Implementing Phicons: Combining Computer Vision with InfraRed Technology for Interactive Physical Icons

Darnell J. Moore, Roy Want**, Beverly L. Harrison**, Anuj Gujar**, Ken Fishkin***

* Center for Signal & Image Processing
Georgia Institute of Technology
Atlanta, Georgia 30332 USA

** Xerox PARC
3333 Coyote Hill Road
Palo Alto, CA 94304 USA

ABSTRACT

This paper describes a novel physical icon [3] or “phicon-” based system that can be programmed to issue a range of commands about what the user wishes to do with hand-drawn whiteboard content. Merely placing a labeled phicon in proximity to the whiteboard content material, automatically executes the chosen command. This is achieved using infrared signaling in combination with image processing and a ceiling-mounted camera system. We leverage existing camera systems that are already used for capturing whiteboard content [4] by further augmenting these systems to detect the presence and location of IR beacons within an image. An HDLC-based protocol and a built-in IR transmitter is used to send these signals.

KEYWORDS: Infrared, computer vision, image processing, physical icons, phicons, tangible user interfaces, ubiquitous computing, physical UI, HDLC

SPECIFYING COMPUTER COMMANDS USING PHICONS

At Xerox PARC, a number of projects are exploring ways to enhance physical media, such as whiteboards and paper, by seamlessly integrating computational processing power (see [4, 5]). Within our recent whiteboard applications [4], we wanted to enable users to process their usual hand drawn content material using simple commands initiated by physical icons. For example, after taking notes on a whiteboard during a meeting, a user could pick up a PRINT phicon and place it in proximity to the content material. After pushing a button on the phicon, an image of the whiteboard would be acquired automatically by a camera and printed out. Similarly, the image could be sent to others using an EMAIL phicon.

Most of the work in this area has used innate objects that provide little, if any, feedback to the user. In particular, providing on-line help or prompts to facilitate interaction where necessary, providing indications of the status of the "request" users may have issued, or other user feedback is typically infeasible via these kinds of phicons. We wished, instead, to develop a more sophisticated physical icon that could be instantiated in many possible form factors and retain the simplicity and affordances of being a physical object. To develop phicons with more interactive communicative abilities, we decided to leverage an existing PARC device (Figure 1) and computer vision.

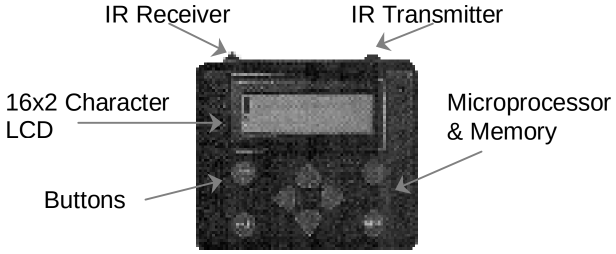

Figure 1. Prototype phicon with interactive UI

PHICON IMPLEMENTATION STRATEGIES

There are a number of methods for implementing phicons. Early approaches used Polhemus 6 degree-of-freedom devices embedded into or onto objects [1, 2]. More recently, others have used various forms of embedded wireless tags, e.g., RFID, or orientation sensors [3, 5]. These approaches often assume either a tethered or wireless tag embedded in the desired innate object and a sensor that can detect its presence or absence. Typical implementation issues include: range of detection, ability to detect location in 2D versus 3D space, ability to detect multiple sensors simultaneously, interference and/or noise in signal detection, and integration of user feedback (i.e., usually on a secondary display not on the invoking innate object). Some approaches have also used computer vision with some form of visual barcode or glyph to tag objects or issue commands to a host system [4,6]. While such highly distinctive and visible encodings improve reliability and processing time, they can dramatically alter the aesthetics of the target object (which may or may not be problematic). Again, typical implementation issues include: system reliability, performance and response time, location within 2D and 3D spaces, and user feedback mechanisms.

IMPLEMENTATION USING ACTIVE PHICONS

We use small, programmable devices called *Minders* as prototypes for phicons in our vision-based system. These battery-powered devices are equipped with IR transceivers for sending data and for downloading code developed and compiled on a PC. The user interface includes a small 16x2 character LCD and several buttons, allowing users to navigate command menus or view/edit the device's settings. When the phicon powers down, its current state is lost, but downloaded code stays resident in memory.

We use an HDLC- (High-level Data Link Control) based protocol for transferring data from the phicon to the vision system. This protocol includes framed packets, unique flags, bit-stuffing, a checksum, and typed data, as shown below.

Delimiter	Address	Data Type	Data	CRC	Delimiter
4-bits	8-bits	4-bits	4-bits	4-bits	4-bits

With a clear line-of-sight to a monochrome CCD-based camera, the IR beacon appears as a bright spot in captured images and is the basis for recovering the phicon's location and data (see Figure 2). In contrast, most color cameras are equipped with filters that diminish the prominence of IR spots. Each phicon is programmed to broadcast a unique packet that is recovered over a sequence of frames. According to Nyquist sampling theory, a bit-rate of ½ the processed frame rate can be expected in systems with a dedicated operating system. However, for an OS where image capture and processing must share resources with other system processes, e.g., Windows NT, Solaris, etc., we developed empirical approaches for establishing frame rate.

A vision-based system offers the spatio-temporal resolution to recover data from multiple phicons simultaneously. Phicons are programmed to repeat IR transmission a fixed number of times, which increases data reliability and addresses potential synchronization issues with image capture.

RESULTS AND FURTHER ISSUES

We have built and successfully tested the vision system using multiple phicons placed on a whiteboard. With data rates averaging 7-8 bits per second, a packet takes about 5 seconds to transmit. On a sufficiently powerful PC, images containing several IR spots (from multiple phicons) can be processed without any perceptible change in the processed frame rate. Broadcasting redundant packets proved to be helpful as some users interrupted transmission by inadvertently breaking the invisible line-of-sight IR path. A distinctive audible tone sounds for each command to verify that the packet was properly received. Data can also be sent to a phicon through an IR transmitter/receiver mounted with the camera system (ensuring the angle is incident with the item). In this way, content can be sent to specific phicons for display on their LCDs and, thus, users can receive prompts, on-line help, and other information. Users responded favorably to the system's ease of use and the phicon's multi-functionality and inconspicuous size (versus drawing a glyph large enough to be seen in an image), but were not enthused with the slow data transfer rate, especially when retransmission was necessary.

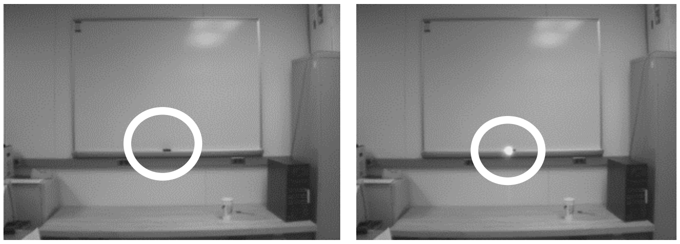

Figure 2. Before and after IR firing from the camera's view.

Advantages of this approach include:

- Needs no barcode/glyph; Can be miniaturized as needed
- Programmable data and user interface
- Several can transmit simultaneously without collisions
- Transmission scheduling is simple

Disadvantages include:

- Sensitive timing issues; requires dedicated CPU time
- Slow data transfer rate
- Susceptible to user occlusion

We plan to explore several usability issues, such as increasing the data rate and adjusting form factors for different applications. In some cases, it may be more appropriate to have a much simpler UI, perhaps with only one button for sending a single command versus the more elaborate, multifunction UI used in our prototype. In fact, the dimensions of the phicon's housing can be miniaturized beyond the point of being easily located in an image. Command specific casings, such as a tiny "envelope" with a send button, are also under development.

REFERENCES

1. Fitzmaurice, G., Ishii, H., and Buxton, W. "Bricks: Laying the Foundations for Graspable User Interfaces," *Proc. of CHI'95*, pp. 442-449.
2. Hinckley, K., Pausch, R., Goble, J.C., and Kassell, N.F."Passive Real-World Interface Props for Neurosurgica Visualization," *Proc. of CHI'94*, pp. 452-458
3. Ishii, H., & Ullmer, B. "Tangible bits: Towards seamless interfaces between people, bits and atoms." *Proc. of CHI'97*, pp. 234-241.
4. Moran, T., Saund, E., van Melle, W., Gujar, A., Fishkin, K., Harrison, B. "Design and Technology for Collaborage: Collaborative Information Collages on Physical Walls." Submitted for publication.
5. Want R., Fishkin, K., Gujar, A. and Harrison, B. "Bridging Physical and Virtual Worlds with Electronic Tags." *Proc. of SIGCHI '99* (Pittsburgh, PA, May 15-20) ACM, New York, 1999.
6. Stafford-Fraser, Q., Robinson, P., Tauber, M.J., Bellotti, V., Jeffries, R., Mackinlay, J.D., and Nielsen, J. "BrightBoard: a video-augmented environment," *Proc. of CHI '96*, pp. 134-41.

ACKNOWLEDGEMENTS

We wish to thank Dan Greene and Eric Saund for their guidance.