

Tangible Augmented Reality

Mark Billinghurst

HIT Laboratory
University of Washington
Box 352-142, Seattle
WA 98195, USA
grof@hitl.washington.edu

Hirokazu Kato

Faculty of Information Sciences
Hiroshima City University
3-4-1, Ozaka-higashi, Asaminami-ku
Hiroshima 731-3194, JAPAN
kato@sys.im.hiroshima-cu.ac.jp

Ivan Poupyrev

Interaction Lab
Sony CSL
3-14-13 Higashi-Gotanda
Tokyo 141-0022, JAPAN
poup@csl.sony.co.

ABSTRACT

This paper advocates a new metaphor for designing three-dimensional Augmented Reality (AR) applications, Tangible Augmented Reality (Tangible AR). Tangible AR interfaces combine the enhanced display possibilities of AR with the intuitive manipulation and interaction of physical objects or Tangible User Interfaces. We define what Tangible AR interfaces are, present some design guidelines and prototype interfaces based on these guidelines. Experiences with these interfaces show that the Tangible AR metaphor supports seamless interaction between the real and virtual worlds, and provides a range of natural interactions that are difficult to find in other AR interfaces.

CR Categories: H.5.2 [User Interfaces] Input devices and strategies; H.5.1 [Multimedia Information Systems] Artificial, augmented, and virtual realities.

Keywords: Augmented Reality, Collaboration, CSCW.

Additional Keywords: Tangible User Interfaces

1 INTRODUCTION

In 1965 Ivan Sutherland built the first head mounted display and used it to show a simple wireframe cube overlaid on the real world [1], creating the first Augmented Reality (AR) interface. We use the term Augmented Reality to mean interfaces in which three-dimensional computer graphics are superimposed over real objects, typically viewed through headmounted or handheld displays.

Today, computers are incomparably faster and graphics are almost lifelike, but in many ways AR interfaces are still in their infancy. As Ishii says, the AR field has been primarily concerned with “..considering purely visual augmentations” [2] and while great advances have been made in AR display technologies and tracking techniques, *interaction* with AR environments has been usually limited to either passive viewing or simple browsing of virtual information registered to the real world. Few systems provide tools that let the user interact, request or modify this information effectively and in real time. Furthermore, even basic interaction tasks, such as manipulation, copying, annotating, dynamically adding and deleting virtual objects to the AR scene have been poorly addressed.

In this paper we advocate a new approach to designing AR interfaces that we refer to as Tangible Augmented Reality (Tangible AR). Tangible AR interfaces are those in which 1) each virtual object is registered to a physical object (Figure 1) and 2) the user interacts with virtual objects by manipulating the corresponding tangible objects. In the Tangible AR approach the physical objects and interactions are equally as important as the virtual imagery and provide a very intuitive way to interact with the AR interface.

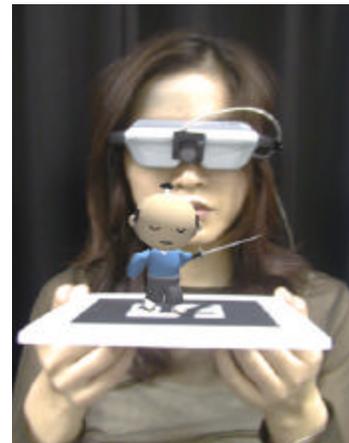


Figure 1: The user interacts with 3D virtual objects by manipulating a tangible object; such as a simple paper card.

Our work derives large part of its inspiration from Ishii's Tangible Media group [3]. The many projects developed by this group are developed around the notion of the Tangible User Interface (TUI) in which real world objects are used as computer input and output devices, or as Ishii puts it “by coupling digital information to everyday physical objects and environments” [2]. The Tangible AR approach builds on the principles suggested by TUI by coupling an AR visual display to a tangible physical interface.

In the remainder of this paper we first review related work and describe the need for new AR interface metaphors. We then describe our notion of Tangible AR in more detail and outline some Tangible AR design principles. Next we show how these design principles have been applied in a variety of prototype applications and describe the technology involved.

Note that some of the interfaces described have previously been written about in other publications. However this is the first time that they have been used together as examples of the Tangible AR design approach. This paper aims to generalize scattered results, report new work, and present the conceptual vision that has been driving our exploration.

2 BACKGROUND

When a new interface medium is developed it typically progresses through the following stages:

1. Prototype Demonstration
2. Adoption of Interaction techniques from other interface metaphors
3. Development of new interface metaphors appropriate to the medium
4. Development of formal theoretical models for predicting and modeling user interactions

For example, the earliest immersive Virtual Reality (VR) systems were used to just view virtual scenes. Then interfaces such as 3DM [4] explored how elements of the desktop WIMP metaphor could be used to enable users to model immersively and support more complex interactions. Next, interaction techniques such as the Go-go [5] or World in Miniature [6] were developed which are unique to Virtual Reality. Most recently, researchers are attempting to arrive at a formal taxonomy for characterizing interaction in virtual worlds that would allow developers to build 3D virtual interfaces in a systematic manner [7].

In many ways AR interfaces have barely moved beyond the first stage. The earliest AR systems were used to view virtual models in a variety of application domains such as medicine [8] and machine maintenance [9]. These interfaces provided a very intuitive method for viewing three-dimensional information, but little support for creating or modifying the AR content. More recently, researchers have begun to address this deficiency. The AR modeler of Kiyokawa [10] uses a magnetic tracker to allow people to create AR content, while the Studierstube [11] and EMMIE [12] projects use tracked pens and tablets for selecting and modifying AR objects. More traditional input devices, such as a hand-held mouse or tablet [13][14], as well as intelligent agents [15] have also been investigated. However these attempts have largely been based on existing 2D and 3D interface metaphors from desktop or immersive virtual environments. This means that in order to interact with virtual content the user uses a special purpose input device and the number of physical input devices are limited.

In our work we are trying to develop interface metaphors that enable AR applications to move beyond being primarily 3D information browsers. In Augmented Reality there is an intimate relationship between 3D virtual models and physical objects these models are attached to. This suggests that one promising research direction may arise from taking advantage of the immediacy and familiarity of

everyday physical objects for effective manipulation of virtual objects.

For over a decade researchers have been investigating computer interfaces based on real objects. We have seen the development of the Digital Desktop [16], ubiquitous computing [17] and Tangible User Interfaces (TUI) [2] among others. The goal of these efforts is to make the computer vanish into familiar real world objects or the environment. As with VR interfaces, recent Tangible User Interfaces employ unique interface metaphors in areas such as 3D content creation [18], and work has begun on developing and testing models of user interaction [19].

Tangible interfaces are extremely intuitive to use because physical object manipulations are mapped one-to-one to virtual object operations, and they follow a space-multiplexed input design [19]. In general input devices can be classified as either space- or time-multiplexed. With a space-multiplexed interface each function has a single physical device occupying its own space. In contrast, in a time-multiplexed design a single device controls different functions as different points in time. The mouse in a WIMP interface is a good example of a time-multiplexed device. Space-multiplexed devices are faster to use than time-multiplexed devices because users do not have to make the extra step of mapping the physical device input to one of several logical functions. In most manual tasks space-multiplexed devices are used to interact with the surrounding physical environment.

Although intuitive to use, with TUI interfaces information display can be a challenge. It is difficult to dynamically change an object's physical properties, so most information display is confined to image projection on objects or augmented surfaces. In those Tangible interfaces that use three-dimensional graphics there is also often a disconnect between the task space and display space. For example, in the Triangles work [20], physical triangles are assembled to tell stories, but the visual representations of the stories are shown on a separate monitor distinct from the physical interface. Presentation and manipulation of 3D virtual objects on projection surfaces is difficult [21], particularly when trying to support multiple users each with independent viewpoints. Most importantly, because the information display is limited to a projection surface, users are not able to pick virtual images off the surface and manipulate them in 3D space as they would a real object.

So we see that current Tangible interfaces provide very intuitive manipulation of digital data, but limited support for viewing 3D virtual objects. In contrast current AR interfaces provide an excellent interface for viewing virtual models, but limited support for interaction and space-multiplexed input devices. We believe that a promising new AR interface metaphor can arise from combining the enhanced display possibilities of Augmented Reality with the intuitive

manipulation of Tangible User Interfaces. We call this combination Tangible Augmented Reality. In the next section we show how Tangible AR supports seamless interaction, and provide some design guidelines.

3 TANGIBLE AUGMENTED REALITY

The goal of computer interfaces is to facilitate seamless interaction between a user and their computer-supported task. In this context, Ishii defines a seam as a discontinuity or constraint in interaction that forces the user to shift among a variety of spaces or modes of operation [22]. Seams that force a user to move between interaction spaces are called *functional* seams, while those that force the user to learn new modes of operation are *cognitive* seams.

In the previous section we described how Tangible User Interfaces provide seamless interaction with objects, but may introduce a discontinuity or *functional* seam between the interaction space and display space. In contrast most AR interfaces overlay graphics on the real world interaction space and so provide a spatially seamless display. However they often force the user to learn different techniques for manipulating virtual content than from normal physical object manipulation or use a different set of tools for interacting with real and virtual objects. So AR interfaces may introduce a *cognitive* seam.

A Tangible AR interface provides true spatial registration and presentation of 3D virtual objects anywhere in the physical environment, while at the same time allowing users to interact with this virtual content using the same techniques as they would with a real physical object. So an ideal Tangible AR interface facilitates seamless display and interaction, removing the functional and cognitive seams found in traditional AR and Tangible User Interfaces. This is achieved by using the design principles learned from TUI interfaces, including:

- The use of physical controllers for manipulating virtual content.
- Support for spatial 3D interaction techniques (such as using object proximity).
- Support for both time-multiplexed and space-multiplexed interaction.
- Support for multi-handed interaction.
- Support for Matching the physical constraints of the object to the requirements of the interaction task.
- The ability to support parallel activity where multiple objects are being manipulated.
- Collaboration between multiple participants

Our central hypothesis is that AR interfaces that follow these design principles will provide completely seamless interaction with virtual content and so will be extremely intuitive to use. In the next section we describe some prototype interfaces that support this hypothesis.

4 TANGIBLE AR INTERFACES

In order to explore the Tangible AR design space we have developed the following prototype interfaces:

Space-Multiplexed Interfaces

Shared Space: A collaborative game

ARgroove: A music performance interface

Tiles: A virtual prototyping application

Time-Multiplexed Interfaces

VOMAR: A scene assembly application

In this section we briefly describe these interfaces, showing how the Tangible AR design principles have been applied. Although each of these interfaces has been developed for a particular application the interaction principles that we have explored in them are generic and can be broadly applied.

SHARED SPACE Space-Multiplexed Interaction in AR

The Shared Space interface is an example of how Tangible AR principles can be used to design simple yet effective multiple user space-multiplexed AR interfaces [23]. The Shared Space was designed to create a collaborative AR game that could be used with no training. Several people stand around a table wearing Olympus HMDs with cameras attached (figure 2). The video from the camera is feedback into the HMD to give them a video-mediated view of the world with 3D graphics overlaid. On the table are cards and when these are turned over, in their HMDs the users see different 3D virtual objects appearing on top of them. The users are free to pick up the cards and look at the models from any viewpoint.



Figure 2: Shared Space: the insert shows user view

The goal of the game is to collaboratively match objects that logically belonged together. When cards containing correct matches are placed side by side an animation is triggered (figure 3). For example, when the card with the UFO on it is placed next to the alien, the alien appears to jump into the UFO and start to fly around the Earth.



Fig 3: The Spatial Interaction Based on Proximity

Although is a very simple application it provides a good test of the usefulness of the tangible interface metaphor for manipulating virtual models. Shared Space is essentially virtual model viewing software that support six degree of freedom viewing with a simple proximity based interaction. However, rather than using a mouse or magnetic tracker to manipulate the model, users just hold cards. The form of the cards encourage people to manipulate them the same way they would use normal playing cards, such as turning them over, rotating them, holding them in the hands, passing them to each other and placing them next to each other. They can hold them in either hand and many cards can be uncovered at once to show a number of different objects.

Shared Space has been shown at Siggraph 99 and Imagina 2000 and the thousands of people that have tried it have had no difficulty with the interface. Users did not need to learn any complicated computer interface or command set and they found it natural to pick up and manipulate the physical cards to view the virtual objects from every angle. Players would often spontaneously collaborate with strangers who had the matching card they needed. They would pass cards between each other, and collaboratively view objects and completed animations. By combining a tangible object with virtual image we found that even young children could play and enjoy the game.

After the Imagina 2000 experience 157 people filled out a short user survey, answering the following questions on a scale of one to seven (1=not very easily, 7=very easily):

Q1: How easily could you play with other people?

Q2: How easily could you interact with the virtual objects?

Table 2 summarizes the results. Users felt that they could very easily play with the other people (5.64) and interact with the virtual objects (5.62). Both of these are significantly higher than the neutral value of 3.5; the t-test value row showing the results from a one-tailed t-test.

	Q1 (n=132)	Q2 (n=157)
Average	5.64	5.62
Std Dev.	1.19	1.20
t-test val.	237.09, p<0.01	278.74, p<0.01

Table 2: Shared Space Survey Results

When users were asked to comment on what they liked most about the exhibit, interactivity, how fun it was, and ease of user were the most common responses. Perhaps more interestingly, when asked what could be improved, people thought that reducing the tracking latency, improving image quality and improving HMD quality were most important.

ARGROOVE: 3D AR Interface for Interactive Music

ARGroove is a tangible AR musical interface that allows people to control interactive electronic musical compositions individually or in a group. Unlike the previous interface ARGroove uses more complex 3D physical motion to control a non-visual modality, and it uses 3D widgets for AR displays.

In the ARGroove music is constructed from a collection of short looped elements, each carefully composed to fit others so they can be interactively mixed. For each individual loop a composer assigns filters and effects, to allow the user to interactively modulate them.

The interface consists of a number of real LP records with markers placed on a table in front of a projection screen. An overhead camera captures the image of the table and shows it on the screen with additional 3D widgets overlaid on each card (figure 7). The user can play, remix and modulate musical elements by picking up and manipulating the real records in space which serve both as physical *musical containers*, grouping together related elements of musical composition, and *tangible AR 3D controllers*, that allow users to interactively modulate music, mix and fade between music pieces. To start playing a musical element, the user simply flips over the associated record so that the overhead camera can identify it and start playing corresponding musical sequence. The user then modifies the sound by translating the record up and down, or rotating and tilting it (Figure 8), these 3D motions are mapped into corresponding modulation functions, e.g. pitch, distortion, amplitude, filter cut-off frequency, and delay mix. Since the system tracks and recognizes several records at the same time, users can play several musical elements simultaneously and collaboratively.



Figure 7: Two users are playing music in the Augmented Groove: the insert shows the view on the projector screen

A three-dimensional virtual controller is overlaid on top of each of the records providing the user with instant visual feedback on the state and progression of musical performance. For each control dimension a corresponding graphical element changes depending on the value of the control (figure 8). For example, as the user raises the record a pyramid in the middle also goes up and when the musical control reaches limit a small, animated character pops up cuing the user. Although virtual controls are not absolutely necessary to control music, they are essential to make the system easier and more enjoyable to use.

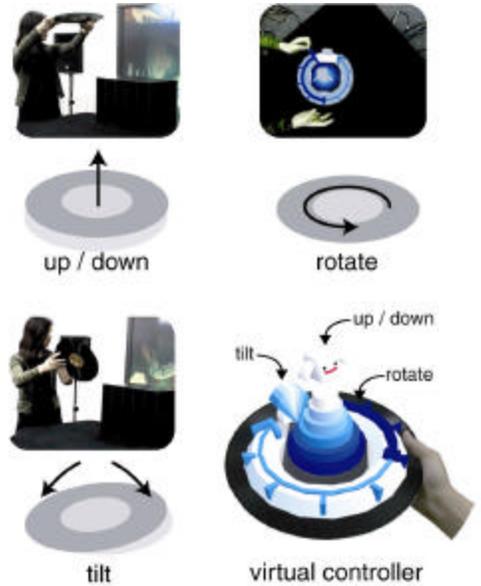


Figure 8: Gestural musical interface in Augmented Groove

ARGroove was demonstrated at SIGGRAPH 2000 where users could perform an electronic composition using three records. Over 1500 people tried the experience and the majority found it very enjoyable. One of the reasons what was that the range of possible musical variations afforded was very large and the ability to simultaneously control multiple sound effects and filters resulted in a very expressive and enjoyable interface. Only a short explanation was usually sufficient for them to be able to effectively control the musical performance. We interviewed 25 users and of these 92% rated the experience as “Excellent”. In terms of ease of learning 40% rated it as “Excellent”, while 52% said it was “Okay” and only 8% as “Poor”.

TILES: Putting Everything Together

Tiles is an AR authoring interface that explores how more complicated behaviors can be supported, including copying, pasting, deleting, and browsing virtual information in AR settings. In Shared Space all of the cards possessed the same functionality, whereas in Tiles we further explore space-multiplexed control by assigning different behaviors to different objects, creating tangible 3D widgets that we first explored in ARGroove. We distribute functionality

across tangible AR widgets (that we called *tiles*) letting the user to choose operation simply by picking a needed tile.

The application domain is rapid prototyping for aircraft instrument panels. The interface consists of a metal whiteboard, a book, and two stacks of magnetic tiles (approximately 15cm x 15cm). Sitting in front of the whiteboard the user wears a lightweight high resolution Sony Glasstron HMD with a video camera attached (fig. 4).



Figure 4: Using the Tiles Interface

The various tangible elements of the interface serve a different purpose. The whiteboard is the working space where users can layout virtual aircraft instruments. The book serves as a *menu object*, and when the user looks through its pages they will see a different virtual instrument model on each page. One stack of tiles serve as *data tiles* and shows no virtual content until virtual objects are copied onto them. The remaining tiles are *operator tiles* and are used to perform basic operations on the data tiles. There is a unique tile for each operation and currently supported operations include *deletion*, *copying* and a *help* function. Each of the operations tiles has a different three-dimensional virtual icon on them to show what their function is and tell them apart from the data tiles (fig 5).



Trashcan delete widget

Talking head help widget

Figure 5: Virtual Widgets on Operator Tiles

As with the Shared Space interface, virtual images appear attached to the physical objects and can be picked up and looked at from any viewpoint. Interaction between objects is also based on physical proximity, however the operation that is invoked by bringing objects next to each other depends on their semantic. For example, to copy a virtual instrument from the menu book to an empty data tile, the tile

is just placed by the appropriate book page. However, touching a data tile that contains a virtual instrument with the trashcan delete tile, removes the virtual instrument, while putting the help tile beside it displays a help message (fig 5). Once virtual instruments have been placed on the data tiles, these can be attached to the whiteboard to layout a prototype virtual instrument panel (figure 6).



Figure 6: Virtual Instrument Panel view by the user

The main difference between this interface and the previous interfaces is the use of different shaped physical objects for different interface properties, and assigning different semantics to different objects. Supporting one interface function per object is similar to the interface models of desktop GUI interfaces, where each icon and tool has unique functionality. Despite this added functionality, the physical interactions are still based on object manipulation and proximity, showing that quite complex AR interfaces can be built from simple physical interactions. The use of different objects for different functions further emphasizes the space-multiplexed nature of the interface.

We have not evaluated the Tiles interface in a formal manner, however we have demonstrated it at the ISAR 2000 conference where over 70 people tried it. Once again these users had no difficulty with the interface, although the wider range of interface functions typically meant that they needed to be shown how the different operator tiles worked before they tried it themselves.

VOMAR – Time-Multiplexed Interaction in AR

The VOMAR project explores how a time-multiplexed Tangible AR interface could be designed. VOMAR uses a single input device that allows the user to perform multiple different tasks in a virtual scene assembly application. To achieve this we explored how complex physical gestures can be used to support natural and effective interaction.

The physical components of the interface comprise a real book, a cardboard paddle the user holds in their hand, a large piece of paper and a lightweight HMD the user wears (figure 10 (a)). As before the form of each of these objects reflects their function; the book serves as a container holding all the virtual models, the paddle is the main interaction device, and the large piece of paper the workspace.

The application is layout of virtual furniture in a room, although the same interface can be applied to many domains. When the user opens the book on each of its pages they see a different set of virtual furniture, such as a set of chairs, rugs etc (fig 10 (b)). The 3D virtual models appear exactly superimposed over the real book pages. Looking at the large piece of paper they see an empty virtual room. They can then copy and transfer objects from the book to the virtual room using the paddle (fig 10 (c,d)).

The paddle is the main interaction device and it is a simple object with an attached tracking symbol. It is designed to be used by either hand and allows the user to make static and dynamic gestures to interact with the virtual objects:

Static

1. Paddle proximity to object
2. Paddle tilt/inclination (e.g. fig 10 (d))

Dynamic

1. Shaking (side to side motion of paddle)
2. Hitting (up and down motion of paddle)
3. Pushing object (fig 10 (e))



(a) The VOMAR interface



(b) A virtual furniture menu



(c) Picking virtual furniture object with a paddle



(d) Placing object in a room by sliding it from the paddle



(e) Moving virtual objects



(f) Constructed scene

Figure 10: The VOMAR interface

To copy an object from the object book onto the paddle the user simply places the paddle beside the desired object and the close proximity is detected and the object copied onto

the paddle (fig 10 (c)). Once a model is on the paddle it can be picked up and viewed from any viewpoint. To drop a model into the virtual scene the paddle is placed at the desired location and tilted until the model slides off (figure 10 (d)). Models in the scene can be pushed around by pushing motions of the paddle (fig. 10 (e)). A shaking motion is used to delete an object from the paddle, while models in the virtual room can be removed by hitting them.

As can be seen these interactions are very natural to perform with a real paddle, so in a matter of a few moments a user can assemble a fairly complex arrangement of virtual furniture (figure 10 (f)). Of course what the user is really doing is interacting with a simple CAD program, but instead of using a mouse or keyboard they are just manipulating a cardboard paddle in very intuitive ways.

5 DISCUSSION

Tangible AR interfaces couple AR displays with tangible user interface controllers and 3D spatial interaction to design a wide variety of powerful AR interfaces. From our prototype interfaces we have found that there are several advantages of Tangible AR interfaces. First, Tangible AR interfaces are *transparent interfaces* that provide for seamless two-handed 3D interaction with both virtual and physical objects. They do not require participants to use or wear any special purpose input devices and tools, such as magnetic 3D trackers, to interact with virtual objects. Instead users can manipulate virtual objects using the same input devices they use in physical world – their own hands - leading to seamless interaction with digital and physical worlds. This property also allows the user to easily use both digital and conventional tools in the same working space.

Tangible AR allows *seamless spatial interaction* with virtual objects anywhere in their physical workspace. The user is not confined to a certain workspace but can pick up and manipulate virtual data anywhere just as real objects, as well as arrange them on any working surface, such as a table or whiteboard. The digital and physical workspaces are therefore continuous, naturally blending together.

Tangible AR interfaces can allow the design of a simple yet effective and *consistent AR interface model*, providing a set of basic tools and operations that allow users, for example, to add, remove, copy, duplicate and annotate virtual objects in AR environments.

An interesting property of Tangible AR interfaces is their *ad-hoc, highly re-configurable nature*. Unlike traditional GUI and 3D VR interfaces, Tangible AR interfaces are in some sense designed by user as they are carrying on with their work. In these interfaces the users are free to put interface elements anywhere they want: tables, whiteboards, in boxes and folders, arrange them in stacks or group them together. How the interface components should be designed for such environments, if they should be aware of

the dynamic changes in their configuration, and how this can be achieved are interesting future research directions.

Another advantage is the use of *physical form-factor* to support interface functions. In our interfaces the physical design of the tangible interfaces elements provide affordances that suggest how they are to be used. So for example, in the Shared Space interface users discover the proximity based interactions because placing cards together is a natural behaviour and suggested by the shape of the cards. Naturally the physical form factor and the computer graphics design of the virtual images attached to the interfaces is important and should correspond to each other.

Finally, Tangible AR interfaces *naturally support face-to-face collaboration* as shown by the Shared Space and ARgroove interfaces. People commonly use the resources of the physical world to establish a socially shared meaning [24]. Physical objects support collaboration both by their appearance, the physical affordances they have, their use as semantic representations, their spatial relationships, and their ability to help focus attention. In a Tangible AR interface the physical objects can further be enhanced in ways not normally possible such as providing dynamic information overlay, private and public data display, context sensitive visual cues, and physically based interactions.

One of the reasons why our interfaces are successful is that the virtual models appear to be attached to the physical objects they are associated with. In the next section we describe the tracking technology that makes this possible.

6 TRACKING TECHNOLOGY

Precise registration of real and virtual objects anywhere in space is a significant research problem in the AR field. Azuma provides a good review of the issues faced in AR tracking and registration [25] and there are a number of possible tracking approaches that could be used in developing Tangible AR interfaces. We have developed a computer-vision based method in which a virtual model can be fixed in space relative to one or more tracking markers [26]. These markers are simple black squares with a unique pattern inside them. Our approach is summarized in figure 13. After thresholding an input image, square markers are extracted and identified. Then pose and position of markers are estimated from coordinates of the 4 vertices. Finally virtual images are drawn on the input image. A more complete explanation is given in Appendix A.

One of the advantages of this method is that in a video-see through AR system, the video frame that is shown in the users display is the same frame that is used to track their viewpoint and so the virtual models can appear exactly overlaid on a real object. These markers are also simple and cheap and can be attached to any flat surface. The tracking works in real time (30 fps on a 766 Mhz Pentium III) and is robust provided the square marker is in view.

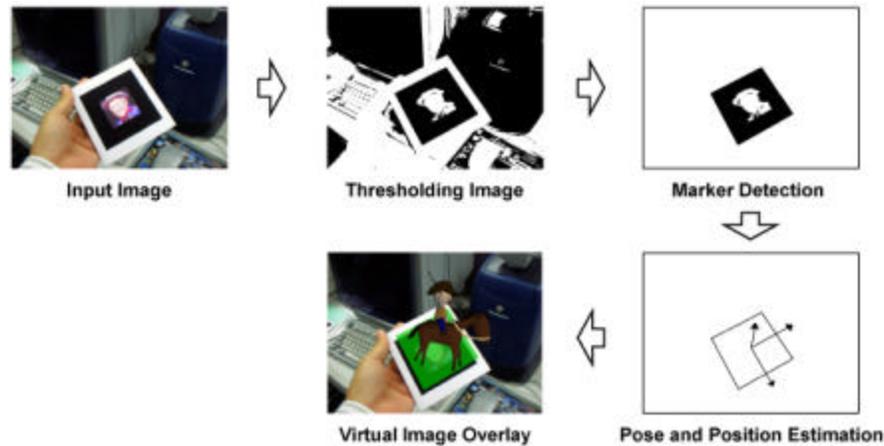


Figure 13: Vision based AR Tracking Process

Our system tracks six-degree of freedom manipulation of the physical marker, allows a user to use either hand to manipulate an object and can track as many objects are there are markers in the field of view.

However there are some limitations. In our system, the camera position and orientation is found in the marker's local coordinate frame. So if there are several markers in view, multiple camera transformations will be found, one for each marker. This use of local coordinate frames prevents measurement of some physical interactions; the distance between two objects can be found by considering the camera to marker distances of the same camera in the two local coordinate frames, but to measure the tilt of an object its orientation must be know relative to some global coordinate frame. We overcome this problem by using sets of markers to define a global coordinate frame. For example, the workspace in the VOMAR interface is defined by the set of six square markers on the large paper mat.

6 CONCLUSIONS

Tangible Augmented Reality is a new approach to designing AR interfaces that emphasizes physical object form and interactions. Using design principles adapted from Tangible User Interfaces we can develop AR interfaces that support seamless interaction and are very intuitive to use. We believe that exploration with Tangible AR interfaces are a first step towards developing new physically-based interface metaphors that are unique to Augmented Reality.

In the future we plan to conduct more rigorous user studies to quantify the benefits of Tangible AR interface, as well as developing a wider range of interaction techniques such as free hand interaction and two-handed gesture input.

REFERENCES

1. Sutherland, I. The Ultimate Display. *International Federation of Information Processing*, Vol. 2, 1965, pp. 506-508.
2. Ishii, H., Ullmer, B. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. In *Proceedings of CHI 97*, Atlanta, Georgia, USA, ACM Press, 1997, pp. 234-241.
3. Tangible Media site: <http://www.media.mit.edu/tangible/>
4. Butterworth, J., A. Davidson, et al. (1992). 3DM: a three dimensional modeler using a head-mounted display. Symposium on Interactive 3D graphics, ACM.
5. Poupyrev, I., Billingham, M., Weghorst, S., Ichikawa, T., The Go-Go Interaction Technique. In *Proc. of UIST'96*. 1996. ACM. pp. 79-80.
6. Stoakley, R., Conway, M., Pausch R. Virtual Reality on a WIM: Interactive Worlds in Miniature. In *Proceedings of CHI 95*, 1995, ACM Press.
7. Gabbard, J. L. *A taxonomy of usability characteristics in virtual environments*. M.S. Thesis, Virginia Polytechnic Institute and State University. 1997. Document available online at <http://www.vpsti.org/jgabbard/ve/framework/>.
8. Bajura, M., H. Fuchs, et al. (1992). Merging Virtual Objects with the Real World: Seeing Ultrasound Imagery Within the Patient. SIGGRAPH '92, ACM.
9. Feiner, S., B. MacIntyre, et al. (1993). "Knowledge-Based Augmented Reality." *Communications of the ACM* **36**(7): 53-62.
10. Kiyokawa, K., Takemura, H., Yokoya, N. "A Collaboration Supporting Technique by Integrating a Shared Virtual Reality and a Shared Augmented Reality", *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC '99)*, Vol.VI, pp.48-53, Tokyo, 1999.
11. Schmalstieg, D., A. Fuhrmann, et al. (2000). Bridging multiple user interface dimensions with augmented reality systems. ISAR'2000, IEEE.
12. Butz, A., Hollerer, T., et.al. Enveloping Users and Computers in a Collaborative 3D Augmented Reality. In

Proceedings of IWAR '99, October 20-21, San Francisco, CA, 1999, pp. 35-44.

13. Rekimoto, J., Y. Ayatsuka, et al. (1998). Augment-able reality: Situated communication through physical and digital spaces. ISWC'98, IEEE.
14. Hollerer, T., S. Feiner, et al. (1999). "Exploring MARS: developing indoor and outdoor user interfaces to a mobile augmented reality system." *Computers & Graphics* **23**: 779-785.
15. Anabuki, M., H. Kakuta, et al. (2000). Welbo: An Embodied Conversational Agent Living in Mixed Reality Spaces. CHI'2000, Extended Abstracts, ACM.
16. Wellner, P. Interactions with Paper on the DigitalDesk. *Communications of the ACM*, Vol. 36, no. 7, July 1993, pp. 87-96.
17. Weiser, M. The Computer for the Twenty-First Century. *Scientific American*, 1991, 265 (3), pp. 94-104.
18. Anderson, D., Frankel, J., et. al. Tangible Interaction + Graphical Interpretation: A New Approach to 3D Modeling. In Proceedings of SIGGRAPH 2000, August 2000, New Orleans, ACM Press, pp. 393-402.
19. Fitzmaurice, G. and Buxton, W. (1997). An Empirical Evaluation of Graspable User Interfaces: towards specialized, space-multiplexed input. Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'97). pp. 43-50. New York: ACM.
20. Gorbet, M., Orth, M., Ishii, H. Triangles: Tangible Interface for Manipulation and Exploration of Digital Information Topography. In *Proceedings of CHI 98*, Los Angeles, CA, 1998.
21. Fjeld, M., Voorhorst, F., Bichsel, M., Lauche, K., Rautenberg, M., H., K. (1999). Exploring Brick-Based Navigation and Composition in an Augmented Reality. In *Proceedings of HUC' 99*, pp. 102-116.
22. Ishii, H., Kobayashi, M., Arita, K. (1994). *Iterative design of seamless collaborative media*. CACM 37(8), 83-97.
23. Billinghurst, M., Poupyrev, I., Kato, H., May, R. (2000) *Mixing Realities in Shared Space: An Augmented Reality Interface for Collaborative Computing*. In Proceedings of the IEEE International Conference on Multimedia and Expo (ICME2000), July 30th - August 2, New York.
24. Gav, G., Lentini, M. Use of Communication Resources in a Networked Collaborative Design Environment. http://www.osu.edu/units/jcmc/IMG_JCMC/ResourceUse.html
25. Azuma, R. "A Survey of Augmented Reality", *Presence*, 6, 4, pp.355-385, 1997

26. ARToolKit tracking software available for download at: http://www.hitl.washington.edu/research/shared_space/download/

Appendix A: A Tangible AR Tracking Algorithm

Figure 14 shows coordinates frames that are used in tracking process. A size-known square marker is used as a base of the marker coordinates in which virtual objects are represented. The goal is to find the transformation matrix from the marker coordinates to the camera coordinates (T_{cm}), represented in eq.1. This matrix consists of the rotation $R_{3 \times 3}$ and the translation $T_{3 \times 1}$. These values are found by detecting the markers in the camera image plane and using the perspective transformation matrix.

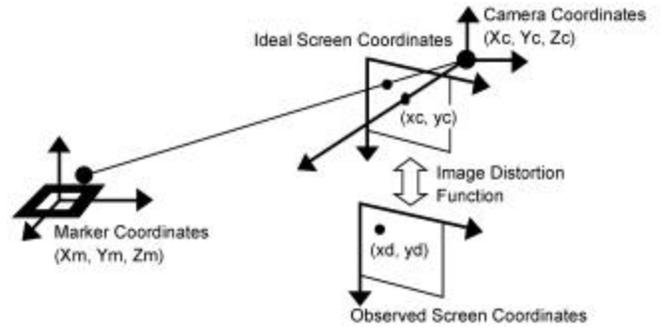


Figure 14: Coordinates System for Tracking

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} = T_{cm} \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} \quad (\text{eq.1})$$

1) Perspective Transformation Matrix

The camera coordinate frame is transformed to the ideal screen coordinates by the perspective matrix P (eq.2). This perspective transformation matrix is found from an initial off-line calibration process.

$$P = \begin{bmatrix} P_{11} & P_{12} & P_{13} & 0 \\ 0 & P_{22} & P_{23} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} hx_c \\ hy_c \\ h \\ 1 \end{bmatrix} = P \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (\text{eq.2})$$

2) Marker Detection

Step 1: Low-level Image Processing

Input images are thresholded by the constant value. Then all regions in the image are labeled.

Step 2: Marker Extraction

The regions whose outline contour can be fitted by four line segments are extracted. Parameters of these four line segments and coordinates of the four vertices of the regions found from the intersections of the line segments are calculated in the ideal screen coordinates and stored.

3) Pose and Position Estimation

Step 1: Estimation of R_{3x3}

When two parallel sides of a marker are projected on the image screen, the equations of those line segments in the ideal screen coordinates are the following:

$$a_1x + b_1y + c_1 = 0 \quad (\text{eq. 3})$$

$$a_2x + b_2y + c_2 = 0$$

For each of markers, the value of these parameters has been already obtained. The equations of the planes that include these two sides respectively can be represented as (eq.4) in the camera coordinates frame by substituting x_c and y_c in (eq.2) for x and y in (eq.3).

$$\begin{aligned} a_1P_{11}X_c + (a_1P_{12} + b_1P_{22})Y_c + (a_1P_{13} + b_1P_{23} + c_1)Z_c &= 0 \\ a_2P_{11}X_c + (a_2P_{12} + b_2P_{22})Y_c + (a_2P_{13} + b_2P_{23} + c_2)Z_c &= 0 \end{aligned} \quad (\text{eq.4})$$

Given that normal vectors of these planes are \mathbf{n}_1 and \mathbf{n}_2 respectively, the direction vector of the parallel two sides of the marker is given by the outer product. Given that two unit direction vectors that are obtained from two sets of two parallel sides of the marker is \mathbf{u}_1 and \mathbf{u}_2 , these vectors should be perpendicular. However, image processing errors mean that the vectors won't be exactly perpendicular. To compensate for this two perpendicular unit direction vectors are defined by \mathbf{v}_1 and \mathbf{v}_2 in the plane that includes \mathbf{u}_1 and \mathbf{u}_2 as shown in figure 15. Given that the unit direction vector which is perpendicular to both \mathbf{v}_1 and \mathbf{v}_2 is \mathbf{v}_3 , the rotation component R_{3x3} in the transformation matrix T_{cm} from marker coordinates to camera coordinates specified in eq.1 is $[V_1^t \ V_2^t \ V_3^t]^t$.

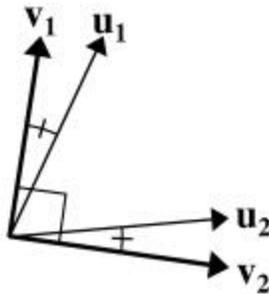


Figure 15: Two perpendicular unit direction vectors

Step 2: Estimation of T_{3x1}

Since the rotation component R_{3x3} in the transformation matrix T_{cm} was given, by using (eq.1), (eq.2), the four

vertices coordinates of the marker in the marker coordinate frame and those coordinates in the ideal screen coordinate frame, eight equations including translation component T_x T_y T_z are generated and the value of these translation component T_x T_y T_z can be obtained from these equations.

Step 3: Modification of R_{3x3} and T_{3x1}

The transformation matrix found from the method mentioned above may include error. However this can be reduced through the following process. The vertex coordinates of the markers in the marker coordinate frame can be transformed to coordinates in the ideal screen coordinate frame by using the transformation matrix obtained. Then the transformation matrix is optimized as sum of the difference between these transformed coordinates and the coordinates measured from the image goes to a minimum. Though there are six independent variables in the transformation matrix, only the rotation components are optimized and then the translation components are re-estimated by using the method in step 2. By iteration of this process a number of times the transformation matrix is more accurately found. It would be possible to deal with all of six independent variables in the optimization process. However, computational cost has to be considered.

Evaluation of Tracking Accuracy

In order to evaluate accuracy of the marker detection, the detected position and pose were recorded while a square marker 80[mm] wide was moved perpendicular to the camera and tilted at different angle. Figure 16 shows errors of position. Accuracy decreases the further the cards are from the camera and the further they are tilted from the camera.

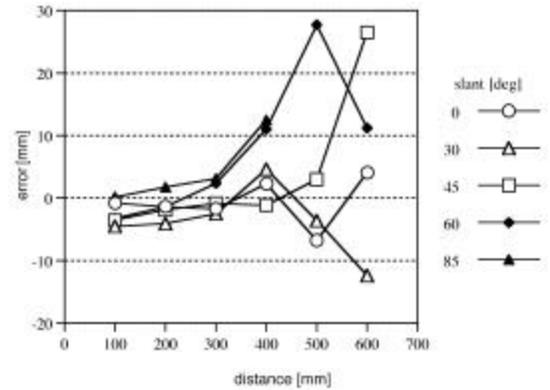


Figure 16: Errors of position

