

## Research Article

# RoboTable: An Infrastructure for Intuitive Interaction with Mobile Robots in a Mixed-Reality Environment

Haipeng Mi,<sup>1</sup> Aleksander Krzywinski,<sup>2</sup> Tomoki Fujita,<sup>1</sup> and Masanori Sugimoto<sup>1</sup>

<sup>1</sup>Interaction Technology Laboratory, Department of Electrical Engineering and Information Systems, University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, 113-8656 Tokyo, Japan

<sup>2</sup>The Interaction Research Group, Department of Information Science and Media Studies, University of Bergen, Fosswinkelsgate 6, 5020 Bergen, Norway

Correspondence should be addressed to Haipeng Mi, mi@itl.t.u-tokyo.ac.jp

Received 17 January 2011; Accepted 9 May 2011

Academic Editor: Kiyoshi Kiyokawa

Copyright © 2012 Haipeng Mi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents the design, development, and testing of a tabletop interface called RoboTable, which is an infrastructure supporting intuitive interaction with both mobile robots and virtual components in a mixed-reality environment. With a flexible software toolkit and specifically developed robots, the platform enables various modes of interaction with mobile robots. Using this platform, prototype applications are developed for two different application domains: *RoboPong* investigates the efficiency of the RoboTable system in game applications, and *ExploreRobot* explores the possibility of using robots and intuitive interaction to enhance learning.

## 1. Introduction

In the past few years, much development has taken place in the research field of human-computer interaction. Several research approaches in this field, including tabletop interaction, tangible user interfaces (TUIs), augmented reality, and mixed-reality, show great promise for bringing new interaction styles to other related research domains. The research presented in this paper is an attempt to integrate several research approaches to create a mixed reality environment for novel human-robot interaction (HRI).

Because the horizontal surface of a table permits the placement of objects, and its large surface area enables spreading, piling, and organization of the items, digital tabletop user interfaces are becoming increasingly popular for supporting natural and intuitive interaction [1–3].

A TUI is a user interface in which a person interacts with digital information via the physical environment. It gives a physical form to digital information and computation, facilitating the direct manipulation of bits [4]. Such physical interactions are very natural and intuitive for human beings, because they enable two-handed input and can provide spatial and haptic feedback [5].

In this paper, we present RoboTable, an infrastructure that combines tabletop interaction with TUIs to support intuitive interaction with mobile robots. This framework can create a mixed-reality environment in which interaction with real robots and virtual objects can be combined seamlessly. This capability also extends the robot entity into the virtual world, enabling rich and complex HRIs to be supported in various applications.

Based on the RoboTable framework, we have developed two prototype applications for proof-of-concept purposes. *RoboPong* is a tabletop game in which a robot player also participates. This game supports touch input for virtual objects and graspable interaction with robots simultaneously. *ExploreRobot* is designed for educational purposes. It enables users to interact with the robot at a behavioral level, enabling the robot to be reconfigured easily for different tasks.

An important motivation for this research is to explore new possibilities for HRI in a tabletop mixed-reality environment. We believe that with these new interaction styles, the RoboTable system will enable us to build attractive games and playful educational applications, which will give instant

and intuitive feedback to the user, further facilitating the entertainment or learning experience.

## 2. Related Work

In the past decade, several applications for HRI using mobile robots have been discussed in the literature. As an alternative solution to traditional HRI, TUIs have been developed in various forms. TUIs bridge the physical world and the digital world to enable users to manipulate information in a natural way [4]. Recent researchers are trying to develop HRI methods that are more intuitive, such as finger-touch control [6] and tangible-object control [7, 8].

Curlybot is an early example of enabling graspable manipulation for interaction with a mobile robot [9]. The curlybot robot has encoders for its motors by which robot movements corresponding to user manipulations can be recorded in the microcontroller. The recorded robot trajectory can be replayed after user definition.

Kato et al. developed a multitouch interface for controlling multiple robots [6]. This system utilizes ceiling-mounted cameras to track the mobile robots on the ground. A multitouch table enables users to control the robots by manipulating the corresponding image of each robot on the table.

Guo et al. discussed a manipulation method using TUIs [7, 8]. In this system, physical toys are used as indicators of robots on a table. Two multicamera systems are used to track the robot space and the toy-indicator space so that a mapping from the toy-indicator space to the robot space can be created. Users can manipulate toys on the table to move the corresponding remote robot to a desired position. A shortcoming of this system is that users and robots are insulated from each other, preventing the perception of physical feedback.

Other projects have investigated the intuitive control and programming of a mobile robot. Furthermore, other researchers have attempted to give users not only an intuitive control experience but also intuitive feedback during their interaction with mobile robots.

The Augmented Coliseum developed by Kojima et al. creates an environment in which a physical robot can be augmented by projection [10]. The robot has light sensors mounted on its top to enable tracking of a specifically designed light pattern projected onto the table. The robot will track and move to follow the projected image's translation and rotation. This configuration scheme realized robot control and its augmentation using only one projector. However, there was neither direct interaction between users and the robot nor between users and the environment.

IncreTable, developed by Leitner et al. [11] is another project utilizing projection-augmented robots. This system uses the same robots as those in the Augmented Coliseum project. Compared with the Augmented Coliseum project, IncreTable has improved user interaction. In this application, a special pen is used, with which the user can create virtual objects on the table. In addition, the system also enables interaction between physical and virtual objects. This system is an effective attempt at implementing a mixed-reality

environment on a tabletop platform. However, because of the limitations of the robots, graspable manipulation with the robots was not supported. Moreover, "placing" virtual objects with a pen cannot be considered intuitive.

Other recent projects have extended the capabilities of tabletop HRI applications. Robot Arena, developed by Calife et al. [12], is an augmented-reality platform for game development. This system utilizes camera tracking and Bluetooth (BT) communication for the robot to make the robot more flexible. This wireless communication enables users to control the robot remotely, and it can be extended to a multirobot system. Because the camera only tracks specifically designed color markers on the robot, and is not sensitive to hand manipulations, users cannot interact directly with the virtual world.

In reviewing these related projects, we found that no project creates a mixed-reality environment in which users can both directly interact with either real robots or virtual objects using familiar techniques and perceive all objects as cosituated in the same space. This consideration inspired and motivated us to develop the RoboTable system. This work tries to go beyond existing projects to explore a mixed-reality tabletop environment in which novel and intuitive HRI is enabled.

## 3. Robotable Implementation

*3.1. Table.* To achieve the research goal of creating a tabletop that supports both robot tracking and multitouch input, we implemented a combination of two different tracking techniques: frustrated total internal reflection (FTIR) [13] and diffused illumination (DI) [14].

Figure 1 shows the hardware setup for our table. A piece of 10 mm clear acrylic board is applied as the table surface. An infrared (IR) LED stripe is placed at the side of the acrylic board to generate the necessary light for FTIR. On top of this surface, we applied two thin layers of silicone-coated plastic film to form a compliant surface and a separate piece of tracing paper to act as a diffuser. Four IR illuminators fixed on the bottom of the baseplate create a light field inside the table for DI tracking. The combination of the two techniques, where DI helped the tracking of objects and FTIR enhanced finger-touch recognition, enables both tracking goals to be achieved.

A short-throw projector (Benq MP522T) is mounted underneath the surface. A piece of IR-block filter is applied to the projector lens to reduce hot spots introduced by the projector lamp. The projection area is 870 mm × 652 mm. A Firefly MV B/W camera (640 × 480 @ 60 fps) is mounted at the center of the baseplate with a piece of IR band-pass filter. In combination with particular tracking software, this table achieves a tracking resolution of 1.36 mm/pixel for both fingers and objects.

*3.2. Robot.* The definitions of "robot" range widely. For the RoboTable project, the robot is a simple mobile robot. Its microcontroller covers low-level motion control, whereas higher-level strategy and path planning take place in the remote console of the RoboTable system. Therefore, the

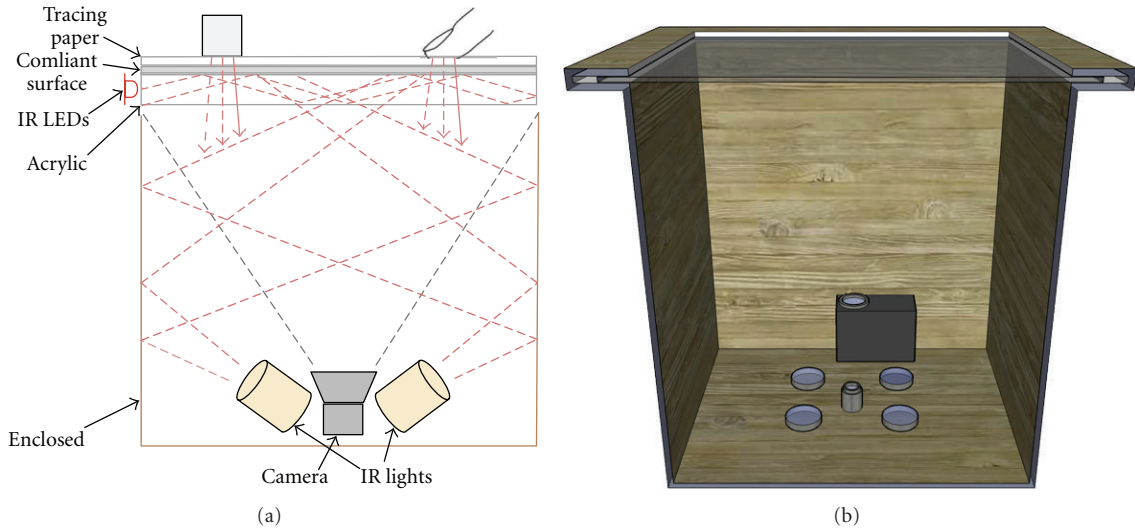


FIGURE 1: Tabletop setup.

robot used in this project can be classified as a nonautonomous mobile robot.

The robot is designed and developed using five design principles.

- (1) It must have a small footprint because of the relatively small size of the table.
- (2) It must be highly maneuverable.
- (3) Its baseplate has to be as close to the table as possible because of the DI tracking.
- (4) It must be able to communicate with the table system.
- (5) It must be aesthetically pleasing and “touchable”.

**3.2.1. Actuation.** The robot is a simple two-wheel design, with each wheel being actuated by an independent DC motor and gearbox unit to enable differential steering. With this arrangement, the robot can move forward or backward and turn with a zero-radius turning circle. The speed of the DC motor is controlled by the microcontroller via power width modulation (PWM).

**3.2.2. Tracking.** Each robot has a ReacTIVision [14] fiducial marker carved into the baseplate, creating a durable, high-contrast symbol that is easily tracked by the camera (Figure 2(a) left). The ReacTIVision fiducial markers are specifically designed and optimized for rapid and high-precision recognition of both position and direction in the tabletop system. The white dots in the fiducial marker have a minimum diameter of 5 mm, and the distance from the marker to the surface is only 0.5 mm. This design ensures stable tracking; the robot can move at high speed without tracking failure. The maximum speed for stable tracking is about 120 mm/sec.

**3.2.3. Communication.** We have chosen BT technology to connect the robot and the server in the RoboTable system,

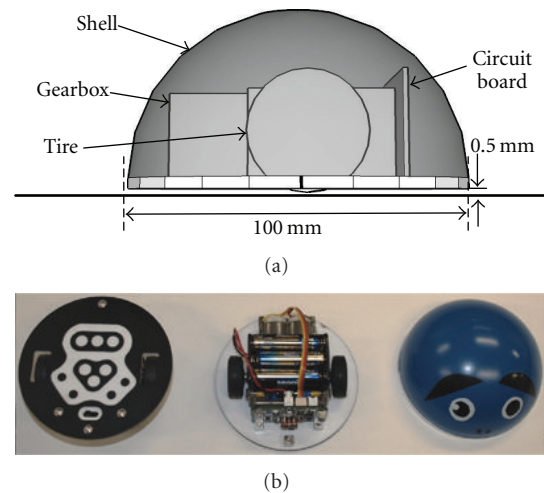


FIGURE 2: Robot.

which gives us a communication channel that has relatively low power consumption, readily available hardware, and software libraries and is easily able to accommodate several units simultaneously. However, most BT devices require manual setup before the connection can be established. For this project, a manual connection setup would significantly reduce the smoothness of the interaction because a robot unit may be introduced into the environment or be removed from the environment by users at any time. To solve this problem, we introduced an automatic connection-setup mechanism.

As shown in Figure 3, when a working robot is put on the surface, the system will recognize the fiducial marker of the robot and will look up the BT address of that particular robot. If the address is found, the system negotiates a connection with that address automatically. The connection will be established seamlessly within seconds of placing the robot on the surface.

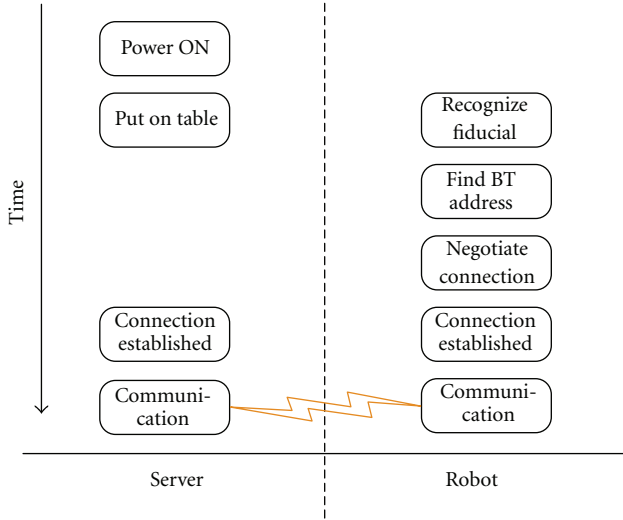


FIGURE 3: Robot connection flow chart.

3.2.4. *Control.* The robot uses a microcontroller (PIC16F886) as the main control unit, which receives commands from, and sends responses to, the server and drives each motor according to the received commands.

In addition, another important task of the microcontroller is to manage the differential steering. Because mechanical tolerances exist for each driving unit (DC motor, gearbox, wheel, and tire), there is always some imbalance between the wheels. To compensate for the deviation angle introduced by the imbalance in differential steering, a gyroscope sensor is also implemented to monitor the angular velocity of the robot in real time.

Figure 4 illustrates the feedback control diagram for the robot using a gyroscope sensor.  $v_y$  and  $\omega_z$  are the desired robot speed and rotation parameters, respectively. A transformation matrix  $A$  transforms  $v_y$  and  $\omega_z$  to left and right motor speeds  $V_L$  and  $V_R$ , which are then sent to the PWM drivers for the two motors. The actual rotation is observed by the gyroscope, and a control algorithm will correct the motion error.

3.2.5. *Integration.* All components of the robot sit on the baseplate, whose diameter is 100 mm (Figure 2(b) center). A hemispherical acrylic shell, which is painted and decorated, covers all components. The compact design fits snugly in a user’s palm (Figure 2(b) right).

3.3. *Software Implementation.* All the RoboTable software components are developed in Java. We used several open-source tools in the project, developing specific drivers and application-programmer interfaces (APIs) for mixed-reality applications based on the RoboTable platform.

3.3.1. *Software Architecture.* The RoboTable API was implemented as a layer above the supporting layer, which comprised several libraries. These toolkits support event-handling and rendering functions, physical simulations and wireless communications. An independent tracking engine

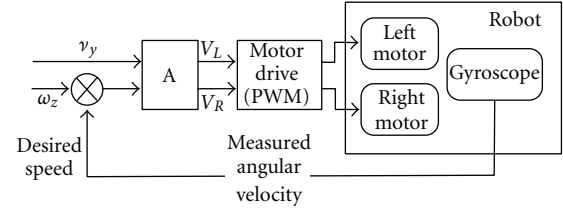


FIGURE 4: Motion control diagram.

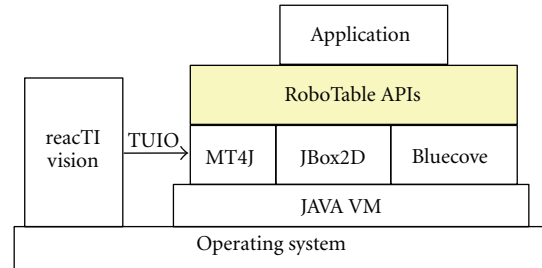


FIGURE 5: Software architecture.

delivers a stream of input events via a TUIO protocol. Figure 5 illustrates the software architecture of the developed infrastructure.

3.3.2. *Tracking.* The main tracking engine in our system was that used in the reactIVision project. The reactIVision engine [14] handles the segmentation, fiducial marker, touch recognition and tracking, and delivers a stream of TUIO-based data to a specific network address and port. Usually, the address is that of a local interface, but configurations where the workload is divided amongst several computers are easily implemented. Therefore, the TUIO stream provides simple add, update and remove events for both cursors (i.e., touches) and objects (i.e., tangibles with fiducial markers).

3.3.3. *Physical Simulation.* To accommodate mixed-reality applications, we must ensure that both the robots and the virtual objects on the screen behave in a way that is intuitive for users, being based on human experience of the real world. Otherwise, users will find it difficult to understand the interaction with the virtual objects on the table, and any advantage in usability leveraged by the familiarity of the user with everyday physical objects is lost. For the flat, horizontal surface of a table, it is most appropriate to simulate the interaction of bodies on a plane. Therefore, we used a 2D physics simulation framework provided by the JBox2D project [15]. (Simulating a third dimension normal to the surface is superfluous, justifying the limitation to two dimensions.) The chosen framework has powerful APIs for defining the properties of physical objects and their interaction with each other when set in motion, enabling us to create rapidly a world in which the objects on the screen collide and exert forces on each other.

3.3.4. *Event Handling and Rendering.* Another open-source framework, MultiTouch for Java (MT4J) [16], is included in

the software toolkit for handling events, managing a canvas and rendering objects on a screen. MT4J is a powerful tool that handles all TUIO events, including both touches and objects. It also provides a rendering engine based on OpenGL technology that supports both 2D and 3D rendering. The MT4J framework has predefined many common gestures such as drag, rotate and zoom. By providing rich APIs, MT4J offers easy extension to different application areas.

**3.3.5. Communications.** We have chosen the Bluecove library [17] for BT communication to handle all the wireless communications between the robots and the RoboTable server.

**3.3.6. The RoboTable API.** For the RoboTable project, the mixed-reality application environment requires seamless management of both on-screen objects and physical robots. For example, an on-screen object will collide with another object if contact is made, whether on-screen or in the real world. In other words, both virtual objects and physical robots should have the same abstraction in the mixed-reality world. The RoboTable API provides such features to upper-layer application development, enabling a specific application with intuitive HRI to be developed easily via the RoboTable infrastructure.

## 4. Interaction

An important feature of the RoboTable system's interaction is that users can interact with both real and virtual objects in an intuitive way. Because the RoboTable system leverages the strong points of multitouch interfaces and TUIs, users can interact seamlessly with all objects involved in RoboTable applications in a mixed-reality environment.

**4.1. Multitouch.** The RoboTable system supports the most popular multitouch and gesture interactions. Users can manipulate on-screen objects simply, in ways they are familiar with. The multitouch interaction feature can be used in many tabletop applications.

**4.2. Robot.** The requirements for interaction with robots depend on the application. Interaction with robots in the RoboTable project falls mainly into two classes, namely, direct interaction and interaction at the behavioral level. These specialized HRIs enable the development of flexible game and learning-assistant applications involving mobile robots.

**4.2.1. Direct Interaction.** The robot used in the RoboTable system is, to some extent, a specific TUI. It has a physical form of input as an ordinary TUI that can be used as a physical token for manipulating virtual information. In addition, the robot also has the ability to transform responses from the virtual world back into the physical world. In RoboTable applications, the robot can be used as a normal TUI, enabling users to manipulate it via everyday actions such as pick up, move or put down. The robot will also

respond to any change in the virtual environment (e.g., collision) by changing its physical properties such as position and movement accordingly.

**4.2.2. Interaction at the Behavioral Level.** The robots also have their own natural properties regarding specific behaviors. A robot with a certain behavior can be used in a variety of applications involving simulation tools, games and learning applications. However, in most of these applications, users expect the robot behavior to be changed easily. In other words, the robot behavior is like an attribution of this special object. Based on this consideration, we argue that if one can interact with the robot at the behavioral level, such as by changing the robot's behavior in a simple way, it can benefit several applications.

The robots used in the RoboTable project are capable of a certain level of autonomous action, and we introduce a behavior arbitration structure for them. The robot has a set of real sensors and actuators, and an additional range of virtual sensors (virtual actuators are also possible, such as those for controlling a virtual robot arm) that can report data about the virtual world in which the robot is cosituated. By being able to attach and detach various virtual sensors easily, the robot becomes reconfigurable. In addition, by having a simple robot behavior-definition feature, the reconfigurable robot can also be redefined (reprogrammed) with a different behavior.

## 5. Application Prototypes

As proof-of-concept tests for the platform, two application prototypes were developed to explore the characteristics of the RoboTable. The first, RoboPong, is a tabletop game that explores the basic interactions in a mixed-reality gaming environment. The second is ExploreRobot, a learning assistant application for school students and programming beginners. This application explores the features of interaction with robots at a behavioral level.

**5.1. RoboPong.** The goal of the development of the RoboPong game prototype was to explore basic interactions in a mixed-reality gaming environment.

The RoboPong game was developed from the classical arcade game of "Pong". The basic version of RoboPong enables two players to play together. Each player uses two fingers to create a paddle on the player's own side of the table, trying to place this paddle where the ball will hit it and bounce back to the opponent's side, hopefully scoring a point in the process.

**Ball.** The ball can be created simply by touching the center ring. Only one ball can be created at a time. Once created, the ball will move in a random direction at a certain speed. If the ball collides with the boundary or a paddle, it will rebound like an actual collision in a real environment.

**Paddle.** A paddle can be created if a player touches the player's own defense area with two fingers. The created

paddle is a straight bar whose endpoints are the player's touch points. (The player can only create a paddle of up to 130 mm in length.) Only one paddle for each player can be created at a time. If a new paddle is created, the old paddle is removed immediately.

*Score.* If the ball reaches the baseline of a player, the player's opponent scores one point. A game will end with the winning player being the first to score five points.

The most important difference between RoboPong and classical Pong is the participation of the robot player. Various behaviors for the robot player were implemented, enabling the robot to influence the game in different ways. Figure 6 illustrates two game modes of RoboPong for which the robot is deployed differently.

*Competitive Mode.* In the competitive mode, a human player plays against a robot player. The robot carries a paddle and moves across the defense area to return the incoming ball. The robot will find the best defending position and automatically move to that position. In this mode, the human player competes with the robot player, aiming to achieve the higher score (Figure 6(a)).

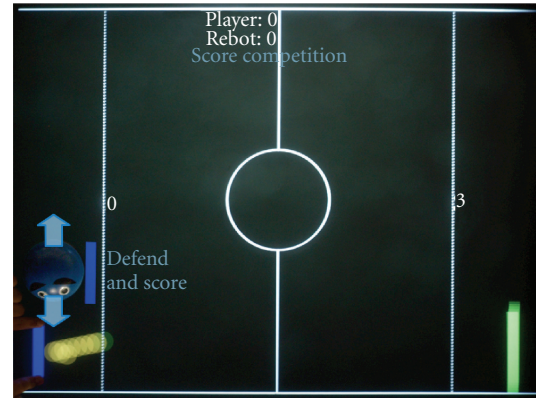
*Cooperative Mode.* In the cooperative mode, two (human) players play with up to two robots. A robot joins one side as a member of that human player's team. The robot tries to chase the ball and return it to the opponent's side. The human player can pick up the robot and place it appropriately whenever necessary. In this mode, the robot effectively cooperates with the human player in playing the game (Figure 6(b)).

5.2. *ExploreRobot.* The goal of the development of the ExploreRobot application was to explore the features of interaction with robots at a behavioral level.

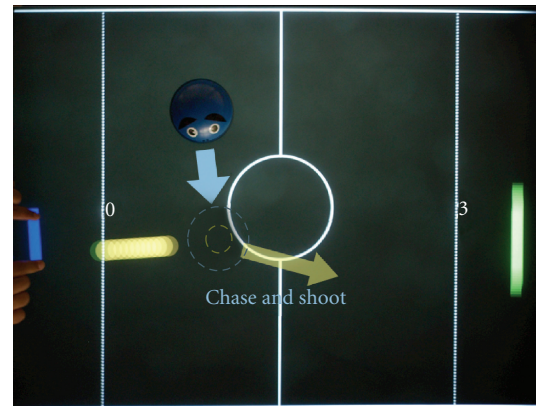
ExploreRobot was a learning assistant application targeting school students and programming beginners. The aim of this application was to help users understand the basic concepts of robot programming. A robot equipped with virtual radar is placed in a virtual maze. An acrylic plate is placed somewhere on the table to indicate the goal of the robot explorer.

The robot has virtual radar that can detect obstacles and the goal. The radar will report the approximate direction of a detected object as left, right, or front. A programming mechanism is introduced that involves radar detection and a simple behavior-definition method for users. As shown in Figure 7(a), the player can move the robot directly on the table to define a motion path. The recorded motion path can be assigned as a default behavior or a special behavior responding to a specific event.

Figure 7(b) illustrates how to assign a motion behavior to a specific event. As the robot is moved to a certain place involving particular radar events, the robot will enter the behavior-defining state and players can then assign the corresponding motion behavior to that specific event using the same method as described above.



(a)



(b)

FIGURE 6: RoboPong game: competitive robot (a) and cooperative robot (b).

After defining the default behavior and response behavior for various cases, the robot-programming phase is completed, with all definitions being stored and organized automatically. When the player puts the robot into executing mode, the robot will start exploring the maze to find the goal, driven by the program created by the player. If the robot reaches the goal successfully, the player wins the game. Alternatively, the player can stop the execution and revise the robot's behavior by simply recalling the stored behaviors and redefining them.

Figure 8 is an example program for the simple maze-explorer robot. While no objects are detected, the default action will be executed repeatedly. If an event occurs that is caused by radar detection, the corresponding action will be executed immediately as an interrupt.

5.3. *Exploratory Evaluation.* As one of the research goal, RoboTable infrastructure is expected to improve users' experience by introducing real robots to create a mixed-reality environment. In order to better understand users' perception playing with real robots in mixed-reality applications compared to traditional graphical applications, we have carried out a preliminary evaluation experiment.

We have modified the RoboPong game as the test-bench which has two different setups. These two setups were almost

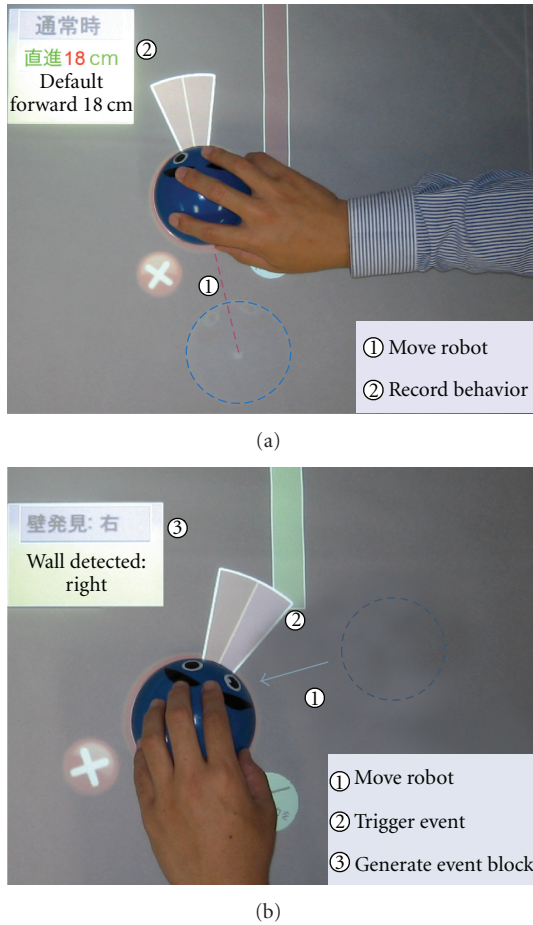


FIGURE 7: ExploreRobot: definition of motion path (a) and definition of event (b).

Default	Goal detected: left	Goal detected: right
Move 15 cm ↑ Rotate 2 deg →	Rotate 2 deg → Move 5 cm ↑	Rotate 2 deg → Move 5 cm ↑

Wall detected: left	Wall detected: right	Wall detected: front
Rotate 5 deg →	Rotate 160 deg →	Move 5 cm ↓ Rotate 30 deg →

FIGURE 8: A simple program for the maze explorer.

the same except the robot deployed in the game. Either a real robot or a virtual robot was deployed for each setup. In order to reduce effects caused by robot appearance and behavior, the virtual robot was presented with a photograph of the real robot and behaved exactly the same as the real robot so that the two robots look quite similar. In other words, one setup was a mixed-reality version of RoboPong game and the other setup was a traditional graphical version.

A number of participants were recruited to experience this game. Each participant was asked to play the game under two different setups. During the experiment, we have taken

video record and memos of special dialogues or actions. After a video analysis and memo summary, we have found some interesting episodes from the evaluation experiment.

(1) *Episode 1.* Participant A was playing with a real robot, but could not defeat against the robot opponent. A heavily patted the robot with anger.

(2) *Episode 2.* Participant B was playing with a real robot, and got scored with the robot's help. B gently patted the robot with happiness.

(3) *Episode 3.* Participant C was playing with a real robot. C had a dialogue with the Experimenter (E):  
(During the game)

C: What's the name of this guy?

E: It has no name yet.

C: Tom? How about Tom?

E: You gave a name to the robot?

C: Yes! It's Tom!

(After the game)

C: May I come to play again? With Tom!

It is a remarkable fact that these episodes were all observed when participants were playing with the real robot. We did not find any special episode in the case of playing with the virtual robot. Because the real robot is cosituated in the same space with players, it is possible for players to express their emotion by direct gestures such as push, pull, or pat. In addition, the dialogue happened in Episode 3 implies a fact that players may treat the real robot as an anthropomorphic co-player.

This result has an implication that mixed-reality games with real robot provide a higher social engagement to players compared to graphical games with virtual robot.

*5.4. Analysis of Results.* In terms of the goals set for this infrastructure, the results were satisfactory and promising, with respect to the techniques implemented in the system. The combination configuration of RoboTable provided stable tracking of both finger touches and objects simultaneously. Using the integrated calibration feature of ReacTIVision, the projection coordinates and the tracking coordinates coincided perfectly, enabling interaction with both virtual objects and the robot to be carried out seamlessly during the application. The robot's flexible maneuverability and rapid response enabled different levels of interaction. In addition, multiple robots worked together simultaneously and successfully. These basic tests of the configuration of the RoboTable infrastructure show great promise for supporting various applications that involve intuitive interaction with mobile robots in a mixed-reality environment.

The RoboPong prototype aimed to test direct interaction with mobile robots in games. As observed in the test, the robot interacts very well with the virtual world. It can

interact not only with virtual objects (e.g., return the incoming ball), but can also respond to virtual environment changes (e.g., move according to the opponent's action). Interactions between virtual objects and the physical robot were successful. The impression that both robot and virtual objects were colocated in the same environment could be perceived.

The ExploreRobot prototype aimed to test interaction at a behavioral level. As observed in the test, a simple reconfigurable robot was achieved successfully. The redefinition method for robot-motion paths was simple and robust. In addition, virtual sensors can work very well with the real robot, and the addition of complexity to the behavior programming was tested successfully.

However, the two applications met problems in some special situations. RoboPong is a fast-paced game. The robot has only a few seconds to move to the desired position for defense against the opponent's action. However, the robot we implemented was not omnidirectional, having only the ability to move in a direction parallel to the wheels. In other words, to move to a desired position, the robot must first rotate to a specific direction and then move forward or backward. This maneuverability problem was sometimes the cause of the robot failing to defend against the incoming ball.

In a later prototype, the robot explorer met the problem of accumulated error. Here, the overall robot behavior can be decomposed into a sequence of motion actions, where each action is a user-defined motion path. However, with every execution of a motion action, a motion error is generated. Although a single error is not significant, repeated motions can generate a significant accumulated error, which can result in uncertain behavior by the robot. For example, if the robot is placed initially at exactly the same position with the same direction, and the program is executed several times, the final position of the robot will be variable. Although this problem was not serious in the ExploreRobot application, it could be a major problem for other applications with more stringent requirements.

Although some problems for robot control and maneuverability still exist, the RoboTable system has performed well overall. As one of the most important research issue, the result of exploratory evaluation implies that real robot and mixed-reality environment provide higher social engagement to users. The feature of integrating interactions with both real and virtual objects shows a great possibility of RoboTable infrastructure to develop several kinds of attractive and interactive applications.

## 6. Discussion

*6.1. Interactions.* Exploring interactions is the main objective of this work. In the RoboTable prototype applications, we uncovered some interesting issues concerning interaction.

Multitouch interaction extends traditional HRI to a mixed-reality environment. In aiming to manipulate virtual objects attached to the robot directly, the robot entity spreads to the virtual world. This enables the user to interact with the robot in new ways, such as moving the robot a little and then adjusting the parameters of the virtual components, or even

performing these actions in an arbitrary order, which enables the perception of manipulating a unified entity, rather than separate objects.

Another interesting issue is related to interaction at a behavioral level. RoboTable applications have the capability for users to reconfigure a robot directly as well as via its behavior. The simple interaction by which the user grasps and moves a robot to define a motion path as a segment of a behavior sequence ensures a direct and intuitive perception of a user wanting to reconfigure the robot. Even though this kind of "programming" is not precise, enabling only approximate definitions, we would argue that this intuitive programming method is suited to games and educational contexts because of its intuitiveness and tangibility.

Although the RoboTable infrastructure is only a combination of existing interaction techniques, we found new possibilities for HRI when we connected the real world and the virtual world, and enabled seamless interaction with all objects involved in the mixed-reality environment. We believe that the RoboTable system can offer users a new experience of HRI.

*6.2. Application Domain.* The RoboTable system creates a mixed-reality environment in which users can interact with mobile robots in an intuitive way. One main consideration is the range of application domains that can benefit from this kind of system. In general, the tabletop mixed-reality environment has three main strong points: a collaborative workspace, interaction with both physical and virtual objects, and reconfigurable robots.

The RoboTable system utilizes a tabletop interface, which offers the advantages of tabletop interaction such as face-to-face communication and a collaborative workspace. These advantages can benefit applications involving games, discussions and collaborative work and learning.

In addition, the mixed-reality environment created by the RoboTable system enables rich interaction with both virtual and real objects. By successfully implementing interactions between the real world and virtual world, all objects involved in RoboTable applications are perceived as colocated in the same environment. Interactions across the mixed-reality environment can be performed seamlessly.

Last but not least, the virtual components extend the capability of the robot by converting it into a reconfigurable robot. This feature widens the range of application domains for the RoboTable system.

With these advantages for the RoboTable system, we believe that the platform offers great benefits in the creation of applications for games, entertainment, and collaborative learning. Other possible application domains would include museum exhibitions and tactical simulation.

*6.2.1. Games and Entertainment.* The RoboTable platform is suited to the creation of a mixed-reality environment for face-to-face gaming. The physical robot as a game protagonist will enhance the attraction of the game, and the visual effects and virtual components greatly extend the boundaries of the physical gaming environment.



*6.2.2. Collaborative Learning.* As reported in some research articles [18], the TUI has benefits for some learning activities because of its physicality. In our experimental configuration of ExploreRobot, the intuitive interaction that helped users create a direct mapping between the program and the behavior of the robot showed great benefit for this specific learning activity. Moreover, collaborative learning can be implemented easily on the tabletop platform, further enhancing learning activities in some situations. In addition to programming-learning applications, the storytelling application domain [19] would be another possibility for collaborative learning.

*6.3. Robot Maneuverability.* Maneuverability is one of the most important characteristics of the mobile robot. The requirements for robot maneuverability depend on the application. Based on the experimental results for our prototype applications, two main issues related to robot maneuverability should be considered carefully.

The first issue concerns the robot architecture. In the RoboTable system, we implemented the robot using a simple two-wheel architecture, which only has maneuverability in two directions. Although a differential steering mechanism is implemented to realize zero-radius rotations, the additional rotation time will limit the robot's capability whenever the robot is expected to move rapidly to a desired position. A possible solution to this problem is to make the robot omnidirectional. However, this method will result in higher cost and greater complexity. Therefore, we consider that an omnidirectional robot should be used only if the application has stringent requirements for the timing of robot movements.

The second issue concerns robot control. Because we implemented the robot using only simple DC motors, the robot lacked a precise control capability. To improve the precision of the robot motion, there are two possible options. The first involves upgrading the motor systems to servomotors with a control unit, enabling the robot to move with a higher level of precision. The second option is to create a global observation and calibration unit from the software side. Because the RoboTable system can track the robot across the whole table-surface area, a prediction mechanism could be used to reduce the accumulated error of motion and calibrate the robot position dynamically. For most RoboTable applications, the latter solution is considered better because of its easy implementation and good performance. It is expected that it will be included in future developments of the system.

## 7. Conclusions

This work has two main achievements. First, we have created a complete framework that can create a mixed-reality environment involving mobile robots and that enables seamless interaction with both physical and virtual objects. The second contribution involves new possibilities for interaction styles.

To accommodate the goal of seamless interaction with different kinds of objects, we have developed the RoboTable

framework by incorporating two approaches. First, we have successfully integrated existing interaction technologies to create an environment in which both robot tracking and finger-touch input could be used simultaneously and with a high degree of responsiveness. In this way, hand manipulation, such as touching and gesturing, for virtual objects is enabled, extending HRI from the real world to the virtual world. The extended robot entity, which now includes physical and virtual components, enables rich and complex interaction between the user and the robot. In addition, the physical simulation enables users to utilize their knowledge of the real world for interaction in the mixed-reality environment. The RoboTable system ensures that users perceive a unified world containing cosituated objects that interact in ways that fit the users' experience and common sense.

The framework provides three main interaction styles for users in the mixed-reality environment. The multitouch feature allows users to interact directly with virtual objects via touching and common gestures. In addition, interaction with the robot bridges the real world and the virtual world, with the robot responding not only to the user's direct physical interaction but also to interactions from the virtual world. Lastly, the robot has the capability of reconfiguration according to user requirements, where the behavior of the robot can be defined easily, using simple gestures.

For proof-of-concept purposes, we have developed two prototype applications. RoboPong is a simple game based on a classical arcade game. A human player uses touch to create a paddle, which can return an incoming ball in competition with an opponent. The robot is deployed in the game as a player with different behaviors. In competition mode, the robot can automatically move a virtual paddle to compete with a human player. In cooperation mode, the robot becomes a team member alongside one of the human players. The second application, ExploreRobot, is a learning-assistant application for school students and programming beginners. It provides simple sensors and programming mechanisms that enable users to redefine a robot's behavior, aiming to reach the goal in a virtual maze.

Robots have played an important role in education for many years, and their presence has proved stimulating for students [20]. The framework presented in this paper enables the creation of an intuitive and powerful interface that enables users to program robots easily for different tasks. Via programming and playful interaction with robots on a tabletop, students can learn concepts and principles in different disciplines such as mathematics and physics at different educational levels while also learning to think creatively, reason systematically, and work collaboratively.

The prototype applications demonstrate the possibility of developing interactive applications to support student learning, using the RoboTable framework. In addition, the advantages of the RoboTable system lead to prospects for several other application domains.

*7.1. Future Work.* The first task we will focus on is the improvement of global robot control. As discussed above, the current system has an accumulated error problem, which will affect applications that have stringent requirements for

precise robot control. Compared with the hardware solution of upgrading the motors and control units inside the robot, the global tracking and calibration method is a more cost-effective solution. We will implement a predictor unit for the robot and a global control module that can correct the robot motion dynamically to enable its arrival at the required position.

Another promising direction for the RoboTable system is the investigation of its possible use in other serious application domains such as transportation simulation and urban planning. Because the robot provides a TUI in addition to physical forms of feedback, it is capable of representing a simulation target in some applications. The intuitive manipulation and physical representation of a simulation process might have benefits in some specific areas.

We will also consider another interesting direction of development for the RoboTable system, namely, implementing remote interaction between two tabletop systems. Because the robots can be treated as both input and output devices, they are capable of connecting two or more remote environments in which users can manipulate the robot as well as perceive physical outputs according to other users' manipulations. Remote interaction has great benefits for games and collaborative workspace applications.

Conducting serious user studies is a further objective, enabling the evaluation of different applications for different target users. Particularly for educational purposes, we hope to analyze the efficiency of the applications as teaching tools, and to collect user feedback that will help us improve the RoboTable system and applications further.

## References

- [1] E. Hornecker, "I don't understand it either, but it is cool"—visitor interactions with a multi-touch table in a museum," in *Proceedings of the IEEE International Workshop on Horizontal Interactive Human Computer System (IEEE TABLETOP '08)*, pp. 113–120, Amsterdam, The Netherlands, October 2008.
- [2] A. Mahmud, O. Mubin, J. R. Octavia et al., "Affective tabletop game: a new gaming experience for children," in *Proceedings of the 2nd IEEE International Workshop on Horizontal Interactive Human-Computer Systems (IEEE TABLETOP '07)*, pp. 44–51, Newport, RI, USA, October 2007.
- [3] M. R. Morris, A. M. Piper, A. Cassanego, A. Huang, A. Paepcke, and T. Winograd, "Mediating group dynamics through tabletop interface design," *IEEE Computer Graphics and Applications*, vol. 26, no. 5, pp. 65–73, 2006.
- [4] H. Ishii, "Tangible bits: beyond pixels," in *Proceedings of the 2nd International Conference on Tangible and Embedded Interaction (ACM TEI '08)*, pp. 15–25, Bonn, Germany, February 2008.
- [5] D. Rosenfeld, M. Zawadzki, J. Sudol, and K. Perlin, "Physical objects as bidirectional user interface elements," *IEEE Computer Graphics and Applications*, vol. 24, no. 1, pp. 44–49, 2004.
- [6] J. Kato, D. Sakamoto, M. Inami, and T. Igarashi, "Multi-touch interface for controlling multiple mobile robots," in *Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems (ACM CHI '09)*, pp. 3443–3448, Boston, Mass, USA, April 2009.
- [7] C. Guo, J. E. Young, and E. Sharlin, "Touch and toys: new techniques for interaction with a remote group of robots," in *Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems (ACM CHI '09)*, pp. 491–500, Boston, Mass, USA, April 2009.
- [8] C. Guo and E. Sharlin, "Utilizing physical objects and metaphors for human robot interaction," in *Proceedings of the Artificial Intelligence and the Simulation of Behaviour Convention (AISB '08)*, AISB Press, Aberdeen, UK, April 2008.
- [9] P. Frei, V. Su, B. Mikhak, and H. Ishii, "Curlybot: designing a new class of computational toys," in *Proceedings of the Conference on Human Factors in Computing Systems 'The Future is Here' (ACM CHI '20)*, pp. 129–136, The Hague, The Netherlands, April 2000.
- [10] M. Kojima, M. Sugimoto, A. Nakamura, M. Tomita, H. N II, and M. Inami, "Augmented coliseum: an augmented game environment with small vehicles," in *Proceedings of the 1st IEEE International Workshop on Horizontal Interactive Human-Computer Systems (IEEE TABLETOP '06)*, pp. 3–8, Adelaide, Australia, January 2006.
- [11] J. Leitner, M. Haller, K. Yun, W. Woo, M. Sugimoto, and M. Inami, "IncreTable, a mixed reality tabletop game experience," in *Proceedings of the International Conference on Advances in Computer Entertainment Technology (ACM ACE '08)*, pp. 9–16, Yokohama, Japan, December 2008.
- [12] D. Calife, J. L. Bernardes, and R. Tori, "Robot arena: an augmented reality platform for game development," *Computers in Entertainment*, vol. 7, no. 1, Article ID 11, 2009.
- [13] J. Y. Han, "Low-cost multi-touch sensing through frustrated total internal reflection," in *Proceedings of the 18th Annual Symposium on User Interface Software and Technology (ACM UIST '05)*, pp. 115–118, Seattle, Wash, USA, 2005.
- [14] M. Kaltenbrunner and R. Bencina, "ReacTIVision: a computer-vision framework for table-based tangible interaction," in *Proceedings of the 1st ACM International Conference on Tangible and Embedded Interaction (ACM TEI '07)*, pp. 69–74, Baton Rouge, La, USA, 2007.
- [15] <http://www.jbox2d.org>.
- [16] [http://www.mt4j.org/mediawiki/index.php/Main\\_Page](http://www.mt4j.org/mediawiki/index.php/Main_Page).
- [17] <http://blucove.org>.
- [18] P. Marshall, "Do tangible interfaces enhance learning," in *Proceedings of the First International Conference on Tangible and Embedded Interaction (ACM TEI '07)*, pp. 163–170, Baton Rouge, La, USA, 2007.
- [19] H. Mi, A. Krzywinski, and M. Sugimoto, "RoboStory: a tabletop mixed reality framework for children's role play storytelling," in *Proceedings of the 1st International Workshop on Interactive Storytelling for Children (ACM IDC '10)*, Association for Computing Machinery, Barcelona, Spain, June 2010.
- [20] T. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots," *Robotics and Autonomous Systems*, vol. 42, no. 3-4, pp. 143–166, 2003.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

