Personal Contextual Awareness through Visual Focus

Li-Te Cheng *Lotus Research*

John Robinson *University of York*

The authors explore options for using body parts as focal points for wearable systems. Their two working systems, Handel and Footprint, demonstrate several possibilities for recognizing visual body cues.

One of the goals of wearable computer technology is to let users move freely in their environments while interacting with the virtual information that their wearable computers associate with the real-world objects around them.[1] Augmentation and mediation technology can help wearables achieve this goal by gathering and acting on sensor readings of the user's activity and environment. This technology, in other words, can help create an awareness of the user's personal context, which is an increasingly important feature for wearable computer interfaces.[2]

However, programming a wearable computer to sense when it is appropriate to enable interaction is a significant challenge that involves analysis of the user's overall environment. Sensing context presents serious complications, especially because a user's context is never stable. However, even in the complete absence of environmental stability, there still is one physical object on which wearable technology can focus: the user's own body.

One of our ideas is to focus, in particular, on the user's hands and feet. We can apply this technique to wearables in several contexts: physical rehabilitation, choreography, pathfinding, sports, and so forth. By focusing wearable technology on hands and feet, we can define virtual annotations and commands by hand or foot gestures. For example, framing a shot for video or photography could be triggered by a two-handed framing gesture, where the size and location of the framing gesture defines the parameters of the snapshot and the placement of a virtual annotation window.[1] The wide range of possible applications of such technology present new opportunities for mobile computing devices.

To demonstrate how this notion of personal context can enhance specific functions in wearable computers, we built two working systems: Handel and Footprint. We designed both to enhance user learning experiences seamlessly by linking instructional overlays to hands and feet.

Personal context

We define *personal context* as the contextual awareness of the user's body as a stimulus and rendering surface for augmentation and mediation technology. While general context is derived from the environment at large, personal context is derived from an awareness of a user's body parts with respect to a particular task. Performing particular tasks requires that we move our hands and feet in certain routine ways, which suggests a good focal point for any virtual information that a wearable computer might present. For example, if a wearable were to use direct sensor measurements or a combination of sensors and pattern recognition, it could derive personal context from the user's body movements. Virtual information could then augment the user's experience through a heads-up display or through audio feedback. The ultimate goal would be to provide such feedback mediated by how relevant that feedback would be to the task at hand.

Augmented reality systems[3] are designed to overlay virtual information onto the real world. Such systems include first-person applications that use head-mounted displays and environmental sensor cues to overlay information onto appropriate objects to help direct a user in a particular task (such as servicing a printer[4] or reading an enhanced book[5]). A personal-context approach to a printer-servicing application or an augmented book would rely on the user's gaze

1094-7167/01/$10.00 © 2001 IEEE

with respect to the hands rather than on some kind of ultrasonic tracking infrastructure[4] or on specially marked book pages.[5]

Having a wearable computer rely on the user's body as a rendering surface does not necessarily imply a body-stabilized interface such as a cylindrical or spherical overlay that surrounds the user.[6] An object-centric interface—where the objects are really parts of the user's body—appears to the user to be world-stabilized. Unlike a true world-stabilized interface,[6] an object-centric interface attaches information to body parts with little or no attempt to assess a complete world model. In these systems, overlay graphics are linked to the user's hands, and tracking is done using simple two-dimensional techniques that don't require knowledge of a user's physical location. A simplified model based on personal context makes tracking algorithms more readily available because they are somewhat easier to implement than complete world models.

Handel and Footprint both use personal context (as we've defined it here) to infer a user's need for augmentation.

Handel: Giving the user a hand

A considerable amount of research exists in the areas of hand-based user interfaces and computer-vision techniques used to locate and recognize hand gestures.[7] Data gloves, magnetic trackers, and optical sensors can all be used to obtain hand orientation. In these cases, however, the hand acts solely as an input device. We designed Handel (*hand*-based *e*nhancement for *l*earning) to rely on hand movements to trigger an augmented-reality overlay onto the user's hands during piano practice. Essentially, Handel creates a "hands-up" display instead of a heads-up display.

There are some preexisting technologies that are similar to Handel. Some of these technologies merge interactive graphics with hands[8] and some even place small displays on hands to ease interaction with large-screen virtual environments.[9] There are also countless piano-teaching tools, including self-help computer software that shows keyboard layouts to guide pianists. Modern acoustic player pianos such as the Disklavier allow direct playback on the keyboard from music files or from captured piano-key action.

In Handel, the pianist is equipped with a wearable computer system and sits at a normal acoustic piano with no sheet music. As the pianist attempts to play a piece from memory, the pianist can look down at the hands to invoke the visual aid. Focusing on the hands is the trigger. If the pianist does not look down, no graphics clutter the screen so that the pianist can concentrate on playing from memory. When the pianist looks at the right hand, Handel shows only the right hand's part of the music at the current position in the piece. Similarly, if the pianist looks directly at the left hand, Handel shows only the music for that hand. Handel uses each hand as an input to trigger the overlay of virtual sheet music. Because Handel presents the music near the relevant hand, the hand also acts as a context-sensitive display window for the sheet music.

Handel uses a head-mounted video camera to perform scene analysis. The pianist's hands are totally unencumbered and free to interact normally with the piano. Handel uses Fast Fourier Transform (FFT) phase-correlation analysis[10] on consecutive video frames to determine whether the pianist's head is looking to the left or to the right. Handel also uses a skin-color table to detect whether a hand is in view or not. Handel sets the skin-color scheme during a training session beforehand. Skin-color detection is sufficient for determining where a pianist is looking, because Handel assumes that the only thing the head-mounted camera will see is the piano. Figure 1 illustrates Handel's general system architecture. On our Pentium 233-MHz subnotebook, Handel runs at about five frames per second.

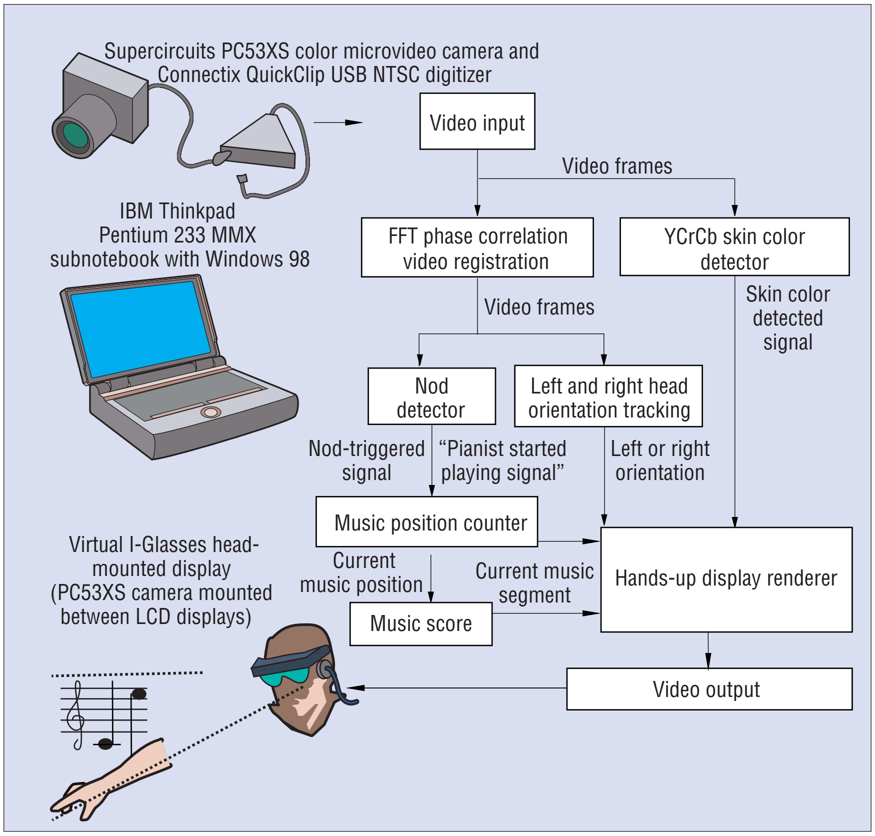
Supercircuits PC53XS color microvideo camera and Connectix QuickClip USB NTSC digitizer
Video input
Video frames
IBM Thinkpad Pentium 233 MMX subnotebook with Windows 98
FFT phase correlation video registration
YCrCb skin color detector
Video frames
Skin color detected signal
Nod detector
Left and right head orientation tracking
Nod-triggered signal
"Pianist started playing signal"
Left or right orientation
Music position counter
Virtual I-Glasses head-mounted display (PC53XS camera mounted between LCD displays)
Current music position
Current music segment
Hands-up display renderer
Music score
Video output

Figure 1. Handel's architecture and components.

The practice session begins with the pianist loading the music score into Handel. For the current implementation, we created a simple score language to store the music in a text file. The pianist dons the head-mounted display and sits in front of the piano. He or she then gives a nod when starting to play the memorized music. Handel uses FFT phase correlation to detect a strong vertical displacement (the nod) to begin incrementing an internal counter to keep track of the current position in the piece. In the current implementation, Handel increments the counter at a predetermined rate.

While the pianist plays the piece, Handel doesn't overlay anything on the pianist's heads-up display (Figure 2a) until it sees skin color. When Handel detects skin, it assumes that the pianist is looking down at the hands. Handel determines what hand to overlay on the

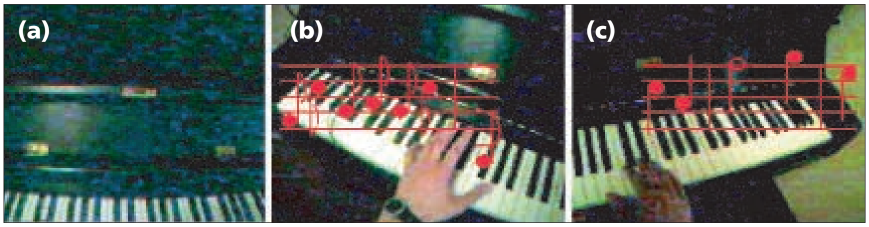

Figure 2. Views from the head-mounted display: (a) nothing overlaid when no hand is in view, (b) left-hand part displayed for the left hand, and (c) right-hand part displayed for the right hand.

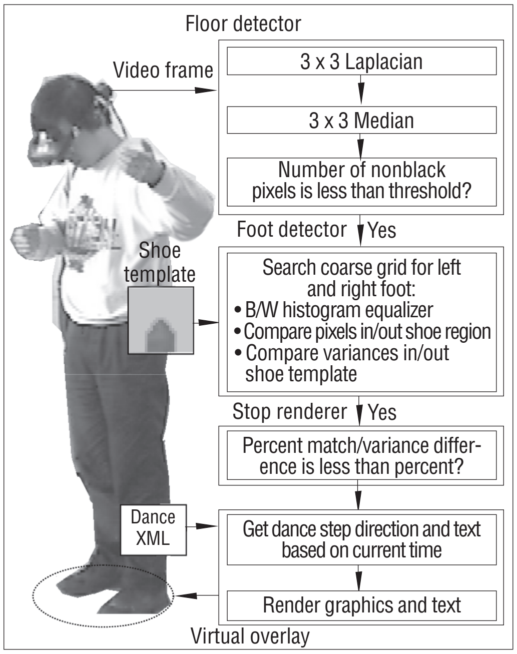
Floor detector
Video frame
3 x 3 Laplacian
3 x 3 Median
Number of nonblack pixels is less than threshold?
Foot detector
Yes
Shoe template
Search coarse grid for left and right foot:
• B/W histogram equalizer
• Compare pixels in/out shoe region
• Compare variances in/out shoe template
Stop renderer
Yes
Percent match/variance difference is less than percent?
Dance XML
Get dance step direction and text based on current time
Render graphics and text
Virtual overlay

Figure 3. Footprint's system architecture.

basis of whether the pianist is looking to the left (Figure 2b) or to the right (Figure 2c). Handel then displays the musical score at the current position, for the given hand, on the head-mounted display. Meanwhile, the software continues to update itself while the pianist is playing. Handel renders the score at a fixed position on the left side of the display for the left hand and on the right for the right hand. Handel doesn't link the score to the hand itself because this would cause the overlayed musical notes to move with the playing hand. The virtual musical score disappears whenever the pianist looks up from the keys.

Footprint: Another step in personal context

Footprint, our second personal-context application, uses the feet as the focal point for computer assistance. Previous work on foot-based user interfaces can generally be classified as hardware-based or vision-based implementations. Applications for such interfaces include dance performance, choreography, motion capture for animation, and interactive entertainment.

Hardware-based schemes often rely on body-mounted magnetic, ultrasonic, or LED devices that monitor the motion of the whole body. Hardware systems can quickly provide great accuracy and a wealth of data but require complex infrastructure or equipment. Computer vision systems make use of a single camera or several cameras fixed in the environment to monitor a specific location for body motion. While some systems rely on body-placed markers to aid visual detection, many analyze the scene with only an a priori model of the human body.[11] These systems are more interested in entire body motion rather than just foot motion. One exception to this rule is a technology[12] that derives 3D motion data from a bicyclist's legs by analyzing specially textured shorts. Computer vision systems often free the users from having to wear any special devices, but they also require good lighting conditions and fast computers to process complex algorithms.

Footprint operates on the same minimal wearable computer system as Handel: a small laptop with a see-through head-mounted display complete with an attached video camera. The user's feet trigger computer interaction when Footprint detects them in-screen. Footprint accomplishes foot detection by analyzing the frames captured by the video camera and exploiting a priori knowledge of the owner's feet.

Footprint can handle basic waltz steps. Figure 3 shows Footprint's architecture. A typical practice session begins when the user starts the application and loads the system settings and dance information. The user activates an internal timer, which allows Footprint to synchronize dance steps to time. The user then performs the dance to music the computer supplies. Whenever the user needs help, he or she simply looks down. As Figure 4 shows, Footprint then presents graphics and text that indicate where the feet should move next. This information disappears when the user looks back up. Looking down at the feet provides a natural means to interact with the computer. As in Handel, Footprint only shows information when the user needs it, which minimizes graphical clutter on the limited-resolution head-mounted display.

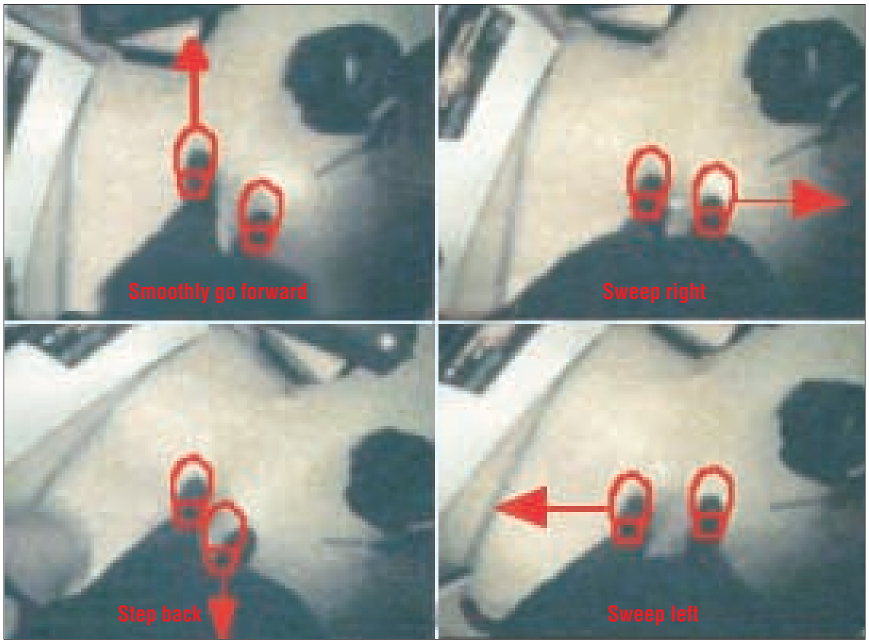

Figure 4. Dance step instructions as seen by the user's head-mounted display.

The feet-detection algorithm assumes that the user is wearing dark shoes and that the floor is fairly uniform in color. Presuming that lack of edges corresponds to uniformity, the algorithm first checks for a fairly uniform background by using a nonlinear spatial activity detector. If the detector senses a cluttered background, Footprint assumes the user is not looking at a uniform floor and will not perform any foot detection.

If the current video frame passes the floor test, Footprint matches a predefined shoe template against a coarse grid on the current frame. The grid is set to the left half of the image to search for the left foot. At each grid position, Footprint equalizes the local rectangular region to be compared against the template and calculates the local variances inside and outside the shoe area. If the difference falls below a threshold (indicating the texture inside and outside the shoe is the same), or if the total difference within shoe area against the template exceeds a threshold (indicating the shoe area does not have a dark shoe), then Footprint does not detect a foot.

Otherwise, Footprint computes a measure proportional to the match against the template divided by the difference of variances. Footprint classifies the grid position with the smallest measure (that still falls under a threshold) as a foot. Footprint then repeats the process to find the right foot, except that Footprint sets the grid to the right of the discovered left-foot position. Figure 5 illustrates the foot-detection algorithm under different lighting and floor conditions. On subsequent steps after the first, the system searches around the last detected coordinates first before performing a full grid search.

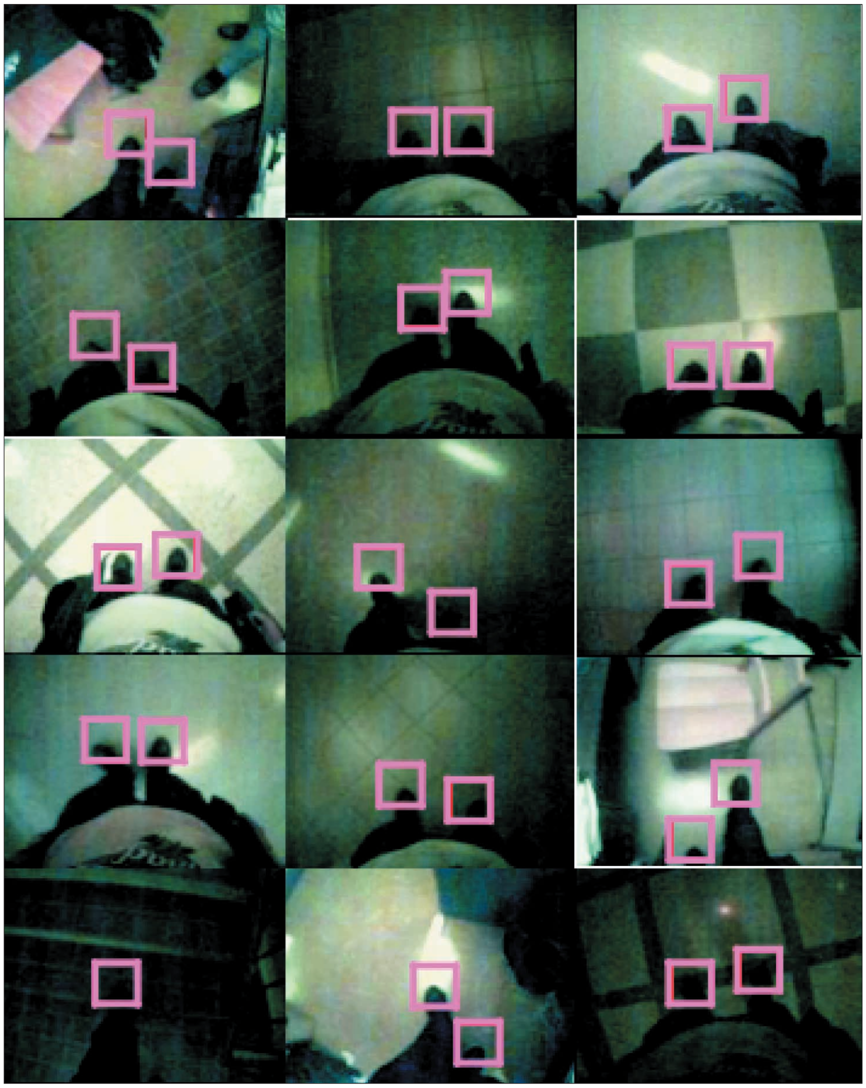

Figure 5. Footprint's foot detection in different lighting and floor conditions, as seen by the head-mounted camera. Detected feet are highlighted by rectangles.

We've represented the dance itself as an XML text file using custom markups. As Figure 6 illustrates, Footprint represents the dance moves clearly. Footprint can present these moves in sequence using common ballroom dance step speed denotations such as "quick" or "slow" along with text descriptions of each movement. This new "dance markup language" is similar to SMIL, a markup language for synchronized multimedia.[13] All the parameters controlling Footprint are stored in another XML text file. Footprint runs at about four frames per second on a Pentium 233 laptop, which includes all the image processing, video capture, and graphics rendering required by the ballroom dancing task. It detects the feet well and runs effectively with the basic waltz.

Future directions

We tested Handel successfully only on an acoustic piano for a short musical piece. While the system proved to be comfortable to use, there are of course numerous improvements that could be made, and we would like to do a formal study to assess Handel's benefit (or detriment) to memorizing piano music. Handel's FFT phase approach combined with skin detection seems to be sufficient for detecting a hand and for determining which hand is currently in focus. An improvement would be to employ projective-based scene analysis technology such as the kind found in wearable camera systems.[1]

Footprint would likely benefit from a faster computer, foot-pose recognition, and additional user tests to optimize the dance instruction presentation. It would be interesting to study other modes of computer-assisted teaching for Footprint, such as having Footprint measure feet movement to assess proper steps. Extending the system to recognize and coordinate with a live partner would also be desirable. Using an XML-based dance step file to represent content and an XML configuration file as a style sheet means that Footprint is, in one sense, a browser for wearable computer interfaces.

Because the dance markup language is a simple description of dance steps, it can be interpreted for different purposes on other platforms. For instance, another wearable computer could create XML-based data from

```
<dance>
    <title>Basic Waltz</title>

    <step name="advance" duration="quick">
         Smoothly go forward
         <leftfoot direction="forward">Left first
              </leftfoot>
    </step>

    <step name="right" duration="quick">
         Sweep right
         <rightfoot direction="right">Right first
              </rightfoot>
    </step>

    <step name="right wait" duration="quick">
         Close
         <leftfoot direction="hold">Left arrives late
              </leftfoot>
    </step>

    <step name="back" duration="quick">
         Step back
         <rightfoot direction="back">Right first
              </rightfoot>
    </step>

    <step name="left" duration="quick">
         Sweep left
         <leftfoot direction="left">Left first
              </leftfoot>
    </step>

    <step name="left wait" duration="quick">
         Close
         <rightfoot direction="hold">Right arrives
              late</rightfoot>
    </step>
</dance>
```

Figure 6. The dance markup file for the basic square-step waltz.

streaming sensor data. A 3D-capable XML desktop browser could translate the dance step file into a dancing avatar that could be incorporated into a virtual reality environment or a computer graphics movie. Online XML database engines could index and catalog the dance step file in a repository, allowing for text-based searches for human gesture and motion.

In general, context-aware applications can exploit XML as a foundation to create readable, portable, and indexable notations for human gesture, motion, and interaction with the real world. Because gesture, motion, and interaction vary over time and depend on different conditions, context-aware notations might adapt properties and behaviors from scripting languages and temporal-based notations.

With only simple computer-vision techniques, Handel and Footprint demonstrate the great possibilities for more natural human–computer interaction. And ,using XML in Footprint illustrates the potential for XML to become a portable format to represent human activity for both wearable and desktop applications. ■

References

1. S. Mann, "Wearable Intelligent Signal Processing," *Proc. IEEE*, vol. 86, no. 11, Nov. 1998, pp. 2123–2151.
2. T. Starner, B. Schiele, and A. Pentland, "Visual Contextual Awareness in Wearable Computing," *Proc. Second Int'l Symp. Wearable Computers*, IEEE CS Press, Los Alamitos, Calif., Oct. 1998, pp. 50–57.
3. R.T. Azuma, "A Survey of Augmented Reality," *Presence*, vol. 6, no. 4, Aug. 1997, pp. 355–385.
4. S. Feiner, B. MacIntyre, and D. Seligmann, "Knowledge-Based Augmented Reality," *Comm. ACM*, July 1993, pp. 53–62.
5. M. Billinghurst, H. Kato, and I. Poupyrev, "The Magicbook: Moving Seamlessly Between Reality and Virtuality," *IEEE Computer Graphics and Applications*, vol. 21, no. 3, May/June 2001, pp. 6–8.
6. M. Billinghurst et al., "An Evaluation of Wearable Information Spaces," *Proc. IEEE Virtual Reality Annual Int'l Symp.*(VRAIS 98), IEEE Press, Piscataway, N.J., Mar. 1998, pp. 20–27.
7. A. Pentland, "Looking at People: Sensing for Ubiquitous and Wearable Computing," *IEEE Trans. Pattern Analysis and Machine Intelligence,* IEEE CS Press, Los Alamitos, Calif., Jan. 2000, pp. 107–118.
8. M.M. Krueger, "Environmental Technology: Making the Real World Virtual," *Comm. ACM*, July 1993, pp. 36–37.
9. J. Miyasato, "See-Through Hand," *Proc. 8th Australian Conf. on Computer Human Interaction (OzCHI 98)*, IEEE CS Press, Los Alamitos, Calif., November 1998.
10. L. Cheng and J. Robinson, "Dealing with Speed and Robustness Issues for Video-based Registration on a Wearable Computing Platform," *Proc. Second Int'l Symp. Wearable Computers*, IEEE CS Press, Los Alamitos, Calif., Oct. 1998, pp. 84–91.
11. J. Ohya, J. Kurumisawa, and R. Nakatsu, "Virtual Metamorphosis," *IEEE Multimedia*, vol. 6, no. 2, April/June 1999, pp. 29–39.
12. F. Lerasle, G. Rives, and M. Dhome, "Tracking of Human Limbs by Multiocular Vision," *Computer Vision and Image Understanding*, vol. 75, no. 3, Sept. 1999, pp. 229–246.
13. W3C World Wide Web Consortium, "Synchronized Multimedia," www.w3.org/AudioVideo (current 18 June 2001).

The Authors

Li-Te Cheng is a research scientist at Lotus/IBM Research. His research interests include collaboration for wearable and ubiquitous computing, virtual reality, and computer-vision–supported augmented reality. He is completing a PhD in Electrical Engineering at Memorial University of Newfoundland. He received an MASc and a BASc in Systems Design Engineering from the University of Waterloo and a Certificate of Space Studies from the International Space University. He is a member of the IEEE Computer and Signal Processing societies and ACM SIGCHI. Contact him at li-te_cheng@lotus.com.

John Robinson is Professor of Media Technology in the Department of Electronics at the University of York. From 1996 to 2000 he held an Industrial Research Chair at Memorial University of Newfoundland, where he performed the work reported in this article. Previously, he worked for Standard Telephones and Cables Ltd., Basildon, UK, Bell-Northern Research Ltd., Verdun, Quebec, Canada, and the University of Waterloo, Ontario, Canada. He received postgraduate degrees in Electronic Engineering, from the University of Essex, and in the Humanities from Memorial University of Newfoundland. He is a Fellow of the IEE and a Member of IEEE. Contact him at www-users.york.ac.uk/~jar11/.