2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation

Catch the Shadow: Person Tracking Under Occlusion with a Single RGB-D Camera

Wei Gai[1], Meng Qi[2*], Lu Wang[1], Chenglei Yang[1,4], Juan Liu[1],
Yulong Bian[1], Gerard de Melo[3], Shijun Liu[1], and Xiangxu Meng[1,4]
[1]*School of Software*
Shandong University, Jinan, China
Email: gw@sdu.edu.cn
[2]*Shandong Normal University, Jinan, China*
Email: qimeng@sdnu.edu.cn
[3]*Rutgers University, New Brunswick, USA*
[4]*Engineering Research Center of Digital Media Technology, MOE, Jinan, China*

***Abstract*—Locomotion in physical space is one of the most natural forms of interaction in applications such as virtual reality systems. Although there are many algorithms to track walking people, existing methods mostly fail to cope with occluded bodies in the setting of multi-person tracking with one camera. This paper proposes a method to overcome this challenge by fusing skeletal with shadow data, both of which are captured by a single RGB-D camera. Skeletal tracking provides the positions of people that can be captured directly, while their shadows are used to track them when they are no longer visible. Our experiments confirm that this method can efficiently handle full occlusions. It thus has substantial value in resolving the occlusion problem in multi-person tracking, even with other kinds of cameras.**

***Keywords*-Multi-person tracking; RGB-D camera; shadow; occlusion**

I. Introduction

In immersive virtual environments, locomotion through the virtual space is among the most crucial forms of interaction. The primary manifestation of human locomotion is walking, and, hence, genuine walking has substantial advantages over both virtual walking and flying as a mode of locomotion, in terms of its simplicity, straightforwardness, and naturalness [1]. Thus, it is not surprising that real walking in the physical space, which can engender greater degrees of flow experience and preference with respect to non-moving modes [2], has emerged as one of the most natural and effective interaction methods in virtual reality systems [3].

There are many algorithms seeking to track genuinely walking people, and visual tracking is a popular form with a long history [4]. Recently, RGB-D cameras such as Microsoft's Kinect, which is based on vision techniques, have enabled many applications. They constitute a non-intrusive and appealing tracking technology due to their low cost and ease of deployment [5]. Unfortunately, one often faces the challenge of occlusion in multi-person tracking with a single front-view camera [6].

In recent years, many methods have sought to address this, including methods based on multiple cameras [7], Kinect setups relying on the ceiling [8], and approaches that fuse Kinect signals with other sensors [9]. However, these approaches may not be suitable in all settings, given issues such as their high cost or inconvenient deployment setup for users. They also do not solve the problem of occlusion during a long period of interaction or the problem of full-body occlusion.

In this paper, we assess to what extent shadows can serve as clues in tracking human movement. This is motivated by the fact that the shadow of a person moves in sync with a person's body. Wang and Yagi also showed that shadows were helpful in pedestrian detection [10]. A person's shadow exists in either indoor or outdoor conditions in most cases. In cases where such shadows are lacking, we can easily bring about shadows by adding a low-cost light source.

We propose a multi-person tracking algorithm fusing shadow signals in the RGB image with skeleton data, both of which are captured solely by a single RGB-D camera without any reliance on other sensors. Our experiments and sample application results show that our algorithm can resolve even long-duration and full-body occlusions using a single Kinect. This in turn helps to improve the tracking capability of the Kinect.

II. Related Work

Body occlusion is an important yet insufficiently well resolved problem in multi-person tracking. In this section, we mainly introduce related work with regard to tracking algorithms based on RGB-D cameras.

In recent years, the arrival of cheap RGB-D devices (such as Microsoft's Kinect) has facilitated the development of many new approaches to multiple person tracking. These sensors can provide color information as well as the estimated depth for each pixel [11]. RGB image and depth data are often used jointly as cues to resolve partial occlusion [12].However, occlusion, especially full occlusion, is still a significant problem in real deployments of single, front-view camera systems.

978-1-7281-4034-6/19/$31.00 ©2019 IEEE
DOI 10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00114

Multiple cameras can be deployed to resolve full occlusions while tracking people [13]. However, installing more cameras has a number of downsides, such as higher costs, difficulty in calibration, and an inconvenient deployment setup for users.

To address this problem, recent work has focused on the single perspective occlusion problem. An optimal camera placement scheme can aid in avoiding the full occlusion problem [8]. However, in settings with a high ceiling or without any ceiling, mounting a camera to obtain a bird's eye view is either unfeasible or inconvenient. Another approach to cope with occlusion under a single perspective is to rely on prediction methods based on motion trajectories, such as particle filters [14]. However, long-duration occlusion from a single camera cause a loss of observation information, and these methods may fail to track the occluded person in the presence of long-duration or full occlusions. Further research has proposed methods to fuse Kinect data with other sensor data [9], [15]. None of the aforementioned methods fully address the long-duration and full occlusion problems adequately.

In this paper, we focus on human tracking solutions that have low cost and are easy to deploy, relying on just a single Kinect without any other sensors. In our tracking algorithm, the shadow of a person serves as a clue. Our approach efficiently tracks human movement by fusing shadow information in the RGB image with skeleton data, both of which are captured solely by a single Kinect, which can resolve occlusion issues even under long-duration or full-body occlusions.

III. Shadow-based Tracking Algorithm (STA)

In this section, we shall introduce the occlusion cases we can deal with, explore the basic idea and principle, and provide the details of our algorithm.

A. Central Idea and Principle

As a popular RGB-D camera, Kinect devices can provide color, depth, and predicted skeleton data. The Kinect SDK provides data in three spaces: the color image space, depth image space, and skeleton space. In many applications, it has been shown that skeleton data can be reliably used to track people. Kinect V2 can predict the skeleton data of up to 6 persons simultaneously. However, with a single Kinect, the skeleton of a person is lost when that person is occluded by others. Although RGB image and depth data of the Kinect can be used together as clues to resolve partial occlusions, these methods cannot handle complete body occlusions particularly well. In Fig. 1, person H_b is occluded by person H_f. In this case, the Kinect fails to detect the skeleton of H_b.

In such cases, we can rely on shadows, which are present in both indoor and outdoor settings and can also be expressly created by adding a light source, as a simple and low-cost

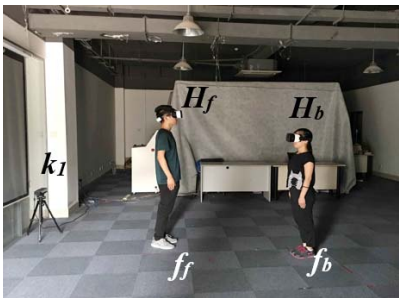


Figure 1. An example of an occlusion event.

solution. Since the shadow of a person always moves in conjunction with that person's body, it can easily be captured from the RGB image of the Kinect, and thus it is possible to evaluate the position of the occluded person by analyzing their shadow.

Fig. 2 show the tracking trajectories of one person computed by her skeleton (blue line) and shadow (red line), wherein the person walks at different speeds along different directions in the coordinate system of the Kinect. The results show that the position of the person computed by her shadow is close to that computed by her skeleton. Thus, we can rely on shadows as a clue to assist capturing a person's position when their skeleton is lost, while relying on the skeleton data to compute the position of a person when their skeleton is available.

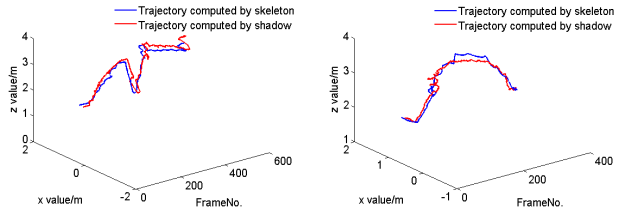


Figure 2. The tracking trajectories of one person computed by her skeleton (blue line) and shadow (red line), respectively, while she walks in arbitrary directions in the coordinate system of the Kinect, (a) at various speeds, and (b) at a constant speed.

Hence, the key idea of our algorithm is as follows: If the skeleton data can be obtained by the Kinect, we use it to track people; otherwise, i.e., in the case that the skeleton data of a person is not available, we make use of their shadow in the RGB image captured by the Kinect to assess the person's position.

When relying on the shadow of a person to evaluate their location, it is necessary to segment the shadow in the RGB image and compute the position in the skeleton space. This involves a conversion between the image space and skeleton space. In particular, Fig. 1 shows the RGB image, in which H_b is occluded by H_f. Fig. 3 shows the skeleton space,

where o is the center of the Kinect infrared camera. G is the plane corresponding to the ground, parallel to the xoz plane. Here, o_1 is the projection point of o on the ground plane G, and p_f as well as p_b represent the positions of the feet of H_f and H_b, respectively. Here, p_f, p_b, o_1 are on G. Since H_b is occluded by H_f, the points o_1, p_f, p_b are on the same line. In this case, H_b is occluded by H_f, but the shadow of H_b is visible for the RGB camera. In Fig. 3, the shadow of H_b is represented by st. Therefore, we can compute the intersection point of o_1p_f and st on the plane G, which can be seen as p_b to represent the position of H_b.

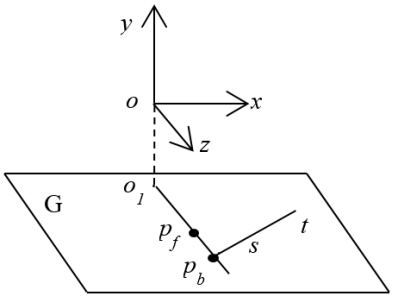


Figure 3. The skeleton space of Kinect and the transformation relationship between RGB image space and skeleton space.

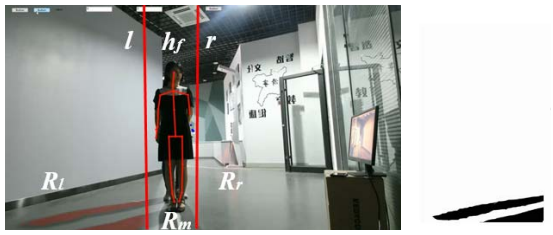
Additionally, to quickly and efficiently extract the shadow of the occluded person, we need to locate the region in which the occluded person's shadow can be found. In this paper, we assume that the light source and Kinect are placed such that the shadow of each person is always on the left (or right) side of their body from the perspective of the Kinect during the tracking process. We let $a = 0$ or 1 designate the left or right side, respectively.

As shown in Fig. 4(a), the lines l and r divide the RGB image into three parts R_l, R_m, and R_r, where R_l and R_r are on the left and right side of H_f, respectively. R_r has no shadow, R_l only has shadows of H_f and H_b, and the bodies and small parts of shadows of H_f and H_b are in the region R_m. Hence, the region R_l only has shadows of H_f and H_b after subtracting the background image (cf. Fig. 4(b)). In this case, it is easier to extract shadows from R_l than from the entire RGB image.

During the tracking process, if the shadow of H_b appears in R_l (when $a = 0$), then it will always exist in R_l. Here, R_l changes along with changes of the position of H_f in each frame (the method to compute these will be introduced in Section IV). Hence, we extract shadows from R_l. Similarly, if the shadow of H_b appears in R_r (when $a = 1$), then it will always exist in R_r. Here, R_r changes along with any change in position of H_f in each frame. Hence, we extract shadows from R_r.

B. Algorithm Overview

In our method, when there is no loss of tracking, we detect the human body and obtain its skeletal model using the standard approach [16]. The skeleton is used to compute the person's position. Otherwise, the shadow is used to track a person whose skeleton is lost in the tracking process.



(a) (b)

Figure 4. The region in which the occluded person's shadows are located. (a) The RGB image is divided into three regions and (b) the difference image is obtained by subtracting the background image from R_l. It includes the occluded person's shadow.

In the initialization stage, the algorithm first captures the background image. For each person, once they are inside the depth-perceiving area of the Kinect, we begin to obtain their position using their skeleton data. Additionally, to quickly and efficiently extract the shadow of the occluded person, we need to determine the region in which the occluded person's shadow is located, and set the value of a.

During the running stage, we obtain their position using a person's skeleton data for each frame. If the skeleton of a person exists, we compute their position. Otherwise, we invoke a Shadow-based Tracking Algorithm (STA) to evaluate their positions.

In the following, we give an overview of our STA algorithm.

Algorithm 1 Shadow-based Tracking Algorithm

Require: A background image, current color image, and the skeleton data of H_f.

Ensure: The position of the occluded person H_b.

1: Find the region R including the shadow in the current color image according to the value of a that we computed in the initialization stage, based on the background image and the skeleton data of H_f;
2: Extract H_b's shadow in R;
3: Compute the position of H_b based on the shadow;
4: **return** The position of H_b

IV. Algorithm Implementation

A. Search Region Identification and Shadow Extraction

In the following, we consider how to find the region R and extract the shadow in accordance with the value of a computed in the initialization stage.

We first obtain the head joint point of H_f based on their skeleton data and transform it into the RGB image space, marked as h_f (cf. Fig. 4(a)).

If $a = 0$, then we consider the line l, which is a perpendicular line across the point $(h_f.x - d_x/2, 0)$. The left region R of l will be used to extract the shadow of H_b. Otherwise, we consider the line r, which is a perpendicular line across the point $(h_f.x + d_x/2, 0)$. The right region R of r will be used to extract the shadow of H_b. Here, d_x is evaluated according to the maximum width of the bodies of H_f and H_b in the initialization stage.

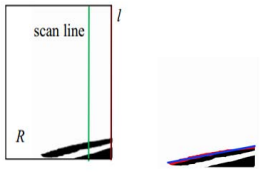
Subsequently, we obtain the difference image C by subtracting the background image B from R:

For each pixel $R(x, y) \in R$, $C(x, y) = |R(x, y) - B(x, y)|$. If $C(x, y) > T$, then $C(x, y)$ is considered as belonging to the shadows of H_f or H_b, and we set $C(x, y) = 1$; otherwise, we set $C(x, y)$=0.

B. Compute the Position of Occluded People via Shadows

Next, we consider how to compute the position of the occluded person. This entails computing the intersection point of o_1p_f and st on the plane G (cf. Fig. 3).

First, we scan R from left to right. For each scan line, we access the pixel in C from top to bottom (cf. Fig. 5(a)). When we encounter the first pixels with value 1, we record this, stop the scan, and initiate the next scan. All such pixels together constitute the upper contour of H_b. Here, S is used to represent the set of points on the upper contour of H_b. Then, we use least squares method to fit this contour of H_b to a line (the blue line in Fig. 5(b)), and map it onto the plane G, which is st. Finally, we set the position of H_b as the intersection point of st and o_1p_f.



(a) (b)

Figure 5. The result of fitting the upper contour of H_b to a line.

V. Experimental Results

A. Experiment Design

For our experiments, we rely on a setup with one main Kinect k_1, and a secondary one k_2 for evaluation purposes. In the tracking process, H_f is always visible from k_1, while for H_b, the device may experience tracking loss. The Kinect k_2 is used to record the position of H_b. Moreover, H_b is always visible from k_2, and the trajectories obtained by k_2 are used as reference values to test our method.

We design two experiments to assess the system. In the first experiment, we specifically evaluate the tracking accuracy when tracking is inhibited due to bodily occlusion. In the second experiment, we evaluate the tracking accuracy when the human participant moves freely, whereby the skeleton may on occasion be tracked successfully, and on occasion may fail to be tracked.

Experiment 1: The first experiment is designed to assess the accuracy of our method when person H_b is occluded. In the experiment, the person is free to walk around, such that x and z values may change. In order to better verify the accuracy of the algorithm, we considered three different runs along different paths:

Path 1: When the points o_1, p_f and p_b are approximately collinear, and the line o_1p_f is parallel to the z-axis, H_b moves back and forth along the z direction, as in Fig. 3. We analyze the tracking accuracy with regard to the z value when H_b is in full and long-duration occlusion.

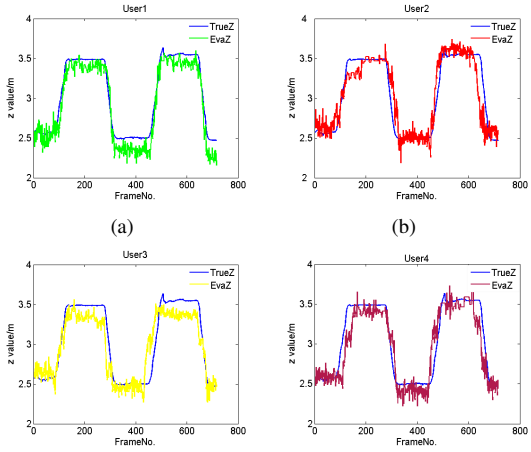
Path 2: When a full-body occlusion occurs, H_f and H_b move back and forth along the x direction simultaneously. We analyze the tracking accuracy of H_b with regard to the x value.

Path 3: When the points o_1, p_f and p_b are approximately collinear and the line o_1p_f is not parallel to the z-axis, H_b moves back and forth along the line o_1p_f. We analyze the tracking accuracy of H_b.

We assess each of these paths 10 times, relying on a pool of 5 human participants to assume the roles of H_b and H_f.

Experiment 2: In the first experiments, the user's motion path was designed in advance. In the second experiment, the participant H_b is instructed to move freely within the space. We verify the effectiveness of our method in various scenarios that may occur during person tracking, including non-occlusion and occlusion.

B. Results



(a) (b) (c) (d)

Figure 6. Comparison between trajectories as tracked by a person's skeleton as opposed to computed using shadows for Path 1, where Users 1, 2, 3, and 4 are randomly selected human participants, and their tracking results correspond to (a), (b), (c), (d), respectively.

1) Tracking Plots: In Experiment 1, there are three different motion paths. The comparison between the position obtained via H_b's skeleton and the position computed via our shadow-based algorithm for Path 1 is given in Fig. 6. Here, H_b moves back and forth along the z direction. The result show that there is only a minor deviation between the trajectories as tracked by the participant's skeleton and computed by our method when the person is in long-term occlusion and full-body occlusion.

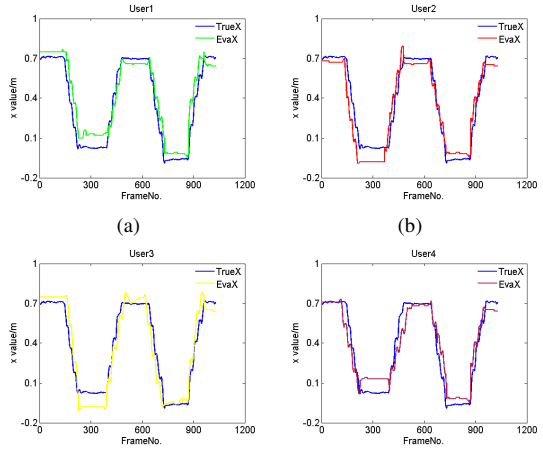


Figure 7. Comparison between trajectories as tracked by a person's skeleton as opposed to computed using shadows for Path 2, where Users 1, 2, 3, and 4 are randomly selected human participants, and their tracking results correspond to (a), (b), (c), (d), respectively.

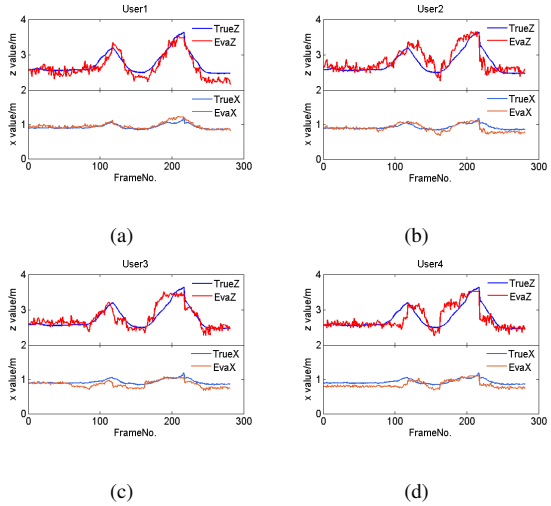


Figure 8. Comparison between trajectories as tracked by a person's skeleton as opposed to computed using shadows for Path 3, where Users 1, 2, 3, and 4 are randomly selected human participants, and their tracking results correspond to (a), (b), (c), (d), respectively.

Similarly, Fig. 7 provides parts of the tracking results for Path 2, recording x values of the occluded person H_b, and Fig. 8 shows the obtained changes in both the x and z directions.

Overall, the results suggest that our algorithm effectively computes the position of people, even when they are completely occluded or occluded for a long time, regardless of whether their position changes along a single axis or along both axes. Moreover, our algorithm is able to compute a person's position effectively regardless of whether they are stationary, in motion, or in either of the two state transitions. This shows that our algorithm is robust in coping with a variety of occlusions.

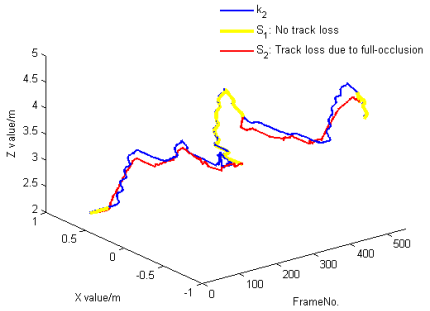


Figure 9. Comparison between trajectories as tracked by a person's skeleton and via their shadow when H_b moves in the tracking area.

In Experiment 2, we compared the tracking results as obtained for the user's skeleton data against the shadow-based tracking results of our algorithm when a person moves freely in the tracking area. As shown in Fig. 9, when a person is in different tracking states, the trajectory obtained by our algorithm is very close to the actual trajectory of that person, which shows the effectiveness of our algorithm.

2) Accuracy: We computed the deviation between the result of our method and the trajectory of the occluded person H_b obtained from k_2 as follows:

$\mathrm{Err}_i^t = \sum_{t=1}^{N_F} \mathrm{err}_i^t / N_F$, $\mathrm{err}_i^t = \sqrt{(p_i^t - q_i^t)^2}$

Here, Err_i^t refers to the error value of the trajectory of H_b at time t, N_F refers to the duration of the entire tracking run, p_i^t and q_i^t respectively refer to the trajectory of H_b at time t computed by our method and captured by Kinect k_2.

Based on this, in order to evaluate the effectiveness of our method, we compute the accuracy as: $\mathrm{acc}_i = e^{-\mathrm{Err}_i^t}$.

Hence, one obtains accuracy values in the range $[0, 1]$ such that the smaller the error value, the higher the accuracy.

First, we computed the tracking deviation of participants along the x and z axes. The tracking deviation along the x axis and the z axis are respectively in the range $[0.14, 0.21]$ and $[0.1, 0.21]$. Subsequently, we measured the tracking accuracy, and its mean value is 0.8, which demonstrates that shadows can indeed be used to track the positions of people when their skeletons are lost.

3) Time Cost: Note that the algorithm is evaluated on a 2.8GHz Intel Core i5 computer. The average time cost

is 67ms for each frame, which is equivalent to about 15 frames per second (fps). This indicates that our proposed method is a feasible choice for real-time applications on modest hardware.

VI. Conclusion

Occlusion has been a persistent problem for multi-person tracking with a single view camera. Although a variety of tracking algorithms have been proposed, they do not effectively and efficiently solve the challenges presented by long-duration and full-body occlusion. In this paper, we explore the novel idea of relying on shadows as additional cues in tracking body movement, rather than merely treating such shadows as noise. Our proposed algorithm fuses shadow and skeletal data to track two persons using just a single Kinect device. Our experiments demonstrate that one can improve the tracking capabilities for people in motion with a single Kinect, without needing to resort to the use of additional sensor devices.

The present study constitutes an initial exploration towards fully resolving long-duration occlusion and full-body occlusion problems. In terms of limitations, the success of this method hinges on an accurate shadow detection, which implies that if the shadow is overly light, it will likely not be captured accurately. Fortunately, in some cases, this problem can be addressed by adjusting the lighting so as to obtain darker shadows. Currently, we only consider the case of a single shadow of a person for a given light source. In settings involving more than one shadow of a person, our method would need to adopt a more elaborate shadow tracking mechanism.

Acknowledgment

This work is supported by the National Key Research and Development Program of China (2018YFC0831003), Shandong Key Research and Development Program (2017CXGC0606), and the National Natural Science Foundation of China (61802232).

References

[1] M. Uson, K. Arthur, M. C. Whitton, R. Bastos, A. Steed, M. Slater, and F. P. B. Jr., "Walking¿walking-in-place¿flying, in virtual environments," in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. New York, NY: ACM, 1999, pp. 359–364.

[2] W. Gai, C. L. Yang, Y. L. Bian, C. Shen, X. X. Meng, L. Wang, J. Liu, M. D. Dong, C. J. Niu, and C. Lin, "Supporting easy physical-to-virtual creation of mobile vr maze games: a new genre," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. New York, NY: ACM, 2017, pp. 5016–5028.

[3] S. Marwecki, M. Brehm, L. Wagner, L. P. Cheng, F. F. Mueller, and P. Basudisch, "Virtualspace - overloading physical space with multiple virtual reality users," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. New York, NY: ACM, 2018, p. 241.

[4] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, no. 4, p. 13, 2006.

[5] M. Nebeling, E. Teunissen, M. Husmann, and M. C. Norrie, "Xdkinect: development framework for cross-device interaction using kinect," in *Proceedings of the 2014 ACM SIGCHI symposium on Engineering interactive computing systems*. New York, NY: ACM, 2014, pp. 65–74.

[6] B. Y. Lee, L. H. Liew, W. S. Cheah, and Y. C. Wang, "Occlusion handling in videos object tracking: A survey," in *Iop Conference Series Earth & Environmental Science*, 2014, pp. 81–93.

[7] S. W. Sun, C. H. Kuo, and P. C. Chang, "People tracking in an environment with multiple depth cameras," *Journal of Visual Communication and Image Representation*, vol. 35, pp. 36–54, 2016.

[8] C. J. Wu, S. Houben, and N. Marquardt, "Eaglesense: Tracking people and devices in interactive spaces using real-time top-view depth-sensing," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. New York, NY: ACM, 2017, pp. 3929–3942.

[9] H. C. Li, P. J. Zhang, S. A. Moubayed, S. N. Patel, and A. P. Sample, "Id-match: A hybrid computer vision and rfid system for recognizing individuals in groups," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. New York, NY: ACM, 2016, pp. 7–7.

[10] J. Wang and Y. Yagi, "Shadow extraction and application in pedestrian detection," *EURASIP Journal on Image and Video Processing*, vol. 1, no. 12, pp. 1–10, 2014.

[11] K. Litomisky, "Consumer rgb-d cameras and their applications," *Rapport technique*, pp. 1–20, 2012.

[12] D. Chrapek, V. Beran, and P. Zemcik, "Depth-based filtration for tracking boost," in *International Conference on Advanced Concepts for Intelligent Vision Systems*. BerlinGerman: Springer, 2015, pp. 217–228.

[13] R. V. Delden, A. Moreno, R. Poppe, D. Reidsma, and D. Heylen, "A thing of beauty: Steering behavior in an interactive playground," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. New York, NY: ACM, 2017, pp. 2462–2472.

[14] K. Meshgi, S. I. Maeda, S. Oba, H. Skibbe, Y. Z. Li, and S. Ishii, "An occlusion-aware particle filter tracker to handle complex and persistent occlusions," *Computer Vision and Image Understanding*, vol. 150, pp. 81–94, 2016.

[15] R. Meng, J. Isenhower, C. Qin, and S. Nelakuditi, "Can smartphone sensors enhance kinect experience," in *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing.* New York, NY: ACM, 2012, pp. 265–266.

[16] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *CVPR*, 2011, pp. 1297–1304.