

# CyberCode: Designing Augmented Reality Environments with Visual Tags

Jun Rekimoto and Yuji Ayatsuka

Interaction Laboratory

Sony Computer Science Laboratories, Inc.

3-14-13, Higashi-Gotanda, Shinagawa-ku, Tokyo 141-0022 Japan

Phone: +81-3-5448-4380

Fax: +81-3-5448-4273

E-Mail: {rekimoto, aya}@csl.sony.co.jp

<http://www.csl.sony.co.jp/person/rekimoto.html>

## ABSTRACT

The CyberCode is a visual tagging system based on a 2D-barcode technology and provides several features not provided by other tagging systems. CyberCode tags can be recognized by the low-cost CMOS or CCD cameras found in more and more mobile devices, and it can also be used to determine the 3D position of the tagged object as well as its ID number. This paper describes examples of augmented reality applications based on CyberCode, and discusses some key characteristics of tagging technologies that must be taken into account when designing augmented reality environments.

**KEYWORDS:** Augmented reality, merging virtual and real, ID-aware interface, CyberCode.

## INTRODUCTION: ID-AWARE AUGMENTED ENVIRONMENTS

In designing augmented reality systems, it is often essential to implement a tagging (ID) system to make a link between physical and digital spaces. Some examples of tagged IDs are barcodes [18, 11, 9, 6], radio-frequency (RF) tags [2, 23], resonant tags [13], and infrared IDs [22].<sup>1</sup> Unlike the digital links or icons on the computer screen, these tags are tangible, can be manipulated physically, and can be used in real-world contexts (e.g., installed in a building, attached to a wall, or printed on a book). When these tags are part of our physical environment, devices with a tag reader can retrieve digital information from them [18, 10], activate associated actions, or attach information to them [17].

These tags can also be used as operands for direct-manipulation operations. For example, a user picks-up data from a printed document, and drops it on the nearby printer. This operation

<sup>1</sup>Location information, based on the GPS for example, can also be regarded as a kind of ID. This paper, however, focuses on tags that can be attached to objects or environments.



Figure 1: Examples of CyberCode visual tags (top: tags used in several contexts. bottom left: earlier prototypes. bottom right: a concatenated type for more "bits").

can be realized by recognizing an ID on the document and an ID on the printer. This is a "physically" extended concept of drag-and-drop, an interaction technique commonly used at computer GUIs.

Each tagging technology has each own advantages and disadvantages (e.g., an RF tag is not printable and an infrared (IR) beacon needs batteries), and these often determines the characteristics of augmented reality applications. Printed tags are probably the least expensive and most versatile tagging technology: They can be easily made by normal printers, can be attached to almost any physical object, and can be recognized by mobile readers. Even when other tagging systems



Figure 2: Examples of camera-equipped mobile devices (left: Sony VAIO notebook PC. right: CASIO Palm-size PC).

are more suitable for a particular application, paper tags often provide as a good starting point for rapid prototyping.

We have designed a visual tagging system called CyberCode, a 2D-barcode system designed to be used in several augmented reality systems, and in this paper we explain some of the features distinguishing this tagging system from other 1D/2D-barcode systems. We also introduce several “real-world” augmented reality applications based on the technology used in this system.

### **CyberCode: A VISUAL TAGGING SYSTEM FOR AUGMENTED REALITY**

Figure 1 shows some examples of CyberCode tags. As in other systems using 2D-barcode tags, information is encoded in a two-dimensional pattern, and can be optically recognized from image data. This section presents several key features of CyberCode that are specific to the context of augmented reality applications.

#### **Using the CCD / CMOS camera of a mobile device as a tag reader**

CyberCode’s 2D pattern is designed to be recognized by the low-cost CCD or CMOS cameras becoming common in mobile devices such as notebook computers and PDAs. Traditional 2D-barcode readers often contain a CCD sensor with relatively high resolution as well as lights and optics that are specific to code recognition (e.g., the focal length adjusted for the ideal recognition distance between the reader and the target). As a result, they can recognize very fine code patterns, patterns that contain several hundreds of bits. The readers can be used only for code recognition, however, and are almost impossible to use as normal cameras.

On the other hand, recent trends in multimedia communication are pushing mobile devices to have small cameras. One sub-notebook computer popular in Japan, for example, the Sony VAIO-C1 (also called the “PictureBook”), has a built-in CCD camera. Other examples include Epson’s Locatio (a PDA with a GPS and a camera), the Sharp Zaurus, and Casio’s palm-size PC (see Figure 2), and many telephone companies are developing cellular phones with cameras. Digital cameras and digital camcorders are already commonplace, and the next generation will be “networked.”

One might thus expect that many, if not, most of tomorrow’s mobile devices will have a built-in camera. It is consequently prudent to design a visual tagging system that can be recognized by these cameras, making many mobile devices “augmented-reality ready.”

Since these mobile devices will also be “network ready”, for connection to a wired or wireless network, we will no longer have to encode a large amount of data in printed tags, if we can attach to an object an ID (a link to digital contents) that is itself a physical object. Once the ID is recognized, the mobile device can easily retrieve all data relating to the object the ID is attached to by accessing a database. A user can also change the content of an ID dynamically by changing the association between the ID and the content. For example, a user might be able to attach a voice memo to the visual tag attached to an office door. Such a memo would act as a “digital post-it.”

To enable the IDs to be read by inexpensive low-resolution CCD cameras, CyberCode encodes fewer bits (24 or 48bits, excluding error correction bits), than other 2D-barcode systems do. A CCD camera (either monochrome or color) that has 100,000 or fewer pixels can therefore be used as a tag reader. Such a camera works well under normal ambient lighting, and with normal optics. The tag shape is designed to compensate for distortion of the visual image, so tags are recognized even if they are not perfectly placed in front of the camera.

#### **Using environmental cameras as tag readers**

CyberCode tags can also be recognized by using nonportable cameras. A zoom camera mounted on the ceiling, for example, can identify an object placed on a desktop. A pan/tilt camera can be used so that objects in various parts of the environment can be recognized by a single camera (e.g., the same camera could be used to detect objects on the table, as well as the wall). The distance between a camera and a tag can also be varied. We think these things cannot be done when other tagging systems are used, and that CyberCode therefore offers more flexibility when augmented reality applications are being designed.

#### **Measures the code position and orientation as well as an ID**

Another advantage of using a camera as a visual tag reader is that the 3D position and orientation of a tag relative to the camera can be estimated – and, consequently, the camera position and orientation relative to the tag [16].

This kind of information can be obtained by measuring, on the tag on the camera image plane, feature points on the tag. Using this information, we can overlay augmented information that is correctly registered to the real world scene. In addition, since the tag encodes ID information, the switching of overlay information according to IDs is straightforward.

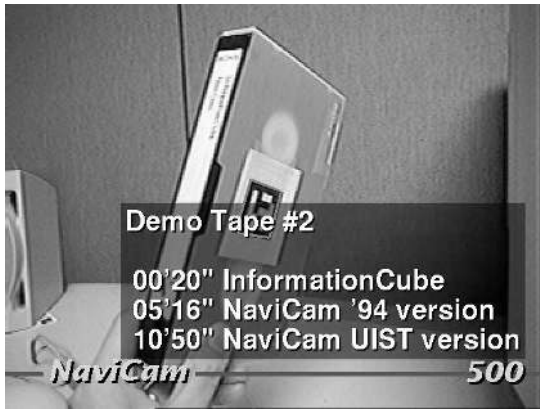


Figure 3: Physically embedded links to the digital content.



Figure 4: A stick-on type of CyberCode sheet.

### Transmitting visual tags through computer displays and TV screens

The "low-resolution" approach of CyberCode also enables a camera to recognize a visual tag displayed on a computer screen or TV display. This means that ID data can be sent through a normal TV broadcasting system. For example, a newscaster can show a flip board with CyberCode tags, and TV watchers with hand-held camera devices can retrieve the ID data from their TV screens. CyberCode ID can also be provided by way of a computer display. For example, a user can pick up the information in a CyberCode tag from a web page by using a hand-held camera, and then drop the information onto another device. Several direct-manipulation techniques (e.g., drag and drop) can thus be used across the boundaries of computers, electronic appliances, and physical objects.

### APPLICATIONS

Using CyberCode, we have developed several systems that combine digital and physical spaces. Some are still experimental, but others are already used in real-world settings. This section briefly introduces these applications and their features.

#### Physically embedded links to the digital information

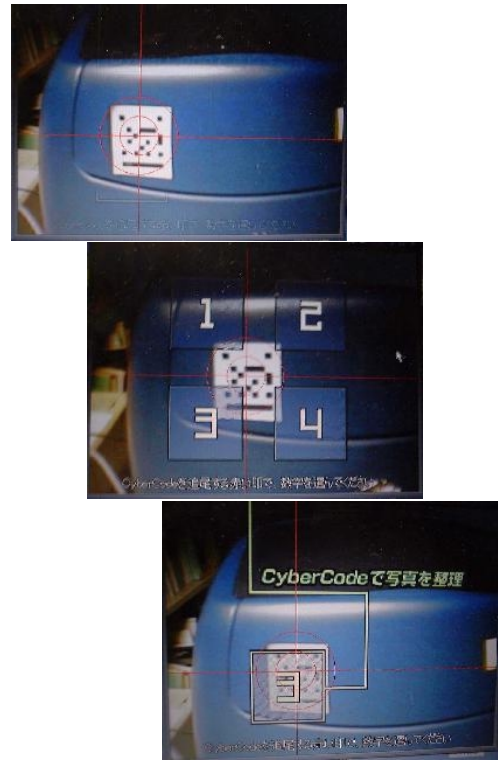


Figure 5: Physical Pop-up Menu: using a physically embedded ID to open context-sensitive menu items (left: a device with a camera approaches the ID. middle: as a result of ID recognition, menu items appear on a display. right: one item is selected by moving the device itself).

The simplest use of a CyberCode tag is as an embedded link to digital information. For example, when a CyberCode tag is attached to a document, a user can retrieve a digital version without referring to a file name or a URL (Figure 3). It is also easy to assign a unique code to each document automatically when it is printed out. When a device with a camera recognizes these IDs, a predefined action – such as opening a specific web page – launching an application, or starting a movie, is activated automatically.

We have also prepared "stick-on" CyberCode tags (Figure 4) that can be used to instantly attach an ID to object. These IDs are originally linked to nothing, and the system asks the user to define an action. Then this information is stored in a local or network database.

The use of a video camera as an ID reader also makes possible interaction techniques that use the position of an ID on the video image. Figure 5 shows one example: using a CyberCode tag to bring up menu items and then selecting one by moving the tag or the device itself. In this example, CyberCode is used as a pointing device as well as a menu ID.

Another application of combining position and ID recognition is paper "dialogue box" cards called "CyberDialogue" (Figure 6). A user can add marks on this card in order to cus-

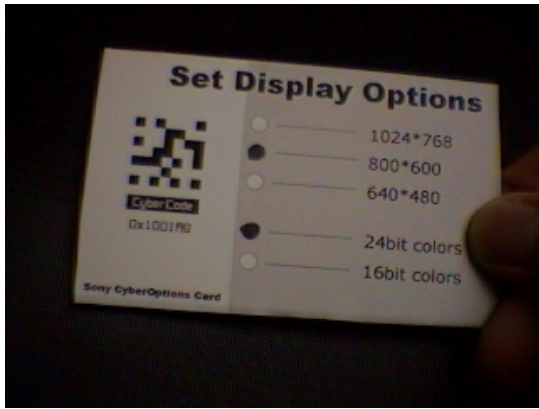


Figure 6: CyberDialogue: a paper card serving as a dialog box.



Figure 7: Browsing information space from the physical environment

tomize it. When a user puts a card in front of the camera, the system recognizes the CyberCode ID, retrieves a definition of the card, and checks the markings added by the user. The position and orientation of CyberCode is also recognized and used to collectively locate positions of markings on the card.

### Indoor navigation systems

Some museums have a navigation system that gives guidance information to visitors. These systems determine the current location of the visitor by asking the visitor to manually enter numbers, or else determine the location automatically do so by using the infrared (IR) signals from beacons. The former case is cumbersome, and for the latter case the cost for installation and maintenance of IR beacons might be a problem. CyberCode can also be used in such indoor guidance systems. If a CyberCode were printed on every label identifying the items in the museum, a visitor would be able to retrieve ID numbers, and get guidance information. If the same IDs were printed on the physical guide book, visitors could also retrieve the same information from it.

We have demonstrated this idea in several "actual" exhibitions. Figure 7, for example, is a snapshot from an exhibition in Tokyo of the works of the architect Neil Denari (September 1996) [4]. Writings, design sketches, and computer graphics by Neil Denari were virtually installed in the physical gallery space by attaching icons to the surfaces of the room. Visitors walked around the space carrying a browsing device called NaviCam, which served as a portal to the information space.

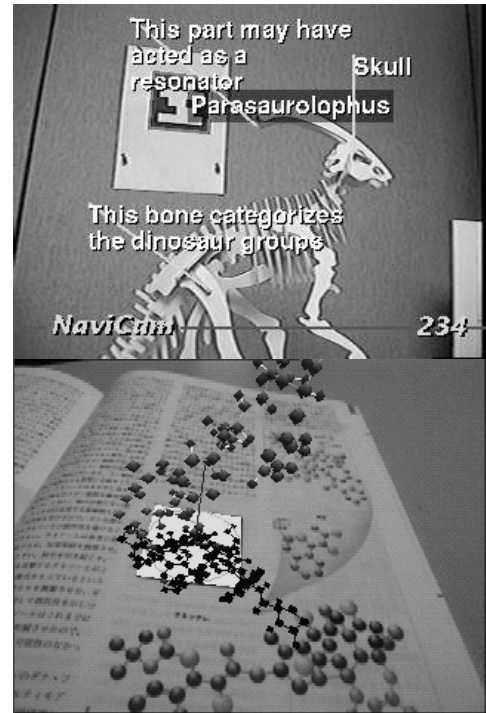


Figure 8: Examples of 3D information annotation (top: 3D annotation on a dinosaur skeleton, bottom: a virtual "pop-up" book showing a molecular model on the page.)

### Annotating the real world by using 3D information

When the CyberCode system identifies a real-world object from the attached code, the corresponding 3D-annotation



Figure 9: A gyro-enhanced camera device and a building navigation application

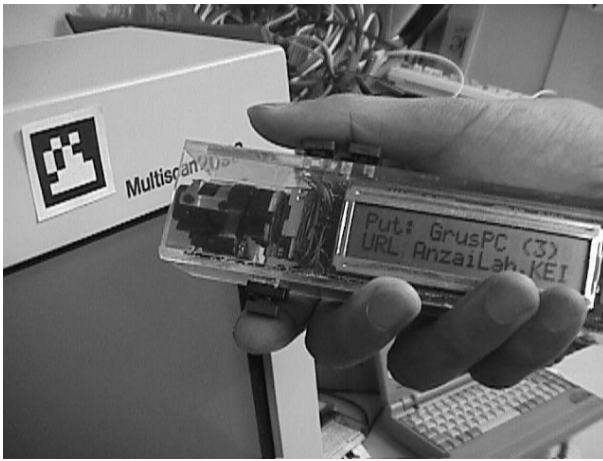


Figure 10: InfoPoint: a direct manipulation device for the physical environment

information is retrieved from a database. The estimated camera position is used to superimpose this information on the video image. The 3D annotation data is stored on the local server, and when the user first encounters a new CyberCode ID, the system automatically downloads the corresponding annotation information. Figure 8 shows some examples of information overlaid by using this technique. An algorithm for this 3D overlay is described in the “Implementation” section of this paper.

#### Combining CyberCode with other sensing technologies

CyberCode recognition technology can also be enhanced by combining it with other sensing technologies. Figure 9 shows an experimental navigation system based on a gyro-enhanced ID recognition device [15]. It demonstrates a typical use of the combination of ID recognition and spatial awareness. In this application, a user first puts the device in front of a nearby CyberCode tag on the wall. The system determines the global location, including orientation information, from the recognized ID and its shape. Then the user can freely look around the environment by moving the device. Even when the CyberCode tag is out of sight of the camera, the system continues to track the relative motion of the device by using the gyro sensor, and displays proper navigation information<sup>2</sup>.

#### Direct manipulation devices for physical environments

CyberCode can also be an “operand” for manipulating physical environments. For example, one can click on an ID, or can virtually “pick-up” an item from one ID, and drop it on another ID. These operations are an extension of the concept of direct manipulation techniques into physical environments. Since these IDs can be embedded in the real-world context, we can manipulate digital objects and physical objects in similar ways. For example, a user could perform a “drag-and-drop” operation between a paper ID and an ID attached on the real printer (or a picture of the real printer). Then a hard copy of that document would be printed. This is a more direct and natural way to specify the source and destination than in the use of traditional GUIs, particularly when the user

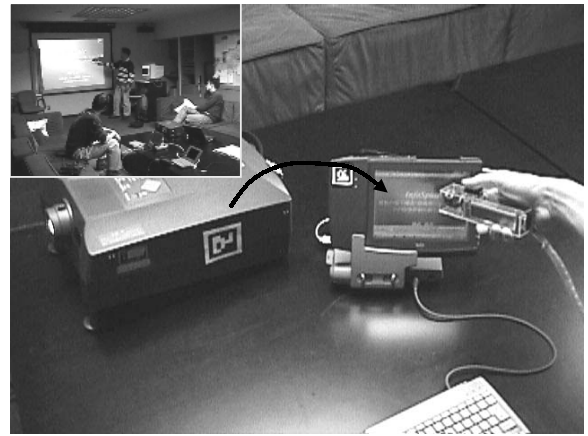


Figure 11: An example of direct-manipulation operations in a physical space: a user performs a “drag-and-drop” between a slide projector and a notebook PC in order to retrieve a currently displayed slide image.



Figure 12: An ID-aware pen is used to perform a “drag-and-drop” between physical objects.

is away from the desktop computer.

Naohiko Kohtake and the present authors have developed a hand-held direct manipulation device based on this concept [8]. Called the “InfoPoint,” (Figure 10) it is a wand type device with a CCD camera for ID recognition, buttons for operation, and a LCD display for showing information about an object. A typical use of this device is for pointing it to a physical object (actually to an attached CyberCode tag), selecting from the display one of the available actions, and activating that action by pressing a button. In this sense, the InfoPoint acts as a “universal commander” for various digital appliances: it changes its functionality automatically according to a recognized ID.

<sup>2</sup>The gyro used with this system (JAE MAX3) is a solid-state inertia tracker with three acceleration sensors and three orthogonal angular rate sensors. It is a 6-DOF tracker and thus can report x-y-z positions as well as orientations (yaw, pitch, and roll).



Figure 13: Retrieving ID information from a TV screen. After this operation, the user can drop the ID on a web browser and open the corresponding web page.



Figure 14: A scene from a TV program using CyberCode to link web information with the program.

It can also be used to perform direct manipulation techniques more complicated than this simple “point-and-click” operation, such as “drag-and-drop,” in a physical environment. The user can press a button to designate the source ID tag, then move toward a destination ID (i.e., can drag that ID), and then drop it on the destination ID tag by releasing the button. Since drag sources and destinations can be either computers (electric devices) or physical objects, this technique can be used in real-world contexts. For example, one can drag-and-drop between a projector and a notebook PC in order to transfer information about the currently projected slide to the computer (Figure 11). In this case, the projector is used as physical landmark to designate the information source, and the actual data transfer occurs in the network. Similarly, one can drag the ID tag printed on a paper document to the printer in order to make a hard copy of that document (Figure 12).

### ID transmission in TV programs

The visual nature of CyberCode allows it to be transmitted as a normal TV signal. For example, a TV screen can display a CyberCode pattern as well as a URL. Instead of manually jotting down a displayed URL, a user with a camera-equipped mobile device can simply point it at the TV screen and the device will recognize the CyberCode ID. The user can then point to the appropriate field on a web browser display, and corresponding web page will appear on the screen. One

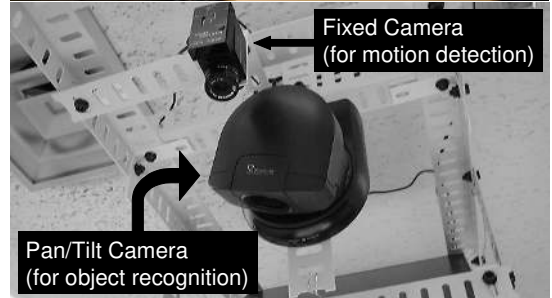


Figure 15: The Augmented Surfaces system and cameras mounted above the table.

advantage of this method is that we do not have to change any of existing broadcasting systems or TV sets. Figure 13 shows experimental ID transmission, and Figure 14 shows an example of an TV program using this technique.

### Object recognition and registration in ubiquitous computing environments

Visual tags can also be used for object identification and position determination in ubiquitous computing [24] environments. Figure 15 is an augmented meeting room, called “InfoRoom,” consisting of a digital table and a wall [19].

A camera mounted above the table is used to recognize objects on the table. Physical objects, such as VCR tapes, for example, can be used as links to the digital space. When a

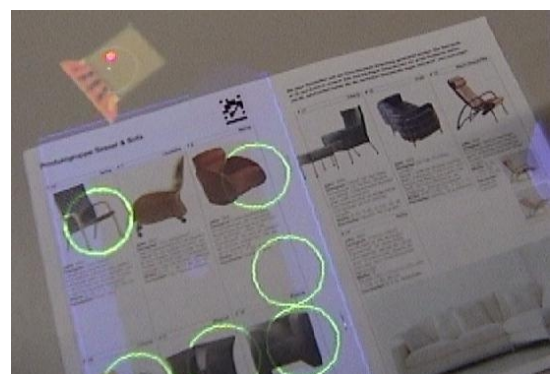


Figure 16: Retrieving information from a printed catalog

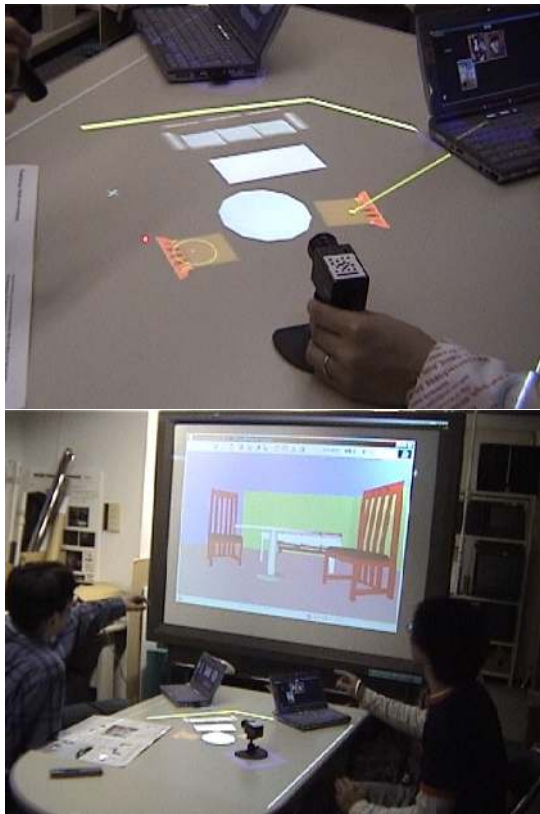


Figure 17: A mock-up camera with a CyberCode tag is used to create a 3D scene.

tagged object is placed on the table, its related information appears on the table automatically. It is also possible to dynamically “bind” digital data to the physical object by simply dragging it to that object with a pointing device such as a laser pointer or a normal mouse.

A similar technique can also be used to make a physical booklet a catalog of digital information. Figure 16 shows how a printed catalog can be used to retrieve a 3D model. Each page has an attached CyberCode tag, and the camera recognizes page position on the table as well as the page number. Then the system indicates embedded links on the page with projected images and a user can pick it up by a pointing device.

It is also possible to determine the position and orientation of physical functional objects (“phicons” [5]) placed on the table. Figure 17 shows one example: a mock-up camera that is used to create a 3D scene of interior layout simulation on the table. When position and orientation of the mock-up camera are recognized, the system creates a VRML model of the scene, and the 3D scene appears on the wall screen.

Figure 18 also shows another example of combining ID and position information. In this example, a user put a notebook computer with a CyberCode tag on the augmented table. Then, a camera mounted above the table recognize the notebook PC’s ID (i.e., IP address) and the position on the table.

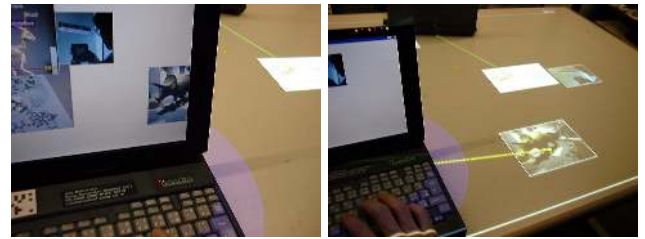


Figure 18: Hyper-Dragging: a notebook computer with a CyberCode tag attached is automatically recognized by the table-type computer, and the table surface becomes a spatial extension of the notebook computer screen.

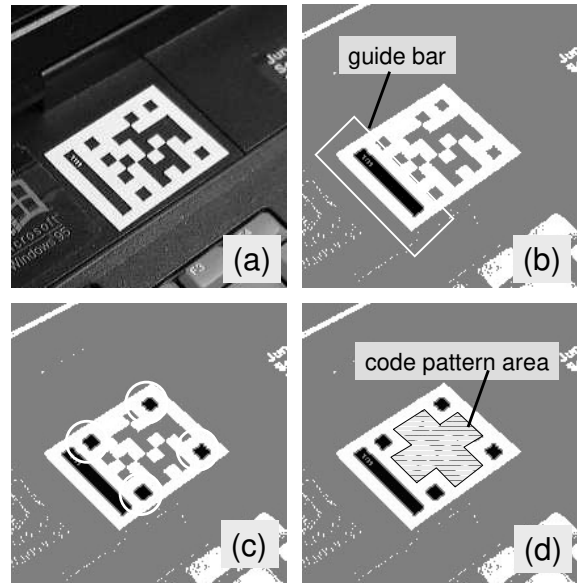


Figure 19: The visual tag recognition steps.

This recognition enables seamless information exchange between the notebook PC and the table. As shown in Figure 18, the user can manipulate a cursor of the notebook PC across the boundary of computers: the user can grab an object on the notebook PC and drag it to the table surface.

Recognition targets are not limited to physical objects. When a notebook PC with an attached CyberCode tag is recognized, the system makes an ad-hoc network connection between the PC and the table. The table surface thereby becomes an extended workspace for the notebook PC. For example a user can drag an item on the notebook PC and transfer it to the table to by moving the cursor across the boundary between the notebook PC and the table. This technique, called “hyper-dragging” is another example of how a direct manipulation technique can be extended into the physical space.

To enable these examples, resolution of consumer-level video cameras are not enough for cover the entire table surface. We use the combination of two cameras for virtually achieving higher-resolution of wider viewing area. The first camera is a fixed camera that is always looking at the entire table surface. This camera detects changes on the table by analyzing the dif-

ference between two consecutive video images. It determines which area has been changed and sends an "area changed" sign to the second camera, which is a computer-controlled pan/tilt camera that can zoom in on the changed area.

### Implementation

In this section we describe the internal details of the augmented reality applications described in the previous section. The CyberCode recognition algorithm consists of two parts, one that recognizes the ID of a tag, and one that determines the 3D position of the tag (in relation to the camera).

### Visual tag recognition algorithm

The CyberCode tag ID is recognized in the following five steps (Figure 19).

- (i) Binarizing the image. We are using an adaptive binarization method [25].
- (ii) Selecting the connected regions that have a specific second-order moment. These regions become candidate guide bars for the tag.
- (iii) Searching for the four corners of the marker region using positions and orientations of guide bars found in step ii.
- (iv) When the guide bar and the four corners are found, the system decodes the bitmap pattern in the tag. Using the positions of the corners of the marker, the system estimates and compensates for the distortion effect caused by camera/object tilting.
- (v) Decoding the code bit pattern. After checking for the error bits, the system determines whether or not the image contains a correct CyberCode.

### 3D position reconstruction algorithm

The recognized code frame is also used for estimating the position and orientation of the camera. From four known points on the image plane, it is possible to calculate a matrix representing the translation and rotation of the camera in a real-world coordinate system. We use the four corners of the CyberCode tag as these reference points.

To ensure that the estimated coordinate system is orthogonal, the algorithm also minimizes the following constraint during estimation:

$$E = (\vec{v}_0 \cdot \vec{v}_1)^2 + (\vec{v}_1 \cdot \vec{v}_2)^2 + (\vec{v}_2 \cdot \vec{v}_3)^2 + (\vec{v}_3 \cdot \vec{v}_0)^2 + (\vec{v}_4 \cdot \vec{v}_5)^2 \rightarrow \min$$

where  $\vec{v}_0 \dots \vec{v}_3$  are the orientation vectors of the four edges, and  $\vec{v}_4 \dots \vec{v}_5$  are the two diagonals of the code frame. Since we can represent  $\vec{v}_0 \dots \vec{v}_5$  by using  $\vec{n}$ , which is a vector normal to the matrix code plane, we can replace with the above equation:

$$E(\vec{n}) \rightarrow \min.$$

We use the downhill simplex method [14] to estimate the  $\vec{n}$  that minimizes  $E$ . Once  $\vec{n}$  is calculated, we can use it to recalculate vectors  $\vec{v}_0 \dots \vec{v}_5$ . A point  $(x, y, z)^T$  in the real world corresponds to the point  $(X, Y, Z)^T$  in the camera coordinate system :

	Visual Tags	RF Tags	IR Tags
Printable	yes	no	no
Line-of-sight	required	not required	required
battery	no	no	required
rewritable	no	yes/no	yes/no

Table 1: Features of tagging systems.

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} \vec{e}_x & \vec{e}_y & \vec{e}_z & \vec{e}_t \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

where  $\vec{e}_x, \vec{e}_y, \vec{e}_z$  represent camera rotation coefficients, and  $\vec{e}_t$  represents camera motion:

$$\begin{aligned} \vec{e}_x &= N(\vec{v}_5 - \vec{v}_4), \\ \vec{e}_y &= N(\vec{v}_4 + \vec{v}_5), \\ \vec{e}_z &= N(\vec{n}), \\ \vec{e}_t &= dist \times N(\vec{p}), \end{aligned}$$

where  $dist$  is the distance from the camera center to the center of the matrix code,  $\vec{p}$  is a vector from the camera center to the center of the matrix code on the image plane, and  $N(\vec{v})$  is a normalization function.

Once the transformation matrix is known, it is easy to overlay spatially correct annotation information and computer graphics on the real world video images. It is also used to find marking positions on the paper card (Figure 6).

### Performance

Based on the above algorithm, A workstation class computer (such as the SGI-O2) using two-part CyberCode recognition algorithm can recognize the code in real time (frames/sec), a rate making feasible several of the 3D composition applications described in the previous section. On mobile PCs (such as the Sony VAIO-C1 with a 200MHz Mobile Pentium MMX, the processing rate is about 15 frames/sec, and on the Mitsubishi AMITY pen computer (with a 50MHz Intel 486) it is 5 frames/sec. Because of the small bandwidth of the connection between the camera and the computer (i.e., PCMCIA), most of the processing time is devoted to simply transferring the video data to the memory. And more than the half of the actual image-processing time is consumed in labeling connected regions of pixels. This implies that performance could be greatly improved by using a computer vision hardware.

### Application programmer interfaces

To make it possible for programmers to develop augmented reality applications quickly, we provide a set of Java classes to control cameras and CyberCode recognition engine. Using Java native method invocation (JNI) mechanism, these classes "wrap" low-level image processing code (which is written in C) and free application programmers from having to deal with the details of image processing. Many of the systems described in the "Applications" section have been developed with this Java interface.



## DISCUSSION

### Comparison with other tagging technologies

As described in the introduction section, there are other possibilities than visual tagging system. Radio-frequency (RF) tags are becoming popular and do not require "line of sight" detection (one can embed them in an object). These tags are not printable, however, so they cannot be used with paper documents. Table 1 summarizes the features of various tagging technologies.

Another alternative is a 1D barcode. The main advantage of using such a tag is that we might be able to exploit existing barcode systems [18], such as EAN for product numbers and ISBN for publishing. Thus, if an application should handle a large number of existing products, it would be better off using a 1D barcode than a more exotic tagging technology. For example, we have developed a bulletin board system based on product barcodes. This system, called the "ThingsBoard," allows users to open a corresponding bulletin board of the Internet by scanning a product barcode<sup>3</sup>. Another example is the "Pocket Bargain Finder" [3], which is a hand-held device with a barcode reader for retrieving price information from actual products.

A drawback of these alternatives, however, is that they require special reader devices. In addition, since these readers normally can only provide ID information, applications that need position information such as those illustrated in Figures 6, 8, 16, 17 and 18 should also use another sensing device. CyberCode uses as an ID reader a normal camera originally designed for taking still pictures or making movies. We think this is an advantage for a tagging system using mobile devices, which are subject to severe size and weight constraints. Some barcode readers emit a laser beam so that they can read codes that are not close to the reader. This, however, makes difficult to use these devices in home environments.

Visibility and social acceptance are other issues. A system designer using RF tags has a great deal of freedom in designing systems because the tags can be embedded in objects. And a 1D barcode makes a rather "industrial" impressions on users, so it might not be suitable for consumer or home applications. CyberCode is between the two, and some users recognize it as a "real world icon" that helps a user to recognize the existence of real-virtual links. We are also exploring several visual design alternatives for use in more entertainment-oriented domains.

### RELATED WORK

Ariel [11] and transBOARD [5] support connections between digital contents and barcode-printed documents or cards. Insight Lab [9] is a computer-supported meeting room that extensively uses barcoded tags as physical-digital links and commands. The "Palette" system uses barcode-printed cards for organizing presentation slides [12]. We think these kinds of applications can be implemented easily, and without requiring dedicated ID readers, by using CyberCode. For ex-

ample, a document camera used for teleconferencing could also be used to select slide IDs.

XAX [7] uses a printed code to embed digital information in paper documents, and a scanning device recognizes the document ID and thus document formats. Our "CyberDialogue" paper-card interface described in this paper is also based on this kind of idea, but it can also be used in more mobile environments. That is, a hand-held camera can be used to recognize the card IDs: users don't have to insert the cards into a fax or a scanner.

Some AR applications [1, 20, 21] also use "colored" visual tags, and our earlier prototype [18]. In our experience, however, using color in the real world is both difficult and unreliable. Lighting conditions change too much, and the color sensitivity of various CCD devices are very different. We thus use only monochrome tags like those in other 2D barcode systems. We expect that in the near future, most of the circuitry needed for processing the recognition algorithm can be embedded in an image-sensing device.

### CONCLUSION

In this paper we introduced CyberCode, a visual tagging system designed for augmented reality, by describing a series of examples. Some of these applications have been "commercialized" as a bundled software tool on Sony notebook PC with a built-in camera. We think that CyberCode is an affordable and flexible technology that can be used to construct a wide variety of augmented reality systems. Even when some other tagging technology is more suitable for a particular application, CyberCode often works as a good "first approximation" and helps system designers to prototype their ideas quickly.

### ACKNOWLEDGMENTS

We thank Taka Sueyoshi, Keigo Ihara, Makoto Nijjima, Tota Hasegawa, Hidenori Karasawa and Takao Yoshimine for making a "product" version of CyberCode. "InfoPoint" was developed by Naohiko Kohtake while he was a student intern at the Sony Computer Science Laboratory. We are in debt to the members of Sony CSL Interaction Laboratory for extensive discussions and to Toshi Doi and Mario Tokoro for supporting our research.

### REFERENCES

1. Toshifumi Arai, Kimiyoshi Machii, Soshiro Kuzunuki, and Hiroshi Shojima. InteractiveDESK: a computer-augmented desk which responds to operations on real objects. In *CHI'95 Conference Companion*, pages 141–142, 1995.
2. Yuji Ayatsuka, Jun Rekimoto, and Satoshi Matsuoka. UbiquitousLinks: Hypermedia links embedded in the real world. In *IPSJ SIGHI Notes*, number 67-4, pages 23–30. Japan Information Processing Society, July 1996. in Japanese.
3. A. Brody and E. Gottsman. Pocket BargainFinder: A disruptive technology for augmented com-

<sup>3</sup><http://www.csl.sony.co.jp/projects/thingsboard/>

- merce. In *Proc. of Handheld and Ubiquitous Computing (HUC'99)*, 1999.
4. Neil Denari. *INTERRUPTED PROJECTIONS*. TOTO Shuppan, 1996.
  5. Hiroshi Ishii and Brygg Ullmer. Tangible Bits: Towards seamless interfaces between people, bits and atoms. In *CHI'97 Proceedings*, pages 234–241, 1997.
  6. Siio Itiro, Masui Toshiyuki, and Fukuchi Kentaro. Real-world interaction using the FieldMouse. In *UIST'99*, pages 113–119, 1999.
  7. Walter Johnson, Herber Jellinek, Leigh Klotz, Ramana Rao, and Stuart Card. Bridging the paper and electronic worlds: the paper user interface. In *INTERCHI'93 Conference Proceedings*, pages 507–512, 1993.
  8. Naohiko Khotake, Jun Rekimoto, and Yuichiro Anzai. InfoStick: An interaction device for inter-appliance computing. In *Workshop on Handheld and Ubiquitous Computing (HUC'99)*, 1999.
  9. Beth M. Lange, Mark A. Jones, and James L. Meyers. Insight Lab: An immersive team environment linking paper, displays, and data. In *CHI'98 Proceedings*, pages 550–557, 1998.
  10. Sue Long, Dietmar Aust, Gregory Abowd, and Chris Atkeson. Cyberguide prototyping context-aware mobile applications. In *CHI'96 conference companion*, page 293, 1996.
  11. W.E. Mackay, D.S. Pagani, L. Faber, B. Inwood, P. Louniainen, L. Brenta, and V. Pouzol. Ariel: augmenting paper engineering drawings. In *CHI'95 Conference Companion*, pages 420–422, 1995.
  12. Les Nelson, Satoshi Ichimura, Elin Ronby Pedersen, and Lia Adams. Palette: A paper interface for giving presentations. In *CHI'99 Proceedings*, pages 354–361, 1999.
  13. Joseph Paradiso and Kai-Yuh Hsiao. Swept-frequency, magnetically-coupled resonant tags for realtime, continuous, multiparameter control. In *CHI'99 extended abstracts*, pages 212–213, 1999.
  14. W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1992.
  15. Jun Rekimoto. The magnifying glass approach to augmented reality systems. In *International Conference on Artificial Reality and Tele-Existence '95 / Conference on Virtual Reality Software and Technology '95 (ICAT/VRST'95) Proceedings*, pages 123–132, November 1995.
  16. Jun Rekimoto. Matrix: A realtime object identification and registration method for augmented reality. In *Proc. of Asia Pacific Computer Human Interaction (APCHI'98)*, July 1998.
  17. Jun Rekimoto, Yuji Ayatsuka, and Kazuteru Hayashi. Augment-able Reality: situated communication through physical and digital spaces. In *Proc. of the Second international symposium on wearable computers (IEEE ISWC'98)*, pages 68–75, 1998.
  18. Jun Rekimoto and Katashi Nagao. The world through the computer: Computer augmented interaction with real world environments. In *Proceedings of UIST'95*, pages 29–36, November 1995.
  19. Jun Rekimoto and Masanori Saitoh. Augmented Surfaces: A spatially continuous workspace for hybrid computing environments. In *Proceedings of ACM CHI'99*, pages 378–385, May 1999.
  20. Thad Starner, Steve Mann, Bradley Rhodes, Jeffrey Levine, Jennifer Healey, Dana Kirsch, Rosalind W. Picard, and Alex Pentland. Augmented reality through wearable computing. *Presence: Teleoperators and Virtual Environments*, (4):386–398, 1997.
  21. John Underkoffler and Hiroshi Ishii. Illuminating Light: An optical design tool with a luminous-tangible interface. In *CHI'98 Proceedings*, pages 542–549, 1998.
  22. R. Want, A. Hopper, V. Falcao, and J. Gibbons. The active badge location system. *ACM Trans. Inf. Syst.*, January 1992.
  23. Roy Want, Kenneth P. Fishkin, Anuj Gujar, and Beverly L. Harrison. Bridging physical and virtual worlds with electronic tags. In *CHI'99 Proceedings*, pages 370–377, 1999.
  24. Mark Weiser. The computer for the twenty-first century. *Scientific American*, pages 94–104, September 1991.
  25. Pierre Wellner. Interacting with paper on the DigitalDesk. *Communication of the ACM*, 36(7):87–96, August 1993.