SymBall - Camera Driven Table Tennis for Mobile Phones
Mika Hakkarainen, Charles Woodward
VTT Information Technology
Tekniikantie 4B, Espoo, P.O. Box 1203, FIN-02044 VTT, Finland
+358-9-20722 6500, +358-9-20722 5629
mika.hakkarainen@vtt.fi, charles.woodward@vtt.fi

ABSTRACT
We present a table tennis game concept, implemented for Symbian OS/Series 60 mobile phones, using the phone's integrated camera as the main game controller. The game demonstrates how the integrated cameras are already usable for interaction on mobile phones, with real time performance. Data communication between the phones is handled using Bluetooth, but the implementation would allow for using slower radio links too.

Categories and Subject Descriptors
H.5.2 [Information Interfaces and Presentation]: User Interfaces – input devices and strategies, interaction styles, I.4.8 [Image Processing and Computer Vision]: Scene Analysis – Color, K.8 [Personal Computing]: General – Games.

General Terms

Keywords
Camera phones, feature detection, mobile games.

1. INTRODUCTION
Mobile games are becoming a huge business, being fuelled by technology advancements such as new processors and 3D graphics hardware on mobile phones. In particular, smart phones with integrated cameras will trigger new ideas and concepts for the development of mobile games. Up to date, however, only a few games have been introduced to make use of the phone’s camera feature for game control and interaction. One of the first commercial camera phone game applications was Mozzies [1] on Siemens SX1 phones, an augmented shooting game where aiming the target is based on movement in the phone’s camera view. A more advanced example of feature-based mobile game control is provided by ARSoccer [2], originally implemented as a single player game on a PDA device.

In this paper we introduce some new ideas for camera-driven mobile games with a virtual table tennis game SymBall. The game is implemented for Symbian OS/Series 60 camera phones. It is derived as a variation of our augmented PC game CamBall [3], where people play table tennis over the Internet using real rackets. With SymBall instead, the players use the camera phones as rackets, with the camera view to control the virtual racket’s location. The users may play the game against a virtual wall, or against each other via Bluetooth connection. We describe the main ideas behind the game’s implementation, as well as directions for future development and research.

2. IMPLEMENTATION AND FEATURES
Figure 1 shows the typical playing situation of the SymBall game. Two players are playing against each other using the red vase to control their virtual rackets. Moving or tilting the phone moves the user’s virtual racket in the phone’s display. The opponent’s racket is shown at the other side of the table, with the virtual ball bouncing in between.

The phones communicate with each other using Bluetooth connection, having one phone as master and the other one as slave. The ball trajectory is calculated in both phones independently, based on the information received from the opponent. The transmitted information consists of racket position, hit events (hit or miss) and hit direction.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ACE ’05 June 15-17, 2005, Valencia, Spain.
Copyright 2005 ACM 1-58113-000-0/00/0004...$5.00.
By calculating the events independently, the game can handle even long delays in message passing. With a slow communication channel, such as GPRS, the user would see the ball disappearing in the horizon while waiting for the opponent’s hit. Thus the opponent might receive the event a second later, only then seeing the game view updated and the ball appearing back in the game.

The ball’s trajectory and bounces are calculated by physical laws and displayed using OpenGL ES graphics. In the current implementation, the hit direction is simply set at random and the racket is always located at the end of the table. For more sophisticated game control, e.g. the joystick could be used to define the racket direction and to introduce spins and slams.

As game options we provide alternatives for ball speed (e.g., slow motion), racket shape/size (to make the game easier), and turning the sounds on/off. The racket’s “follow mode” option determines whether the virtual racket moves in the same direction as the phone, or in the reverse direction as preferred by some players. Further options are available for determining the racket’s control color in the camera view, as explained next.

3. RACKET CONTROL
The position of the player’s virtual racket is controlled by feature detection from the phone’s camera view. For a good combination of reliability and speed, we have chosen to use a color based feature detection method. The method detects the largest, user-defined color area from the camera view by region growing. The method is simple enough so it can be implemented using just integer arithmetics.

The control color can be selected by the user from predefined ranges of red/green/blue, or by automatic method. The automatic method analyses the view for distinct color areas, and selects the most suitable color to be used for the game at the time.

The advantage of the color based method is that game does not require any pre-defined marker, shape or special pattern. Suitable color objects are practically always easy to find in the player’s natural environment, e.g., colored books, pens, candy bars or whatever – we’ve even used tomato slices on a sandwich to define the red pattern!

4. PERFORMANCE
With our implementation on Nokia 6600 phone, the total run time memory requirement for the game is about 500 kB. The color pattern detection method can in principle analyze up to 50 fps 160x120 resolution images with a usable pattern. The phone provides only 15 fps from the camera so this leaves enough processing power also for other game events handling. Due to event prediction, we can update the game view even at higher speeds than what the camera provides. Overall, with the Bluetooth connection the SymBall game refresh rate with the varies between 15-18 fps.

5. CONCLUSIONS AND FUTURE WORK
According to our experience, people learn to play SymBall in a minute, and they enjoy playing the game even for long periods of time. Compared to using the joystick, the camera-driven interaction paradigm clearly adds interest to the game. With SymBall, another part of the game’s physical aspect is finding optimal color patterns that help to beat the opponent.

To make the game technically more challenging, hit force based on color pattern size, camera movement speed and direction, as well as making use of the joystick for extra effects could be applied. We may also envisage more advanced feature detection methods to allow for improved controlling of the racket, for adding spins and/or force of hit without joystick interaction.

Smart-phones with twin cameras would give to opportunity to stream live image of the opponent’s face to the game view (similar to CamBall [3]). This way the game would gain a new, mixed reality aspect with increased physical presence. Similar concepts could be implemented also with single camera phones, having the color pattern attached to a mirror in front of the player. Another, “fighting” game concept is obtained with the current implementation simply by using the opponent’s clothes as the color pattern for game control; see Figure 2.

The GPRS/3G version would allow the players to situate even in different continents while playing against each others. The implementation would require a special relay server for fixed network that handles player pairing and data passing between them.

Other ongoing and future work includes adding improved 3D graphics to the game, e.g. an avatar following the opponent’s movements at the other side of the table, virtual audience, and so forth. 3D graphics enhancements to SymBall are developed in co-operation of VTT and the OpenGL ES development company Hybrid Graphics Ltd in Finland.

6. REFERENCES