table tennis balls with dot pattern and corresponding 3D printed stencil.

With our method, we can estimate spins up to 175 rps with a 350 fps camera and a 60x60 ball image resolution. It provides great accuracy as it can be seen in fig. 2. Failure cases come from angle unwrapping errors for extremely high or low spins.

The estimated spin can be used to adapt the racket's orientation for the returning stroke. Our method also enabled us to empirically check the standard assumption in table tennis robotics that the spin is constant while the ball is airborne. The spin dampening coefficient was estimated to be $k = 0.091 \pm 0.03$ with $\omega(t) = \omega(t_0)\exp(-kt)$.

3.2 Multi-modal calibration

In Gossard et al. [23], we introduced **eWand**, a wand-based calibration approach for frame-based and event-based cameras supporting wider baselines

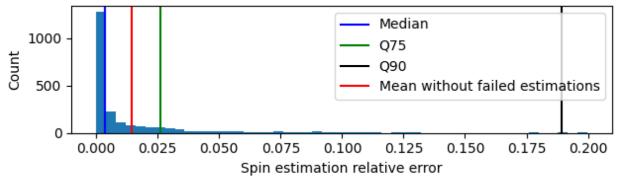

Figure 2: Spin estimation benchmark (failed estimation are those where the relative)

and cameras facing different fields of view. Our setup is described in fig. 3.
Camera calibration for conventional cameras is a well-studied subject, with literature published over

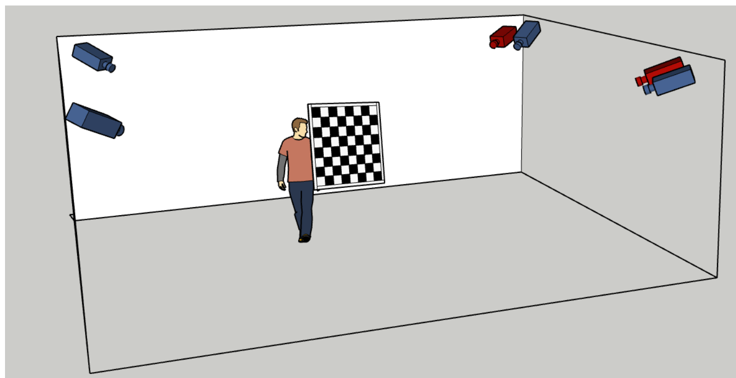

Figure 3: Our camera setup consisting of four frame-based cameras (in blue), one high frame-rate frame-based cameras for ball spin estimation (in orange) and two event-based cameras (in red) with baselines of 3m to 5m. Schematic is up to scale.

multiple decades [24][25][26]. However, not many of these approaches are applicable for our setup. Also, since event-based cameras only report illumination changes, such approaches cannot directly be applied to event-based cameras. To overcome these limitations, we propose **eWand** [23], a new method that uses blinking LEDs inside opaque spheres instead of a printed or displayed pattern. Our method provides a faster, easier-to-use extrinsic calibration approach that maintains high accuracy, listed in table 1, for both event- and frame-based cameras.

Camera	BA eWand (our)	kalibr [26] (circleboard)	kalibr [26] (checkerboard)	BA circleboard	BA checkerboard
frame_0	0.325 ± 0.257	0.274 ± 0.060	0.390 ± 0.325	0.218 ± 0.219	0.225 ± 0.204
frame_1	0.320 ± 0.271	0.311 ± 0.070	0.398 ± 0.335	0.201 ± 0.178	0.201 ± 0.185
frame_2	0.401 ± 0.361	0.168 ± 0.072	0.320 ± 0.320	0.178 ± 0.178	0.213 ± 0.193
frame_3	0.396 ± 0.389	0.159 ± 0.062	0.301 ± 0.243	0.199 ± 0.188	0.180 ± 0.165
event_0	0.603 ± 0.448	0.269 ± 0.068	0.513 ± 0.396	0.398 ± 0.365	0.459 ± 0.363
event_1	0.550 ± 0.431	0.295 ± 0.070	0.515 ± 0.405	0.469 ± 0.447	0.402 ± 0.370

Table 1: Reprojection error (MAE) in pixels (mean and std.) for each calibration approach and camera, after the calibration. "BA" represents a bundle adjustment approach using OpenCV and Ceres.

3.3 SNN for event-based ball detection

Real-time table-tennis ball tracking, crucial for enabling a robotic arm to rally the ball back successfully, demands both fast and accurate ball detection. Previous approaches have employed frame-based cameras with CNNs or traditional computer vision methods. We enter a different avenue and combine event-based cameras and a Spiking Neural Network (SNN) for ball detection.

SNNs mimic the spiking behavior of biological neurons, offering a biologically inspired approach to neural network computation. Unlike traditional artificial neurons that produce real-valued outputs, spiking neurons accumulate input in a membrane potential until a threshold is reached, triggering a spike. This binary output format of SNNs is a perfect fit for event-based cameras and enables energy-efficient computations.

In our approach, we treat x and y positions as two independent classification tasks. In this case, the ball's x position can be one of 128 classes, as can the y position. A visualization of the network's output is provided in fig. 4.

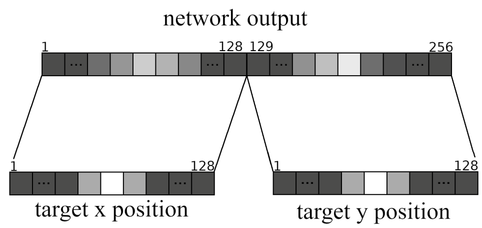

Figure 4: The network has 256 output neurons, which are split into two populations. Each of the neurons in the first population represents a x position, each of the neurons in the second a y position. The target is also split into a x and y target, each setting one as the target activity for the correct neuron, 0.5 for the two adjacent neurons and zero for all others. Values are represented using brightness, with larger values being brighter.

The network consists of four layers, with the first two being convolutional layers and the last two being linear layers. The heaviside step function is used as activation function.

While the standard approach would be to use the cross entropy loss function to train a classification network, the mean squared error (MSE) proved more suitable to this problem. The cause of this is not poor accuracy when training with cross entropy loss, but the tendency of the cross entropy loss to produce networks with very large activations. Large activations result in more spikes, causing more synaptic operations.

In table 2 we list the results of the SNN based ball detection. As a comparison, in Tebbe et al. [1],

Time steps	Accuracy [pixels]	SynOps
8	1.00 ± 0.04	$1.43 \cdot 10^5$
16	0.92 ± 0.03	$2.97 \cdot 10^5$
32	$\mathbf{0.89 \pm 0.03}$	$6.06 \cdot 10^5$

Table 2: Average Euclidean distance in pixels and standard deviation across 10 networks. Also shown are the average number of synaptic operations (SynOps) per forward pass. Numbers in bold mark the best performing method.

we achieved an accuracy of 1.53 pixels with a classical computer vision approach on frames from a frame-based camera.

4 Conclusion

In this paper, we started with an introduction of our table tennis robot system, originally presented in [1]. We then highlighted our recent advancements, ranging from a novel calibration approach to calibrate our multimodal perception system, over a more accurate spin estimation approach, to a Spiking Neural Network (SNN) approach for ball detection with event-based cameras. We hope that this work will foster the discussion, how table tennis robot systems can be further improved and maybe compete against professional players at one point.

Acknowledgments

We would like to thank Kin Man Lee and Zulfiqar Zaidi for approaching us and their encouragement to submit our recent developments. We also appreciate the valuable feedback from Timon Höfer, Thomas Ziegler, and Philipp Riedel. Special thanks to Sony AI for funding this project.

References

[1] J. Tebbe, Y. Gao, M. Sastre-Rienietz, and A. Zell. A table tennis robot system using an industrial KUKA robot arm. In *Lecture Notes in Computer Science*, pages 33–45. Springer International Publishing, 2019. doi:10.1007/978-3-030-12939-2_3. URL `https://doi.org/10.1007/978-3-030-12939-2_3`.

[2] D. D'Ambrosio, N. Jaitly, V. Sindhwani, K. Oslund, P. Xu, N. Lazic, A. Shankar, T. Ding, J. Abelian, E. Coumans, G. Kouretas, T. Nguyen, J. Boyd, A. Iscen, R. Mahjourian, V. Vanhoucke, A. Bewley, Y. Kuang, M. Ahn, D. Jain, S. Kataoka, O. Cortes, P. Sermanet, C. Lynch, P. Sanketi, K. Choromanski, W. Gao, J. Kangaspunta, K. Reymann, G. Vesom, S. Moore, A. Singh, S. Abeyruwan, and L. Graesser. Robotic table tennis: A case study into a high speed learning system. In *Robotics: Science and Systems XIX*. Robotics: Science and Systems Foundation, July 2023. doi:10.15607/rss.2023.xix.006. URL `https://doi.org/10.15607/rss.2023.xix.006`.

[3] S. Gomez-Gonzalez, Y. Nemmour, B. Schölkopf, and J. Peters. Reliable real-time ball tracking for robot table tennis. *Robotics*, 8(4):90, Oct. 2019. doi:10.3390/robotics8040090. URL `https://doi.org/10.3390/robotics8040090`.

[4] T. Ding, L. Graesser, S. Abeyruwan, D. B. D'Ambrosio, A. Shankar, P. Sermanet, P. R. Sanketi, and C. Lynch. Learning high speed precision table tennis on a physical robot. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Oct. 2022. doi:10.1109/iros47612.2022.9982205. URL `https://doi.org/10.1109/iros47612.2022.9982205`.

[5] J. Billingsley. Robot ping pong. *Practical Computing*, 6(5), 1983.

[6] Y. Huang, D. Xu, M. Tan, and H. Su. Trajectory prediction of spinning ball for ping-pong player robot. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Sept. 2011. doi:10.1109/iros.2011.6095044. URL `https://doi.org/10.1109/iros.2011.6095044`.

[7] R. Xiong, Y. Sun, Q. Zhu, J. Wu, and J. Chu. Impedance control and its effects on a humanoid robot playing table tennis. *International Journal of Advanced Robotic Systems*, 9(5):178, Nov. 2012. doi:10.5772/51924. URL `https://doi.org/10.5772/51924`.

[8] D. Buchler, S. Guist, R. Calandra, V. Berenz, B. Scholkopf, and J. Peters. Learning to play table tennis from scratch using muscular robots. *IEEE Transactions on Robotics*, 38(6):3850–3860, Dec. 2022. doi:10.1109/tro.2022.3176207. URL `https://doi.org/10.1109/tro.2022.3176207`.

[9] K. Mülling, J. Kober, O. Kroemer, and J. Peters. Learning to select and generalize striking movements in robot table tennis. *The International Journal of Robotics Research*, 32(3): 263–279, Jan. 2013. doi:10.1177/0278364912472380. URL `https://doi.org/10.1177/0278364912472380`.

[10] H.-I. Lin, Z. Yu, and Y.-C. Huang. Ball tracking and trajectory prediction for table-tennis robots. *Sensors*, 20(2):333, Jan. 2020. doi:10.3390/s20020333. URL `https://doi.org/10.3390/s20020333`.

[11] W. Gao, L. Graesser, K. Choromanski, X. Song, N. Lazic, P. Sanketi, V. Sindhwani, and N. Jaitly. Robotic table tennis with model-free reinforcement learning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Oct. 2020. doi:10.1109/iros45743.2020.9341191. URL `https://doi.org/10.1109/iros45743.2020.9341191`.

[12] J. Tebbe. *Adaptive robot systems in highly dynamic environments: A table tennis robot*. PhD thesis, University of Tübingen, 2021.

[13] H. Li, H. Wu, L. Lou, K. Kuhnlenz, and O. Ravn. Ping-pong robotics with high-speed vision system. In *2012 12th International Conference on Control Automation Robotics & Vision (ICARCV)*. IEEE, Dec. 2012. doi:10.1109/icarcv.2012.6485142. URL `https://doi.org/10.1109/icarcv.2012.6485142`.

[14] J. Tebbe, L. Klamt, Y. Gao, and A. Zell. Spin detection in robotic table tennis. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2020. doi:10.1109/icra40945.2020.9196536. URL `https://doi.org/10.1109/icra40945.2020.9196536`.

[15] T. Tamaki, T. Sugino, and M. Yamamoto. Measuring ball spin by image registration. *Proc. 10th Frontiers of Computer Vision*, pages 269–274, 2004.

[16] C. Theobalt, I. Albrecht, J. Haber, M. Magnor, and H.-P. Seidel. Pitching a baseball. *ACM Transactions on Graphics*, 23(3):540–547, Aug. 2004. doi:10.1145/1015706.1015758. URL `https://doi.org/10.1145/1015706.1015758`.

[17] S. Furuno, K. Kobayashi, T. Okubo, and Y. Kurihara. A study on spin-rate measurement using a uniquely marked moving ball. In *2009 ICCAS-SICE*, pages 3439–3442, 2009.

[18] A. Szep. Measuring ball spin in monocular video. In *Proc. 16th Comput. Vis. Winter Workshop*, pages 83–89. Citeseer, 2011.

[19] T. Tamaki, H. Wang, B. Raytchev, K. Kaneda, and Y. Ushiyama. 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)*. IEEE, Mar. 2012. doi:10.1109/icassp.2012.6288166. URL `https://doi.org/10.1109/icassp.2012.6288166`.

[20] J. Glover and L. P. Kaelbling. Tracking the spin on a ping pong ball with the quaternion bingham filter. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2014. doi:10.1109/icra.2014.6907460. URL `https://doi.org/10.1109/icra.2014.6907460`.

[21] Y. Zhang, R. Xiong, Y. Zhao, and J. Wang. Real-time spin estimation of ping-pong ball using its natural brand. *IEEE Transactions on Instrumentation and Measurement*, 64(8):2280–2290, Aug. 2015. doi:10.1109/tim.2014.2385173. URL `https://doi.org/10.1109/tim.2014.2385173`.

[22] T. Gossard, J. Tebbe, A. Ziegler, and A. Zell. Spindoe: A ball spin estimation method for table tennis robot. In *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*. IEEE, 2023. doi:10.48550/ARXIV.2303.03879. URL `https://arxiv.org/abs/2303.03879`.

[23] T. Gossard, A. Ziegler, L. Kolmar, J. Tebbe, and A. Zell. ewand: A calibration framework for wide baseline frame-based and event-based camera systems, 2023.

[24] T. A. Clarke and J. G. Fryer. The development of camera calibration methods and models. *The Photogrammetric Record*, 16(91):51–66, Apr. 1998. doi:10.1111/0031-868x.00113. URL `https://doi.org/10.1111/0031-868x.00113`.

[25] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*. IEEE, 1999. doi:10.1109/iccv.1999.791289. URL `https://doi.org/10.1109/iccv.1999.791289`.

[26] J. Rehder, J. Nikolic, T. Schneider, T. Hinzmann, and R. Siegwart. Extending kalibr: Calibrating the extrinsics of multiple IMUs and of individual axes. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2016. doi:10.1109/icra.2016.7487628. URL `https://doi.org/10.1109/icra.2016.7487628`.