

Article

Development of a Mobile Robot That Plays Tag with Touch-and-Away Behavior Using a Laser Range Finder

Yoshitaka Kasai ¹, Yutaka Hiroi ^{2,*} , Kenzaburo Miyawaki ³ and Akinori Ito ⁴ 

¹ Graduate School of Robotics and Design, Osaka Institute of Technology, Kita-ku, Osaka 530-8568, Japan; m1m21r08@st.oit.ac.jp

² Faculty of Robotics and Design, Osaka Institute of Technology, Kita-ku, Osaka 530-8568, Japan

³ Faculty of Information Sciences and Technology, Osaka Institute of Technology, Kitayama, Hirakata City, Osaka 573-0196, Japan; kenzaburo.miyawaki@oit.ac.jp

⁴ Graduate School of Engineering, Tohoku University, Aoba-ku, Sendai 980-8579, Japan; aito@spcom.ecei.tohoku.ac.jp

* Correspondence: yutaka.hiroi@oit.ac.jp; Tel.: +81-6-6147-6513

Abstract: The development of robots that play with humans is a challenging topic for robotics. We are developing a robot that plays tag with human players. To realize such a robot, it needs to observe the players and obstacles around it, chase a target player, and touch the player without collision. To achieve this task, we propose two methods. The first one is the player tracking method, by which the robot moves towards a virtual circle surrounding the target player. We used a laser range finder (LRF) as a sensor for player tracking. The second one is a motion control method after approaching the player. Here, the robot moves away from the player by moving towards the opposite side to the player. We conducted a simulation experiment and an experiment using a real robot. Both experiments proved that with the proposed tracking method, the robot properly chased the player and moved away from the player without collision. The contribution of this paper is the development of a robot control method to approach a human and then move away safely.

Keywords: LRF; playing robot; play tag; mobile robot; human–robot interaction



Citation: Kasai, Y.; Hiroi, Y.; Miyawaki, K.; Ito, A. Development of a Mobile Robot That Plays Tag with Touch-and-Away Behavior Using a Laser Range Finder. *Appl. Sci.* **2021**, *11*, 7522. <https://doi.org/10.3390/app11167522>

Academic Editor: Oleg A. Bukin

Received: 29 July 2021

Accepted: 13 August 2021

Published: 17 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The realization of robots that can play physical games with humans is a challenging research topic in the field of human–robot interaction research. Many robots that play physical games with humans have been developed to date, for games such as table tennis [1], playing catch [2,3], handball [4], volleyball [5], and air hockey [6,7]. Such a robot can be applied not only for entertainment purposes [2,7–9] but also in special needs education, such as helping in the development of children with autism [10,11]. For example, we developed a robot that plays “Daruma-san ga Koronda” [12,13], a children’s game similar to “Red light, Green light.” To extend this aspect of a robot that can play physical games with children, we considered a more active game such as play tag.

One of the difficulties in playing such games is detecting and tracking players who actively move in the field. Matsumaru et al. developed a robot system to play tag [14]; however, in their system, the robot’s role was fixed on the fleeing player, so the robot did not need to track the player. In our system, the robot is intended to be both “it” and the other player. Therefore, the robot needs to detect and track other players, chase other players, and touch them when the robot is being “it”, and run away from “it” by avoiding collisions with other players or objects when the robot is not being “it”.

This paper focuses on developing a robot that plays the role of “it”, which chases other players and touches the nearest player. Usually, the goal of mobile robots that move in a crowd is to arrive at the specified location without colliding with other persons and obstacles [15]. On the other hand, the goal of a robot that plays tag with the role of “it” is not

to avoid other players but to approach one of the players and touch them without colliding with the players. To achieve this task, the robot needs to track multiple players around it, focus on one player, approach the player, determine the target point at which the robot touches the target player, and move away from the player. We assumed a two-wheeled robot as a platform for this work. We could use a biped robot or an omnidirectional mobile robot—however, compared with these types of robots, a two-wheeled robot is easier to control than a biped robot, and mentally safer than an omnidirectional mobile robot because a player can predict the direction in which the robot is about to move.

We already evaluated a method for tracking a single player that moved quickly [16]. As a next step, we developed a method to approach a player and pass by the player within a distance of the robot's arm length. This paper proposes a robot control method that realizes the robot's "touch-and-away" behavior.

2. Related Works

There have been a number of methods to program a robot to keep a distance from humans. One application of such a technique is *Follow Me*, a robot's behavior to follow a walking person [17]. In this task, the robot observes the target person's position using sensors, such as laser range finders (LRF) [18,19], cameras [20,21], or RGB-D sensors [22]. Then, the robot moves towards the target person, keeping a distance from the target and avoiding obstacles. As for the choice of sensors, the frame rate of the camera and RGB-D sensor is typically up to 30 (frame/s) [23], which is comparable to the scan rate of an LRF (10 to 40 (frame/s)) [12,24,25]. The advantage of LRF is that the person recognition using a 2D LRF requires less computation than a depth image [26,27], which is more advantageous for our application that needs to track quickly-moving players. For example, Jung et al. [28] used a 2D LRF for high-speed person tracking for a robot that follows a running person. Moreover, the distance range of an LRF [25] is much larger than that of an RGB-D sensor [29], and an LRF is robust against illumination condition change. Those advantages of an LRF are crucial for our application.

A robot should avoid physical contact between the robot and a human for safety in a usual situation [30]. Most of the robots mentioned above that play with humans do not directly contact other players. However, there have been several works that designed robots that contact a human physically. For example, the humanoid robot KASPAR [31] interacts with children by physically touching them. High-fiving and handshaking are other human–robot interaction topics that need physical contact [32–34]. Kamezaki et al. [35] proposed a robot that contacts humans in a crowded situation to clear the robot's path. In this situation, the robot and other passengers move slowly; thus, the effect of physical contact is limited to a psychological one.

On the other hand, both the robot and players are moving quickly in our situation, making the touch behavior more difficult to realize.

3. Realization of "Touch-and-Away" Behavior

3.1. Rule of Play Tag

First, we describe the rule of playing tag. There are many variants of tag games [36], but we employed the simplest rule. There are two kinds of players: one "it" and the other players. The "it" chases the other players, and when "it" touches another player, the touched player becomes the next "it".

We realized a part of the game where the robot is "it" and chases another player and touches them. We supposed that there is only one player other than the robot, who runs away from "it".

3.2. The "Touch-and-Away" Behavior

The robot needs to make physical contact with a player for touching. However, the physical contact between a player and the robot should be executed carefully because

collision could cause injury to the player or damage to the robot. Thus, we designed the “Touch-and-Away” behavior to touch the player with minimum risk of harm.

Figure 1 shows the “Touch-and-Away” behavior. The robot approaches the player and moves towards the virtual circle surrounding the player. The radius of the circle is the same as the robot’s arm length. When the robot moves in the circle, it extends its arm, touches the player, and moves away from the player.

This behavior is realized by setting the tangent point between the virtual circle and the tangent line from the robot. The robot calculates the tangent point in real-time and moves towards the tangent point. After arriving at the point, the robot continues to move to outrun the player.

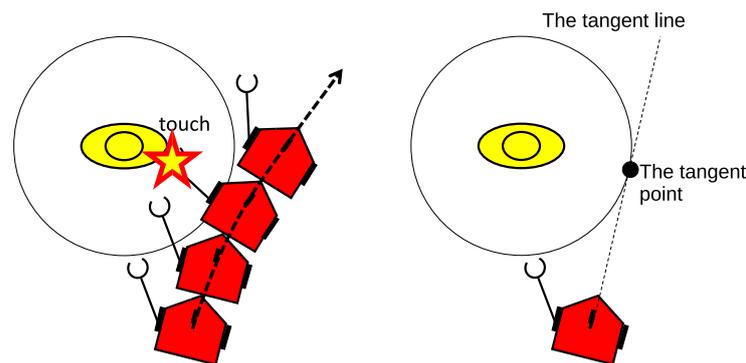


Figure 1. The “Touch-and-Away” behavior.

Here, we consider two tangent points on the virtual circle. We need to consider that the player is also moving. If the player is also moving towards the tangent point, the robot is likely to collide with the player. Thus, as shown in Figure 2, the robot should move towards the tangent point on the opposite side to the player’s moving direction.

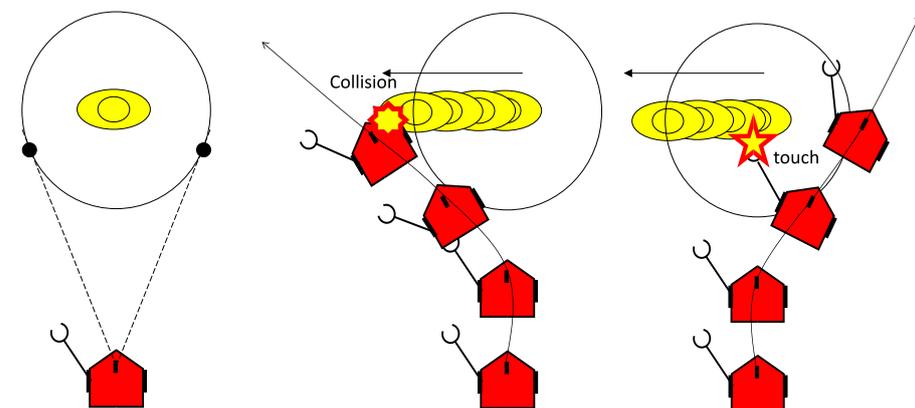


Figure 2. The robot’s movement to avoid collision.

3.3. Player Detection and Tracking

To approach the circle, we need to detect and track the player. We used the person tracking method using an LRF [12,26]. Using this method, we can recognize human bodies, obstacles, and their center coordinates.

Figure 3 shows human detection using an LRF [26]. The LRF scans the field from right to left. We let $D(\theta)$ be the measured distance at the angle step θ from the LRF to the obstacle. Then, let $\Delta D(\theta)$ be the distance difference, such as

$$\Delta D(\theta) = D(\theta) - D(\theta - 1). \tag{1}$$

When observing a smooth object, $|\Delta D(\theta)|$ may have relatively large values at the left and right edge of the object. After detecting the edge points of the object, we can decide whether the object is a human body or not based on the object's size. We calculate the body's center point as the center of the left and right edge points.

As pointed out by Nakamori et al. [12], it is a problem that the player's arms are misrecognized as bodies. To solve the problem, we employed the robust object boundary detection method described in [12].

Once a player is detected, the robot continues to track the player. Figure 4 shows how we track the player [16]. Let $P(t)$ be the position of a player at time t , where the unit of time is the frame of the LRF scan. At the next scan, we observe the players' positions again, then calculate the position difference. When the distance between two observations is small enough, the player at $P(t + 1)$ is regarded as the same player as $P(t)$, meaning that:

$$\|P(t) - P(t + 1)\| < r \tag{2}$$

where the threshold radius $r = 600$ (mm).

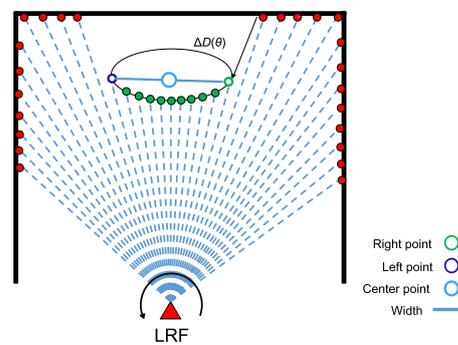


Figure 3. Detection of human body [26].

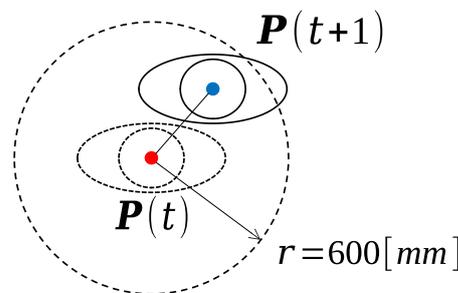


Figure 4. Person tracking [16].

3.4. Initial State and Starting the Tracking

First, the robot is in the initial state, in which the robot waits to detect a player in a pre-defined area (the first detection area). Figure 5 shows the initial state. At the initial state, the robot is not moving, observing the rectangular area in front of the robot. When it observes a player entering the area, the robot starts to chase the player.

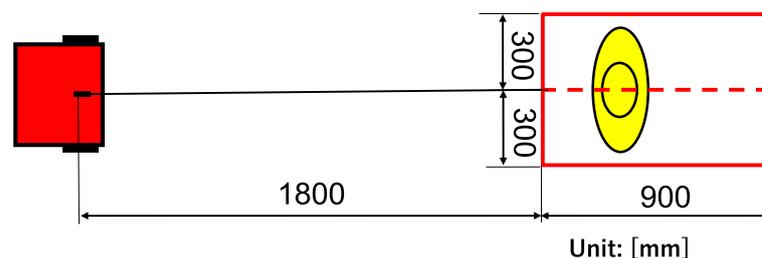


Figure 5. The robot's initial state and the first detection of the player.

3.5. Calculation of Tangent Points

After determining the target player’s center coordinate, we need to calculate the position of the tangent point towards which the robot tries to move, as shown in Figure 1.

The parameters are shown in Figure 6. Let the origin of the coordinate system be the center of the robot. The center point of the target player is $\mathbf{P} = (x_P, y_P)$. We consider a virtual circle around \mathbf{P} with a radius of $r_{\text{touch}} = 1$ (m). When the robot is outside the virtual circle, we can consider two tangent lines from the origin to the circle. Let the tangent points at the left side be $\mathbf{S}_L = (x_L, y_L)$ and that at the right side be $\mathbf{S}_R = (x_R, y_R)$.

Since we measure the object using the LRF, we obtain the angle of the target player θ_P and the distance d_P . Then, the coordinate of \mathbf{P} is calculated as

$$\begin{pmatrix} x_P \\ y_P \end{pmatrix} = d_P \begin{pmatrix} \cos \theta_P \\ \sin \theta_P \end{pmatrix}. \tag{3}$$

The distance to the tangent points d_S is:

$$d_S = \sqrt{d_P^2 - r_{\text{touch}}^2}. \tag{4}$$

Since $\tan \Delta\theta = \frac{r_{\text{touch}}}{d_S}$, we can calculate:

$$\Delta\theta = \tan^{-1} \frac{r_{\text{touch}}}{d_S}. \tag{5}$$

Then, the coordinates of \mathbf{S}_L and \mathbf{S}_R are calculated as follows:

$$\begin{pmatrix} x_L \\ y_L \end{pmatrix} = d_S \begin{pmatrix} \cos(\theta_P + \Delta\theta) \\ \sin(\theta_P + \Delta\theta) \end{pmatrix} \tag{6}$$

$$\begin{pmatrix} x_R \\ y_R \end{pmatrix} = d_S \begin{pmatrix} \cos(\theta_P - \Delta\theta) \\ \sin(\theta_P - \Delta\theta) \end{pmatrix} \tag{7}$$

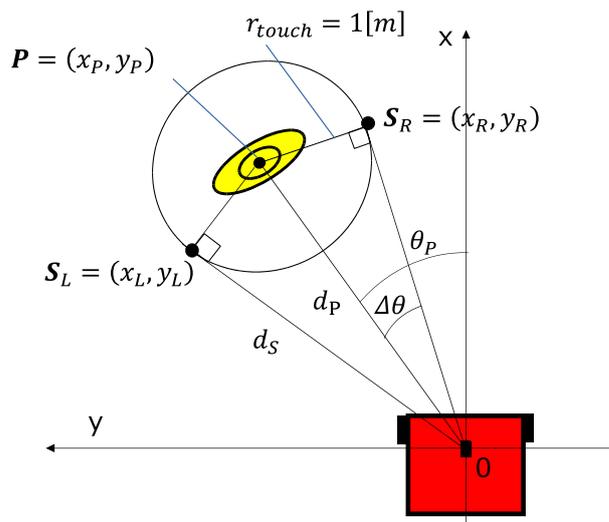


Figure 6. Calculation of the tangent points.

3.6. Selection of the Robot’s Behavior

We then need to choose which tangent point to approach. As shown in Figure 2, the basic idea is that the robot moves towards the opposite side of the player’s motion direction. Let the direction of the player at time t be $\theta_P(t)$. Then, we calculate $\delta(t) = \theta_P(t) - \theta_P(t - 1)$. When $\delta(t) \geq 0$, the player is moving towards left—thus, the robot moves towards \mathbf{S}_R . Otherwise, it moves towards \mathbf{S}_L .

However, this simple behavior is not enough to ensure robust and safe movement because it is likely that the robot’s movement becomes unstable if the selection of the tangent point rapidly changes. In addition, if the robot moves nearer than r_{touch} , the robot cannot calculate the tangent points. Thus, we divided the field around the robot into six areas, as shown in Figure 7. The definitions of the areas are in Table 1.

When the robot chases the player, the player is supposed to be in area A . In this case, the robot chooses the tangent point according to $\delta(t)$. When the player is in area B_L or B_R , it is difficult for the robot to change the tangent point since it needs to rotate rapidly. Thus, the robot moves towards the tangent point that can be targeted with less movement. When the player is in area C , the robot fixes the target tangent point (S_L or S_R) and does not change the target even when the player moves sideways because the robot’s movement becomes unstable if it switches the target tangent point at this distance. Area D is within the length of the robot’s arms. Thus, if the player is in this area, the robot regards the players as touched. Then it moves away from the player. Area E is the emergency zone. If the player enters this area, the robot stops immediately.

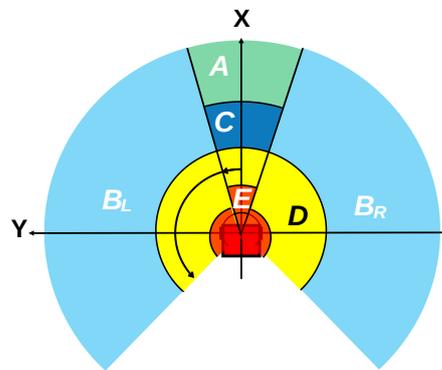


Figure 7. Control areas.

Table 1. Definition of the areas and the robot’s behavior in each area (unit of d_p is (m)).

Area	Definition	Behavior
A	$ \theta_p \leq 15^\circ$ and $d_p > 1.7$	Choose the tangent point according to $\delta(t)$
B_L	$15^\circ < \theta_p < 135^\circ$ and $d_p > 1.2$	Choose S_R
B_R	$-135^\circ < \theta_p < -15^\circ$ and $d_p > 1.2$	Choose S_L
C	$ \theta_p \leq 15^\circ$ and $1.2 < d_p \leq 1.7$	Choose the previously selected tangent point
D	$d_p < 1.2$ and $P \notin E$	Pass through the player and stop
E	$(\theta_p \leq 15^\circ$ and $d_p < 0.85)$ or $d_p < 0.5$	Stop immediately

3.7. Move to the Target

Once the tangent point is determined, the robot moves towards the point. The control algorithm is based on Hiroi et al. [26] with a slight modification. We assume that the robot is two-wheeled. Let V be the robot’s translational velocity, and ω be the angular velocity. Then:

$$\theta_S = \begin{cases} \theta_p + \Delta\theta & \text{Target is } S_L \\ \theta_p - \Delta\theta & \text{Target is } S_R \end{cases} \quad (8)$$

$$V = V_0 + K_v d_S \quad (9)$$

$$\Delta V = K_t \theta_S + K_{tD} \dot{\theta}_S \quad (10)$$

$$\omega = -\frac{2\Delta V}{W_R} \quad (11)$$

Here, V_0 is the minimum speed, which was set to 1.0 (m/s) in the later experiment. W_R is the wheel track (the distance between the center of two wheels) of the robot. $K_v, K_t,$ and K_{tD} are parameters for control, which were set to 0.5, 0.2, and 0.01, respectively, [26].

3.8. Moving Away from the Player and Emergency Stop

When the robot enters area D , the robot decides that it can touch the player and switch to the “moving away” state.

We examined two strategies for moving away from the player. One strategy is “onward.” Here, the robot moves straight for two seconds at velocity V_0 and then stops. During the motion, the robot does not consider the position of the player.

However, after the robot touches a player, the player becomes “it” and starts to chase the robot; the robot needs to rapidly move away from the player. Thus, we designed another strategy, called “parallel,” in which the robot moves towards the other side of the player.

There is an issue with designing this behavior. The robot cannot suddenly change its moving direction because the robot is driven by two wheels. Thus, we designed the behavior of moving away from the player, as shown in Figure 8.

We consider two points, H_L and H_R , at the r_{touch} left and right side of the robot. Then, we observe the angles from the two points to the player, θ_L and θ_R , as follows:

$$\theta_L = \tan^{-1}\left(\frac{x_P}{r_{touch} - y_P}\right) \tag{12}$$

$$\theta_R = \tan^{-1}\left(\frac{x_P}{r_{touch} + y_P}\right) \tag{13}$$

Then, we calculated the robot’s direction of movement. When $\max(\theta_L, \theta_R) > 90^\circ$, it means that the player is on the left or right side of the robot; thus, the robot just moves straight. The same applies when the angles θ_L and θ_R are negative. Otherwise, the robot moves to the other side of the player (angle θ_W) as follows:

$$\theta_W = \begin{cases} \theta_L - 90^\circ & \text{if } \theta_L > \theta_R \\ 90^\circ - \theta_R & \text{if } \theta_L < \theta_R \end{cases} \tag{14}$$

Then, the angle (in degrees) is directly used as the angular velocity (degree/s) of the robot:

$$\omega = \theta_W \tag{15}$$

The translational velocity is set to $\min(V_0, 0.8d_P)$ (m/s).

In addition, if the player enters area E , the robot stops immediately for safety.

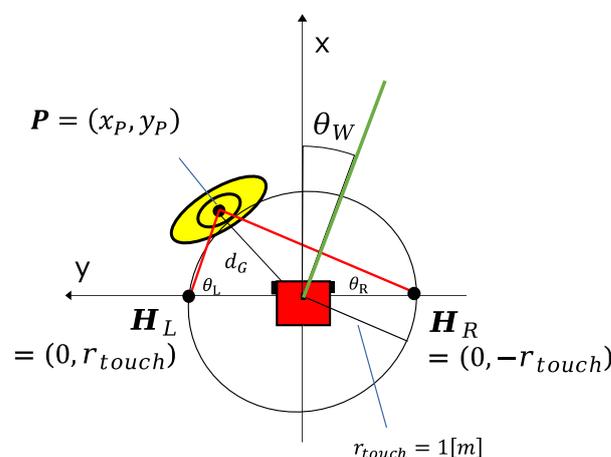


Figure 8. Moving away from the player.

3.9. Total Behavior of the Robot

In the above sections, we explained how the proposed robot works. Here, we summarize the scenario of the play and the behavior of the robot. Figure 9 shows the pseudo-code of the total behavior of the robot. First, the robot is in the initial state and waits for a player to come into the first detection area. Once the robot observes a player, the robot transits to the chasing state and moves towards the tangent point of the target player. If the player enters area D , the robot transits to the “moving away” state, and moves away from the player for two seconds and stops.

In the experiment, we only observed the robot’s trajectory, and the robot did not touch the player because we had not implemented the robot’s arms yet. However, after implementing the arms, the robot would extend an arm of the player’s side and touch the player before transiting to the “moving away” state.

```

(Initial state)
Wait until a player enters into the first detection area
(Chasing state)
Let  $\mathbf{P}(0)$  be the coordinate of the player
Let  $\alpha \in \{L, R\}$  be the choice of left or right tangent point
for  $t = 1, \dots$  do
  Observe  $\theta_P(t)$  and update  $\mathbf{P}(t)$ 
  Calculate  $\mathbf{S}_L(t)$  and  $\mathbf{S}_R(t)$ 
  if  $\mathbf{P}(t) \in E$  then Stop robot
  else if  $\mathbf{P}(t) \in D$  then break
  else if  $\mathbf{P}(t) \in B_L$  then  $\alpha \leftarrow R$ 
  else if  $\mathbf{P}(t) \in B_R$  then  $\alpha \leftarrow L$ 
  else if  $\mathbf{P}(t) \in A$  then
     $\delta(t) \leftarrow \theta_P(t) - \theta_P(t-1)$ 
     $\alpha \leftarrow \begin{cases} R & \text{if } \delta(t) \geq 0 \\ L & \text{otherwise} \end{cases}$ 
  end if
  Move towards  $\mathbf{S}_\alpha(t)$ 
end for
(Moving-away state)
Move away from the player and stop

```

Figure 9. Behavior of the robot.

4. Simulation Experiment

We conducted two experiments, the simulation experiment and the experiment using a real robot. In this section, we explain the details of the simulation experiment.

4.1. Experimental Conditions

We conducted a simulation experiment using the Stage software on ROS. The detail of the platform is shown in Table 2. The supposed robot size is shown in Figure 10.

Table 2. Specification of OS and PC.

PC	Mouse Computer DAIV 5N
CPU	Intel(R) Core(TM) i7-9750H CPU @ 2.60 GHz
Memory	32 GB
OS	Windows 10 with WSL1 Ubuntu 18.04 LTS ROS melodic

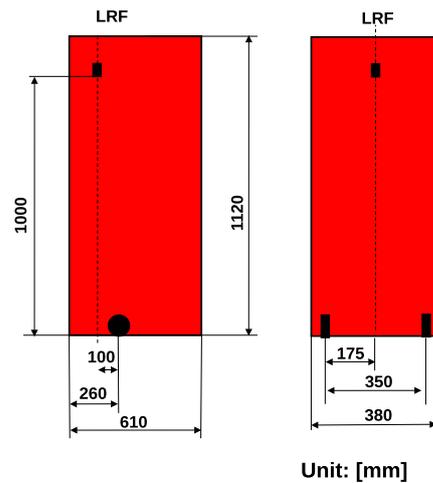


Figure 10. The robot size.

We prepared three paths, as shown in Figure 11. An operator moved the “human” in the Stage manually at 0.74 (m/s) along the prepared path in time to the sound of the metronome. Experiments were conducted three times for combinations of two strategies (straight and parallel) and three trajectories (straight, curved, and zigzag).

4.2. Results

The example of the simulation experiment is shown in Supplemental material Video S1. Figure 12 shows examples of trajectories of the robot and human with the “onward” motion. The dotted lines show the correspondence of a specific time between the robot and the human. In the curved trajectory, we can see that the robot keeps its distance from the player. On the other hand, in the zigzag trajectory, the robot moves closer to the human at the end of the trajectory, which shows that the robot and the human were about to collide.

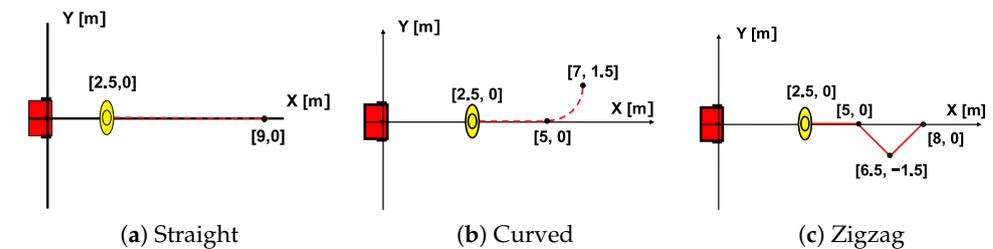


Figure 11. The experimental paths of the simulation experiment.

To visualize all the experimental results, we plotted the trajectories of the robot relative to the player (i.e., $-P(t)$), as shown in Figure 13. In these figures, the player’s position is fixed to the origin. When we employ the “onward” motion, the human and robot become close to each other for the “Curved 1” and “Zigzag 2” trials. On the other hand, we can observe that the robot succeeded in keeping its distance from the player in all trials when the “parallel” motion was employed.

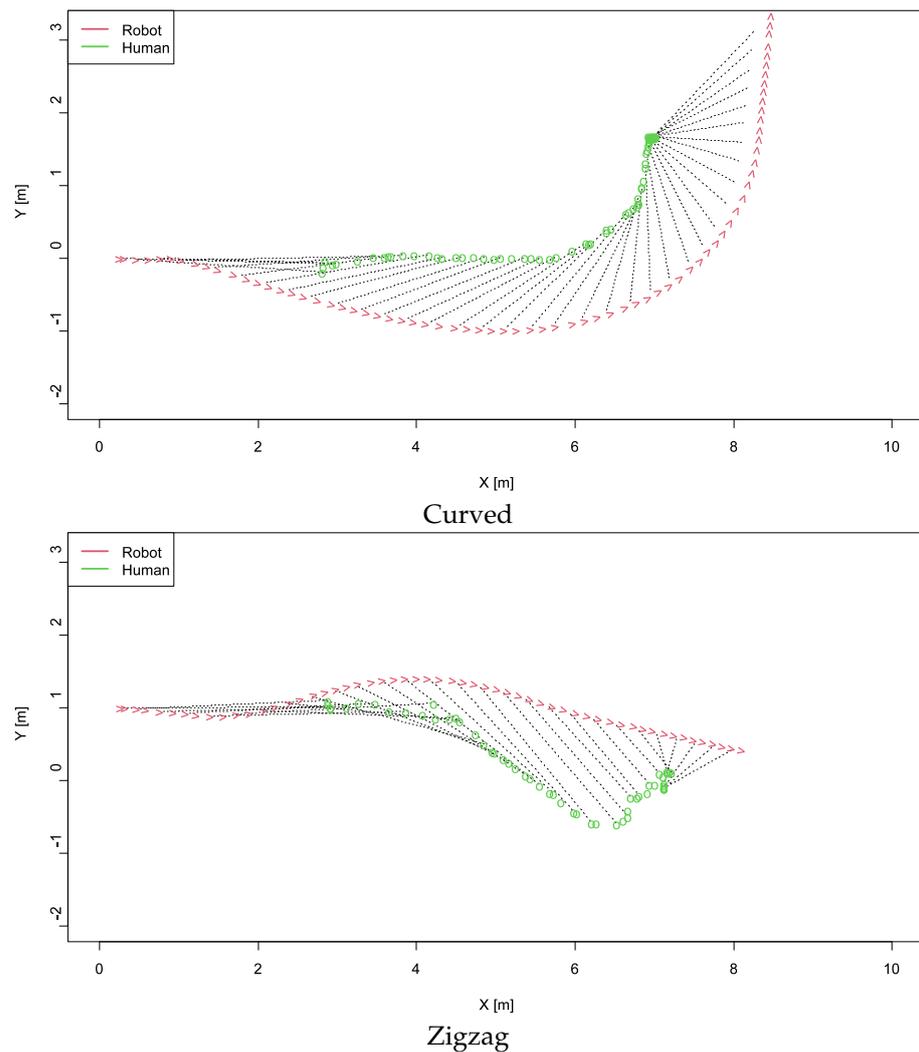


Figure 12. Examples of human and robot trajectories (simulation).

In addition to the previous experiment, we conducted additional simulation experiments. In these experiments, we conducted ten trials for each of all combinations of three trajectories (straight, curved, and zigzag) and two move-away methods (onward and parallel), then investigated the minimum distances between the robot and the player. Figure 14 shows the results. As shown in this figure, the minimum distances in the “parallel” condition are larger than that of “onward” conditions, showing that the proposed move-away method succeeded in keeping distance while approaching the player and moving away. In fact, when we employed the “onward” motion, the robot and the player collided in two out of thirty trials, and the robot stopped in eight trials because the player entered area E. The smallest distance when using the “parallel” motion was 0.70 (m). In this case, the player was still in area D, and the robot succeeded in moving through the side of the player.

We conducted a statistical test to confirm the improvement. We conducted the two-way layout ANOVA, where trajectories and move-away methods are factors ($N = 60$). As a result, trajectories were not significant ($p = 0.191$), while the move-away methods were significant ($p = 8.4 \times 10^{-6}$).

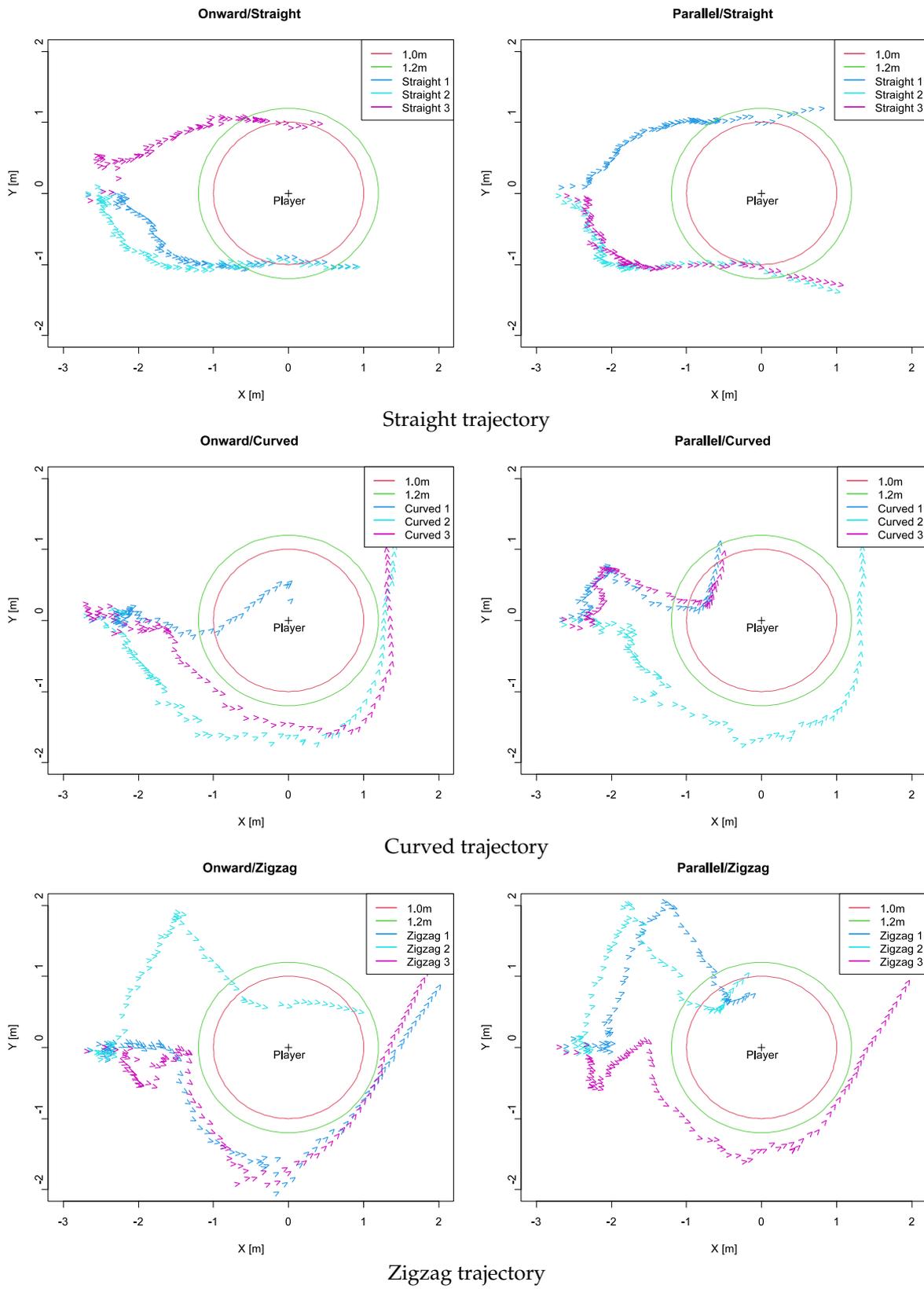


Figure 13. Summary of experimental results (simulation).

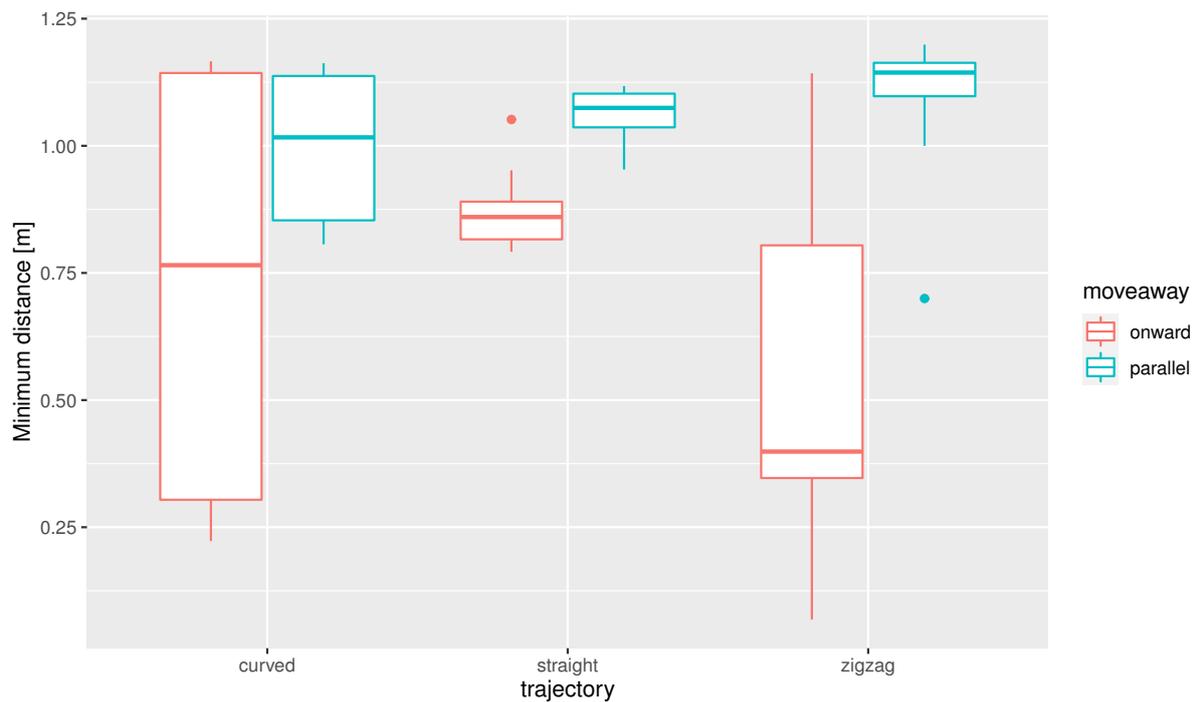


Figure 14. Distributions of minimum distances between the robot and the player.

5. Experiment by a Real Robot

We then conducted an experiment using a real robot, as shown in Figure 15. The mobile robot was based on Pioneer 3DX, an opposed two-wheeled mobile robot. The maximum hardware speed of the robot is 1.2 (m/s) and its maximum angular velocity is 300 (deg/s). The robot's size is $0.38 \times 0.61 \times 1.1$ (WDH:m), and the wheel track is 0.35 (m). The weight is 23 (kg). The LRF is a UTM-30LX made by Hokuyo Electric, which can measure the distance up to 30 (m) in two dimensions within a range of 270 (deg). A laptop PC is installed at the rear of the robot to control the entire robot, which is the same as shown in Table 2. An emergency stop switch is located at the top rear of the robot.

The experiment was conducted in a foyer with carpet flooring. In this experiment, we only used a zigzag path, as shown in Figure 11c. A male participant walked along the path in time to the sound of the metronome so that the walking speed was 0.74 (m/s). In this experiment, the robot's position was estimated using odometry, and the player's position was estimated by combining the robot's odometry and the observation using the LRF.

The example of the experiment can be watched in the supplemental material Video S1. Figure 16 shows the summary of the results. As shown, when we employed the onward motion, the player and the robot collided in the trial Zigzag 2. On the other hand, the collision never happened when we employed the parallel motion. Thus, we think we could show the effectiveness of the "parallel" strategy when moving away from the player.

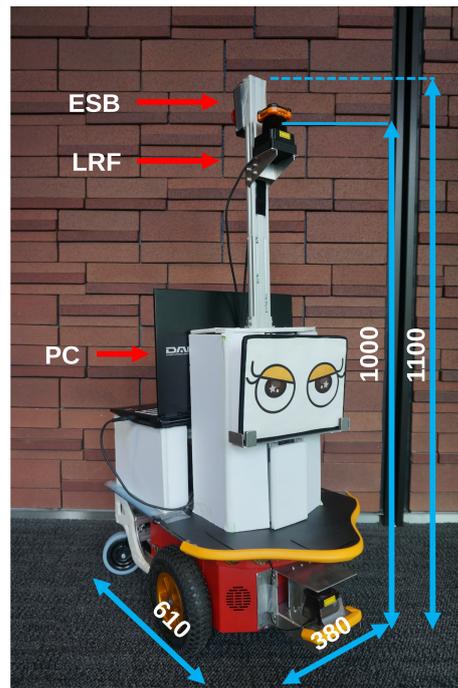


Figure 15. The robot used in the experiment.

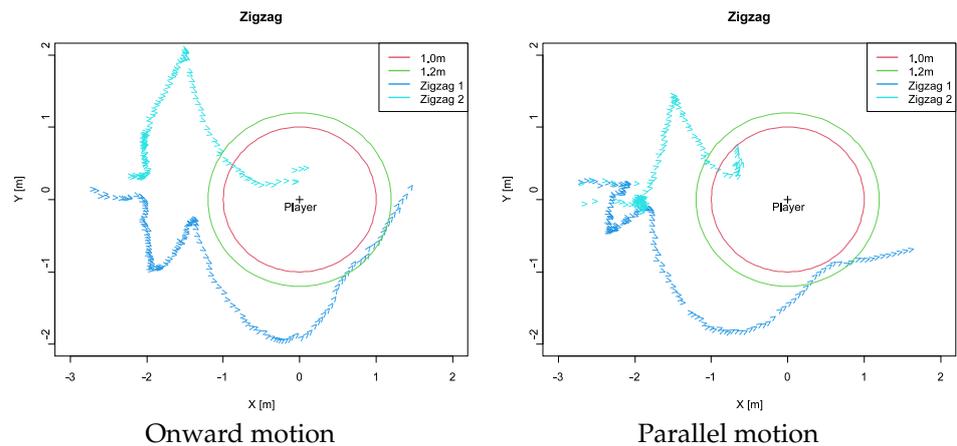


Figure 16. Summary of experimental results (real).

6. Conclusions

We developed a robot that plays “it” from the game “tag”, which approaches a player and passes by the player within a distance of the robot’s arm length. To achieve this system, we employed an LRF to detect and track the player and developed a control method with which the robot moves towards a tangent point of the circle surrounding the player. In addition, we developed a method to control the path after touching the player and moving away from them. The experiments using a simulation environment and a real robot revealed that the proposed method correctly tracked and approached the player. Moreover, the proposed “moving away” method (the “parallel” strategy) worked properly to avoid collision and move away from the player. The contribution of this paper is the development of a robot control method to approach a human and then safely move away.

There is a limitation to our proposed method. It is difficult to avoid collision with a player who intentionally approaches the robot. Thus, we need to develop a method to avoid other players while chasing a player.

There are many issues to realize a robot that actually plays the game. In the current implementation of the robot, we assume that there is only one player other than the robot.

Therefore, the robot needs to detect and track multiple players to play with two or more players. The multiple player tracking method [12] could be applied to this purpose.

In addition, we need to consider safety issues in future works. We also need to install arms on the robot. Since the arm contacts the player, it needs to be equipped with safety functions such as a torque limiter. At the same time, we need to consider that the player could collide with the robot. To ensure safety, we need to cover the robot body with soft materials.

We also need to consider the psychological effect of the robot on other players. For example, it is known that a robot's appearance, moving speed, and distance affect the psychological threat perceived by the human [37]. Thus, we need to design the robot's appearance and motion considering the mental safety of the players.

Supplementary Materials: The following are available online at <https://www.mdpi.com/article/10.3390/app11167522/s1>, Video S1: Comparison of “moving away” methods by a simulation and real experiments.

Author Contributions: Conceptualization, validation, Y.H.; software K.M. and Y.H.; methodology, formal analysis, investigation, Y.K. and Y.H.; resources, Y.H.; data curation, Y.H.; writing—original draft preparation, Y.H.; writing—review and editing, A.I.; visualization, A.I.; supervision, A.I.; project administration, Y.H.; funding acquisition, Y.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by JSPS Kakenhi JP20K04389.

Institutional Review Board Statement: This study was conducted according to the guidelines of the Declaration of Helsinki, and approved by the Life Science Ethics Committee of Osaka Institute of Technology (approval number 2015-12-6, 24 March 2021).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Acknowledgments: Hironobu Wakabayashi contributed to conducting the experiment.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mülling, K.; Kober, J.; Peters, J. A biomimetic approach to robot table tennis. *Adapt. Behav.* **2011**, *19*, 359–376. [[CrossRef](#)]
2. Kober, J.; Glisson, M.; Mistry, M. Playing catch and juggling with a humanoid robot. In Proceedings of the 2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012), Osaka, Japan, 29 November–1 December 2012; pp. 875–881. [[CrossRef](#)]
3. Carter, E.J.; Mistry, M.N.; Carr, G.P.K.; Kelly, B.A.; Hodgins, J.K. Playing catch with robots: Incorporating social gestures into physical interactions. In Proceedings of the 23rd IEEE International Symposium on Robot and Human Interactive Communication, Edinburgh, UK, 25–29 August 2014; pp. 231–236. [[CrossRef](#)]
4. Santiago, C.B.; Sousa, A.; Reis, L.P.; Estriga, M.L. Real Time Colour Based Player Tracking in Indoor Sports. In *Computational Vision and Medical Image Processing: Recent Trends, Computational Methods in Applied Sciences 19*; Tavares, J.M.R.S., Jorge, R.M.N., Eds.; Springer: Berlin/Heidelberg, Germany, 2011. [[CrossRef](#)]
5. Sato, K.; Watanabe, K.; Mizuno, S.; Manabe, M.; Yano, H.; Iwata, H. Development and assessment of a block machine for volleyball attack training. *Adv. Robot.* **2017**, *31*, 1144–1156. [[CrossRef](#)]
6. Ogawa, M.; Shimizu, S.; Kadogawa, T.; Hashizume, T.; Kudoh, S.; Suehiro, T.; Sato, Y.; Ikeuchi, K. Development of Air Hockey Robot improving with the human players. In Proceedings of the IECON 2011–37th Annual Conference of the IEEE Industrial Electronics Society, Melbourne, Australia, 7–10 November 2011; pp. 3364–3369. [[CrossRef](#)]
7. AlAttar, A.; Rouillard, L.; Kormushev, P. Autonomous Air-Hockey Playing Cobot Using Optimal Control and Vision-Based Bayesian Tracking. In *Towards Autonomous Robotic Systems*; Althoefer, K., Konstantinova, J., Zhang, K., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 358–369. [[CrossRef](#)]
8. Cheng, C.Y.; Ke, C.H.; Huang, L.W.; Lin, C.C.; Wang, J.D. Design and implementation of a ball-shooting robot with IR-based embedded vision. In Proceedings of the 2011 8th Asian Control Conference (ASCC), Kaohsiung, Taiwan, 15–18 May 2011; pp. 441–446.
9. Laue, T.; Birbach, O.; Hammer, T.; Frese, U. An Entertainment Robot for Playing Interactive Ball Games. In *RoboCup 2013: Robot World Cup XVII*; Behnke, S., Veloso, M., Visser, A., Xiong, R., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 171–182. [[CrossRef](#)]

10. Ferrari, E.; Robins, B.; Dautenhahn, K. Therapeutic and educational objectives in robot assisted play for children with autism. In Proceedings of the 18th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), New Delhi, India, 14–18 October 2009; pp. 108–114. [\[CrossRef\]](#)
11. Suzuki, R.; Lee, J. Robot-play therapy for improving prosocial behaviours in children with Autism Spectrum Disorders. In Proceedings of the 2016 International Symposium on Micro-NanoMechatronics and Human Science (MHS), Nagoya, Japan, 28–30 November 2016; pp. 1–5. [\[CrossRef\]](#)
12. Nakamori, Y.; Hiroi, Y.; Ito, A. Multiple player detection and tracking method using a laser range finder for a robot that plays with human. *ROBOMECH J.* **2018**, *5*, 25. [\[CrossRef\]](#)
13. Hiroi, Y.; Ito, A. Realization of a robotic system that plays “Darumasan-Ga-Koronda” Game with Humans. *Robotics* **2019**, *8*, 55. [\[CrossRef\]](#)
14. Matsumaru, T.; Horiuchi, Y.; Akai, K.; Ito, Y. Truly-Tender-Tailed Tag-Playing Robot Interface through Friendly Amusing Mobile Function. *J. Robot. Mechatron.* **2010**, *22*, 301–307. [\[CrossRef\]](#)
15. Cheng, J.; Cheng, H.; Meng, M.Q.H.; Zhang, H. Autonomous Navigation by Mobile Robots in Human Environments: A Survey. In Proceedings of the 2018 IEEE International Conference on Robotics and Biomimetics (ROBIO), Kuala Lumpur, Malaysia, 12–15 December 2018; pp. 1981–1986. [\[CrossRef\]](#)
16. Ikemoto, K.; Hiroi, Y.; Ito, A. Evaluation of Person Tracking Methods for Human-Robot Physical Play. In Proceedings of the 2020 IEEE/SICE International Symposium on System Integration (SII), Honolulu, HI, USA, 12–15 January 2020; pp. 416–421. [\[CrossRef\]](#)
17. Islam, M.J.; Hong, J.; Sattar, J. Person-following by autonomous robots: A categorical overview. *Int. J. Robot. Res.* **2019**, *38*, 1581–1618. [\[CrossRef\]](#)
18. Sakai, K.; Hiroi, Y.; Ito, A. Teaching a robot where objects are: Specification of object location using human following and human orientation estimation. In Proceedings of the 2014 World Automation Congress (WAC), Waikoloa, HI, USA, 3–7 August 2014; pp. 490–495. [\[CrossRef\]](#)
19. Kim, J.; Jeong, H.; Lee, D. Single 2D lidar based follow-me of mobile robot on hilly terrains. *J. Mech. Sci. Technol.* **2020**, *34*, 3845–3854. [\[CrossRef\]](#)
20. Weber, T.; Triputen, S.; Danner, M.; Braun, S.; Schreve, K.; Rättsch, M. Follow Me: Real-Time in the Wild Person Tracking Application for Autonomous Robotics. In *RoboCup 2017: Robot World Cup XXI*; Akiyama, H., Obst, O., Sammut, C., Tonidandel, F., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2018; pp. 156–167. [\[CrossRef\]](#)
21. Chen, B.X.; Sahdev, R.; Tsotsos, J.K. Person Following Robot Using Selected Online Ada-Boosting with Stereo Camera. In Proceedings of the 2017 14th Conference on Computer and Robot Vision (CRV), Edmonton, AB, Canada, 17–19 May 2017; pp. 48–55. [\[CrossRef\]](#)
22. Do, M.Q.; Lin, C.H. Embedded human-following mobile-robot with an RGB-D camera. In Proceedings of the 2015 14th IAPR International Conference on Machine Vision Applications (MVA), Tokyo, Japan, 18–22 May 2015; pp. 555–558. [\[CrossRef\]](#)
23. Jing, C.; Potgieter, J.; Noble, F.; Wang, R. A comparison and analysis of RGB-D cameras’ depth performance for robotics application. In Proceedings of the 2017 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), Auckland, New Zealand, 21–23 November 2017; pp. 1–6. [\[CrossRef\]](#)
24. Krejsa, J.; Vechet, S. The Evaluation of Hokuyo URG-04LX-UG01 Laser Range Finder Data. In Proceedings of the 23rd International Conference on Engineering Mechanics, Svatka, Czech Republic, 15–18 May 2017; pp. 522–525.
25. Cooper, M.A.; Raquet, J.F.; Patton, R. Range information characterization of the hokuyo UST-20LX LIDAR. *Photonics* **2018**, *5*, 12. [\[CrossRef\]](#)
26. Hiroi, Y.; Matsunaka, S.; Ito, A. A mobile robot system with semi-autonomous navigation using simple and robust person following behavior. *J. Man Mach. Technol.* **2012**, *1*, 44–62.
27. Dekan, M.; Frantisek, D.; Andrej, B.; Jozef, R.; David, R.; Josip, M. Moving obstacles detection based on laser range finder measurements. *Int. J. Adv. Robot. Syst.* **2018**, *15*, 1–18. [\[CrossRef\]](#)
28. Jung, E.J.; Lee, J.H.; Yi, B.J.; Park, J.; Yuta, S.; Noh, S.T. Development of a Laser-Range-Finder-Based Human Tracking and Control Algorithm for a Marathoner Service Robot. *IEEE/ASME Trans. Mechatron.* **2014**, *19*, 1963–1976. [\[CrossRef\]](#)
29. Corti, A.; Giancola, S.; Mainetti, G.; Sala, R. A metrological characterization of the Kinect V2 time-of-flight camera. *Robot. Auton. Syst.* **2016**, *75*, 584–594. [\[CrossRef\]](#)
30. Kulić, D.; Croft, E.A. Real-time safety for human–robot interaction. *Robot. Auton. Syst.* **2006**, *54*, 1–12. [\[CrossRef\]](#)
31. Wainer, J.; Robins, B.; Amirabdollahian, F.; Dautenhahn, K. Using the Humanoid Robot KASPAR to Autonomously Play Triadic Games and Facilitate Collaborative Play Among Children With Autism. *IEEE Trans. Auton. Ment. Dev.* **2014**, *6*, 183–199. [\[CrossRef\]](#)
32. Okamura, E.; Tanaka, F. Design of a robot that is capable of high fiving with humans. In Proceedings of the 2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Lisbon, Portugal, 28 August–1 September 2017; pp. 704–711.
33. Lee, D.; Ott, C.; Nakamura, Y. Mimetic Communication Model with Compliant Physical Contact in Human–Humanoid Interaction. *Int. J. Robot. Res.* **2010**, *29*, 1684–1704. [\[CrossRef\]](#)

34. Prasad, V.; Stock-Homburg, R.; Peters, J. Advances in Human-Robot Handshaking. In *Social Robotics*; Wagner, A.R., Feil-Seifer, D., Haring, K.S., Rossi, S., Williams, T., He, H., Sam Ge, S., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 478–489. [[CrossRef](#)]
35. Kamezaki, M.; Kobayashi, A.; Yokoyama, Y.; Yanagawa, H.; Shrestha, M.; Sugano, S. A Preliminary Study of Interactive Navigation Framework with Situation-Adaptive Multimodal Inducement: Pass-By Scenario. *Int. J. Soc. Robot.* **2019**, 1–22. [[CrossRef](#)]
36. O'Brien, S. *Old Fashioned Children's Games*; McFarland & Company, Inc.: Jefferson, NC, USA, 1999.
37. Hiroi, Y.; Ito, A. Effect of the size factor on psychological threat of a mobile robot moving toward human. *Kansei Eng. Int.* **2009**, 8, 51–58. [[CrossRef](#)]