PingPongPlus: Design of an Athletic-Tangible Interface for Computer-Supported Cooperative Play

Hiroshi Ishii, Craig Wisneski, Julian Orbanes, Ben Chun, and Joe Paradiso*
Tangible Media Group
*Physics and Media Group
MIT Media Laboratory
20 Ames St., Cambridge, MA 02139, U.S.A.
{ishii, wiz, joules, benchun, joep}@media.mit.edu

ABSTRACT
This paper introduces a novel interface for digitally-augmented cooperative play. We present the concept of the "athletic-tangible interface," a new class of interaction which uses tangible objects and full-body motion in physical spaces with digital augmentation. We detail the implementation of PingPongPlus, a "reactive ping-pong table", which features a novel sound-based ball tracking technology. The game is augmented and transformed with dynamic graphics and sound, determined by the position of impact, and the rhythm and style of play. A variety of different modes of play and initial experiences with PingPongPlus are also described.

Keywords
tangible interface, enhanced reality, augmented reality, interactive surface, athletic interaction, kinesthetic interaction, computer-supported cooperative play.

INTRODUCTION
When an expert plays ping-pong, a well-used paddle becomes *transparent,* and allows a player to concentrate on the task – playing ping-pong. The good fit of grasp is vital to making a paddle transparent [10]. To achieve a "good fit," a user has to choose a paddle of the right size, right form, and right weight for his or her hand and style of play. To achieve a "better fit," the user has to *customize* the tool by scraping the edge of the paddle with a knife and sandpaper. The "best tit" is, however, achieved by using a paddle over a long period of time.

Figure 1 shows the author's paddle and the traces of the body left on it [4]. After twenty years of use, the grip of the paddle has captured the traces of his right hand (marks of the thumb and index finger in front and marks of the middle finger on back). The right-bottom picture shows the dent made on the back of the paddle by a strong grasp with the tip of the middle finger.

The ping-pong paddle, which can co-evolve with a user by changing its physical form and being united with the human hand, suggests an important direction for HCI – transparent physical extensions of our body and mind into both physical and digital worlds.

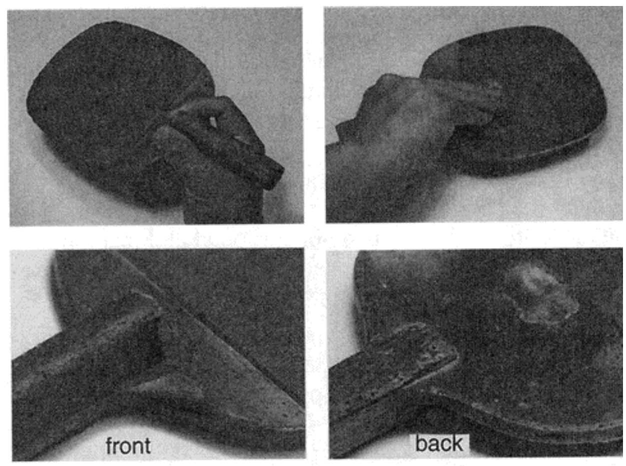


Figure 1 Traces of grasping hand left on the well-used ping-pong paddle

Moreover, the full-body motion, speed, and rhythm of a ping-pong game make the interaction very engaging and entertaining. Kinesthesia is one of the keys of what makes ping-pong enjoyable.

Modem graphical user interface (GUI) technologies provide very limited, generic physical forms (e.g. mouse, keyboard, and monitor) and allow limited physical motions (only clicking and typing). Thus, the GUI is difficult to adapt to human bodies and to take advantage of kinesthesia.

Goals of the PingPongPlus Project
We have designed PingPongPlus on top of the classic game of ping-pong [21]. Its goals are:

1. to demonstrate an instance of *an athletic-tangible interface,* developed on top of existing skills and protocols of familiar competitive/cooperative play.
2. to develop an underlying technology for an "interactive architectural surface" which can track the activities happening on the surface.
3. to study the impact of digital augmentation on the competitive/cooperative nature of play.

COMPUTER-SUPPORTED COOPERATIVE PLAY
Sport is an activity governed by a set of rules or customs that involves skill and physical exertion. It is often

undertaken competitively against opponents, while it is played cooperatively within a team. By playing sports, people can not only learn athletic skills and develop physical strength, but they can also develop social communication and coordination skills.

Computer support is gradually embedding itself in, and transforming the way we play sports and games. Traditional computer games are now extending their reach out from the sole domain of the keyboard, mouse, joystick, and twitch-controllers [8]. Children can create and teach robots, interact with their dolls, and experience complex skiing and motorcycle simulators. With the rise of networks, in the home and in the arcade, play can occur cooperatively more than ever before.

We may give a generic label "CSCP" (Computer-Supported Cooperative Play) to uses of computer technology that enhance physical exertion, social interaction, and entertainment in sport and play. Our research interests in CSCP encompass both the *augmentation* and *transformation* of sports and games. We expect that CSCP research will guide us to design a new form of HCI that we call the "athletic-tangible interface." This refers to a new class of interaction that uses tangible objects and full-body motion in physical spaces with digital augmentation. We believe that a person's physical prowess and sense of kinesthesia can be leveraged to strengthen the quality of a collaborative play experience in physical/digital domain.

Our athletic-tangible interface research looks at augmentation and transformation of *real* sports and games, rather than partial simulations of them. Arcade simulation games, while moving in very promising physically-based directions, can only imitate portions of real experience. Immersive virtual environments, such as VIDEOPLACE [7] and ALIVE [9], allow users to use unencumbered full body motion. Although these systems are engaging, they are designed to provide only a simulated experience and the interaction is limited to simple gesturing.

We see the opportunity to explore the design of new games and play experiences where physical interaction is of central importance. We have begun to explore this by adding digital layers of graphics and sound on top of existing skills and protocols of classic games.

DESIGN OF PINGPONGPLUS

We have chosen ping-pong as a target sport of our athletic-tangible interface research, and have designed a computer-augmented version called "PingPongPlus." PingPongPlus is a digitally enhanced ping-pong game using a "reactive table" that incorporates sensing, sound, and projection technologies. The table displays graphics patterns as a game is played, and the rhythm and style of play drives accompanying sound.

Figure 2 shows a snapshot of PingPongPlus in the water ripples mode, and Figure 3 shows the system architecture of PingPongPlus. In the water ripples mode, a bouncing ball leaves images and the sound of rippling water.

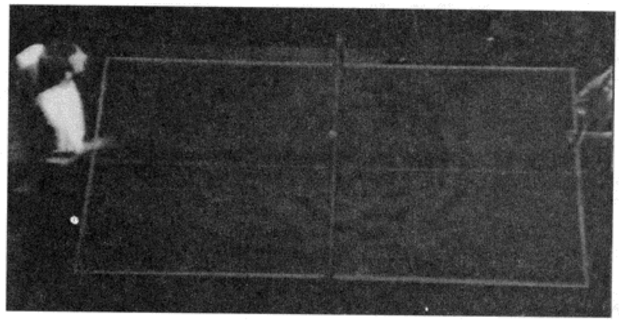


Figure 2 PingPongPlus in water ripples mode

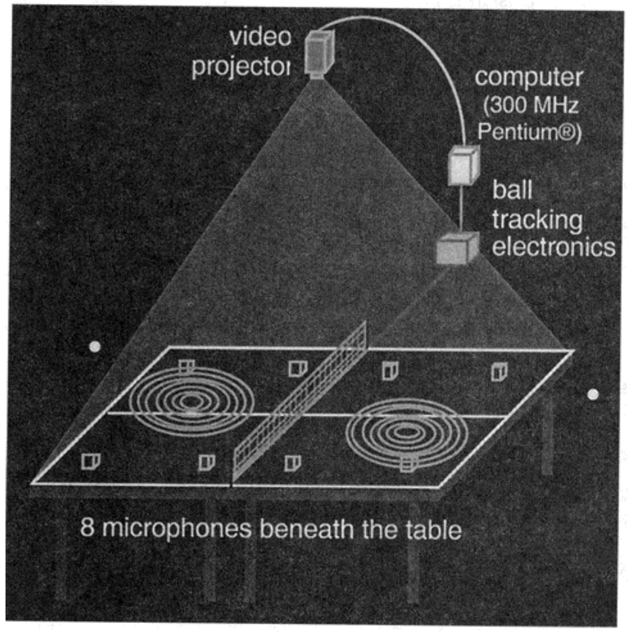


Figure 3 System architecture of PingPongPlus

A series of "tangible interfaces" have been created which give physical form to online digital information [3, 5, 16]. In these projects, users can directly *grasp* and *manipulate* digital information by coupling graspable objects and online digital information. We have also demonstrated the concept of an *interactive surface* that can sense and track the graspable objects on it and project digital shadows [15, 17].

In PingPongPlus, we are extending this notion of tangible interfaces by integrating the kinesthesia of athletic interaction. With PingPongPlus, users experience dynamic and athletic interactions using the full-body in motion, a paddle in hand, a flying ball, and a reactive table. PingPongPlus requires sophisticated realtime coordination among the body, paddle, ball, and digital effects of graphics and sound.

IMPELMENTATION TECHNOLOGY

The PingPongPlus system consists of ball-tracking hardware, software algorithms for ball-hit location detection, and a graphics projection system. The technology behind creating "interactive surfaces" is of utmost importance to this system, and is further described here.

Ball Tracking System

We have developed a sound-based ball tracking system. When a ball hits, the sound travels through the table at roughly twice its speed in air. Eight microphones mounted on the underside of the table pick up the sound. When a microphone detects a hit, a time value is assigned to that microphone, and it is sent to a computer through a custom-made electronic circuit. The time values are evaluated on a 300 MHz PC by an algorithm that determines the location of the hit. The algorithm we have developed can pinpoint the ball's position within a few inches in a matter of milliseconds, which is good enough for our application.

Figure 4 shows a schematic diagram of a ball hit. The four microphones (ml, m2, m3, and m4) on the underside of each table top pick up the ball hit sound at different times (tl, t2, t3, and t4). Given this information, there are a few different algorithms that can determine the original location of a ball hit. We implemented two different methods along with the necessary hardware.

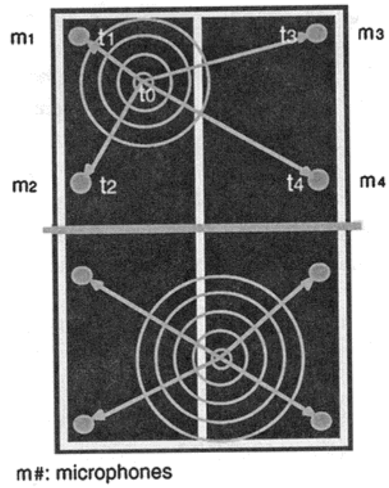


Figure 4 Ball tracking algorithms

Hardware Implementation

A custom-built hardware circuit connects the ping-pong table to the computer via the serial port (Fig. 5). This circuit only outputs a microphone number (ml, m2, m3, or m4) along with its associated time value (tl, t2, t3, t4). Software running on a host PC does the rest of the work.

The hardware is realized by doing peak thresholding on signals from the microphones. The microphones themselves are electret pickups, which output a voltage around 0.25 volts for a typical hit. First, their signal is passed through an op-amp which increases their gain by a factor of 20, such that there is a signal between 0 and 5 volts, quiescently at 2.5 volts. This signal is sent through two comparators and an or-gate that compare the signal's absolute value (relative to the 2.5 volt center) against a threshold voltage (both high and low). The comparator/or-gate pair returns true to a PIC chip if there is an impact. This PIC chip is running at 20 MHz, and polls its input about 100,000 times a second. If there is a hit, the PIC chip assigns a time value to that microphone input, and sends this information out a serial connection. Fig. 6 shows a photo and a block diagram of the electronic circuit.

Including the microphones, the total cost for this hardware is nominal. A future improvement to this system is to implement peak detection and to match the various incoming waveforms (as opposed to simple thresholding) to more accurately determine the time differences, and perhaps enable us to extract impact characteristics. It is expected that this will produce significant gains in accuracy and reliability.

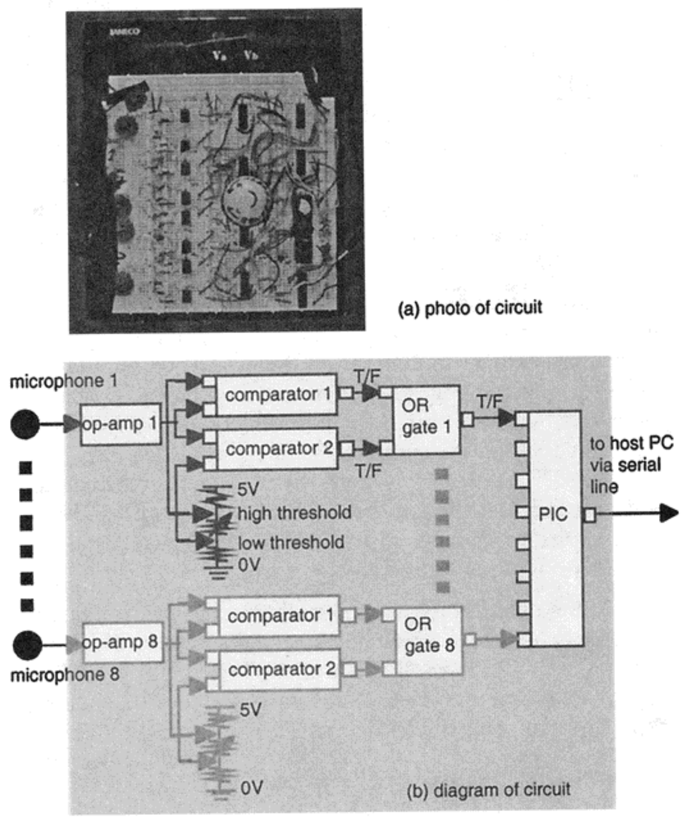


Figure 5 Ball tracking electronic circuit

Software Algorithms for Location Detection

Given the hit timing information from the hardware, the software can calculate a ball-hit coordinate in a number of different ways.

The first algorithm we implemented is by a direct inspection of the time differences. If the ball lands directly at a midpoint between two microphones, the time differences between the two points will be the same (tl = t2, for instance), and you can infer that the ball landed on a straight line equidistant from those points. If the ball lands closer to one microphone than another, it can be inferred that the ball landed on a hyperbolic shaped curve between the two points.

The time differences between many microphones can be compared, which results in a system of hyperbolas that intersect at different points. This system of equations can be solved to yield these points. By throwing out intersection points that do not occur on the table, and looking at which points have the largest number of intersecting hyperbolas, a very good approximation of the original hit location can be made. This method is efficient, as no calibration whatsoever is required.

This algorithm, however, has drawbacks. First, it requires solving equations for two variables that go out to infinity. This is computationally expensive. Second, this method is sometimes not accurate. Sometimes there might be multiple intersection points, or possibly, no points at all. In these cases, a best guess must be made based on the data, and the system is fairly prone to error.

While hyperbolic locator algorithms have been further refined in the literature (e.g. [2]), we have developed a much simpler algorithm to calculate the ball hit position that is better suited to this application. This method is based on a comparison of the time-difference data to a set of model parameters that are acquired by a linear least-squares fit of calibration/training data. The model for this method is:

$$AX = Y$$

Where:

Y = the ball landing coordinate vector (x,y)

X = sensor data vector (time differences information)

A = model parameters (matrix obtained by linear least-squares fit)

When an impact occurs, the sensor values, X, are multiplied by the model parameters, A, which returns a ball landing coordinate, Y. Matrix A, the model parameters, is set through a calibration routine. This calibration routine, however, only needs to be performed once in the life of the table, unless the microphone placement is changed.

Training data is acquired by dropping a ping-pong ball on certain known spots on the table a number of times. In our case, we chose to calibrate the table with 18 distinct points; the A matrix was then calculated through a least-squares fit to this data [14].

Although it involves a linear approximation to hyperbolic relation, this method works well here for a variety of reasons. Since it is a simple matrix multiplication, it is very fast. Also, the linear least-squares fit error metric in the creation of the model parameters makes the system somewhat adaptive to imperfect tables. Performance does not degrade as drastically around edges as compared to the first algorithm (This is important, as most hard surfaces have different kinds of edge effects.). Using this method makes the sensing system more portable to other kinds of tables and surfaces. Although the linear approximation introduces some distortion, it provided accuracy on the order of a few inches, while being fast enough to appear perceptually instant.

At the early stage of this PingPongPlus project, we evaluated the use of computer vision technology for ball tracking, but we concluded that it was slower, more complicated, and computationally more expensive than sound-based tracking technology. Computer-vision, however, is attractive because the system can capture not only the ball but also the motion of players with paddles. Computer vision could be a reasonable and more interesting alternative technology when the computation speed becomes fast enough and the price drops.

Creation and Projection of Graphics

The graphics are created in accordance with the ball tracking information. They are written in Visual C++ with a custom-made graphics package. In the following APPLICATION section, we describe several patterns of graphics we have developed.

A projector suspended 20 ft. above the table displays the graphics on to its surface. We used a Mitsubishi LCD projector LVP-G1A for the experiments, but the brightness of this projector was not enough. To see the graphics on the surface of ping-pong table, we had to darken the room, making it difficult for human eyes to track the ball. We expect the next generation of brighter video projection technology and, potentially, "e-ink" technology [6] to resolve this problem.

In order to make the graphics less "pixelated," we out-focused the video projector slightly so that the image became softer and naturally merged into a wooden table surface.

APPLICATIONS

We have designed and implemented over a dozen different application modes on the PingPongPlus table. The goal of our application design was to explore the design space characterized by the two axes: 1) augmentation vs. transformation, and 2) competition vs. collaboration.

We had two phases of application development.

Phase 1: 1997 Summer-Fall

Artistic and collaborative play modes: water ripples, thunderstorm, spots, painting, comets, etc.

Phase 2: 1998 Spring-Summer

An enhanced artistic mode (school of fish) and a new competitive game mode (Pac-Man®).

PingPongPlus was demonstrated from October 1997 until July 1998 at the MIT Media Lab to the faculty, students, and sponsors. In July 1998, PingPongPlus was exhibited at SIGGRAPH '98 Enhanced Realities in Orlando [20].

Although we have not yet conducted formal experiments to evaluate those applications, informal feedback from casual users was reflected in the iterative design of these applications. In this section, we illustrate and discuss seven examples of those applications.

Water Ripples mode

The *Water Ripple* mode is a simple, causal augmentation. When a ball hits the table, an image of a water ripple flows out from the spot the ball landed (Fig. 2). Players found this to be one of the less distracting applications from the normal game of ping-pong, allowing them to concentrate

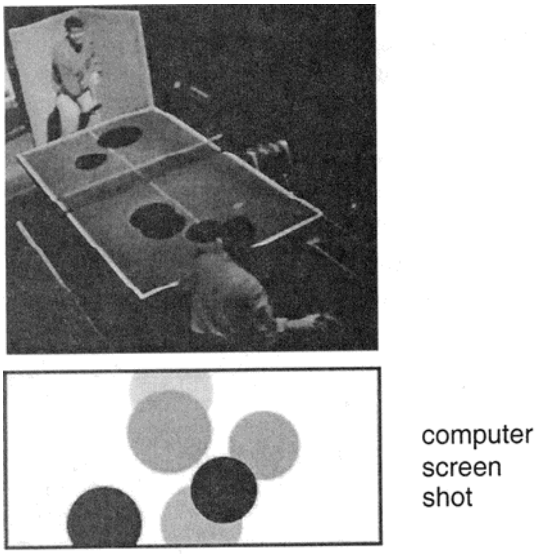


Figure 6 Spots mode

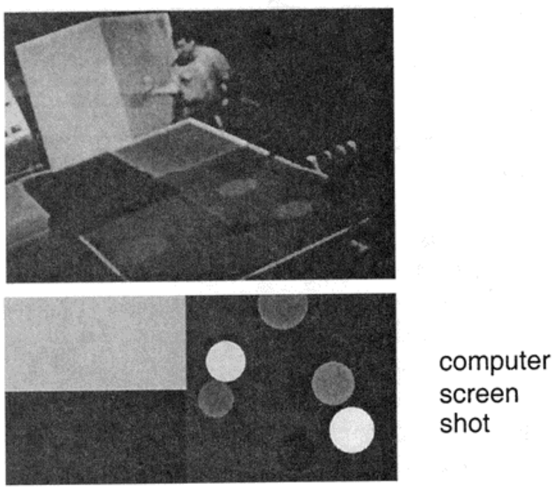


Figure 7 Painting mode

on the game at hand, yet augmenting the game in a non-traditional sense. People often played with curiosity, rather than competitiveness, trying to examine what kinds of interference wave patterns they could create on the table. Once a child even climbed up on the table and created water ripple with his foot. When a player makes an error by hitting a ball into the net, it is usually disappointing. However, in water ripples mode, it turns into an opportunity to enjoy a sequence of small water ripples making a beautiful pattern of interference and sound.

Spots mode
The *Spots* mode was originally intended to be played in a completely dark room where the only light source is the bright white projection on the table. In this mode, a large black spot appears wherever the ball hits, effectively "taking light away" from the other person's side of the table (Fig. 6). The removal of light can be used strategically, changing the strategies employed in a game.

Painting mode
The *Painting* mode was derived from spots mode. The *Painting* mode was designed to explore the collaborative aspects of PingPongPlus. *In Painting* mode, one side of the table is a blank canvas, and the other is a black and white "ink" pallet. When a ball hits the black area of the "ink," it leaves a black spot on the canvas ((Fig. 7). Accordingly, when it hits the white "ink," it leaves a white spot on the canvas side of the table. Through collaboration on color choices and placements by expert players, an interactive artwork can be made on the canvas. There is a shift here away from normal ping-pong to a collaborative painting game. The object is not to win a game, but to create an image. This suggests digital augmentation can not only change the nature of the game, but also change the object of the game itself,

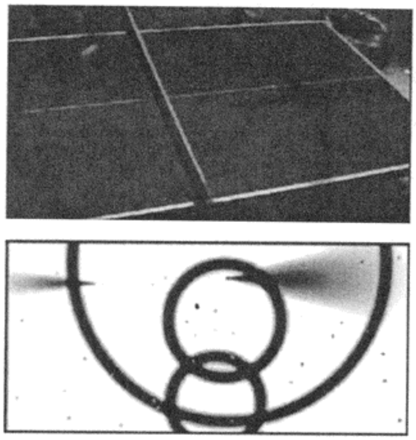
In practice, however, the precise control of the ball is too difficult for most users. They could not succeed in painting what they intended. Rather than coordinating the ball movement to create images, they simply enjoy painting visual effects. This motivated us to design the *Comets* mode.

Comets mode
In the *Comets* mode, when a ball hits the table, it "releases a comet" which travels up towards the net (Fig 8). When the comet hits the net, it creates a sound that is mapped to the place on the table from which the comet originated. Experts using this mode could potentially use PingPongPlus to create/play music. We are planning to further explore the integration of playing music and ping-pong by using the speed of play as a metronome that controls the tempo of music being played.

Thunderstorm mode
The *Thunderstorm* mode was designed to encourage collaboration by continuing to rally rather than scoring points. By keeping the ball in play, rallying back and forth, players "build up a thunderstorm." At the beginning of a point, calm, flowing waves appear on the table (Fig. 9 top). As the rally duration increases, a sound of a heartbeat in the background gets faster, wind whips around the sound space, and waves speed up. If the ball is kept in play for a long time, lightning bolts shoot from one side of the table to the other, connecting the ball's last two locations (Fig. 9 bottom).

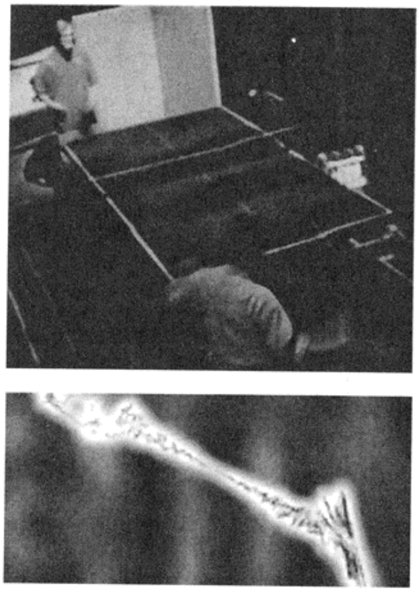
in this mode, we found that the way people play is changed due to the additional effects of the thunderstorm. When the wind picks up and the heartbeat gets faster, players tend to be more nervous and hit the ball faster and harder. Players try to rally until they see the lightning. The lightning at the end of a long rally encourages players to cooperate.

Pac-Man® mode
In *Pac-Man®* mode, the Namco classic video game is reinterpreted for the PingPongPlus environment. (Fig. 10). The ball serves the same functions as Pac-Man® did in the video game; it is controllable by the players and. results in the scoring of points, which is the goal of the game. Points are awarded for accuracy in hitting the various fruit targets, and points are taken away for hitting the ghosts.



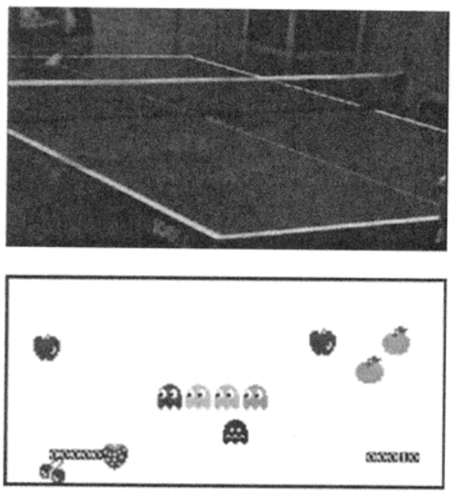
computer screen shot

Figure 8 Comets mode



computer screen shot

Figure 9 Thunderstorm mode



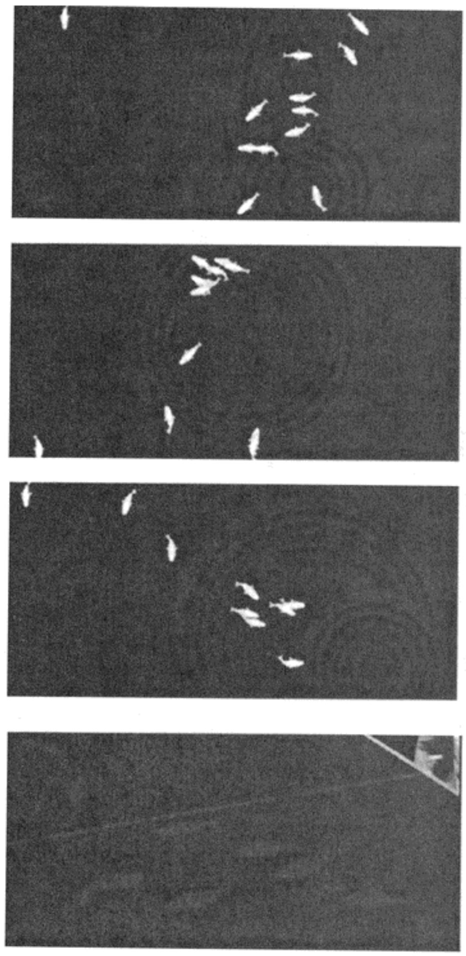
computer screen shot

Figure 10 Pac-Man® mode

We designed this ***Pat-Man*** mode to see if we could transform ping-pong into a very different engaging, competitive game. However, it was found that it was difficult to divide visual attention between tracking a ball and watching the Pat-Man screen on a table. This indicates that highly detailed display elements on the table do not work as well as simple visual patterns. The best results seem to occur when a simple visual pattern is combined with some level of complexity to keep the game interesting. The School of Fish mode is a good example of this concept.

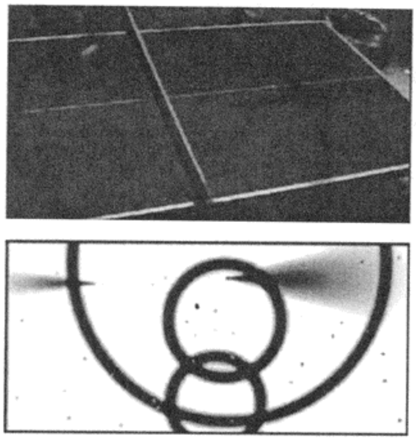
School of Fish mode

The school of fish with water ripples seemed to be the most popular mode for players. In this mode, a school of fish swims on the table (Fig. 11) following a behavior pattern set forth from the algorithms that Craig Reynolds developed for flock behavior [12]. (Top three pictures are the images from a computer screen, and the bottom picture is a picture from the installation.) The ball causes a splash and a ripple in the "water" where it hits, scaring the fish. In time, the fish, following their individual behavior models, school back together. The simplicity of the visual display, combined with the complexity of the emergent activity from a behavior model made this mode continually compelling, even after days of play.



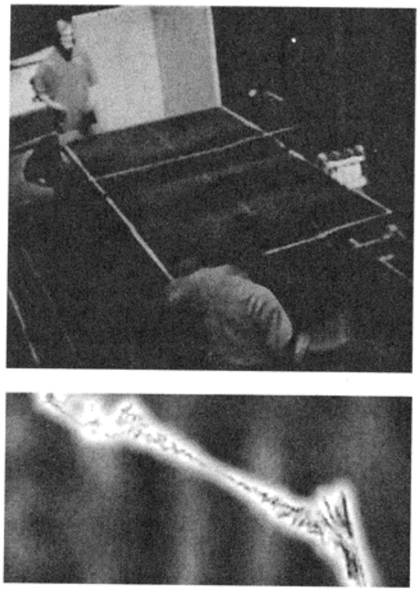
computer screen shots

Figure 11 School of fish and water ripples mode



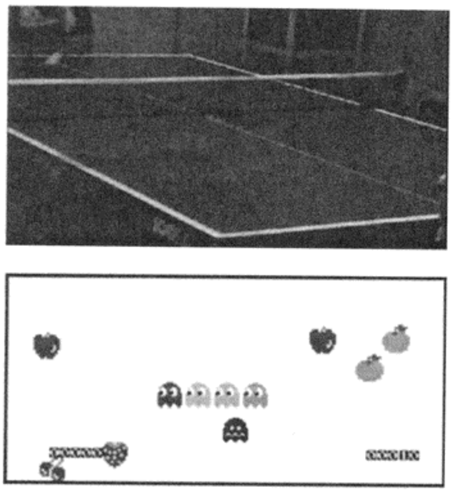
computer screen shot

Figure 8 Comets mode



computer screen shot

Figure 9 Thunderstorm mode



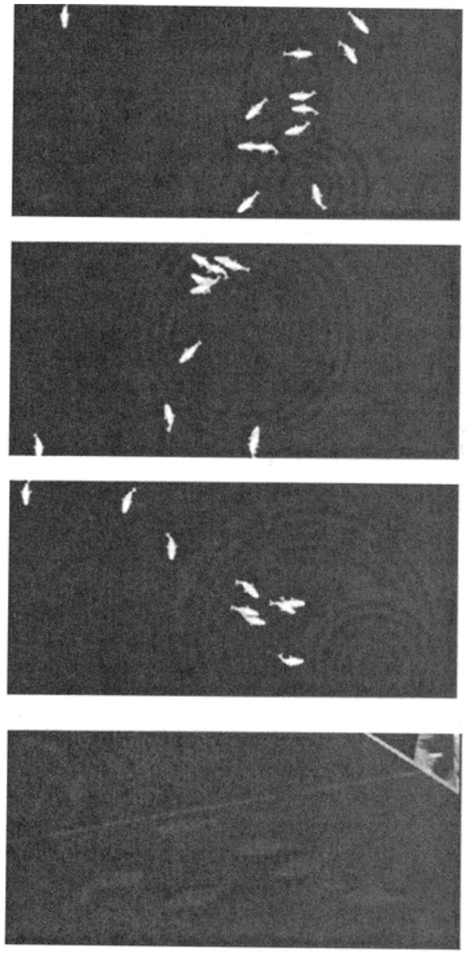
computer screen shot

Figure 10 Pac-Man® mode

We designed this ***Pat-Man*** mode to see if we could transform ping-pong into a very different engaging, competitive game. However, it was found that it was difficult to divide visual attention between tracking a ball and watching the Pat-Man screen on a table. This indicates that highly detailed display elements on the table do not work as well as simple visual patterns. The best results seem to occur when a simple visual pattern is combined with some level of complexity to keep the game interesting. The School of Fish mode is a good example of this concept.

School of Fish mode

The school of fish with water ripples seemed to be the most popular mode for players. In this mode, a school of fish swims on the table (Fig. 11) following a behavior pattern set forth from the algorithms that Craig Reynolds developed for flock behavior [12]. (Top three pictures are the images from a computer screen, and the bottom picture is a picture from the installation.) The ball causes a splash and a ripple in the "water" where it hits, scaring the fish. In time, the fish, following their individual behavior models, school back together. The simplicity of the visual display, combined with the complexity of the emergent activity from a behavior model made this mode continually compelling, even after days of play.



computer screen shots

Figure 11 School of fish and water ripples mode

We expect PingPongPlus to suggest new directions to integrate athletic recreation and social interaction with engaging digital enhancements. By the augmentation and transformation of physical games, new, engaging interactions can be developed in the physical/digital world.

ACKNOWLEDGMENTS

We thank the members of Tangible Media Group and our colleagues in the Digital Life and Things That Think Consortia at the MIT Media Laboratory for their support and collaboration. Thanks are also due to Rich Gold at Xerox PARC and Michael Naimark at Interval Research for their valuable comments on an early prototype of PingPongPlus.

REFERENCES

1. Azuma, R., A Survey of Augmented Reality, Presence, Vol. 6, No. 4, August 1997, pp. 355-385.
2. Chan, Y.T., A Simple and Efficient Estimator for Hyperbolic Location, IEEE Transactions on Signal Processing, Vol. 42, No. 8, August 1994, pp. 1905-1915.
3. Gorbet, M., Orth, M. and Ishii, H., Triangles: Tangible Interface for Manipulation and Exploration of Digital Information Topography, in *Proceedings of Conference on Human Factors in Computing Systems (CHI '98)*, (Los Angeles, April 1998), ACM Press, pp. 49-56.
4. Ishii, H. "The Last Farewell": Traces of Physical Presence. interactions 5, 4 (July + August 1998), ACM, pp. 55-56.
5. Ishii, H. and Ullmer, B., Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms, in Proceedings of Conference on Human Factors in Computing Systems (CHI '97), (Atlanta, March 1997), ACM Press, pp. 234-241.
6. Jacobson, J., et al., The Last Book. IBM Systems Journal, Vol. 36, No. 3, 1997, pp. 457-463.
7. Krueger, M., Artificial Reality II, Addison-Wesley, 1990.
8. Levy, Steven., Hackers: Heroes of the Computer Revolution, Delta Books; February 1994.
9. Maes, P., Darrell, T., Blumberg, B., and Pentland, A., The ALIVE System: Wireless, Full-Body Interaction with Autonomous Agents, *ACM Multimedia Systems*, Special Issue on Multimedia and Multisensory Virtual Worlds, ACM Press, Spring 1996.
10. MacKenzie, C. and Iberall, T., The Grasping Hand, North-Holland, 1994.
11. Ohshima,T., Satoh, K., Yamamoto, H., and Tamura, H., AR^2 Hockey, in Conference Abstracts and Applicatinos, SIGGRAPH '98, ACM, July 1998, pp. 110.
12. Reynolds, C. Flocks, Herds, and Schools: A Distributed Behavior Model, in Proceedings of SIGGRAPH '87, ACM Press, pp. 25-34.
13. Rheingold, H., Virtual Reality, Summit Books, 1988.
14. Strang, G., Introduction to Applied Mathematics, Wellesley-Cambridge Press, 1986
15. Ullmer, B. and Ishii, H., The metaDESK: Models and Prototypes for Tangible User Interfaces, in Proceedings of Symposium on User Interface Software and Technology (UIST '97), (Banff, Alberta, Canada, October, 1997), ACM Press, pp. 223-232.
16. Ullmer, B., Ishii, H. and Glas, D., mediaBlocks: Physical Containers,Transports, and Controls for Online Media, in *Proceedings of SIGGRAPH '98*, (Orlando, Florida USA, July 1998), ACM Press, pp. 379-386.
17. Underkoffler, J. and Ishii, H., Illuminating Light: An Optical Design Tool with a Luminous-Tangible Interface, in Proceedings of Conference on Human Factors in Computing Systems (CHI '98), (Los Angeles, April 1998), ACM Press, pp. 542-549.
18. Weiser, M. The Computer for the 21st Century. Scientific American, 1991, 265 (3), pp. 94-104.
19. Wellner, P., Mackay, W., and Gold, R. Computer Augmented Environments: Back to the Real World. *Commun. ACM*, Vol. 36, No. 7, July 1993
20. Wisneski, C., Orbanes, J. and Ishii, H., PingPongPlus, in Conference Abstracts and Applications, SIGGRAPH '98, ACM, July 1998, pp. 111.
21. Wisneski, C., Orbanes, J. and Ishii, H., PingPongPlus: Augmentation and Transformation of Athletic Interpersonal Interaction (short paper), in Summary of Conference on Human Factors in Computing Systems (CHI '98), (Los Angeles, April 1998), ACM Press, pp. 327-328.