OverTrack: Overhead Camera Tracking Tool for Child-Robot Interaction

Ameer Helmi[1], Connor M. Phillips[1], Fernando Castillo[1], Samuel W. Logan[2], and Naomi T. Fitter[1]

***Abstract*— Child-robot interaction offers benefits from social skill training to physical activity promotion, but many analyses in this research area require labor-intensive post hoc video coding. One reason for this need is that modern computer vision tools are trained with a dearth of child data. In this paper, we present OverTrack, a semi-automatic tool for overhead-video-based position tracking during child-robot interaction. OverTrack can be used for both post hoc video analysis and real-time position sensing. We evaluated OverTrack's performance on child and robot position tracking by comparing its post hoc accuracy against popular open-source object detection algorithms. OverTrack yielded significantly higher accuracy than the considered alternatives. The products of this work can benefit child development researchers and roboticists interested in child-robot interaction.**

I. Introduction

Assistive robotics for early interventions is a growing area of research that includes our own work for promoting physical activity through social play with a robot [1], [2]. A common standard in behavior analysis for this type of intervention, both in our own work (e.g., [1]) and the efforts of others (e.g., [3]), is manual video coding, a process which is both time- and resource-intensive. Computer vision offers one faster processing solution for both post hoc and real-time analysis. However, current computer vision solutions are commonly trained on datasets with a paucity of child data. Based on this current gap, our central research goal in this paper is *to design and evaluate a computer vision tool which can track child and robot position during child-robot interactions*. In our research context, we sought to achieve our goal with an overhead camera. To this end, we designed and evaluated OverTrack, a semi-automatic overhead region-of-interest (ROI) tracker for use in post hoc and real-time analysis of child-robot interaction. We plan to use the system with our custom assistive robot GoBot, shown in Fig. 1, which promotes physical activity during free play with children [1].

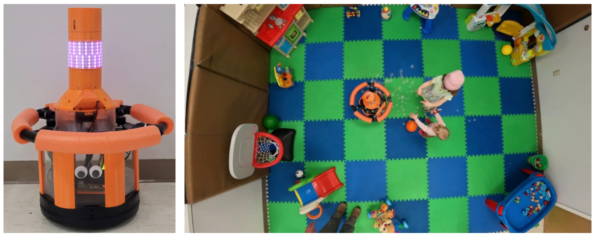

Fig. 1. *Left:* GoBot, our custom assistive robot for encouraging physical activity. *Right:* Overhead view of play space with GoBot and two children.

The performance of existing tools on our overhead tracking task elucidates current challenges in this space. State-of-the-art object detection algorithms such as YOLO [4], OpenPose [5], and R-CNN [6] are trained on side-view camera images and struggle with identifying people in overhead views [7]. Aerial unmanned vehicles commonly use overhead views for object detection but require large datasets for reliably identifying new targets of interest [8], [9]. One openly available tool, RAPiD, was trained on images of adults for object detection with an overhead fisheye lens [10]; we consider this option as a comparison point in our work. In child-robot interaction, the availability of training data is scarce, and varying outfits, mobility aids, and behaviors lead to challenges in constructing individual user profiles. One example used the SMIL model to identify spontaneous bouts of infant motion, but this case was trained on infants who were not yet mobile [11]. Separate approaches have required participants to wear a specific color outfit [12] or an AR tag outfit [13] to support overhead position tracking; however, these approaches have relied on having the consistent outfits to enable tracking. As presented in this paper, our tool offers advantages over competing alternatives, as OverTrack does not require child training data, multiple camera setups, or specific child attire to function.

This paper documents the design and early evaluation of post hoc positional analysis by our OverTrack tool. OverTrack is publicly available for use on the OverTrack repository [14]. We describe the tool design in Section II. In Section III, we describe the methods and results of our post hoc testing; overall OverTrack showed high accuracy and performance according to our metrics. We summarize key conclusions, strengths, and limitations of the work in Section IV. Our main contribution in this paper is the collaborative design and early evaluation of OverTrack, a tool which we believe may benefit early childhood researchers.

II. OverTrack Tool Design

To create a system that would fit our tracking needs, we first discussed tool design with kinesiology collaborators and then prepared software to satisfy identified requirements.

A. Design Criteria

In early discussion among the paper authors, who span the fields of robotics and kinesiology, we sought to determine requirements of the tracking system, child behaviors that we

[1] Collaborative Robotics and Intelligent Systems (CoRIS) Institute, Oregon State University, Corvallis, Oregon, 97331, USA(helmia, fittern)@oregonstate.edu

[2] Oregon State University Disability and Mobility Do-It-Yourself Co-op, College of Public Health and Human Sciences, Oregon State University, Corvallis, Oregon, 97331, USA(Sam.Logan)@oregonstate.edu

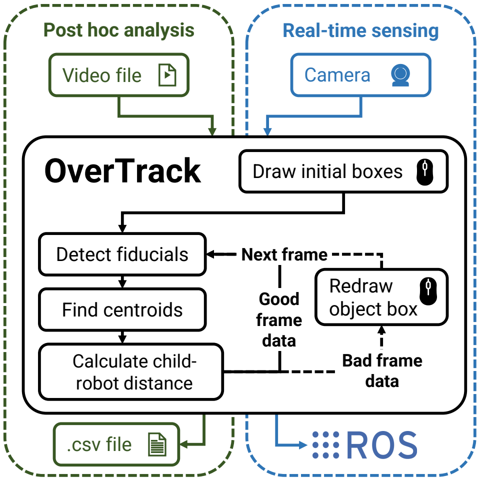

Fig. 2. Flowchart showing OverTrack system operation. The green dashed region corresponds with post hoc use (the part validated in this paper), while the blue dashed region corresponds with real-time use. User-inputted items are denoted with a mouse icon.

would want to detect with our system, and feasible protocols for use. Child-robot playgroups which are held as part of our collaborative research efforts, and other playgroups hosted by the Oregon State University Disability and Mobility Do-It-Yourself Co-op, typically occur with some incorporated data collection protocols. The most common data collection method has been overhead video recording via a GoPro camera. Some efforts also include side-view video recordings and/or wearable inertial sensor data collection. Side-view camera recordings can offer advantages for skeleton tracking applications, but due to the number of toys and other children in the play area, occlusion can be a serious obstacle from this view. Likewise, wearable inertial sensors are convenient for certain types of activity tracking, but they are difficult to use for general position estimation. Thus, we chose to work with overhead camera views to design our general tracking tool.

The identified behaviors that our collaborators from the Disability and Mobility Do-It-Yourself Co-op usually hand-code in post hoc video annotation are as follows:

- Child position and orientation
- Child movement level and speed
- Social play occurrence, evaluated by inter-child spacing
- Directed social interaction, evaluated by inter-child spacing
- Posture and activity level

Most of these categories can be quantified exclusively by tracking position; thus, we focused on position tracking while crafting our overhead camera tracking tool.

Finally, we worked with our collaborators to determine appropriate protocols for system use during an intervention. In robotics research, autonomous function is often the desired operating state of sensing systems; however, it has been far more common for human-in-the-loop tools to be used in past work by our collaborators (i.e., the Disability and Mobility Do-It-Yourself Co-op) and others working in related kinesiology spaces. This fact gave us flexibility in our design in which we could seek an autonomous system, but if necessary, dedicate a human operator's effort to lightly monitoring and supplying any needed input to the tracker.

B. *Resulting Tracker Tool*

Based on the need to track position from overhead camera data, OverTrack is an OpenCV-Python [15] implementation of a semi-automatic multi-object ROI tracker. Figure 2 shows a flowchart of system operation, including user-inputted and automated steps. The system can be used for both post hoc and real-time video analysis. OverTrack can employ any of the following openly available OpenCV multi-object trackers: CSRT [16], KCF [17], MedianFlow [18], MOSSE [19], Boosting [20], and MIL [21]. Related work evaluated each of the above trackers and found that CSRT and KCF were the best in terms of precision, MedianFlow and MOSSE had the highest frames per second (FPS), and Boosting and MIL had the highest success rate [22]. Based on common needs in robotics research, we added support for both real-time tracking with communication through Robot Operating System (ROS) and tracking of ArUco tags that may be on a robot or other objects in the environment. OverTrack requires intermittent input from a human operator, which is a common constraint of the tool operation modes from our collaborators.

The system operator performs four main steps. First, the user is prompted to draw a bounding box surrounding the play space so that only regions-of-interest (ROI) (i.e., the robot and children) within the play space are tracked. The boundary is displayed to the user as a black rectangle, as shown in Fig. 3. Next, the user sets the scale which will serve as a known reference distance for calculating Euclidean distance between bounding boxes. In our videos, we set the scale based on checkered 2ft×2ft (0.61m×0.61m) colored mats in the play space. The user is then able to use specific keys to select and draw a bounding box for each object of interest (e.g., robots, children, toys) in the play space, as further explained in the software package documentation [14]. As shown in Fig. 2, for each frame, the positions and velocities of the drawn bounding boxes are tracked and saved in a Pandas [23] DataFrame. This information is subsequently exported to a csv file or transmitted in real time via ROS. If the child or robot leaves the overhead camera view and later returns (or if the tracker loses track of any ROI), the corresponding bounding box must be redrawn.

Figure 3 shows the system in use with the boundary engaged, the scale set, and bounding boxes selected for GoBot and a child participant from our past study data. Instructions for downloading and using OverTrack are available on the OverTrack repository [14].

III. Post Hoc Performance Evaluation

To assess OverTrack's performance in a post hoc data analysis context, we compared its performance with common computer vision tools on video recordings from a past study.

A. *Video Dataset and Ground Truths*

Our evaluation dataset came from a recent study with $n = 4$ participants, wherein our assistive robot encouraged

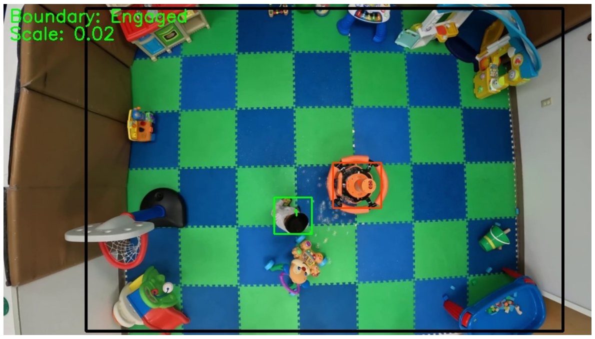

Fig. 3. A frame from OverTrack use during post hoc data extraction. The thick black rectangle surrounding the play space denotes the play space boundary, the orange bounding box marks the robot, and the green bounding box marks the participant. The text in the top left corner indicates that the boundary is set and shows the scale in use for distance measurements.

physical activity during seven weekly play sessions. All study procedures were approved by Oregon State University under protocol #IRB-2020-0723. During each weekly session, we recorded 10 minutes of overhead child-robot interaction video using a GoPro HERO10 running at 30 FPS. An example frame from the overhead view appears in Fig. 3. Due to a technical error during the study trials, one video was not successfully recorded, but the other 27 videos were included in the analysis.

Ground-truth positional data was collected by a trained coder who marked the bounding box for the assistive robot and any present children on a randomly selected 3,126 frames across all videos, which represents 1.3% of the overall captured video frames. A second trained coder likewise marked bounding boxes on 10% of the 3,126 frames. Using metrics further described below, we found an average root-mean-square error (RMSE) of 0.3ft (0.09m) between the centroids of bounding boxes marked by the coders and an inter-rater reliability of 91% for identifying the correct number of ROIs. Agreement of 85% or higher is considered acceptable in observational studies of children [24].

B. Tool Evaluation Methods

For the OverTrack evaluation, we needed tracking results from our proposed tool, points of comparison from other common computer vision resources, and metrics for the eventual comparison. During the post hoc evaluation process, we tested and confirmed the compatibility of OverTrack with Windows 10, Mac OS Maverick, and Linux PCs running Ubuntu 16.04 through Ubuntu 20.04.

1) OverTrack Settings for the Evaluation: For the evaluation, we extracted post hoc ROI position data from all 27 available study videos using OverTrack with the MedianFlow tracker [18]. We used the MedianFlow tracker due to its high FPS with relatively high success rate, as demonstrated in past related work [22].

2) Computer Vision Benchmarks: We compared the performance of OverTrack against multiple relevant and openly available algorithms, including YOLOv7 [25], OpenPose [5], and RAPiD [10]. We conducted an initial analysis of OpenPose and YOLOv7 on two of the study videos, but they both yielded close to 0% accuracy in detecting the children and robots from the available overhead camera view, so we did not perform data extraction or analysis for the rest of the videos. For every video, we ran both OverTrack and RAPiD using a Linux PC running Ubuntu 20.04 with an NVIDIA GTX 1080 Ti GPU, recording the centroids of the robot and child participants, frame rate, and the total number of ROIs detected per frame. We used the author-recommended confidence threshold of 0.3 for object detection with RAPiD [10]. For clarity, we note that both OverTrack and RaPiD were used directly, without any transfer learning step involving our own video datasets.

3) Performance Metrics: Based on the raw information collected from the trackers (i.e., centroids of ROIs, frame rate, and number of ROIs), we identified metrics (i.e., positional accuracy, runtime, and success rate) for comparisons grounded in related computer vision literature.

We aimed to understand the *positional accuracy* of both RAPiD and OverTrack by computing the positional distance RMSE between the ground-truth bounding box centroids and the tool-outputted (i.e., RAPiD and OverTrack) bounding box centroids across the study video data. Because we could not be sure which bounding box from RAPiD was meant to represent the child or robot, we chose the closest RAPiD-outputted bounding box to each ground-truth bounding box, and excluded frames in which RAPiD did not identify any ROIs. We calculated the mean RMSE across each participant's video dataset and the total mean RMSE across all frames. Based on the recommendation of our kinesiology collaborators and a threshold we calculated based on related computer vision work [26], we determined that a mean RMSE below 0.5ft would signal success for using OverTrack as a post hoc analysis tool.

For *runtime* comparison, we computed the average FPS rate for each tracker. This metric provided an idea of how quickly each tool could potentially run in post hoc analysis.

To better understand the *success rates* of the trackers, we compared the number of objects identified by each tool in ground-truth frames. A correct (true positive) identification was defined as when the number of objects identified by the tracker matched the ground-truth count. A false positive was defined as when more objects were identified by the tracker compared to the ground truth. A false negative was marked if fewer objects than the correct count for robots and children were marked. Using the aggregated true positive, false positive, and false negative counts, we calculated the average accuracy, precision, recall, and F1 score of object count across all videos [27].

C. Tool Evaluation Results

Our analysis showed low success for RAPiD and high success for OverTrack, according to our metrics. Performance results of RAPiD and OverTrack appear in Table I and Table II. Across all participants and all frames, OverTrack had a lower RMSE and therefore a higher positional accuracy. The OverTrack RMSE value of roughly 0.5ft indicates success.

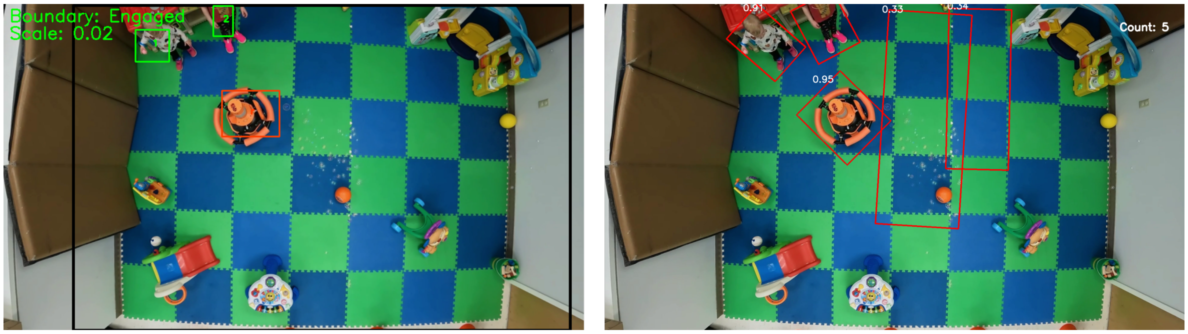

Fig. 4. Representative frames from OverTrack (left) and RAPiD (right) with bounding box renderings from post hoc position-tracking comparisons. In this case, RAPiD incorrectly identified portions of the play floor as ROIs.

TABLE I
RAPiD AND OVERTRACK POSITIONAL ACCURACY (IN FEET), IN THE FORM $Mean \pm SD$.

	P1 RMSE	P2 RMSE	P3 RMSE	P4 RMSE	Mean RMSE
RAPiD	2.5±2.9	2.7±3.1	2.6±3.2	2.6±2.8	2.6±3.0
OverTrack	0.5 ±0.3	0.5±0.2	0.5±0.2	0.4±0.3	0.5 ±0.3

TABLE II
RESULTS FOR THE RAPID AND OVERTRACK RUNTIME, ROI COUNT ACCURACY, PRECISION, RECALL, AND F1 SCORE.

	FPS	Count Accuracy	Precision	Recall	F1 Score
RAPiD	9.5	28%	0.34	0.59	0.43
OverTrack	38.8	92%	0.97	0.94	0.96

RAPiD ran at an average of 9.5 FPS while OverTrack ran at an average of 38.8 FPS during data collection. For accuracy, precision, recall, and F1 score, RAPiD yielded below 0.6 while OverTrack achieved higher than 0.9. Across the 3,126 ground-truth frames, we found 1,607 false positives and 573 false negatives for RAPiD compared to 80 false positives and 171 false negatives for OverTrack.

Qualitatively, it is also helpful to note the types of errors that tended to occur with each tracker. We saw two main types of errors with RAPiD. Uninhabited areas of the play space were often misidentified as ROIs by RAPiD, as shown in Fig. 4. Additionally, RAPiD occasionally generated a bounding box that was much larger than the actual ROI. We observed that OverTrack failures arose in some cases when children moved quickly across the play space. The tracker would lose the child, and the delay until the human operator could redraw the bounding box would lead to a decline in accuracy. False positives and false negatives in OverTrack object count occurred in some scenarios when the trained coders took different approaches to annotating in cases of occlusion.

IV. Discussion

Our OverTrack efforts have been a collaboration between roboticists and kinesiologists to craft a tool for faster post hoc data extraction and potential real-time overhead position tracking during child-robot interaction sessions. Due to a lack of sufficient datasets for training trackers with children, current open-source fully automated solutions were not yet viable for use in our child-robot interaction application. We tested three fully automated and openly available tools that were not as accurate at tracking ROIs when compared with OverTrack. Even with no custom retraining, OverTrack was effective at tracking the positions of GoBot and children in our dataset. This result can be valuable for researchers working with similar datasets, as OverTrack is openly available [14] and can effectively capture video-based metrics for interaction and engagement during child-robot interaction. The most typical errors to expect with OverTrack include losing track of an ROI that is partially occluded or that moves quickly.

Key strengths of this work include the low RMSE and high precision, recall, and F1 score for OverTrack when compared with openly available automated tools. The most current implementation of OverTrack, accessible via the repository, works in real time as part of a robot's sensing suite.

Limitations of this work include that OverTrack requires a human in the loop to operate; however, the considered fully autonomous options do not yet yield sufficient performance levels in the child-robot interaction context. Although our work uses bounding boxes to represent child location, other approaches such as skeleton tracking may give different centroid results. There is also room for more representation (e.g., different flooring types) within our test dataset.

In *conclusion*, we designed and tested OverTrack, a publicly available semi-autonomous tool for overhead video tracking. OverTrack provides high accuracy with reasonable frame rates in post hoc analysis and can be used on most computers with minimal setup. Our future work with OverTrack will focus on integrating OverTrack with GoBot's current sensing capabilities and conducting further studies to validate OverTrack's real-time performance levels. Researchers working in child-robot interaction and child development can benefit from this work.

V. Acknowledgments

We thank Katelyn Chung, Christine Zhan, and the U.S. National Science Foundation (award CMMI-2024950) for their support of this work.

References

[1] J. R. Vora, A. Helmi, C. Zhan, E. Oliviares, T. Vu, M. Wilkey, S. Noregaard, N. T. Fitter, and S. W. Logan, "Influence of a socially assistive robot on physical activity, social play behavior, and toy-use behaviors of children in a free play environment: A within-subjects study," *Frontiers in Robotics and AI*, 2021.

[2] A. Helmi, T.-H. Wang, S. W. Logan, and N. T. Fitter, "Harnessing the power of movement: A body-weight support system & assistive robot case study," in *Proc. of the Int. Consortium on Rehabilitation Robotics (ICORR)*, 2023.

[3] M. Rueben, M. Syed, E. London, M. Camarena, E. Shin, Y. Zhang, T. S. Wang, T. R. Groechel, R. Lee, and M. J. Matarić, "Long-term, in-the-wild study of feedback about speech intelligibility for k-12 students attending class via a telepresence robot," in *Proc. of the Int. Conf. on Multimodal Interaction (ICMI)*, 2021, pp. 567–576.

[4] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[5] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh, "OpenPose: Realtime multi-person 2D pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[7] S. Li, M. O. Tezcan, P. Ishwar, and J. Konrad, "Supervised people counting using an overhead fisheye camera," in *Proc. of the IEEE Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS)*, 2019, pp. 1–8.

[8] A. Gaszczak, T. P. Breckon, and J. Han, "Real-time people and vehicle detection from UAV imagery," in *Intelligent Robots and Computer Vision XXVIII: Algorithms and Techniques*, vol. 7878. SPIE, 2011, pp. 71–83.

[9] J. Lee, J. Wang, D. Crandall, S. Šabanović, and G. Fox, "Real-time, cloud-based object detection for unmanned aerial vehicles," in *Proc. of the IEEE Int. Conf. on Robotic Computing (IRC)*, 2017, pp. 36–43.

[10] Z. Duan, O. Tezcan, H. Nakamura, P. Ishwar, and J. Konrad, "RAPiD: rotation-aware people detection in overhead fisheye images," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision and Pattern Recognition Workshops (CVPR)*, 2020, pp. 636–637.

[11] N. Hesse, S. Pujades, M. J. Black, M. Arens, U. G. Hofmann, and A. S. Schroeder, "Learning and tracking the 3D body shape of freely moving infants from RGB-D sequences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 10, pp. 2540–2551, 2019.

[12] E. S. Short, E. C. Deng, D. Feil-Seifer, and M. J. Matarić, "Understanding agency in interactions between children with autism and socially assistive robots," *Journal of Human-Robot Interaction*, vol. 6, no. 3, p. 21–47, dec 2017.

[13] E. Kokkoni, E. Mavroudi, A. Zehfroosh, J. C. Galloway, R. Vidal, J. Heinz, and H. G. Tanner, "GEARing smart environments for pediatric motor rehabilitation," *Journal of Neuroengineering and Rehabilitation*, vol. 17, no. 1, pp. 1–15, 2020.

[14] A. Helmi, C. Phillips, C. Zhan, and N. T. Fitter, "OverTrack software repository," July 2022. [Online]. Available: https://github.com/shareresearchteam/OverTrack

[15] K. Pulli, A. Baksheev, K. Kornyakov, and V. Eruhimov, "Real-time computer vision with OpenCV," *Communications of the ACM*, vol. 55, no. 6, pp. 61–69, 2012.

[16] A. Lukezic, T. Vojir, L. Cehovin Zajc, J. Matas, and M. Kristan, "Discriminative correlation filter with channel and spatial reliability," in *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6309–6318.

[17] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2014.

[18] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *Proc. of the IEEE Int. Conf. on Pattern Recognition (ICPR)*, 2010, pp. 2756–2759.

[19] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 2544–2550.

[20] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via online boosting," in *Proc. of the British Machine Vision Conf. (BMVC)*. BMVA Press, 2006, pp. 6.1–6.10, doi:10.5244/C.20.6.

[21] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 983–990.

[22] A. Brdjanin, N. Dardagan, D. Dzigal, and A. Akagic, "Single object trackers in opencv: A benchmark," in *Proc. of the IEEE Int. Conf. on INnovations in Intelligent SysTems and Applications (INISTA)*, 2020, pp. 1–6.

[23] W. McKinney *et al.*, "pandas: a foundational python library for data analysis and statistics," *Python for High Performance and Scientific Computing*, vol. 14, no. 9, pp. 1–9, 2011.

[24] S. W. Logan, M. Schreiber, M. Lobo, B. Pritchard, L. George, and J. C. Galloway, "Real-world performance: Physical activity, play, and object-related behaviors of toddlers with and without disabilities," *Pediatric Physical Therapy*, vol. 27, no. 4, pp. 433–441, 2015.

[25] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 7464–7475.

[26] Z. Dong, S. Liu, T. Zhou, H. Cheng, L. Zeng, X. Yu, and H. Liu, "PPR-Net:point-wise pose regression network for instance segmentation and 6D pose estimation in bin-picking scenarios," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2019, pp. 1773–1780.

[27] R. Brégier, F. Devernay, L. Leyrit, and J. L. Crowley, "Symmetry aware evaluation of 3D object detection and pose estimation in scenes of many parts in bulk," in *Proc. of the 2017 IEEE Int. Conf. on Computer Vision Workshops (ICCV)*, 2017, pp. 2209–2218.