Foundations and Trends® in Human-Computer
Interaction
Vol. 8, No. 2-3 (2014) 73–272

DOI: 10.1561/1100000049

A Survey of Augmented Reality

Mark Billinghurst, Adrian Clark, and Gun Lee

The Human Interface Technology Laboratory New Zealand
University of Canterbury
Christchurch, New Zealand
{mark.billinghurst, adrian.clark, gun.lee}@hitlabnz.org

Contents

Abstract

This survey summarizes almost 50 years of research and development in the field of Augmented Reality (AR). From early research in the 1960's until widespread availability by the 2010's there has been steady progress towards the goal of being able to seamlessly combine real and virtual worlds. We provide an overview of the common definitions of AR, and show how AR fits into taxonomies of other related technologies. A history of important milestones in Augmented Reality is followed by sections on the key enabling technologies of tracking, display and input devices. We also review design guidelines and provide some examples of successful AR applications. Finally, we conclude with a summary of directions for future work and a review of some of the areas that are currently being researched.

M. Billinghurst, A. Clark, and G. Lee. *A Survey of Augmented Reality*. Foundations and Trends® in Human-Computer Interaction, vol. 8, no. 2-3, pp. 73–272, 2014.
DOI: 10.1561/1100000049.

1

Introduction

In 1977 many moviegoers were amazed as a small robot projected a three-dimensional image of a woman in mid air. With the words "Help me Obiwan-Kenobi, you're my only hope", a recording of Princess Leia delivered a message that would change Luke Skywalker's life forever. In this Star Wars[1] scene, special effects were used to create the magical impression that three-dimensional virtual content was appearing as part of the real world. The movie forecast a future where people could interact with computers as easily as interacting with the real world around them, with digital and physical objects existing in the same space.

Thirty years later, in the 2008 US presidential campaign, a version of technology was shown for real. During the CNN election coverage reporter Wolf Blitzer turned to an empty studio and suddenly a life sized three-dimensional virtual image of reporter Jessica Yellin appeared beamed in live from Chicago[2]. Just like Princess Leia, she appeared to be part of the real world, but this time it was real and not through movie special effects. Wolf was able to talk to her as easily

---

[1] `http://www.starwars.com`

[2] `http://edition.cnn.com/2008/TECH/11/06/hologram.yellin/`

as if there was there face to face, even though she was thousands of miles away. It had taken only thirty years for the Star Wars fantasy to become reality.

The CNN experience is an example of technology known as Augmented Reality (AR), which aims to create the illusion that virtual images are seamlessly blended with the real world. AR is one of the most recent developments in human computer interaction technology. Ever since the creation of the first interactive computers there has been a drive to create intuitive interfaces. Beginning in the 1960's, computer input has changed from punch cards, to teletype, then mouse and keyboard, and beyond. One overarching goal is to make the computer interface invisible and make interacting with the computer as natural as interacting with real world objects, removing the separation between the digital and physical. Augmented Reality is one of the first technologies that makes this possible.

Star Wars and CNN showed how the technology could enhance communication and information presentation, but like many enabling technologies, AR can be used in a wide variety of application domains. Researchers have developed prototypes in medicine, entertainment, education and engineering, among others. For example, doctors can use AR to show medical data inside the patient body [Navab et al., 2007, Kutter et al., 2008], game players can fight virtual monsters in the real world [Piekarski and Thomas, 2002a], architects can see unfinished building [Thomas et al., 1999], and students can assemble virtual molecules in the real world [Fjeld and Voegtli, 2002]. Figure 1.1 shows a range of applications.

The potential of AR has just begun to be tapped and there is more opportunity than ever before to create compelling AR experiences. The software and hardware is becoming readily available as are tools that allow even non-programmers to build AR applications. However there are also important research goals that must be addressed before the full potential of AR is realized.

The goal of this survey is to provide an ideal starting point for those who want an overview of the technology and to undertake research and development in the field. This survey compliments the earlier surveys of

(a) ARQuake outdoor AR game [Piekarski and Thomas, 2002a]

(b) AR architecture by Re+Public `http://www.republiclab.com`

(c) AR in medicine [Kutter et al., 2008]

Figure 1.1: Typical AR applications.

Azuma [1997], Azuma et al. [2001], Van Krevelen and Poelman [2010] and Carmigniani et al. [2011] and the research survey of Zhou et al. [2008]. In the next section we provide a more formal definition of AR and related taxonomies, then a history of the AR development over the last 50 years. The rest of this survey gives an overview of key AR technologies such as Tracking, Display and Input Devices. We continue with sections on Development Tools, Interaction Design methods and Evaluation Techniques. Finally, we conclude with promising directions for AR research and future work.

2

Definition and Taxonomy

In one of the most commonly accepted definitions, researcher Ron Azuma says that Augmented Reality is technology that has three key requirements [Azuma, 1997]:

1) It combines real and virtual content

2) It is interactive in real time

3) It is registered in 3D

The CNN virtual presenter satisfies these requirements. The virtual image of Jessica Yellin appears in a live camera view of the studio, she is interactive and responds to Wolf Blitzer in real time, and finally, her image appears to be fixed or registered in place in the real world.

These three characteristics also define the technical requirements of an AR system, namely that it has to have a display that can combine real and virtual images, a computer system that can generate interactive graphics the responds to user input in real time, and a tracking system that can find the position of the users viewpoint and enable the virtual image to appear fixed in the real world. In the later sections of this survey we explore each of these technology areas in more depth. It

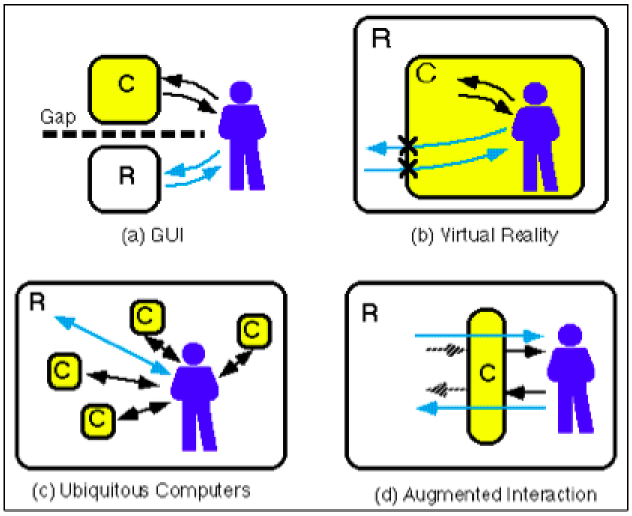


Figure 2.1: Rekimoto's comparison of HCI styles (R = real world, C = computer). [Rekimoto and Nagao, 1995]

should be noted that Azuma's definition doesn't place any limitations on the type of technology used, nor is it specific to visual information, and some AR systems provide an audio or haptic experience.

In the larger context, Augmented Reality is the latest effort by scientists and engineers to make computer interfaces invisible and enhance user interaction with the real world. Rekimoto distinguishes between traditional desktop computer interfaces and those that attempt to make the computer interface invisible [Rekimoto and Nagao, 1995]. As Figure 2.1(a) shows, with a traditional desktop computer and desktop WIMP (Windows, Icons, Menus, Pointer) [Costabile and Matera, 1999] based graphical user interface (GUI) there is a distinct separation between what the on-screen digital domain and the real world. One approach to overcome this is through the approach of Ubiquitous Computing [Weiser, 1993] (Figure 2.1(c)) where computing and sensing technology is seamlessly embedded in the real world.

An alternative approach is through Virtual Reality (VR) [Burdea and Coiffet, 2003] (Figure 2.1(b)) when the user wears a head mounted display and their view of the real world is completely replaced by

Figure 2.2: Typical Virtual Reality system with an immersive head mounted display, data glove, and tracking sensors. [Lee et al., 2002]

computer-generated graphics (Figure 2.2). In a VR system the user is completely separated from the real world, isolated in their head mounted display, and so the computer is again invisible to the user. In contrast, AR interfaces are designed to enhance interactions in the real world (Figure 2.1(d)).

As can be seen from Rekimoto's diagram, Augmented Reality is complimentary to immersive Virtual Reality (VR). Many of the component technologies used are the same, such as head mounted displays, tracking systems, and handheld input devices. However there are some important differences between AR and VR systems. The main goal of a Virtual Reality system is to use technology to replace reality and create an immersive environment. In contrast, the main goal of an Augmented Reality system is to enhance reality with digital content in a non-immersive way. This means that in a VR system the display device should be fully immersive with a wide field of view (FOV), and the 3D graphics shown as realistic as possible. The field of view is the amount of the users visual space filled with computer graphics and so the greater the FOV the higher the level of immersion. Since the user can no long see the real world, viewpoint tracking in the VR system does not have to be very accurate relative to the real world.

Table 2.1: Virtual Reality and Augmented Reality technology requirements.

	Virtual Reality Replacing Reality	**Augmented Reality** Augmenting Reality
Scene Generation	requires realistic images	minimal rendering okay
Display Device	fully immersive, wide FOV	non-immersive, small FOV
Tracking and Sensing	low accuracy is okay	high accuracy needed

In contrast, in an AR system the display can be non-immersive with a smaller FOV and use minimal graphics. For example, wearable displays often have a FOV of less than 30 degrees, but some AR navigation application applications can work well on these displays by using very simple map and arrow graphics. However, in an AR application, the tracking must be as accurate as possible to create the illusion that the virtual content is fixed in the real world. In a see-through AR display it it very easy for the human eye to perceive a mismatch between real and virtual graphics of even a few millimeters. Table 2.1 shows the complimentary differences between AR and VR systems.

Another way of defining Augmented Reality is in the context of other technologies. Milgram and Kishino [1994] introduced the concept of "Mixed Reality", which is the merging together of real and virtual worlds, and a Mixed Reality continuum which is a taxonomy of the various ways in which the "virtual" and "real" elements can be combined together (see Figure 2.3). On the right end is the *Virtual Environment (VE)*, where the user's view of the world is completely replaced by computer generated virtual content. On the opposite left end is the *Real Environment (RE)* where none of the user's view is replace by virtual content. Towards the VE end is Augmented Virtuality where most of the user's view is replaced by computer graphics, but there is still a view of the real world available. Finally, Augmented Reality is closer to the RE end, where virtual cues enhances the user's view of the real world. As more or less virtual content is added to the AR scene the interface moves closer or further away from the VE or RE endpoints. The main lesson from this taxonomy is that AR interfaces don't exist as a discrete point between Real and Virtual experiences, but can appear anywhere along the Mixed Reality continuum.

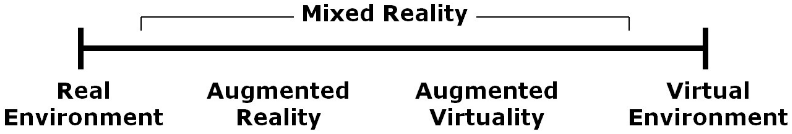


Figure 2.3: Milgram's Mixed Reality continuum. [Milgram and Kishino, 1994]

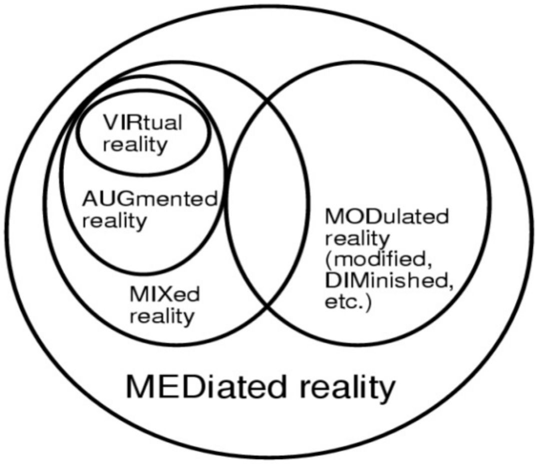


Figure 2.4: Mann's Mediated Reality. [Mann, 1994]

Milgram's Mixed Reality continuum is a one-dimensional array from the Real Environment to the Virtual Environment. However this can be extended along a second dimension. Mann [1994] defines the concept of Mediated Reality as filtering or modifying the view of the real world, rather than just adding to it as is done with Augmented Reality. For example, warping video of the real world to compensate for visual disability, or removing unwanted advertising billboards. Mann's concept of Mediated Reality extends the earlier definitions of AR, VR and MR as shown in Figure 2.4. A Mediality Continuum [Mann, 2002] can be constructed to compliment Milgram's Mixed Reality (or Virtuality Continuum), see Figure 2.5. In this case the vertical axis represents the amount of mediation or filtering that is being performed in the user view of the real or virtual environment. For example, a severely mediated virtuality application would be a VR system in which the

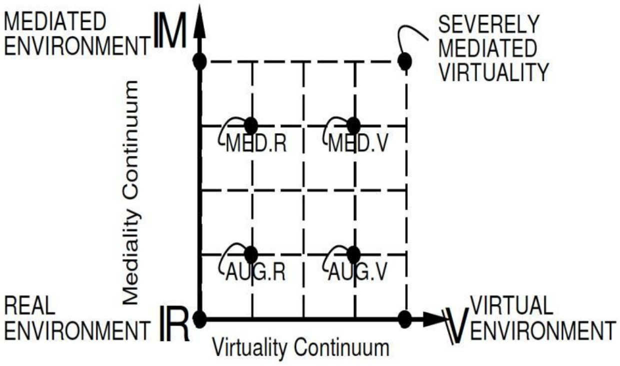


Figure 2.5: Mann's Mediality/Virtuality Continuum. [Mann, 2002]

user's view was filtered in some way. Like Milgram, Mann extends the concept of Augmented Reality and places it in the context of other interface technologies.

The Metaverse roadmap[1][Smart et al., 2007] presents another way of classifying the AR experience. Neal Stephenson's concept of the Metaverse [Stephenson, 1992] is the convergence of a virtually enhanced physical reality and a physically persistent virtual space. Building on this concept the Metaverse roadmap is based on two key continua (i) the spectrum of technologies and applications ranging from augmentation to simulation; and (ii) the spectrum ranging from intimate (identity-focused) to external (world-focused). These are defined as follows:

- *Augmentation*: Technologies that add new capabilities to existing real systems
- *Simulation*: Technologies that model reality
- *Intimate*: Technologies focused inwardly, on the identity and actions of the individual External: Technologies are focused outwardly, towards the world at large

The technologies of Augmented Reality, Virtual Worlds, Life Logging, and Mirror Worlds can be arranged within these continua (see Figure 2.6). They represent another way of classifying AR, alongside other

---

[1] `http://metaverseroadmap.org`

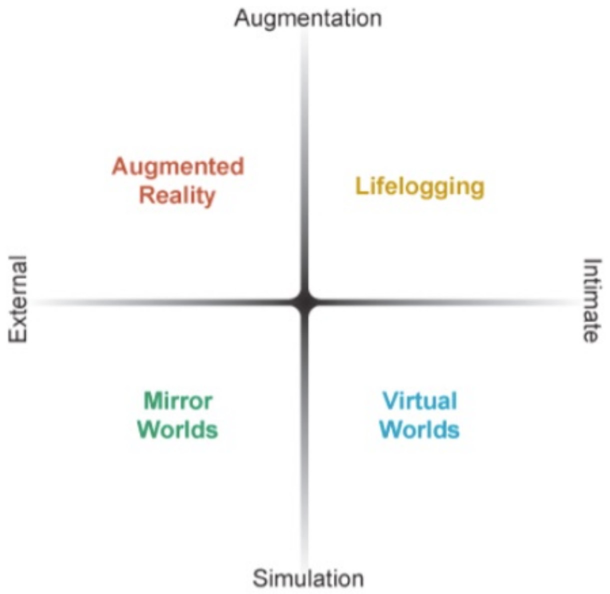


Figure 2.6: Augmented Reality in the Metaverse taxonomy.
`http://www.metaverseroadmap.org/overview`

technologies such as Lifelogging [Achilleos, 2003] and Mirror Worlds [Gelernter, 1991]. Lifelogging interfaces are those used by a person to capture their everyday experience, such as the SenseCam [Hodges et al., 2006] camera that continuously records audio and video from the wearer. In this way they combine intimacy and augmentation. Mirror World technologies are those that are externally focused and try to create a copy or simulation of the real world. For example, Google's Streetview[2] provides a panorama view of a street or location at a particular point in time.

More recently, a number of other AR classification schemes have been proposed. For example, Hugues et al. [2011] presents a taxonomy of AR environments based on functional purpose. They divide environments into two distinct groups, the first concerned with augmented perception of reality, and the second with creating an artificial environment. Similarly, Braz and Pereira [2008] present TARCAST, a

---

[2] `https://www.google.com/maps/views/streetview`

taxonomy based on the idea that any AR system can be made of six subsystems that can characterized using existing taxonomies. These subsystems include; Image Acquisition, Virtual Model Generator, a Mixing Realities Subsystem, Display, Real Manipulator, and a Tracking Subsystem. Finally, Normand et al. [2012] provides an AR taxonomy based on four axes; (1) tracking required, (2) augmentation type, (3) content displayed by the AR application, and (4) non-visual rendering modalities. These taxonomies are not as well established as the earlier work of Azuma, Milgram and Mann, but provide alternative perspectives from which to characterize AR experiences.

In summary, Ron Azuma provides a useful definition of the characteristics of Augmented Reality, which can help specify the technology needed to provide an AR experience. However, to fully understand the potential of AR it is important to consider it in the broader context of other taxonomies such as Milgram's Mixed Reality continuum, the Metaverse Taxonomy or Mann's Augmented Mediation. In the next section we review the history of Augmented Reality, and show that researchers have been exploring AR for many years.

3

History

Although Augmented Reality has recently become popular, the technology itself is not new. For thousands of year people have been using mirrors, lenses and light sources to create virtual images in the real world. For example, beginning in 17th Century, theatres and museums were using large plates of glass to merge reflections of objects with the real world in an illusion that became known a "Pepper's Ghost" [Brooker, 2007]. However, the first truly computer generated AR experience can be traced back to computer interface pioneer, Ivan Sutherland.

Ivan Sutherland is well known for developing Sketchpad, the world's first interactive graphics application at MIT in 1963 [Sutherland, 1964]. Shortly after that he moved to Harvard University, and in 1968, with Bob Sproull, he created the first prototype AR system [Sutherland, 1968] (see Figure 3.1). This combined a CRT-based optical see-through head mounted display, with a ceiling mounted mechanical tracking system connected to a PDP-11 computer and custom graphics hardware. Later the cumbersome mechanical tracker was replaced by an ultrasonic system. Thus their system combined the necessary display, tracking and computing components to provide an AR experience.

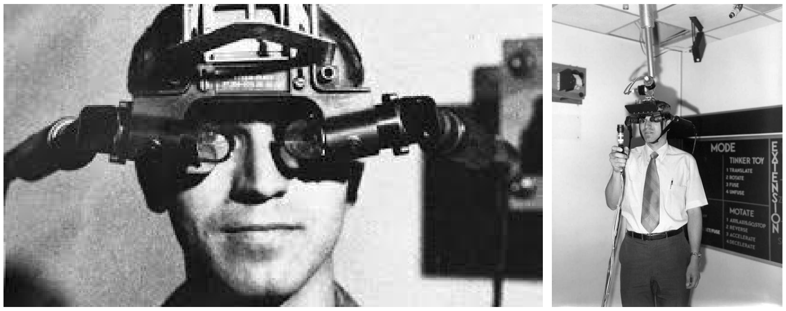


Figure 3.1: Sutherland's AR system. [Sutherland, 1968]

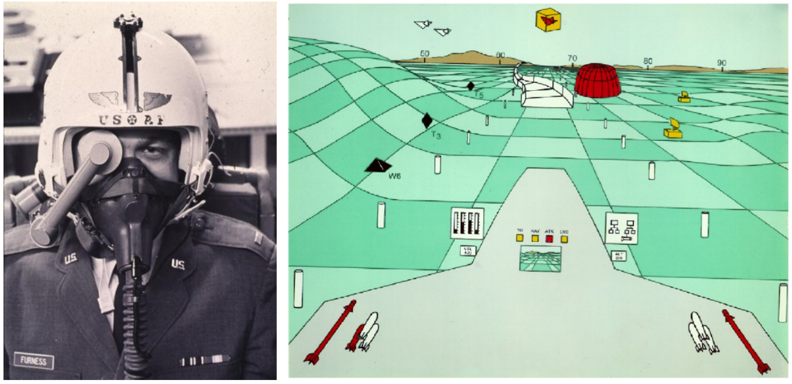
Although primitive, the system was capable of creating three-dimensional graphics that appearing to be overlaid on the real world. In this paper, Sutherland says: "The user has a 40 degree field of view of the synthetic information displayed on the miniature cathode ray tubes. Half-silvered mirrors in the prisms through which the user looks allow him to see both the images from the cathode ray tubes and objects in the room simultaneously. Thus displayed material can be made either to hang disembodied in space or to coincide with maps, desk tops, walls, or the keys of a typewriter." [Sutherland, 1968].

Sutherlands vision was of an Ultimate Display [Sutherland, 1965] in which "..the computer can control the existence of matter. A chair displayed in such a room would be good enough to sit in. Handcuffs displayed in such a room would be confining, and a bullet displayed in such a room would be fatal. With appropriate programming such a display could literally be the Wonderland into which Alice walked." His Augmented Reality research was a first step in this direction.

Sutherland's work went on to play a significant role in influencing virtual reality researchers, but for the next couple of decades most AR research continued in military and government research labs, rather than academic or industrial settings.

While Sutherland was developing his prototype Ultimate Display, Tom Furness at the Wright Pattern Air Force based was beginning research on the Super-Cockpit program for the US Air Force [Furness III, 1969]. From the mid-sixties until the mid-eighties, Furness and others were working on new ways to present complex flight details to pilots so that they wouldn't get overloaded with information. The SuperCockpit was envisioned as a ".. crew station which would exploit the natural perceptual, cognitive and psychomotor capabilities of the operator" [Furness III, 1986].

In the prototype developed the user wore a head mounted display in the aircraft cockpit (see Figure 3.2(a)). During daylight conditions, computer graphics appeared superimposed over part of the pilot's real world (Figure 3.2(b)). During night and in bad weather conditions the display could show graphics appears as a "substitute" for the real world, replacing the user's view. These early experiments evolved into the head mounted AR displays used in modern aircraft such as the helmet targeting system in the Apache attack helicopter. The early HMD developments of Sutherland and Furness were continued by various government organizations and private companies, such as Honeywell and Hughes to produce a wide variety of designs.



(a) User wearing the head mounted display **(b)** Artist's drawing of the view seen through the display

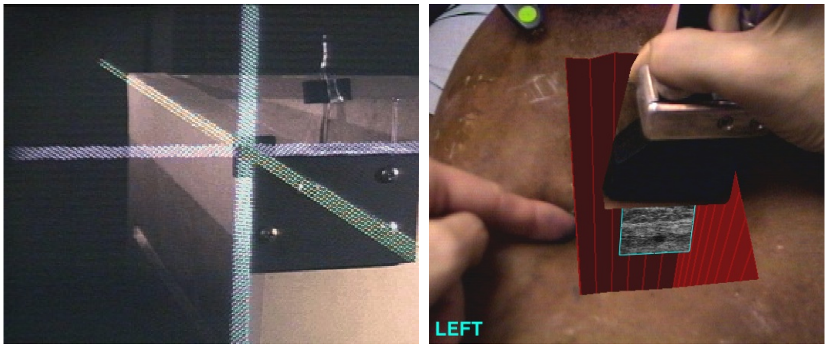
Figure 3.2: The US Air Force *Super Cockpit* system. [Furness III, 1986]

Inspired by Furness, in 1981 the National Aeronautics and Space Agency (NASA) built their own HMD from LCDs taken out of cheap Sony Watchman TV's. This was called the Virtual Visual Environment Display (VIVED) and was connected to a magnetic tracking system and graphics computer to create a fully immersive virtual environment.

This became the VIEW (Virtual Interface Environment Workstation) project and was used to explore possible user interface for astronauts and immersive visualizaton [Fisher et al., 1987]. A key part of this system was a glove interface developed by VPL Research, a company founded by Jaron Lanier and Jean-Jacques Grimaud in 1985 to provide Virtual Reality technology [Kalawsky, 1993]. Lanier coined the term "Virtual Reality" and in the middle and late 1980s VPL became famous for making the DataGlove glove and the EyePhone HMD. This meant that by the late 80's, technologies were becoming available to enable people to conduct AR research in academic and industrial settings.

Furness left the Air Force in 1989 to found the influential Human Interface Technology Laboratory at the University of Washington, transferring key Virtual and Augmented Reality technology into an academic setting. Other key academic research groups studying Virtual and Augmented Reality around that same time included Frederick Brooks' group at the University of North Carolina, Steve Feiner's group at Columbia University, and Paul Milgram's group at the University of Toronto. The UNC group conducted early research in see-through head mounted displays [Rolland et al., 1995a, Kancherla et al., 1996], registration and tracking techniques [Azuma, 1993, Azuma and Bishop, 1994] and medical applications [Bajura et al., 1992, Fuchs et al., 1996] (see Figure 3.3). Feiner published one of the first academic papers on AR, describing the KARMA, a knowledge-based AR system [Feiner et al., 1993b]. His group also researched interaction methods [Feiner et al., 1993a], and developed the first mobile AR systems[Feiner et al., 1997]. Milgram was conducting research on overlaying virtual cues on stereo video [Drascic and Milgram, 1991, Drascic et al., 1993], using AR to enhance tele-operation [Zhai and Milgram, 1992], and AR methods to enahance human robot interaction [Milgram et al., 1993].

At the same time, researchers Dave Mizell and Tom Caudell at Boeing were exploring the use of Augmented Reality in an industrial setting. They were trying to solve the problem of how to enable workers to more efficiently create wire harness bundles, and developed an AR system that showed which real wires should be bundled together through the use of AR cues. Tom Caudell published the first academic



(a) AR tracking and registration [Azuma and Bishop, 1994]

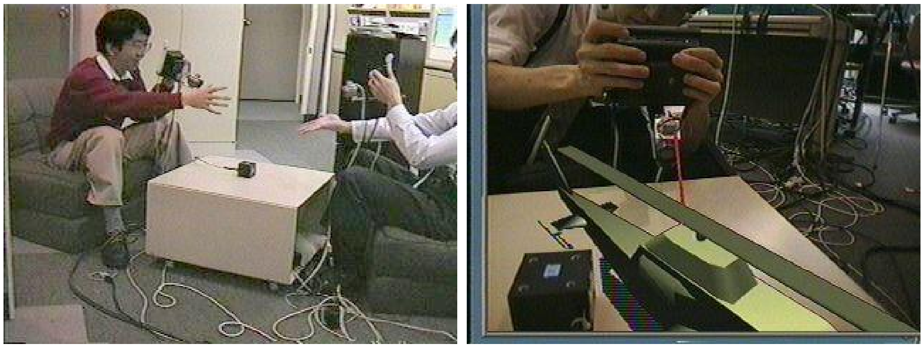
(b) Medical AR system [Fuchs et al., 1996]

Figure 3.3: AR researches at the University of North Carolina.

paper with the term "Augmented Reality" in it [Caudell and Mizell, 1992] and is often credited with coining the phrase. This research led to a number of other projects at Boeing using AR, and a number of other companies began researching industrial applications.

By the mid 1990's research was well underway on some of the key enabling technologies for AR, such as tracking, display and interaction. Using these technologies a number of interesting application areas were being explored. For example, Rekimoto [Rekimoto, 1996b], Billinghurst [Billinghurst et al., 1996, 1997] and Schmalstieg [Schmalstieg et al., 1996, Szalavári et al., 1998] were exploring how AR could be used to enhance face-to-face collaboration, developing systems that allowed people in the same location to see and interact with shared AR content (see Figure 3.4). A range of different medical applications of AR were being explored, such as for visualization of laparoscopic surgery [Fuchs et al., 1998], X-ray visualization in the patient's body [Navab et al., 1999], and for image guided surgery [Leventon, 1997]. Azuma et al. [2001] provides a good summary of key applications being researched before 2000.

One related area of research emerging at the same time was wearable computing. After several decades of exploration by early inventors, researchers at CMU [Smailagic and Siewiorek, 1994] and MIT [Starner,

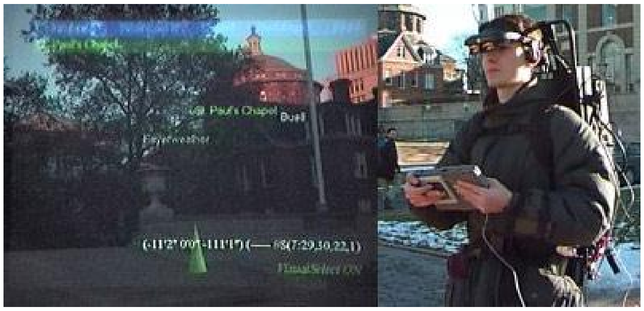


(a) Transvision [Rekimoto 96]

(b) Studierstube [Schmalsteig 96]

Figure 3.4: Early collaborative AR systems.

1996], began developing computer systems that could be worn all day. With a wearable computer, a head mounted display is an obvious choice for information presentation, and so the wearable systems are a natural platform for Augmented Reality. Starner et al. [1997] showed how computer vision based tracking could be performed on a wearable computing platform, and used as the basis for AR overlay on real world markers. Feiner et al. [1997] combined wearable computers with GPS tracking to produce a number of outdoor AR interfaces for showing information in place in the real world. Thomas et al. [1998] also investigated terrestrial navigation application of outdoor wearable AR interfaces, and later demonstrated using them for viewing and creating CAD models on-site [Thomas et al., 1999, Piekarski and Thomas, 2003],



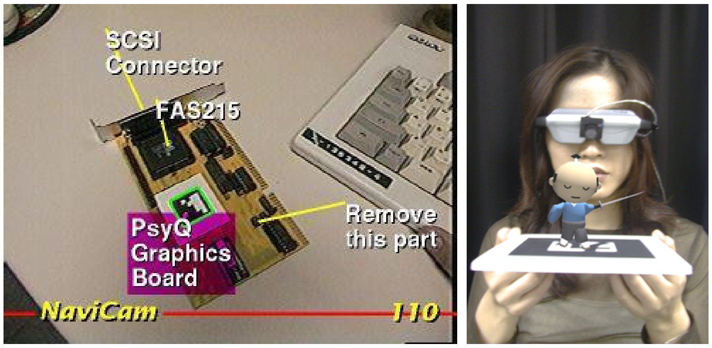
(a) Feiner's Touring Machine [Feiner et al., 1997]

(b) Thomas and Piekarski's wearable AR systems in 1998, 2002, and 2006. `http://www.tinmith.net/backpack.htm`

Figure 3.5: Early wearable AR systems.

and for game playing [Thomas et al., 2002, Piekarski and Thomas, 2002a]. These wearable systems were typically very bulky due to the size of the custom computer hardware, batteries, tracking sensors and display hardware although they eventually became more wearable (see Figure 3.5).

As discussed in the next section, one of the most significant challenges of Augmented Reality is viewpoint tracking. Early AR systems such as those of Azuma [Azuma, 1993, Azuma and Bishop, 1994] and Rekimoto [Rekimoto and Nagao, 1995] used expensive magnetic tracking, or complex computer vision tracking based on InfraRed LEDs [Bajura et al., 1992]. In 1996 Jun Rekimoto developed a simple computer vision tracking system based on a printed matrix code [Rekimoto, 1996a, 1998], that evolved into CyberCode [Rekimoto and Ayatsuka,



(a) CyberCode
[Rekimoto and Ayatsuka, 2000]

(b) ARToolKit
[Kato et al., 2000]

Figure 3.6: Computer vision-based AR marker tracking.

2000], printed tracking pattern software that provided both marker identification and real time camera pose estimation (see Figure 3.6). Soon after, Kato and Billinghurst developed the ARToolKit tracking library that also provided real time computer vision tracking of a square marker [Kato and Billinghurst, 1999]. ARToolKit[1] was released as open source software in 2000 and became one of the most widely used AR tracking libraries, making it significantly easier for researchers and developers to build their own AR applications. ARToolKit was important because it solved two fundamental problems; (i) tracking the user's viewpoint, (ii) enabling interaction with real world objects.

In addition to wearable AR, researchers were also exploring handheld AR systems. The first handheld AR interfaces involved LCD screens tethered to desktop computers that provided the tracking and graphics generation. For example, Fitzmaurice's Chameleon system allowed a person to move an LCD display over a piece of paper and see virtual content [Fitzmaurice, 1993], and Rekimoto's NaviCam [Rekimoto and Nagao, 1995] superimposed virtual tags on real objects using handheld and headworn displays (see Figure 3.7(a)). By 2003 researchers such as Geiger [Geiger et al., 2001] and Assad [Assad et al., 2003] were exploring wireless data transfer of images from handheld

---

[1] `http://artoolkit.sourceforge.net`

(a) NaviCam
[Rekimoto and Nagao, 1995]

(b) PDA based AR
[Wagner and Schmalstieg, 2003]

(c) Mobile phone AR [Mohring et al., 2004]

Figure 3.7: Handheld Augmented Reality.

PDAs to a remote PC for image processing and AR overlay. Handheld devices eventually became powerful enough to run AR computer vision tracking algorithms, and the first self contained handheld AR application was demonstrated by Wagner et al. in 2003 [Wagner and Schmalstieg, 2003]. Mobile phones received the first cameras in 1997 and the first mobile phone based AR application was demonstrated by Mohring and Bimber in 2004 [Mohring et al., 2004] (see Figure 3.7(c)). This was followed soon after by the porting of ARToolKit over to mobile phones [Henrysson and Ollila, 2004]. These developments meant that for the first time millions of people had technology in their pocket that would allow them to have an Augmented Reality experience.

Some of this early research was spurred on by several significant large national research projects in AR. One of the most important was the Mixed Reality Systems Laboratory in Japan, a research laboratory

Figure 3.8: ARVIKA AR product maintenance. [Weidenhausen et al., 2003]

that ran from 1997 to 2001 as a joint venture between Canon and the Japanese Government. The lab received over $50 million USD for research into Mixed Reality technologies for four years and had a staff of around 30 researchers. The focus of the research was on technologies for 3-D imaging and display for Mixed Reality systems, content creation, tools for real time seamless fusion of the physical space and cyberspace, and other related topics. For example, the AR2Hockey AR interface they developed [Ohshima et al., 1998] was one of first collaborative AR systems. Since 2001 Canon took over research for Mixed Reality technology, and continues to conduct research in this area, targeting manufacturing applications.

In Germany, significant research on the application of AR technologies for manufacturing was conducted through the ARVIKA consortium[2]. This was a German government supported research initiative with 23 partners from the automotive and airplane industry, manufacturing, SMEs, universities and other research providers. The project ran for four years from 1999 to 2003 with the focus on developing AR technology that could improve manufacturing in the automobile and aerospace industries. Figure 3.8 shows a prototype product maintenance application developed in the project. Weidenhausen et al. [2003] provides an excellent summary of the lessons learned and overview of the key research outputs.

In 1998 the first research conference dedicated to Augmented Reality began, the IEEE/ACM International Workshop on Augmented Re-

---

[2] `http://www.arvika.de/www/index.htm`

ality (IWAR). This was followed by the International Symposium on Mixed Reality (ISMR) from 1999, and the International Symposium on Augmented Reality (ISAR) in 2000. All of these meetings combined into the International Symposium on Mixed and Augmented Reality (ISMAR)[3] from 2002, which remains the leading research conference in AR, where the most prominent researchers present their latest work.

The development of PC based AR tracking also led to the creation of the first dedicated AR companies. The first of these was Total Immersion[4], founded in 1998, a French company that initially focused on providing AR for location based experiences and events and marketing campaigns. ARToolworks[5] was established in 2001 to commercialize the ARToolKit software and became the first AR company focusing on proving AR tools for developers. Metaio[6] was created in 2003 from an initial industrial AR project with Volkswagen in the ARVIKA project, and has since grown to provide a AR platform for both desktop and mobile AR applications. All three of these pioneering companies are still in business, and have since been joined by dozens of others in the growing commercial space.

At this time most AR experiences either existed in the research laboratory or in museums or theme parks. The technology required specialized display and tracking hardware and high end computing to provide a good user experience. For example, the ARCO project [Wojciechowski et al., 2004] explored how AR technology could be used to show virtual artifacts in museums. The Virtual Showcase project [Bimber et al., 2001b] demonstrated how projection tables could be used to create high-end AR enhanced shared experiences (see Figure 3.9), and the HIT Lab NZ's Black Magic AR kiosk was seen by over 250,000 people at the 2005 America's Cup [Woods et al., 2004]. A number of theme parks opened AR enhanced rides, such as the AR Adventure Game opened in the Guandong Science and Technology Center, China using wide area tracking technology described in [Huang et al., 2009] and see-through HMDs described in [Weng et al., 2009].

---

[3] `http://ismar.vgtc.org`

[4] `http://www.t-immersion.com`

[5] `http://www.artoolworks.com`

[6] `http://www.metaio.com`

(a) ARCO
[Wojciechowski et al., 2004]

(b) Virtual Showcase
[Bimber et al., 2001b]

Figure 3.9: Museum and theme park AR.

One of the first widely available consumer AR experiences was the Sony PlayStation 3 game *The Eye of Judgement*[7], released in October 2007. This card-based battle game used the PS-3 camera and a derivative of Sony's original CyberCode[8] to show AR characters on physical playing cards. The game sold over 300,000 copies, making it the most widely used AR experience at the time. Other consumers experienced AR technology through virtual enhancements on live broadcast TV, with companies such as PVI/Sportsvision[9] providing AR cues on sports broadcasts. For example, the "yellow-line" that appeared in US football games to show the first down line, or the virtual flags that appeared behind swimmers in the Olympics to identify their country. In this case, the broadcast cameras were fixed and careful calibration could be performed to create the illusion that the virtual content is part of the real world. A similar approach was used to create virtual billboards and advertising that could be included into live footage.

The year 2009 represented a turning point for Augmented Reality as it moved from the research laboratory and specialist applications, to widely available commercial technology. Figure 3.10 shows the Google Trends[10] graph of Google searches that contain the term "Augmented

---

[7] `http://www.playstation.com/en-us/games/the-eye-of-judgment-ps3`

[8] `http://www.sonycsl.co.jp/en/research_gallery/cybercode.html`

[9] `http://www.sportvision.com`

[10] `https://www.google.com/trends`

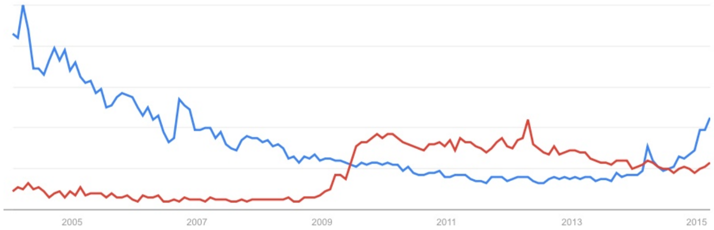


Figure 3.10: Relative Google search terms: "Virtual Reality" in blue and "Augmented Reality" in red. `https://www.google.com/trends`

Reality" compared to "Virtual Reality". Since 2004 there was a steady decrease in searches on Virtual Reality, while activity in Augmented Reality remained low. Then interest in Augmented Reality quickly rose in 2009, passing Virtual Reality searches by June and being twice as popular by the end of the year. This was mainly due to three factors (1) the emergence of Flash-based AR, (2) smart phone based AR, and (3) use of AR for global marketing campaigns. These combined to mean that from 2009 onwards billions of people have had the technology to have an AR experience and many millions have had one. More recently, in June of 2013 searches for VR passed AR, as the Oculus Rift and other cheap HMDs revived interest in Virtual Reality.

Flash based AR arose at the end of 2008 when Adobe added camera support to its popular Flash platform. A pair of Japanese developers nick-named Sqoosha and Nyatla ported the ARToolKit library to Flash creating FLARToolKit , and for the first time people could have an AR experience from within their web browsers. This lead to a large number of web-based AR experiences. For example, the GE Smart Grid website in early 2009 allowed people to see a virtual windfarm displayed over a printed marker with windmill blades that spun faster as people blew on them[11]. Developing for Flash is significantly easier than writing native C or C++ applications and so FLARToolKit enabled many more developers to begin creating AR experiences.

A second important factor was the rise of smart phone. Although early mobile phone based experiences were shown several years earlier

---

[11] `https://www.youtube.com/watch?v=NK59Beq0Sew`

Figure 3.11: Wikitude AR browser showing AR tags over the real world.

these were run on phones that were difficult to develop for, had slow processing and graphics power, and limited sensors. The launch of the iPhone in 2007 provided a smart phone platform that was easy to develop for, with a processor fast enough for real time computer vision tracking and powerful 3D graphics. However it was the release of the first Android phone in October 2008 that provided a significant boost to mobile AR. The Android platform combined the camera and graphics of the iPhone with GPS and inertial compass sensors, creating the perfect platform for outdoor AR. Taking advantage of this platform, Austrian company Mobilizy released the Wikitude[12] AR browser for Android devices in late 2008. Wikitude allowed users to see virtual tags overlaid on live video of real world, providing information about points of interest surrounding them (see Figure 3.11). Since that time a number of other AR browsers have been released, such as Sekai Camera[13], Junaio[14], and Layar[15], and have been used by tens of millions of people.

The third factor contributing to the rise in popularity and awareness of AR was use of the technology in advertising. AR provides a

---

[12] `http://www.wikitude.com`
[13] `http://www.sekaicamera.com`
[14] `http://www.junaio.com`
[15] `http://www.layar.com`

(a) Zoo campaign **(b)** Living Sasquatch

Figure 3.12: AR in advertising.

compelling, almost 'magical' experience that grabs user's attention in a way that makes it easy to draw attention to the marketing message. The HIT Lab NZ created one of the world first mobile AR campaigns in 2007 where they worked with Saatchi and Saatchi to show virtual animals appearing over a printed newspaper advertisement for a local zoo (see Figure 3.12(a)). Since that time hundreds of mobile campaigns have been created for companies such as Nike, Coke, Ford and others. Similarly, many online advertising campaigns have been run. For example, Boffswana produced a virtual Sasquatch that users could animate and place in the real world[16], and found that people would spend over three times as long on the website compared to a traditional website (see 3.12(b)). AR enhanced print media also became increasingly common. The December 2009 edition of the Esquire magazine used AR[17] to provide virtual content overlaid on twelve of its pages, including a virtual video of Robert Downey-Jr on its cover. Following that there have been many examples of printed advertisements or article with AR content in magazines (e.g. Red Bull Bulletin, Colors, Wired), or even entire books.

At the time of this article writing (2015) the AR industry is growing strongly. There are a number of platforms available that make it pos-

---

[16] `https://vimeo.com/4233057`

[17] `https://www.youtube.com/watch?v=LGwHQwgBzSI`

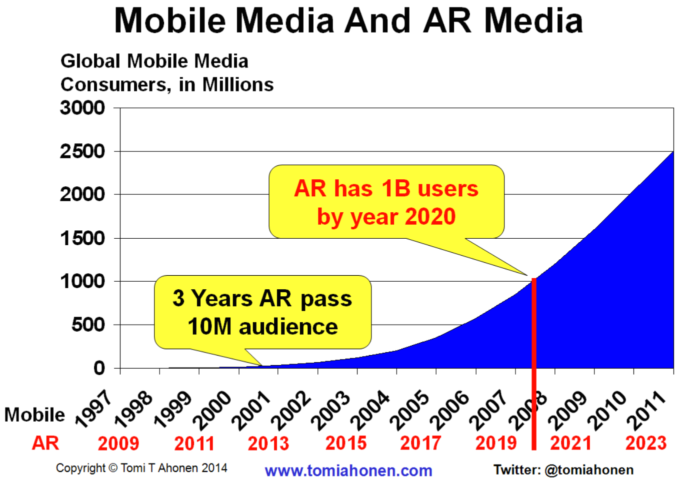


Figure 3.13: AR market prediction on global AR users by Tomi Ahonen (2014).

sible for users to easily access the technology, ranging from the web, to smart phones, and head worn displays such as Google Glass[18]. It is easier than ever before to develop AR applications with free available tracking libraries such as ARToolKit and Qualcomm's Vuforia[19], and authoring tools like BuildAR[20] that make is possible for even non-programmers to create AR experiences. This ease of access has lead to the use of AR technology in one form or another by hundreds of companies, and a fast growing commercial market. Industry analysts have predicted that the mobile AR market alone will grow to be a more than $6 billion USD by 2017 with an annual growth rate of more than 30%[21]. Comparing growth of AR users to early cell phone adoption, Tomi Ahonen has predicted that by 2020 there will be more than a Billion AR users[22] (see Figure 3.13).

---

[18] `http://glass.google.com`
[19] `http://www.vuforia.com`
[20] `http://www.buildar.co.nz`
[21] `http://www.juniperresearch.com`
[22] `http://www.tomiahonen.com`

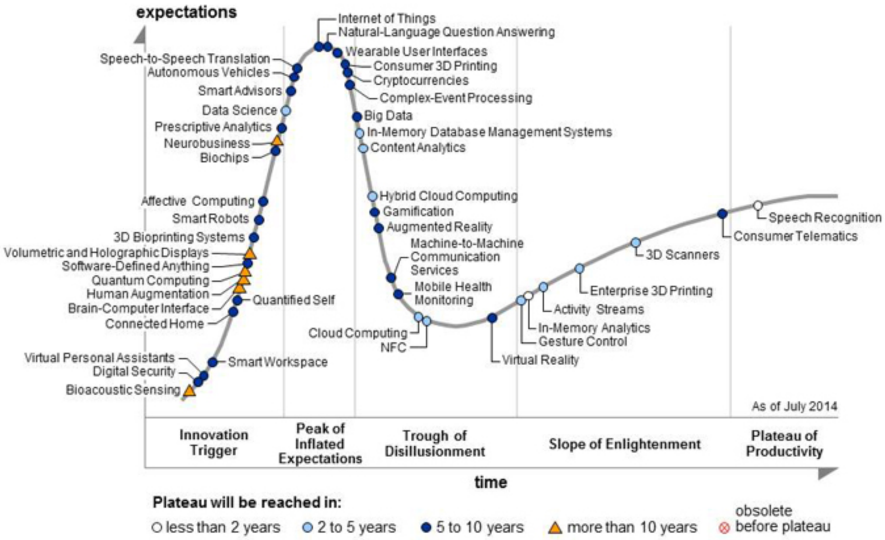


Figure 3.14: Gartner's 2014 Hype Cycle for Emerging Technologies.

Every year the Gartner group provides a graph of its Hype Cycle[23], predicting where certain technologies are on the path to mass-market adoption and how long it is before they are widely accepted. As Figure 3.14 shows, in July 2014 they were predicting that AR was about to reach the lowest point on the Trough of Disillusionment, suggesting a coming period of retrenchment in the commercial sector. However they are also forecasting that it will only take 5 to 10 years to reach the Plateau of Productivity where the true value of the technology is realized.

In summary, as can be seen, the history of AR research and development can be divided into four phases:

1) Pre-80's: Experimentation: Early experimentation that help define the concept of Augmented Reality and show the types of technology require

2) 1980's ~ mid-90's: Basic Research: Research into enabling technologies such as tracking, displays, and input devices

---

[23] `http://www.gartner.com/technology/research/hype-cycles`

3) mid 1990's ~ 2007: Tools/Applications: Using AR enabling technologies to develop early applied and explore interaction techniques, usability, and design theory

4) 2007 ~ present day: Commercial Applications: Widespread AR available in a number of application areas such as gaming, medicine, mobile and marketing

Over the last 50 years the technology has moved from being buried in Government and Academic research labs to the verge of widespread commercial acceptance. It is clear that the near future will bring exciting development in this fast growing field.

In the remainder of this survey we will first review basic enabling technologies such as tracking and displays, and then discuss interaction techniques, design guidelines for AR experiences, and evaluation methods. Finally we close with a review of important areas for research in Augmented Reality and directions for future work.

4

AR Tracking Technology

As discussed in section 2, for a system to meet Azuma's definition of Augmented Reality system [Azuma, 1997], it must fulfill three main requirements:

1) It combines real and virtual content
2) It is interactive in real time
3) It is registered in 3D

The third requirement, "Registered in 3D", relates to the ability of a system to anchor virtual content in the real world such that it appears to be a part of the physical environment. This section focuses on technologies that aim to fulfill this requirement.

In order to register virtual content in the real world, the pose (position and orientation) of the viewer with respect to some "anchor" in the real world anchor must be determined. Depending on the application and technologies used, the real world anchor may be a physical object such as a magnetic tracker source or paper image marker, or may be a defined location in space determined using GPS or dead-reckoning from inertial tracking.

Depending on which technology is used, the process of registering the system in 3D may be comprised of one or two phases; (1) a registration phase, which determines the pose of the viewer with respect to the real world anchor, and (2) a tracking phase, which updates the pose of the viewer relative to a previously known pose. In accordance to common terminology, in this document we use the term "Tracking" to refer to the combined application of both phases.

The following sections explore some of the common tracking techniques used for registering virtual content in the real world for the purposes of Augmented Reality.

4.1 Magnetic Tracking

Magnetic trackers are devices that use properties of magnetic fields in order to calculate the pose of a receiver with respect to a transmitter, which is used as the real world anchor. In a magnetic tracker the transmitter produces alternating magnetic fields, which are detected by one or more receiving units. By measuring the polarization and orientation of the magnetic field detected, the pose of each receiver can be calculated at high speed.

When used for augmented reality, the magnetic tracker transmitter acts as the origin of the virtual coordinate system, and by mounting a receiver to the viewer, the position and orientation can be determined [Caudell and Mizell, 1992].

Magnetic trackers have high update rates, are invariant to occlusion and optical disturbances, and the receivers are small and lightweight. However, the strength of magnetic fields fall of with the cube of distance and resolution falls off with the fourth power of distance [Nixon et al., 1998], and thus magnetic trackers have a limited working volume. Magnetic trackers also are prone to measurement jitter, and are sensitive to magnetic materials and electromagnetic fields in the environment [Bhatnagar, 1993]. Figure 4.1 shows the resolution of a Polhemus magnetic tracking device as a function of the distance between the transmitter and receiver. As can be seen, at up to 50 inches separation the resolution of the position and orientation measurement is

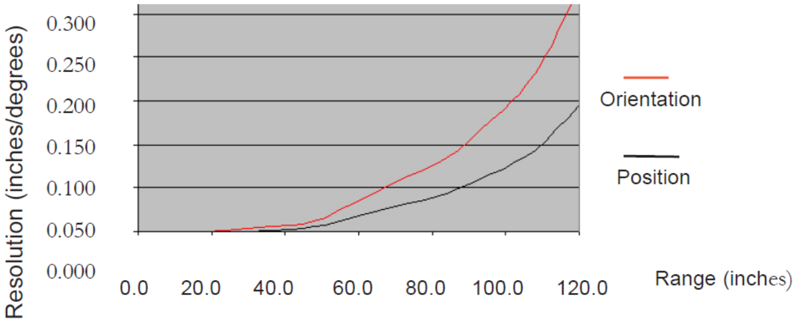


Figure 4.1: Resolution of position and orientation measurements for a magnetic tracker as a function of range between the transmitter and receiver. (Image courtesy of Polhemus `http://www.polhemus.com`)

very low, but this resolution steadily degrades at an increasing rate as the distance between transmitter and receiver increases.

Magnetic tracking technology has been used in a range of AR systems, with applications ranging from manufacturing [Caudell and Mizell, 1992] to maintenance [Feiner et al., 1993a] to medicine [Bajura et al., 1992] (Figure 4.2). Further discussion of Magnetic Tracking approaches can be found in Rolland et al.'s survey [Rolland et al., 2001].

4.2 Vision Based Tracking

We define vision based tracking as registration and tracking approaches that determine camera pose using data captured from optical sensors. These optical sensors can be divided into three main categories: Infrared Sensors, Visible Light Sensors, and 3D structure sensors.

Vision based tracking for augmented reality has become increasingly popular in recent times due to the minimal hardware requirements, improved computational power of consumer devices, and the ubiquity of mobile devices such as smart phones and tablets which feature both a camera and screen, making them ideal platforms for augmented reality technologies.

In the following sub sections, we discuss various vision based tracking techniques, categorized by the sensor hardware that is used.

Figure 4.2: 2D slices of ultrasound data is overlaid on a patient's abdomen. Both the HMD and ultrasound transducer are tracked using a Polhemus magnetic tracker. [Bajura et al., 1992]

4.2.1 Infrared Tracking

Some of the earliest vision based tracking techniques employed the use of targets which emitted or reflected light, which made detection easy due to their high brightness compared to the surrounding environment [Bajura and Neumann, 1995]. The targets, which emitted their own light, were also robust to adverse illumination effects such as poor ambient lighting or harsh shadows. These targets could either be attached to the object which was being tracked with the camera external to the object, known as "outside-looking-in" [Ribo et al., 2001], or external in the environment with the camera mounted to the target, known as "inside-looking-out" [Gottschalk and Hughes, 1993].

With comparable sensors, the inside-looking-out configuration offered a higher accuracy of angular orientation, as well as greater resolution than that of the outside-looking-in system (3). A number of systems were developed using this inside-looking-out configuration [Azuma and Bishop, 1994, Welch et al., 1999, Ward et al., 1992, Wang et al., 1990], typically featuring a head mounted display with external facing camera, and infrared LEDs mounted on the ceiling (see Figure 4.3). The LEDs were strobed in a known pattern, allowing for recognition of individual LEDs and calculation of the position and orientation of the user's head in the environment. The main disadvantages

Figure 4.3: Conceptual drawing of inside-looking-out system (Left). Roof mounted infrared led arrays (Middle) and the Hi-ball camera tracker (Right). [Welch et al., 1999]

of the inside-looking-out configuration were the considerable setup required to add all the LED sources to the environment and the additional weight added to the wearable device from the camera.

A major disadvantage of using light emitting targets is the required power and synchronization of the light sources. Many systems swapped light emitting targets in favour of retro-reflective markers [Dorfmüller, 1999], which reflect back infrared light provided from an external light source, typically located by the camera.

While infrared tracking systems are scalable and offer precision and robustness to illumination effects, they are complex, expensive, and invasive due to the physical infrastructure required.

4.2.2 Visible Light Tracking

Visible light sensors are the most common optical sensor type, with suitable cameras being found in devices ranging from laptops to smartphones and tablet computers, and even wearable devices. For video-see-through AR systems, these sensors are particularly useful as they can be both used for the real world video background that is shown to the users, as well as for registering the virtual content in the real world.

Tracking techniques that use visible light sensors can be divided into three categories: Fiducial tracking, Natural Feature tracking, and Model Based tracking. We discuss these techniques in the following sub sections.

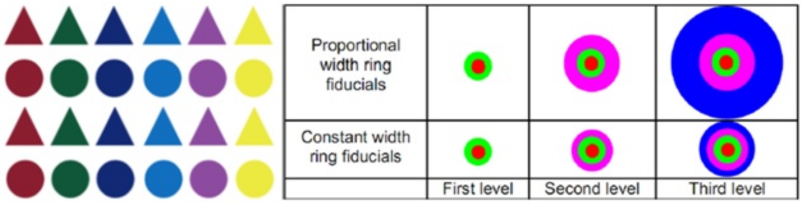


Figure 4.4: Colour sticker fiducial landmarks (left), Multi-Ring fiducial landmarks (right). [Cho et al., 1997, 1998]

Fiducial Tracking

We define fiducials as artificial landmarks that are added to the environment to aid in registration and tracking. Depending on the application and the technology used, the complexity of fiducials varies considerably. Early systems typically used small coloured LEDs or pieces of paper (see left of Figure 4.4), which could be added to the environment and detected using colour matching [Neumann and Cho, 1996]. The pose of the camera could be determined if enough fiducials of known position were detected in the scene. This approach had an additional benefit that the workspace could be dynamically extended by introducing additional fiducials to the scene at runtime and then estimating their position based on the position of other known fiducials [Cho et al., 1997, Neumann and Park, 1998]. Later works expanded on the concept and complexity of fiducials, adding features like multi-rings to allow for the detection of fiducials at greater distances [Cho et al., 1998] (right of Figure 4.4).

In order to calculate the pose of a viewer in the real world, a minimum of four points of known position need to be located in the environment [Fischler and Bolles, 1981]. With these simpler fiducials, the design of the space had to be carefully considered to ensure enough points were visible at any point in time, which required additional effort to physically place the fiducials into the environment and record their real-world positions. A simpler alternative which fulfilled the four point requirement was to use a single planar fiducial featuring a quadrilateral shape, where the corners of the quadrilateral serve as the known reference points [Comport et al., 2006]. Additional identification information can then be encoded inside the quadrilateral, allowing for multiple unique fiducials to be used in the same application [Rekimoto and Ayatsuka, 2000].

The planar quadrilateral fiducial proved to be an extremely popular technique for AR tracking due to its simplicity to use and high accuracy of detection. One of the most popular planar fiducial systems was the ARToolkit, which spawned a number of successors, and whose fiducial design still informs most planar fiducial registration techniques used today.

ARToolKit The ARToolkit, developed by Kato and Billinghurst in 1999, is a widely known and used fiducial marker based registration system [Kato and Billinghurst, 1999]. The fiducial markers are characterized by a thick black outlined square with a pattern inside for identification and orientation recognition.

The ARToolkit system uses binary thresholding and partial line fitting to determine potential edges of the marker (see Figure 4.5). All regions which are enclosed by four line segments are considered potential marker candidates, and are used to calculate a six degree of freedom (6DOF) matrix transformation for the marker with respect to the viewing camera. With this transformation known, the marker can be rectified so that it appears parallel to the camera's sensor plane, and the internal pattern is matched against a database of known patterns. This pattern matching allows to identify whether it is a valid marker, which marker it is (in the case of multiple marker systems), and also to determine the orientation of the marker around the marker's vertical axis.

Similar to simpler fiducials, multiple ARToolkit markers can be spaced through the environment to extend the tracking range. Because the 6DOF transformation of each marker relative to the camera can be determined, the transformation between individual markers can be calculated dynamically at run time [Uematsu and Saito, 2005], allowing for dynamic extension of the tracking environment.

The ARToolkit system has been applied to a range of different research areas such as Teleconferencing [Billinghurst and Kato,

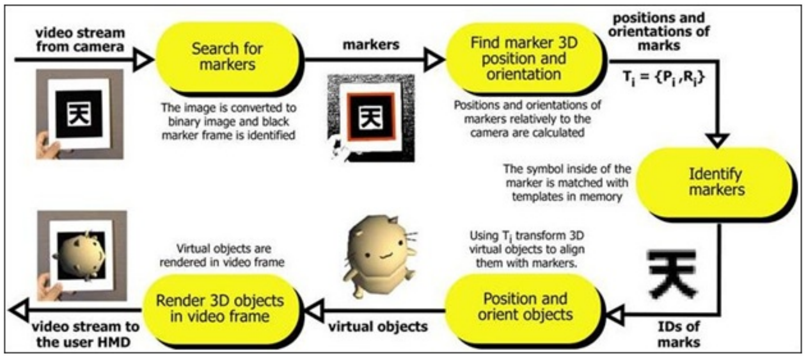


Figure 4.5: The ARToolkit registration process. [Billinghurst et al., 2000]

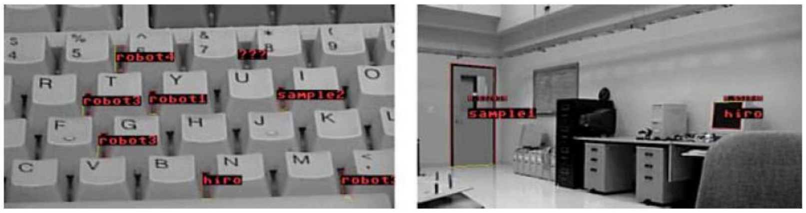


Figure 4.6: The ARToolkit falsely identifying real world features as tracking markers .[Fiala, 2004]

2000], wide area navigation [Wagner, 2002], and hand position tracking [Piekarski and Thomas, 2002b], and ported to run on mobile devices [Henrysson et al., 2005].

Despite the ARToolkit's popularity, there were a number of shortcomings. The partial line fitting algorithm is highly susceptible to occlusion, such that even a minor occlusion of the edge of the marker will cause registration failure. Although it provides flexibility in the appearance of the markers, the pattern matching on the internal image is prone to cause false positive matches, causing the AR content to appear on the wrong markers, or even on non-marker, square shaped regions in the environment, as shown in Figure 4.6.

ARTag The ARTag system [Fiala, 2005a] was developed to address a number of the limitations of ARToolkit. In particular, the system was designed to improve occlusion invariance, reduce the occurrence of false

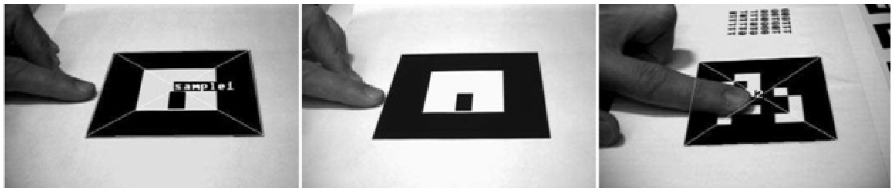


Figure 4.7: Occlusion susceptibility of ARToolkit (Middle) compared to that of ARTag (Right). [Fiala, 2004]

positive matches caused by inaccurate pattern matching, and improve performance in challenging lighting. Figure 4.6 shows an example of the ARToolKit falsely identifying visual features in the real environment as correct ARToolKit markers.

In ARTag, the partial line fitting algorithm was replaced by a quadrilateral heuristic search algorithm, which allowed for breaks in the continuity of the lines which define the edges of the marker. Figure 4.7 shows how this makes the tracking robust to edge occlusion in the tracking image. In addition to improving robustness to occlusion, this algorithm also improved the detection of markers in the presence of poor lighting, shadows and specular lighting reflections. The occurrence of false positive matches was reduced by replacing the image-based template matching with a binary barcode system inside the marker outline. Although reducing the flexibility for the appearance of the markers, the six by six binary grid allowed a greater accuracy in determining the marker ID, and a built in checksum provided redundancy and error correction to add further invariance to the effects of poor lighting or occlusion of the marker. When using multiple markers, the ARTag system defines the order which markers should be used to maximize the difference between the markers, further reducing the likelihood of false positive matches. Fiala [Fiala, 2005b, 2004] provided a very good comparison between ARTag and ARToolKit and how ARTag provides significant improvements.

ARToolkitPlus and Studierstube ES ARToolkitPlus [Wagner and Schmalstieg, 2007] was a ground up rebuild of the original ARToolkit, focusing on performance optimization and various image processing

Figure 4.8: Markers supported by Studierstube ES: Template Markers, BCH Markers, DataMatrix Markers, Frame Markers, Split Markers, and Grid Markers. [Schmalstieg and Wagner, 2009]

improvements to provide a fiducial marker tracking system that could run on the mobile devices of the time. The ARToolkitPlus followed the example set by ARTag of using binary barcodes to improve marker identification.

ARToolkitPlus was later re-implemented from scratch and integrated into the existing PC-based Studierstube collaborative AR environment [Schmalstieg et al., 2002]. The implementation used in Studierstube ES (Embedded Systems) [Schmalstieg and Wagner, 2007], the handheld device component of the Studierstube environment, supported several different marker configurations, including template markers similar to the original ARToolkit markers, ARTag themed binary markers, as well as Frame markers, which encoded all the identifying information for the marker in the frame, allowing for more visually pleasing markers. Figure 4.8 shows the range of different tracking makers supported by Studierstube.

Natural Feature Tracking

The planar fiducials used in the ARToolkit, ARTag and ARToolkit-Plus frameworks overcame difficulties faced when using simpler colour fiducials, and the frame and split fiducial markers in Studierstube ES allowed for more visually pleasing experience. However, all these fiducial types still require modification of the real environment, which may not be desirable or even possible, particularly when the real world anchor is valuable, such as a painting in an art gallery.

As the computational power of devices used for AR applications improved, it became possible to register the pose of the camera, in real-time, using features which already exist in the natural environment. Complicated image processing algorithms are used to detect features in the captured images which are unique in their surroundings, such as points, corners, and the intersections of lines. For each of these features a unique "descriptor" is calculated which allows for identification and differentiation of each feature. By matching features detected in the scene with those detected in the object to be tracked, the pose can be computed using similar algorithms as those used in the fiducial marker techniques.

In the following subsections, some of the more common natural feature detection and description algorithms are discussed, including SIFT, SURF, BRIEF, ORB, BRISK and FREAK.

SIFT The Scale Invariant Feature Transform (SIFT) is a natural feature detector and descriptor which was inspired by the function of neurons in the inferior temporal cortex of primates [Lowe, 1999]. First published in 1999, it remains as one of the most referenced natural feature algorithms, and is still commonly used as a benchmark when evaluating new techniques.

SIFT allows for scale invariant detection of features using the concept of scale spaces (see Figure 4.9). At each scale, known in the paper as an "octave", images are convolved using a Gaussian operation, and the difference between consecutively processed images is found, known as the Difference of Gaussian (DoG). The DoG emphasizes important structures in the image, and functions as an approximation of the Laplacian of Gaussian, a common technique used to find edges and corners in an image. With the DoG calculated, pixels which are the maxima/minima amongst neighbouring pixels in the same scale space are selected as good feature candidates. Each feature is recorded with the scale that it was detected at, and one or more orientations based on local image gradient directions [Lowe, 2004], providing scale and rotation invariance respectively.

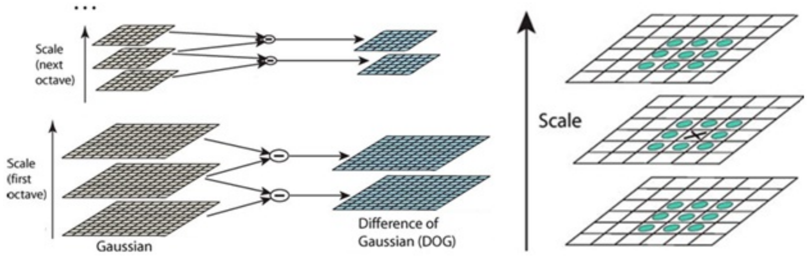


Figure 4.9: Calculating the Difference of Gaussian (DoG) images across different scales (Left). Calculating the maxima and minima of a pixel and it's neighbouring pixels in the same scale space (Right). [Lowe, 2004]

For each feature detected, a descriptor is calculated for identification and differentiation. The SIFT descriptor was modelled on the function of complex neurons in the visual cortex, which are hypothesized to allow for matching and recognition of 3D objects from a range of viewpoints. SIFT calculates the gradient magnitudes and orientations for all points in a 16x16 window around the feature point, with the magnitudes Gaussian weighted relative to distance from the feature point so that points further away from the feature point have less impact on the descriptor. The 16x16 points are split into 4x4 regions, and their gradient magnitudes placed into an 8 bin histogram based on their gradient orientation. The resulting 8 histogram values are used to calculate one component of the descriptor, resulting in a complete SIFT descriptor of 128 values (4x4 regions x 8 histogram values per region), which is then normalized for illumination invariance.

Figure 4.10 illustrates how the process works. A 16x16 window, divided into 4x4 regions, is selected (Left). For every pixel in the region, the gradient magnitude and orientation is calculated, and these are then put into an 8 bin histogram based on the orientation (Middle). Finally, a 128 dimensional vector descriptor is generated, based on the 8 histogram values for each of the 4x4 regions.

Comparison of feature points is done by calculating the distance between their descriptors, either as the sum of square differences between each element, or using a nearest neighbour approach such as Best-Bin-First [Beis and Lowe, 1997].

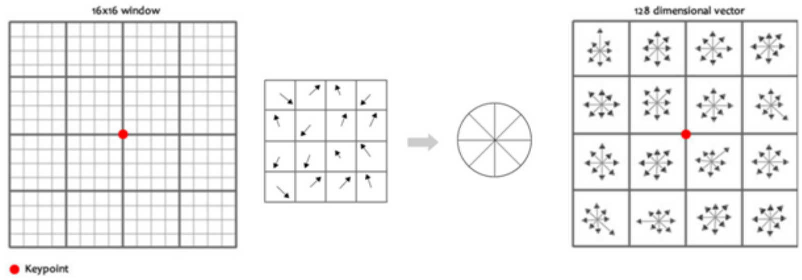


Figure 4.10: Calculating the SIFT Descriptor.

The main strength of SIFT is its invariance to a number of common visual recognition challenges, including changes in marker scale, rotation, perspective, as well as scene illumination. Unfortunately this robustness comes at a cost, and the usefulness of SIFT for Augmented Reality is limited due to the high computational requirements for feature detection, feature description, and feature matching. Additional research aimed to resolve this, including Fast Approximated SIFT [Grabner et al., 2006], which uses approximations of the DoG and orientation histograms algorithms to increase feature detection and feature description speeds, and PCA-SIFT [Ke and Sukthankar, 2004], which uses principal component analysis to reduce the descriptor length from 128 to 36 to increase feature matching speeds, however the algorithm was still too computationally intensive to be practical for real time natural feature tracking.

Although SIFT was unsuitable for real time natural feature tracking, it served as inspiration for SURF, a natural feature detector and descriptor capable of real-time performance.

SURF SURF, or Speeded Up Robust Features [Bay et al., 2006], is a natural feature detector and descriptor algorithm that was designed to be faster to compute than the state of the art algorithms, while maintaining a comparable level of accuracy. The authors chose to focus on invariance to scale and rotation only, leaving the general robustness of the descriptor to minimize the effects of other image transforms and distortions.

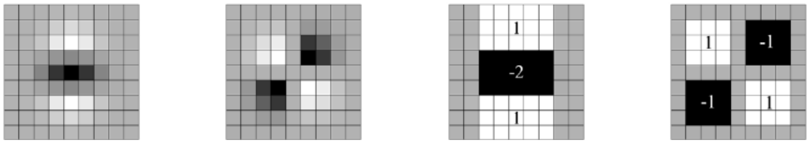


Figure 4.11: Gaussian second order partial derivatives in y-direction and xy-direction (Left) and the SURF box filter approximations (Right). [Bay et al., 2008]

To reduce the computational complexity required for feature detection and description, SURF features a number of approximations of previously used techniques. The feature detector is based on the Hessian matrix, with the determinant of the Hessian used to select for both location and scale. The Gaussian filters used by SIFT to provide scale space invariance were replaced by box filters operating on integral images, as shown in Figure 4.11. This removes the need to create scale spaces using image pyramids; instead box filters of different sizes can be run on the original integral image.

The SURF feature descriptor was based on similar properties to that of SIFT, but with reduced complexity. For each feature point, x and y direction Haar-wavelet responses are calculated for a circular neighbourhood around the point. The radius of circular neighbourhood is relative to the scale that the feature was detected at, and the results are Gaussian weighted by distance from the interest point. Using a sliding orientation window, the vertical and horizontal responses are summed, and the longest resulting vector is used as the orientation of the keypoint.

The orientation information is then used to create an oriented square region around each interest point, from which the descriptor is computed. Haar wavelet responses, oriented to the orientation calculated previously, are computed over 4x4 square sub-regions, and Gaussian weighted by distance from the interest point, as shown in Figure 4.12. The wavelet responses dx dy are summed over each region, and combined with the sum of the absolute value of the responses $|dx|$ $|dy|$, resulting in a four dimensional vector $\mathbf{v} = (\Sigma dx, \Sigma dy, \Sigma|dx|, \Sigma|dy|)$ for each of the 4x4 sub-regions, resulting in a descriptor vector of length 64.

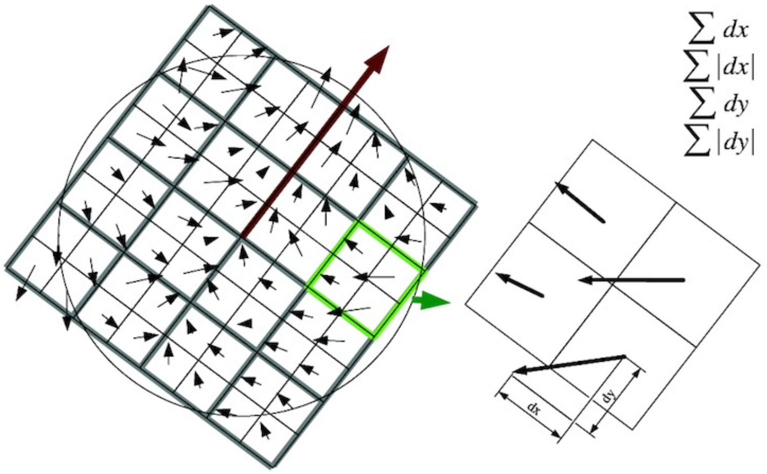


Figure 4.12: The SURF Feature descriptor operates on an oriented square region around a feature point. Haar-wavelets are computed over 4x4 sub-regions, and each sub-region's descriptor field is comprised of the resulting responses $\Sigma dx, \Sigma dy, \Sigma|dx|, \Sigma|dy|$. [Bay et al., 2008]

In the authors evaluation, the number of features found by the SURF detector was on average very similar to that of SIFT and other feature detectors. The repeatability scores of SURF were comparable or better, while being three to ten times faster than the competition [Bay et al., 2008]. The SURF descriptor outperformed other tested descriptors in recall, and was up to approximately 3 times faster than SIFT [Bay et al., 2006].

BRIEF The speed and robustness of SURF allowed for real-time natural feature based registration for augmented reality. However, the complexity of SURF was still too great for real time performance on mobile devices with limited computational power.

BRIEF [Calonder et al., 2010], the Binary Robust Independent Elementary Features descriptor, attempted to reduce the computational time by reducing the dimensionality of the descriptor and changing from a rational number descriptor to a binary descriptor. These changes significantly sped up matching, which previously had to be done iteratively or using techniques like approximate nearest neighbours [Indyk and Motwani, 1998], but could now be performed using the Hamming distance, which can be computed very quickly on modern devices. An additional benefit of the smaller descriptors was that they reduced the amount of memory required by the algorithms, making it more suitable for mobile devices where memory capacity is often less than on a desktop PC.

The BRIEF descriptor was inspired by feature classifiers, which transform the challenge of feature description and matching into one of classification. For the feature classifier, every feature in the marker underwent a number of pairwise pixel-intensity comparisons, which were used to train Classification trees [Lepetit and Fua, 2006] or Naive Bayesian classifiers [Ozuysal et al., 2010] as an offline process. These classifiers were then able to quickly classify new features at runtime with a minimal number of binary comparisons. BRIEF disposed of the classifier and trees, and instead used the results of the binary tests to create a bit vector.

To reduce the impact noise has on the descriptor, the image patches for each feature are pre-smoothed using a Gaussian kernel. The spatial arrangement of the pairwise pixel comparisons was determined by evaluating a set of arrangements for descriptor lengths of 128, 256 and 512 bits (known in the paper by their byte amounts as BRIEF-16, BRIEF-32 and BRIEF-64). The arrangements for the 128 bit descriptor are shown in Figure 4.13, where the first four arrangements were generated randomly, using uniform (1) and gaussian distributions (2,3) and spatial quantization of a polar grid (4), and the fifth arrangement taking all possible values on a polar grid. For the 128 bit descriptor, the second arrangement had the highest recognition rate.

When compared with the SURF feature descriptor, The BRIEF-64 outperformed SURF in all sequences where there was not a significant amount of rotation, as the BRIEF descriptor is not rotation invariant. Shorter descriptor lengths such as BRIEF-32 performed well for easy test cases; however BRIEF-16 was at the lower bound of recognition. The main advantage was the increase in speed, with BRIEF descriptor computation being on average 35 to 41 times faster than SURF, and BRIEF feature matching being 4 to 13 times faster than SURF.

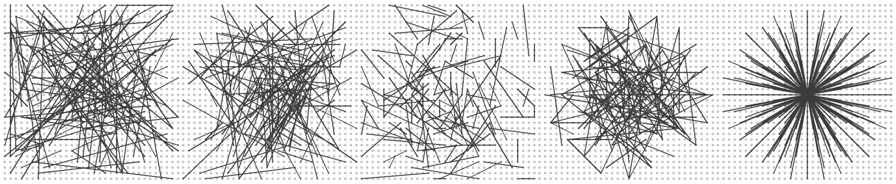


Figure 4.13: Tested spatial arrangements for the pairwise pixel comparisons for the 128 bit descriptor. The first four were randomly generated used various distributions, while the final one uses all possible values on a polar grid. The second arrangement was chosen as it had the greatest recognition rate. [Calonder et al., 2010]

The main disadvantage of BRIEF is its lack of rotation invariance. Resolving this was the main aim of ORB [Rublee et al., 2011], or Oriented FAST and Rotated BRIEF. ORB combined the FAST feature detector [Rosten and Drummond, 2006], modified to determine orientation of a feature using an intensity centroid [Rosin, 1999], with a version of BRIEF that uses lookup tables of pre-computed rotated spatial arrangements. Including all the additional improvements, an evaluation showed ORB is, on average, an order of magnitude faster than SURF and two orders of magnitude faster than SIFT.

The success of BRIEF lead to a number of other feature detector and descriptors which aimed to drastically reduce the computational time required for feature detection, description and matching. Most notable were BRISK (Binary Robust Invariant Scalable Keypoints) [Leutenegger et al., 2011], which used a feature detector inspired by AGAST [Mair et al., 2010] and a feature descriptor inspired by DAISY [Tola et al., 2010], and FREAK [Alahi et al., 2012] (Fast Retina Keypoint), which was designed to mimic features of the human visual processing system, and has been shown to outperform SIFT, SURF and even BRISK.

Model Based Tracking

Although not as popular as planar fiducial or natural feature tracking, there has been some interest in tracking real world objects using a known 3D structure, such as a CAD model. In early works, the 3D model of the object to be tracked was typically created by hand, and the structure of the object was often approximated as a combination of primitives, such as lines, circles, cylinders and spheres [Comport et al., 2003]. Edge filters were used to extract structure information about the scene, and these were matched to the primitive objects to determine pose [Wuest et al., 2005].

Combining natural feature tracking with the edge based tracking approaches provided additional robustness [Vacchetti et al., 2003], and also allowed for large trackable spaces and even tracking in outdoor environments [Comport et al., 2006]. Later works extended the primitive models to include texture [Reitmayr and Drummond, 2006] and introduced the concept of keyframes [Vacchetti et al., 2003], which allowed for even greater robustness in complex and variable environments.

Most recently, there has been significant interest in techniques that are able to simultaneously create and update a map of the real environment while localizing their position within it. The original motivation behind SLAM (Simultaneous Localization and Map Building) was for robot navigation in unknown environments [Dissanayake et al., 2001]; however the technique was adapted for use in Augmented Reality [Davison et al., 2007], where it allowed for drift free tracking of unknown environments. Further optimizations of the process led to PTAM, or Parallel Tracking and Mapping [Klein and Murray, 2007], where the tracking of the camera and mapping of the environment components were separated (see Figure 4.14). PTAM was specifically designed for AR, and improved both the accuracy of tracking as well as overall performance, although due to its diminished ability to close large loops like SLAM it is more suited for small environments.

4.2.3 3D Structure Tracking

In recent years, commercial sensors capable of detecting 3D structure information from the environment have become very affordable, beginning with the introduction of the Microsoft Kinect [Zhang, 2012]. These sensors commonly utilise technologies such as structured light [Scharstein and Szeliski, 2003] or time-of-flight [Gokturk et al., 2004] to obtain information about the three dimensional positions of points in the scene.

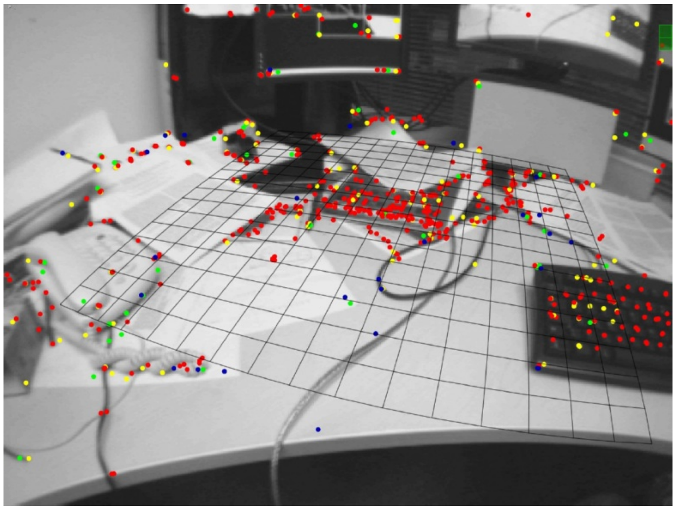


Figure 4.14: PTAM tracking from real world feature points. [Klein and Murray, 2007]

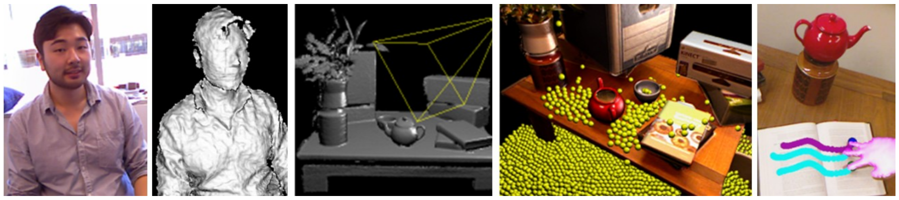


Figure 4.15: KinectFusion. From left to right: RGB image, 3D mesh created from single depth image, Reconstructed model of environment, AR particle effects, Multi-touch interaction. [Izadi et al., 2011].

These new technologies have opened up new possibilities for tracking and mapping the real environment using depth information [Henry et al., 2010]. The most well known approach for Augmented Reality currently is the KinectFusion system [Newcombe et al., 2011], developed by Microsoft. KinectFusion uses data obtained from the Kinect's structured light depth sensor to create high quality three dimensional models of real objects and environments, and these models are also used for tracking the pose of the Kinect in the environment. Figure 4.15 shows some examples of KinectFusion based tracking and interaction [Izadi et al., 2011].

In addition to tracking for AR, KinectFusion offers 3D reconstruction of real objects and the environment, allowing for simulated physical interactions between real and virtual objects, as shown in Figure 4.15. By combining tracking of the environment with tracking of the users limbs, KinectFusion can also detect multi-touch user interactions with the environment.

4.3 Inertial Tracking

Inertial tracking uses Inertial Measurement Unit (IMU) sensors such as accelerometers, gyroscopes and magnetometers to determine the relative orientation and velocity of a tracked object. Inertial tracking allows the measurement of three rotational degrees of freedom (orientation) relative to gravity, and the change in position of the tracker can be determined using the inertial velocity and time period between tracker updates.

Inertial sensors have "no range limitations, no line-of-sight requirements, and no risk of interference from any magnetic, acoustic, optical or RF interference sources. They can be sampled as fast as desired, and provide relatively high-bandwidth motion measurement with negligible latency." [Foxlin et al., 2004].

Unfortunately, inertial sensors are very susceptible to drift over time for both position and orientation. This is especially problematic for position measurements, as position must be derived from measurements of velocity. The effects of drift can be reduced with filtering, although this can reduce the update rate and responsiveness of the trackers [Foxlin, 1996]. Due to these issues with drift, applications requiring accurate tracking should aim to combine inertial sensors should with other sensors for correction, such as optical trackers or ultrasonic range measurement devices.

4.4 GPS Tracking

GPS technology allows for positional tracking in outdoor environments over the majority of the earth's surface. The current average accuracy of satellite based GPS is less than 3 metres, with the accuracy scheduled to improve as new advancements and satellite technology becomes available. In addition, GPS enhancement technologies such as RTK (Real Time Kinematic), which uses the GPS signal's carrier wave for measurement, have the potential to improve accuracy to the centimetre level.

The first Augmented Reality system using GPS was Feiner et al.'s Touring Machine [Feiner et al., 1997], which featured differential GPS positional tracking accurate to 1 metre and magnetometer/inclinometer orientation tracking. The interface was designed in such a way that the inaccuracy of the GPS data did not significantly affect usability, however the authors found that loss of GPS signal due to trees and buildings blocking line of sight to satellites caused significant problems.

GPS tracking for AR has since been applied to a number of application areas, including military [Julier et al., 2000], gaming [Piekarski and Thomas, 2002a, Thomas et al., 2002], and visualisation of historical data [Lee et al., 2012]. However, due to its low accuracy and only supporting positional tracking, GPS tracking has typically only been utilised in applications where accurate pose registration is unimportant, or as an input into a hybrid tracking system.

4.5 Hybrid Tracking

Hybrid tracking systems fuse data from multiple sensors to add additional degrees of freedom, enhance the accuracy of the individual sensors, or overcome weaknesses of certain tracking methods.

Early applications utilising vision based tracking approaches often included magnetic trackers [Auer and Pinz, 1999] or inertial trackers [You and Neumann, 2001]. This allowed the system to take advantage of the low jitter and drift of vision based tracking, while the other sensor's high update rates and robustness would ensure responsive graphical updates and reduced invalid pose computation [Lang et al., 2002]. An additional benefit was that, as magnetic and inertial trackers do not require line of sight, they can be used to extend the range of tracking. Figure 4.16 shows the system diagram of You's hybrid tracking system [You et al., 1999] and the tracking results compared to using the inertial

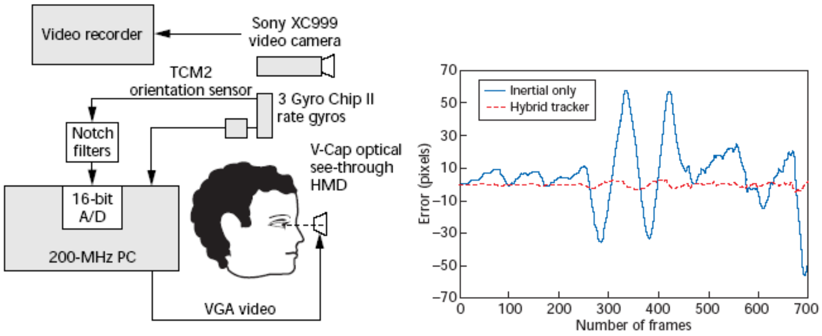


Figure 4.16: Hybrid tracking system (left) and tracking error over time (right). [You et al., 1999]

tracking by itself. As can be seen, the combined hybrid tracking has far less error than just using inertial tracking.

Due to their low accuracy and only providing positional information, GPS tracking systems are often combined with inertial sensors and/or vision based tracking in order to obtain full 6DOF pose estimation [Azuma et al., 1998]. Furthermore, to overcome issues with loss of signal due to occlusions or being inside a building, vision based tracking is often used as for backup tracking when GPS fails [Thomas et al., 2002].

Inertial trackers are often paired with ultrasonic rangefinders [Foxlin et al., 1998] or optical tracking systems [Foxlin et al., 2004] to reduce the amount of positional drift caused by the differential measurement approach. As inertial sensors use forces such gravity as a reference, hybrid tracking also becomes important when the application involves being on a moving platform, such as in a vehicle or simulator Foxlin [2000].

Mobile devices such as smartphones and tablets present new opportunities for hybrid tracking. Many mobile devices include cameras, accelerometers and gyroscopes [Lang et al., 2002], GPS [Höllerer et al., 1999], and wireless networking Evennou and Marx [2006], which can be combined to provide highly accurate pose estimation for both indoor and outdoor AR tracking Newman et al. [2004].

4.6 Summary

In summary, in this section we have shown that a wide variety of tracking approaches have been used create the illusion that AR virtual cues are fixed in space. The most widely used approaches currently use computer vision techniques and can provide marker based and markerless tracking at real time frame rates. However there are limitations with systems just relying on computer vision and it is becoming increasingly obvious that truly robust and accurate AR tracking will require hybrid tracking. In the next section we summarize display technology available to achieve the first of Azuma's AR characteristics, combing real and virtual content together. Combined with AR tracking, AR displays create the illusion that virtual content is really part of the user's real world.

5

AR Display Technology

Returning to Azuma's definition, in order to combine real and virtual images so that both are seen at the same time, some sort of display technology is needed. In this section we review various type of display technologies used in AR systems and applications. First we look into different approaches used for combining and compositing images of virtual scenes with the physical scene. Next we review a range of AR displays categorized based on where they are located between the user's eye and the physical scene. Although AR displays could be used for visualizing 3D graphics as is with generic 3D display technology (e.g. such as stereoscopic displays or volumetric displays), here we focus on combining the images of virtual and physical scenes on displays that are specifically designed for AR visualization.

5.1 Combining Real and Virtual View Images

Combining images of the real and virtual scenes for AR visualization requires a number of procedures including camera calibration, registration, tracking and composition. Camera calibration is a procedure to match the camera parameters of the virtual camera to that of the

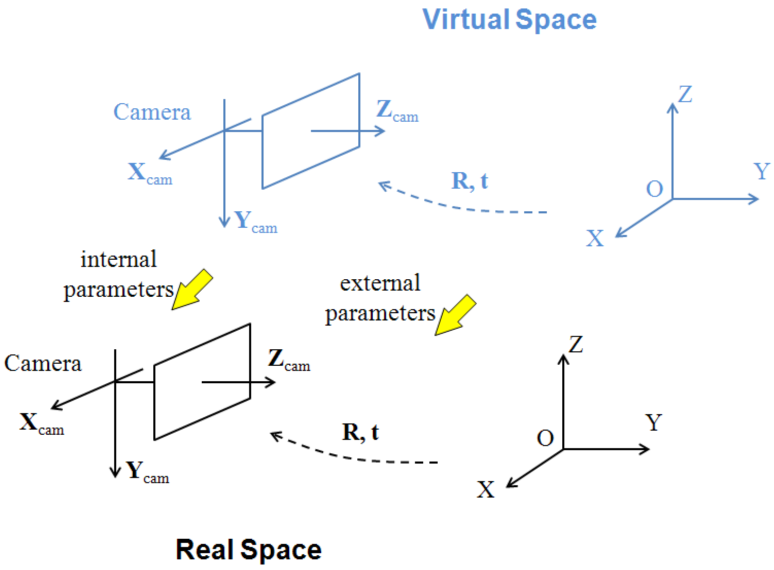


Figure 5.1: Camera calibration: matching internal and external parameters of the virtual camera to the physical view.

physical camera (or an optical model of the user's view defined by eye-display geometry), so that the computer rendered image of a virtual scene is correctly aligned to the view of the real world (see Figure 5.1). There are two types of camera parameters: internal and external parameters. Internal parameters are those that determine how a 3D scene is projected into a 2D image, while the external parameters define the pose (position and orientation) of the camera in a known coordinate frame.

Internal parameters of a physical camera can be calculated using a process that involves taking a set of pictures of a calibration pattern that has known geometric features and calculating the parameters based on the correspondence between the 3D structure of this geometric features and its projection on the 2D image [Tsai, 1987]. This is usually performed off-line before the AR system is in operation, but some AR systems do this on the fly [Simon et al., 2000]. In case of not using a physical camera to capture the real world scene, the internal parameters are decided by a geometrical model of the relationship between the user's eye and the display's image plane (e.g. as a pinhole camera model) [McGarrity and Tuceryan, 1999, Fuhrmann et al., 1999, Tuceryan and Navab, 2000].

External parameters are decided by tracking the pose (position and orientation) of the physical camera. When the physical scene is static, only the pose of the camera relative to the world's reference frame needs to be tracked, while in dynamic scenes the poses of each dynamic object of interest should be also tracked so that the change in the real world can be reflected in the virtual scene. As described in the previous section, there are various technologies for tracking physical objects, ranging from using 6 DOF sensors to computer vision techniques.

Tracking technologies provide the position and orientation of target objects (either the camera or other physical objects of interest) relative to a coordinate frame set by the tracking system. To make the virtual scene correctly aligned with the real world environment, the coordinate frame used for rendering the virtual scene should be matched with the tracking coordinate frame in the physical environment in a process called registration.

Once the virtual camera is calibrated, and the virtual space is registered to (or correctly aligned with) the physical space, rendering the virtual scene based on the tracking update generates images of the virtual scene that corresponds to the users' real world view. Composition of the generated virtual view image (image of the virtual scene) into the real world view can be achieved both digitally or physically (or optically) depending on the display system configuration used in the AR system.

AR displays can be categorized into mainly four types of configuration depending on how they combine the virtual view image with the real world view: (1) video based, (2) optical see-through, (3) projection onto a physical surface, and (4) eye multiplexed.

5.1.1 Video based AR Displays

Video based AR displays use digital processes to combine virtual view images with video of the real world view. This type of display first

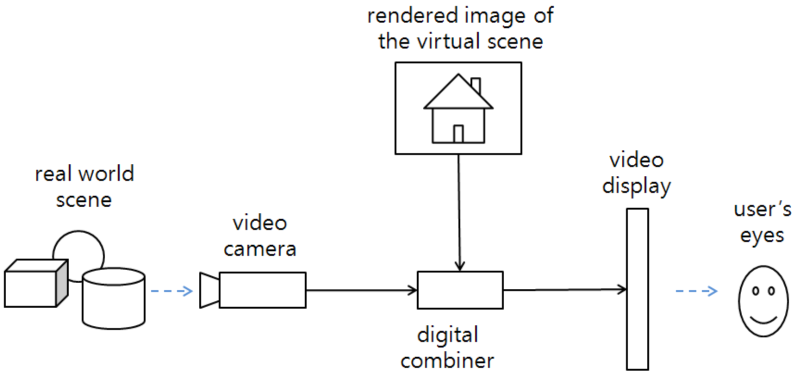


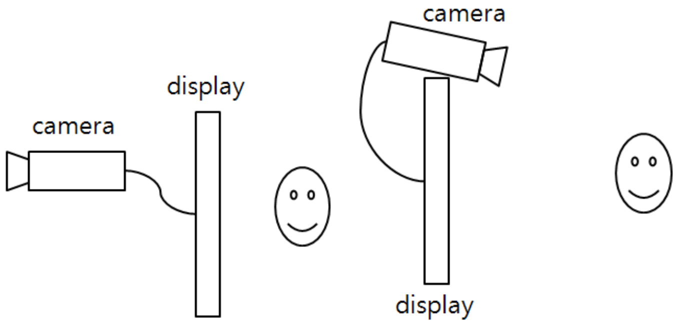
Figure 5.2: Structure of video based AR displays.

digitizes the real world scene using a video camera system, so that the image of the real environment could be composited with the rendered image of the virtual scene using digital image processing technique. Figure 5.2 shows the structure of video based AR displays.

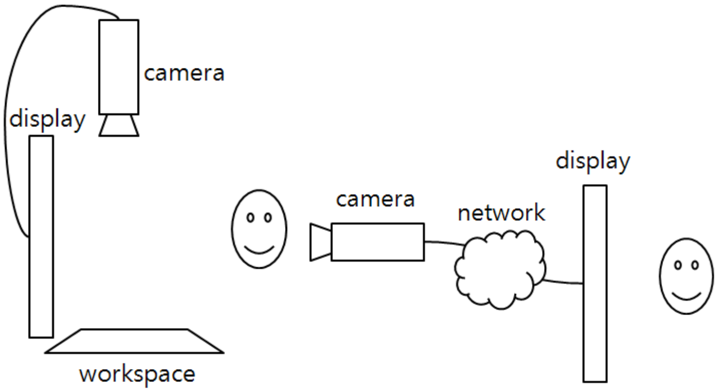
In many cases, the video camera is attached on the back of the display, facing towards the real world scene, which the user is looking at when watching the display. These types of displays create the digital illusion of seeing the real world "through" the display, hence they are called "video see-through" displays. In other cases, the camera can be configured in different ways such as facing the user to create virtual mirror type of experience, providing a top down view of a desk, or even placed at a remote location. Figure 5.3 shows various configurations.

Video based AR displays are the most widely used systems due to the available hardware. As digital camera become popular in various types of computing devices, video based AR visualization can be easily implemented on a PC or laptop using webcams and recently even on smartphones and tablet computers.

Video based AR also has an advantage of being able to accurately control the process of combining the real and virtual view images. Advanced computer vision based tracking algorithms provide pixel accurate registration of virtual objects onto live video images (see section 4). Beyond geometric registration of the real and virtual worlds,



(a) Video see-through **(b)** Virtual Mirror

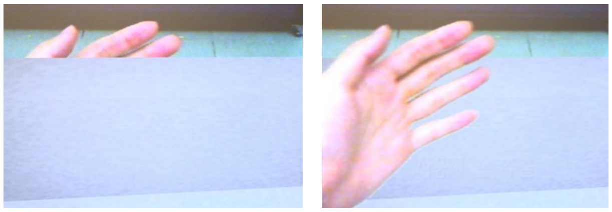


(c) Augmented Desk **(d)** Remote AR

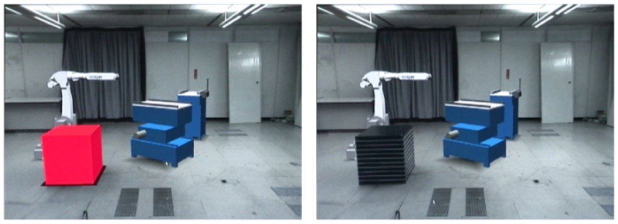
Figure 5.3: Different configurations of placing camera in Video based AR visualization.

with video based AR displays, real and virtual image composition can be controlled with more detail (e.g. correct depth occlusion or chromatic consistency) using digital image processing.

One of the common problems in composition of virtual and real world images is incorrect occlusion between real and virtual objects due to virtual scene image being overlaid on top of the real world image. Video based AR displays can easily solve this problem by introducing depth information from the real world, and doing depth tests against the virtual scene image which already has depth data. The most naive approach is to get depth information of the real world image based



(a) Chroma-keying (user's hand with a virtual structure) [Lee et al., 2010a]



(b) Using mask objects (a real box with a virtual robot) [Lee et al., 2008a]

(c) Using depth camera (a real book with a virtual toy car) [Clark and Piumsomboon, 2011]

Figure 5.4: Compositing AR images with correct depth occlusion.

on heuristics and use chroma-keying techniques when compositing the two images. For instance, an AR system can identify the region in the real world image with skin colour to prevent the user's real hand being occluded by virtual objects (see Figure 5.4(a)).

Advanced methods can estimate or measure the depth information of the real world image. Breen et al. [1996] proposes two ways of obtaining depth image of the real world scene: (1) using mask objects and (2) depth cameras. Mask objects (a.k.a. ghost objects) are virtual objects that represent corresponding physical objects, and by rendering them only into the depth-buffer of the computer graphics pipeline the system can obtain the depth information for the real world scene (see

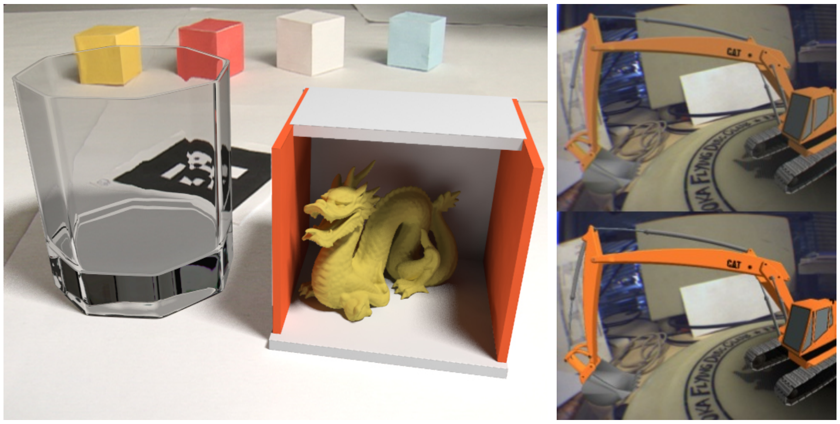


Figure 5.5: Compositing AR image with correct chromatic consistency. (left: a virtual glass and a virtual box with a dragon [Kán and Kaufmann, 2013], right: a virtual excavator [Klein and Murray, 2010])

Figure 5.4(b)). Compared to using mask objects that needs manual modelling of the real world scene, depth cameras are becoming widely available (e.g. Microsoft Kinect) and they can capture the depth image of the real world scene together with the RGB image (see Figure 5.4(c)).

Differences in lighting [Agusanto et al., 2003, Kán and Kaufmann, 2013] and colour space [Klein and Murray, 2010] between the real and virtual scenes can be also reduced relatively easily in video based AR systems during image composition process, as shown in Figure 5.5.

Other benefits of video based AR displays include having no temporal gap between the real world image and the rendered virtual scene, and being more flexible with controlling the field of view of the AR scene using wide FOV lenses on the camera.

The most prominent problem of video based AR displays is having an indirect view of the real world. Since the real world view is provided by video image captured through a camera, it has limitations in terms of resolution, distortion, delay, and eye displacement. These problems can be critical in safety demanding applications where direct view of the real world is necessary. There are efforts to reduce these problems, such as using a camera with higher resolution and faster update rate, undistorting the image while rendering, and using a specially designed optical system to match the eye displacement [Takagi et al., 2000]. However, there will be certain amount of limitations remaining compared to other types of AR displays.

A final limitation of video based AR displays is requiring more computational power. While recent advance in computing hardware made even mobile phones able to run AR applications in real time, compared to other types of displays that use optical systems for image composition, video based AR demand more computing power for combining real and virtual view images.

5.1.2 Optical See-through AR Displays

Optical see-through AR displays use optical systems to combine virtual view images with the real world view. The optical system in this type of AR display usually includes beam splitters (e.g. half mirrors or combined prisms) that combine the real world view seen through the splitter with the reflection of image from a video display (see Figure 5.6). Head up displays (HUD) on airplane cockpits or modern cars are typical examples of optical see-through AR displays using beam splitters in this way. Virtual mirrors based on optical systems combine the real and virtual view images in the other way around. In this case, it is the real world scene that is reflected on the half mirror while the user can see the video display through the mirror.

Transparent projection films are another type of optical component used for optical see-through AR displays. This type of projection film not only diffuses the light to show the projected image, but is also semitransparent so that the viewer can also see what is behind the screen through it. See Figure 5.7 for examples of optical see-through displays.

Most optical see-through AR displays use optical combiners, separate from the displays, for combining the real and virtual view images, however recently there have been technical advances in making the display itself transparent. With the development of transparent flat panel displays (e.g. LCD), their use in AR visualization is being actively investigated. Using transparent flat panel displays is expected to contribute to simplifying and miniaturizing the structure and size of optical see-through AR displays.

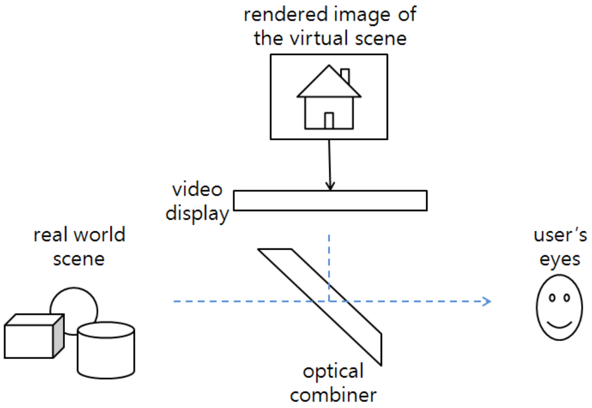


Figure 5.6: Structure of optical see-through displays.

Compared to video based AR displays, the most prominent advantage of optical see-through AR displays is providing a direct view of the real world. With a direct view of the real world, optical see-through displays do not suffer from limitations in resolution, lens distortion, eye displacement, or time delay. This could be a very important feature for safety demanding applications that require a direct view of the real object, such as medical or military applications. Optical see-through AR displays also needs simpler electronic components and demands less processing power since the composition of real and virtual view image is achieved physically.

The main problems in optical see-through AR displays are less accurate registration between the real and virtual view images. In many cases, optical see-through displays require manual calibration process which ends up with relatively poor quality of registration compared to automated computer vision based approach in video based AR displays. Since the calibration parameters are dependent on the spatial relationship between the user's eyes and the display's image plane, there is a higher chance of parameters changing over time (e.g. wearable displays sliding off from the original position) and eventually causing misalignment between the real and virtual view images. Hence, accurate 3D eye tracking (relative to the display) becomes more important to visualize a correctly aligned AR scene with optical see-through displays.

Figure 5.7: Optical see-through displays using half-mirror [Hilliges et al., 2012] (top), transparent projection film (lower left, `http://www.laser-magic.com/transscreen.html`), and transparent LCD (lower right).

Temporal delay between the real world and the virtual views is another challenging problem in optical see-through displays. Even with a spatially accurate tracking system, there always would be temporal delay for tracking physical objects. As the virtual view is updated based on the tracking results, unless the physical scene and the user's viewpoint is static, the virtual view will have temporal delay compared to the direct view of the real world in optical see-through displays.

In many cases, visualizing correct depth occlusion between the real and virtual view images is more challenging with optical see-through displays. Due to the physical nature of optical combiners, users see semi-transparently blended virtual and real view images with most of the optical see-through displays, neither of the images fully occluding the other. To address this problem Kiyokawa et al. [2001] developed an electronically controlled image mask that masks the real world view where the virtual objects are to be overlaid. By closing the pixels of the LCD panel on the region where the virtual objects are visualized, the real world view was optically occluded and the virtual view image from the display became more vivid.

Lighting conditions in the real world environment can also affect perceived brightness of optical see-through displays. In many cases, optical combiners have a fixed physical attribute of amount of transparency that can lead to unbalanced brightness between the real world view and virtual view image depending on the lighting conditions in the real world. In outdoor environments, the virtual view image will appear relatively dim compared to the bright real world view. To overcome this problem, some types of head mounted display provide a replaceable cover with various level of transparency, while some researchers also experimented using grayscale LCD shutters to control the brightness of the real world view.

5.1.3 Projection based AR Displays

While other types of displays combine the real and virtual world view at the display's image plane, projection based AR displays overlay virtual view images directly on the surface of the physical object of interest (see Figure 5.8), such as real models [Raskar et al., 1999, 2001] or walls [Raskar et al., 2003]. In combination with tracking the user's viewpoint and the physical object, projection based AR displays can provide interactive augmentation of virtual image on physical surface of objects [Raskar et al., 2001].

In many cases, projection based AR displays use a projector mounted on a ceiling or a wall. While this could have an advantage of not requiring the user to wear anything, this could limit the display to be tied with certain locations where the projector can project images. To make the projection based AR displays more mobile, there have been efforts to use projectors that are small enough to be carried by the user [Raskar et al., 2003]. Recent advance in electronics miniaturised the size of the projectors so that they could be held in hand [Schöning et al., 2009] or even worn on the head [Krum et al., 2012] or chest [Mistry and Maes, 2009].

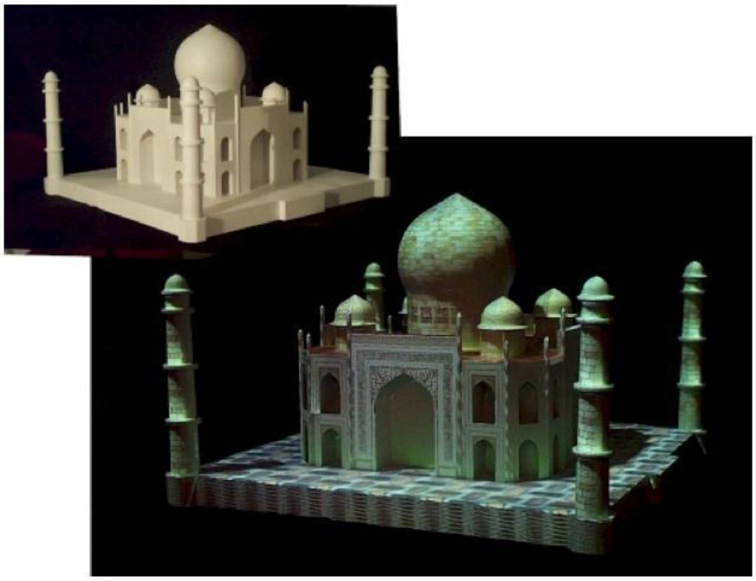


Figure 5.8: A physical model of Taj Mahal augmented with projected texture. [Raskar et al., 2001]

As projection based AR displays display virtual images directly on real object surface, it requires a physical surface where the image can be projected onto. This could limit its application to augmenting only objects near to the user, as it might not be appropriate to projecting images on far away objects as in outdoor AR applications. Other limitations of projection based AR displays include being more vulnerable to lighting conditions, and suffering from shadows created by other physical objects (such as user's body). While stereoscopic projectors can be used for presenting virtual objects that pop out from the projection surface, providing correct occlusion between another physical object (such as user's hand) and the virtual object could be more challenging compared to video or optical see-through displays.

5.1.4 Eye Multiplexed AR Displays

While the three methods above provide the final combined AR image to the user, another approach for visualization in AR applications is to

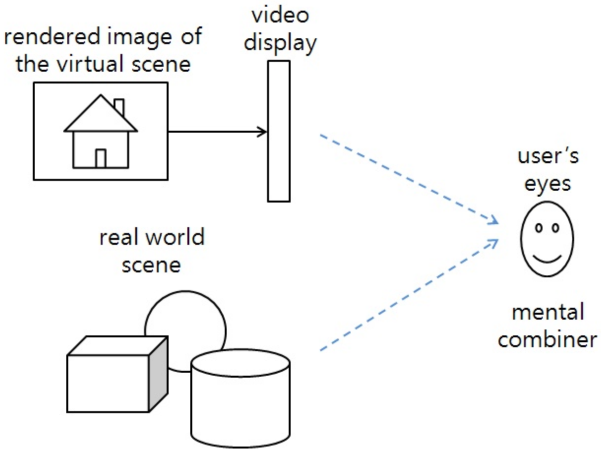


Figure 5.9: Structure of Eye Multiplexed AR Displays.

let users combine the views of the two worlds mentally in their minds. We refer to such a way of using displays as "eye multiplexed" AR visualization. As illustrated in Figure 5.9, with eye multiplexed AR displays, the virtual scene is registered to the physical environment, hence the rendered image shows the same view of the physical scene that the user is looking at. However, the rendered image is not composited with the real world view, leaving it up to the user to mentally combine the two images in their minds. To ease the mental effort of the user in this case, the displays are needed to be placed near to the user's eye and should follow the user's view, so that the virtual scene shown on the display could appear as an inset into the real world view (see Figure 5.10).

As in the case of optical see-through and projection based AR displays, eye multiplexed AR displays do not require digital composition of the real world view and the virtual view image hence demands less computing power compared to video based AR displays. Furthermore, eye multiplexed AR displays are more forgiving of inaccurate registration between the real and virtual scene. As the virtual view is shown next to the real world view, pixel accurate registration is not necessary but showing the virtual scene from the perspective of the real world view is similar enough for the user to perceive it as the same view.

Figure 5.10: Eye multiplexed AR visualization of virtual buildings on Google Glass as an inset in the top right corner. `http://arforglass.org`

On the other hand, as the matching between the real world view and the virtual view is up to the users' mental efforts, the visualization is less intuitive compared to other types of AR displays that show the real and virtual scene in one single view.

5.2 Eye-to-World spectrum

While AR displays use different approaches to combine the real and virtual views, they can also be categorized based on where the display is placed between the user's eye and the real world scene. Bimber and Raskar [2006] summarises this in a diagram shown in Figure 5.11.

5.2.1 Head-attached Displays

Head-attached displays present virtual images right in front of the users' eyes. No other physical objects come between the eyes and the virtual image from the display, which guarantees the virtual image not getting occluded by other physical objects. Head-attached displays vary in size from a helmet or a goggle to the size of lightweight glasses (see Figure 5.12). As the technology advances, the displays are becoming

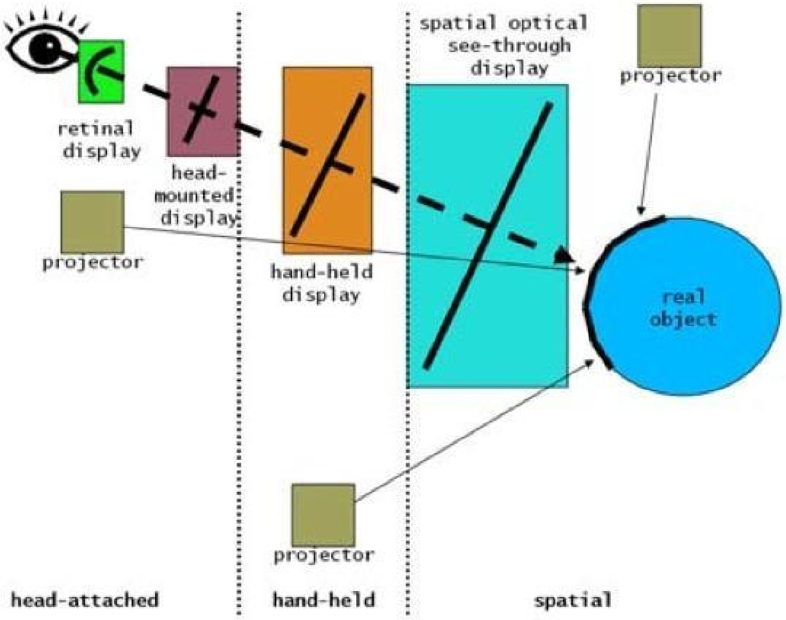


Figure 5.11: AR displays on eye-to-world spectrum. [Bimber and Raskar, 2006]

lighter to wear while providing a wider and brighter image on the display.

Head Mounted Displays (HMD) are the most common type of display that are used in AR research. HMDs mostly are in a form similar to a goggle which users can wear on their head and the display is placed in front of the user's eyes. In some cases they are also referred to as Near Eye Displays or Face Mounted Displays. While video and optical see-through configurations are most commonly used when using HMDs for AR display, head mounted projector displays are also actively researched as the size of the projectors became small enough to wear on user's head.

Recently, with technical developments in wearable devices, smart glass type devices, such as Google Glass or Recon Jet, are also actively investigated for their use in AR applications. While many of the smart glass devices are capable of working as both video or optical see-through AR displays, due to the narrow field of view of the display, eye

Figure 5.12: Various types of head-attached displays. From left to right: video and optical see-through HMD from Trivisio (`www.trivisio.com`), and Google Glass (`http://www.google.com/glass`)

multiplexed AR visualization is also considered as an alternative way of visualizing AR scene on smart glasses.

While most of the head-attached displays use lenses on an eyepiece similar to telescopes or microscopes that does not directly contact with the user's eyeball, more direct methods are under development. For example, in the Virtual Retinal Display [Tidwell et al., 1995] project virtual images are beamed directly onto the retina of the user's eye.

5.2.2 Handheld and Body-attached Displays

Head-attached displays provide a high mobility and visually immersive experience, but in many cases they have certain limitations in terms of wearability (feeling comfortable while wearing), safety issue with indirect real world view, and social acceptance. Handheld or body-attached displays are considered to be mobile and personal, yet sharable with others as needed. They are also more socially acceptable compared to head-attached displays.

Early researchers [Rekimoto and Nagao, 1995, Mogilev et al., 2002] used small LCD monitors tethered to a computer for experimenting with handheld AR experiences. With advances in mobile device technology, handheld computing devices became powerful enough to process AR visualization, and various devices have been used as AR displays, including UMPC (Ultra Mobile PC) [Reitmayr and Drummond, 2006], PDA (Personal Digital Assistant) [Wagner and Schmalstieg,

Figure 5.13: A smartphone with depth imaging camera from Google Tango project.

2003], Tablet computers [Klein and Drummond, 2004], and cell-phones [Mohring et al., 2004].

Today, smartphones and tablet computers are widely adopted and have powerful graphics processors, cameras, and various sensors that can run AR applications. Most recently, even depth imaging sensors are experimented on smartphones and tablets (e.g. Google Project Tango[1]) for advanced tracking and visualization (see Figure 5.13).

While video-based AR display configurations have been widely investigated on handheld devices, handheld optical see-through AR displays are still experimental design concepts, as the electronic components are still under development. Some electronics companies have been producing transparent LCD and OLED panels, and it is expected to be able to use these for building handheld optical see-through AR displays in the near future.

Micro projectors are also another type of devices used as mobile AR displays. While most of the use cases involve holding the projector as a handheld device similar to a flashlight [Beardsley et al., 2005, Schöning et al., 2009], there is also various research on wearing the device on different parts of the body such as on the chest [Mistry and Maes, 2009], shoulder [Harrison et al., 2011], or on a wrist [Blasko et al., 2005].

---

[1] `https://www.google.com/atap/projecttango/`

5.2.3 Spatial Displays

Compared to head mounted and handheld displays, spatial displays are limited in mobility and are usually installed at a fixed location. As spatial displays tend to provide a larger image in many cases, they are more applicable as public displays which multiple users share.

Typical examples of spatial displays are those using a beam splitter (e.g. half mirror) to create optical see-through configuration for AR visualization. A half mirror workbench [Poston and Serra, 1994, Bimber et al., 2001a, Gunn et al., 2003, Hilliges et al., 2012] are one of the examples of an AR display using this configuration, which allows for close interaction with virtual objects and hence are widely used in combination with haptic interfaces (see Figure 5.14). Other work using a similar configuration of optical see-through AR displays include a virtual showcase which can overlay virtual images on a physical object inside a pyramid or cone shaped half mirror case [Bimber et al., 2001b], and using multiple autostereosopic displays above and below a slanted half mirror [Kim et al., 2012].

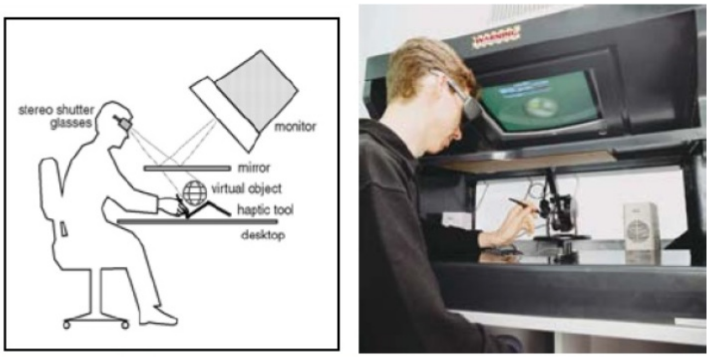


Figure 5.14: AR Haptic workbench. [Gunn et al., 2003]

Recently, with advances in transparent active display components, such as transparent LCDs and OLEDs, it is expect to see more wide adoption and application of AR displays used in show cases or even show windows (see Figure 5.7). Another common optical element used for creating optical see-through spatial AR displays is transparent projection film as described in §5.1.2. These types of screens are also known as 'holographic projection film' that are transparent yet diffuses the projected image onto the surface of the screen so that the user can see both the projected virtual image and the physical scene behind the screen.

Instead of using a transparent projection screen, projecting the virtual image directly onto the surface of physical objects is another common approach used for implementing spatial AR display. As described in §5.1.3, projection based AR displays overlay virtual images directly on the surface of the physical object of interest (see Figure 5.8), which can be physical models [Raskar et al., 2001, 1999] or walls [Raskar et al., 2003].

Spatial displays can also use video-based AR visualization. A desktop monitor paired with a video camera pointing on the desk in front of the monitor is one of the widely used setup, a.k.a. Desktop AR or Video-Monitor AR, for demonstrating AR applications. Another example of using video-based AR visualization on a spatial display would be virtual mirror like setup where a large screen (either a monitor or a projection screen) with a camera is pointing towards the user standing in front, showing the real world around the user and overlaying virtual objects into the scene.

Using other types of 3D display technology such as volumetric 3D displays [Favalora, 2005] and autostereoscopic displays [Dodgson, 2005] for AR visualization is expected to be an interesting topic to investigate in the future, together with using a combination of different AR display technologies together [Lee et al., 2006] to provide richer AR experience.

5.3 Other Sensory Displays

While most of the research in AR focuses on visual augmentation of the real world, there is also work on AR displays for the other senses, such as audio or haptic. These types of non-visual displays are mostly adopted from the field of Virtual Reality (VR) where the main focus is to replace all human sensory input with computer generated virtual signals. A lot of this work in VR can be applicable in AR research as well, while there are also certain specific topics that arise in AR applications. In this section we briefly introduce the non-visual sensory displays that are used in AR research and some of the work that focuses on their use in AR applications.

Compared to visual augmentation, computer generated sounds are mixed with real world sound by nature and the main challenge of providing augmented sound to the user is spatializing the sound in the 3D space. This has been widely investigated using various technologies including stereo headphones with HRTF (Head-Related Transfer Function) [Zotkin et al., 2003] or multi-channel speaker systems. While traditional multi-channel 3D surround speaker systems use dedicated speakers for each direction of the sound that has to be placed in space around the user, digital sound projectors [Takumai, 2011] use an array of speakers to generate 3D sound utilizing interference and reverberation of sound in a room.

Traditional spatialized 3D sound technology can be easily applied in AR systems, and there are attempts to investigate AR specific issues in audio displays. For instance, Lindeman et al. [Lindeman et al., 2007] proposed a concept of "hear-through" and "mic-through" Augmented Reality analogous to optical and video see-through AR visualization, and demonstrated its feasibility through a user study with bone conduction headset.

Haptic displays have been also well investigated in the VR research field. Among various haptic devices, robotic arm based mechanical devices (as shown in Figure 5.14) and vibro-tactile feedback available on various modern mobile devices (e.g. smartphones) are the most widely investigated in haptics research, and are also applied in various AR research. Extending Milgram's Reality-Virtuality continuum [Milgram and Kishino, 1994], Jeon and Choi [2009] proposes a taxonomy for haptic AR based on a composite visuo-haptic reality-virtuality continuum and summarize various approaches for AR systems using haptic interfaces to let the user feel a real environment augmented with synthetic haptic stimuli.

Augmenting olfactory and gustatory senses is rarely explored compared to other sense, yet considered as promising research area to complement the other sensory experience. Narumi et al. [2011] investigated

Figure 5.15: Meta cookie - augmenting olfactory and gustatory experience. [Narumi et al., 2011]

augmenting olfactory and gustatory sense through visual and olfactory augmentation of cookie eating experience. The system built by the research team, named Meta Cookie (see Figure 5.15), uses a plain cookie with AR marker printed on it (with edible colour) to visually augment different toppings to the cookie and as the user eats the cookie, a specially designed scent distribution device provides olfactory stimuli to reproduce the different flavours of the cookie.

5.4 Summary

In this section we have provided an overview of a variety of different types of displays that can be used to combine real and virtual images. Although most early AR systems used head mounted displays, the most common display type currently used is a handheld device such as a mobile phone or tablet. However this may change with the emergence of lower cost head worn wearable computers and displays, and also with new research in different display systems. In the next section we review tools for creating AR systems using the tracking and display technology discussed.

6

AR Development Tools

In the previous sections we have reviewed important tracking and display technologies that can be used to create Augmented Reality experiences. However these are hardware devices and require AR application software to provide that experience. In this section we discuss software libraries and tools for developing AR applications. As Table 6.1 shows, there are numerous tools for implementing AR systems that can be arranged in a hierarchy of decreasing programming skill required to use them; from low level software libraries that require consider skill on the part of the developer, to simple authoring tools for novice users that assume no programming ability. In this section we will review each of these types of tools in term, covering both research and commercially available products.

AR development tools also have different target application platforms. Some tools are for developing desktop AR applications while others target mobile AR application. Some support computer vision based tracking for developing AR application, mainly for indoor use, while others use sensors on a mobile device for developing outdoor AR application. So each of the types of tools will also be discussed in terms of the platforms that they are designed for.

Table 6.1: Hierarchy of AR development tools from most complex to least complex.

Type of Tool	Skill Required	Example
Low-level software library/framework	Strong programming/coding ability	ARToolkit, osgART, Studierstube, MXR-ToolKit
Rapid prototyping tools	Some programming ability, but design/prototyping skills	FLARManager, Processing, OpenFrameworks
Plug-in for existing developer tool	Skill with the developer tool that the plug-in works with	DART, AR-Media plug-ins, Vuforia and Metaio Unity plug-ins
Stand-alone AR authoring tools	No programming ability, but can learn stand-alone tool	BuildAR, Metaio Creator, Layar Creator, Wikitude Studio

In the rest of this section we provide an overview of tools available for developing AR applications. We first look into low-level software development tools for those who need full control of the technology with programming skills. Then we present plug-ins that work with existing applications, followed by stand-alone AR authoring tools for users with a non-programming background. We next present rapid prototyping tools that can be used to quickly develop AR concepts, and finally research into the next generation of AR authoring tools. The tools presented will be both novel tools published in many research papers, and those available in public domain that readers can access.

6.1 Low Level software libraries and frameworks

Low level software libraries and frameworks for AR typically provide access to the core tracking and display functionality need to build an AR application. For example, in the previous section we introduced ARToolkit[1] one of the most widely adopted software library for developing AR applications. ARToolKit provides marker based tracking in the open source version, while the commercial version from ARTool-

---

[1] `http://artoolkit.sourceforge.net`

Works[2] also supports tracking natural features on an arbitrary image. In this way it provides a solution for the key problem of being able to know where the user view is currently. ARToolkit is written in the C language and it supports various platforms including Windows, Linux, and Mac OS X desktop operating systems. However, the main function of ARToolKit is to provide tracking and so there is only low-level OpenGL based rendering, and no built-in support for interaction techniques or complex graphics. With ARToolKit alone it is difficult and time consuming to develop a complete AR application.

This limitation is can be overcome by using the osgART[3] library, which provides a link between ARToolkit tracking and the OpenSceneGraph[4] library for scene graph based real-time rendering. Scene graphs represent computer graphics scenes as a node based graph, and provide one way to manage the functionality of complex graphics applications, such as being able to sort objects according to their visibility, remove objects that are out of the camera field of view, and provide state management. They enable the developer to worry about interactions between virtual objects rather than low-level object rendering. OpenSceneGraph is open source and is one of the most widely used scene graph libraries. It is written in C++ and uses OpenGL for it's graphics rendering, and has support for a large number of plug-ins that provide support for additional functionality such as model loading, shadow rendering, and physics.

By combing ARToolKit and OpenSceneGraph together, osgART makes it very easy for developers to create interactive AR applications with advanced rendering techniques. The osgART library includes nodes for video capture, video viewing and for camera tracking. Figure 6.1 shows a simple osgART scene graph, showing nodes for video input and tracking, combined with nodes for graphics rendering. The end result is a simple AR scene with a 3D model of a cow overlaid on an ARToolKit tracking marker. The osgART library is designed using a modular approach meaning that it is easy to replace one module for an-

---

[2] `http://www.artoolworks.com`

[3] `http://www.osgart.org`

[4] `http://www.openscenegraph.org`

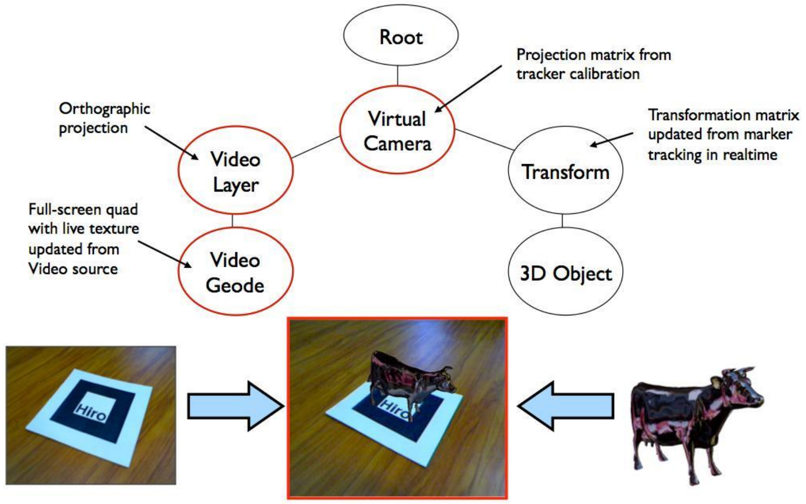


Figure 6.1: Simple osgART scene graph for AR scene representation.
`http://www.osgart.org`

other in the application. So, if developers don't want to use ARToolKit as the base tracking library, they can replace it with another tracking solution, such as the BazAR[5] image tracking library, or PTAM[6].

Oda and Feiner's Goblin XNA platform[7] uses a similar approach, combining the artTag tracking library with the Microsoft XNA game platform[8] to create an infrastructure for developing AR games, on top of an existing non-AR game development environment. GoblinXNA includes modules for AR tracking, 3D user interface elements, video capture, physics based interaction, and displaying output into AR head mounted displays. Using Goblin XNA it was relatively easy to modify an existing XNA car racing game, and make an AR version of it [Oda et al., 2008].

Another AR framework developed for building complete systems was Studierstube[9] [Schmalstieg et al., 2002]. This was based on a het-

---

[5] `http://cvlab.epfl.ch/software/bazar`
[6] `http://www.robots.ox.ac.uk/~gk/PTAM`
[7] `http://graphics.cs.columbia.edu/project/goblinxna`
[8] `http://xbox.create.msdn.com/en-US`
[9] `http://www.cg.tuwien.ac.at/research/vr/studierstube`

erogeneous distributed architecture that allowed the user easily incorporate multiple different display types (AR HMDs, projected displays, desktops, etc) with a variety of input devices and reconfigurable tracking hardware. Studierstube was designed from the ground up to support collaborative applications using a distributed shared scene graph and so many of the early application were for shared AR experiences. Another of its advantages was that it was very easy to rapidly incorporate different interaction techniques such as Personal Interaction Panels [Szalavári and Gervautz, 1997], 3d viewports, or mobile interfaces.

While Studierstube, osgART and GoblinXNA are freely available as academic research projects, there are also commercial solutions available. For example, the Metaio SDK[10], provides a comprehensive set of AR tracking technologies, including marker tracking, image tracking, 3D object tracking, face tracking, and even an external Infrared tracker sold by Metaio. In terms of visualization, it provides a simple VRML renderer and an option to use OpenSceneGraph together.

For mobile applications, the Studierstube platform originally developed for desktop applications was ported over to the Windows CE and a PDA platform [Wagner et al., 2005], and then Android, with the Studierstube Light library. However there were many components of Studierstube that did not have equivalent libraries on the PDA platform and so had to be written from scratch. This included a mobile rendering library (KLIMT), tracking (OpenTracker) and libraries for hardware abstraction (PocketKnife), as well as existing libraries for mobile SceneGraphs (Coin) and communication (ACE). This framework was then used to build many mobile AR applications, including The Invisible Train [Wagner et al., 2004], the first self-contained handheld collaborative AR experience. Figure 6.2 shows the typical components used to build a mobile AR application using Studierstube Light and the Invisible Train application.

ARToolkit for Mobile[11] is a version of the ARToolkit library available for mobile platforms including iOS and Android. It is also integrated with the OpenSceneGraph library for rich real-time computer

---

[10] `http://www.metaio.com/products/sdk/`

[11] `https://www.artoolworks.com/products/mobile/artoolkit-for-mobile`

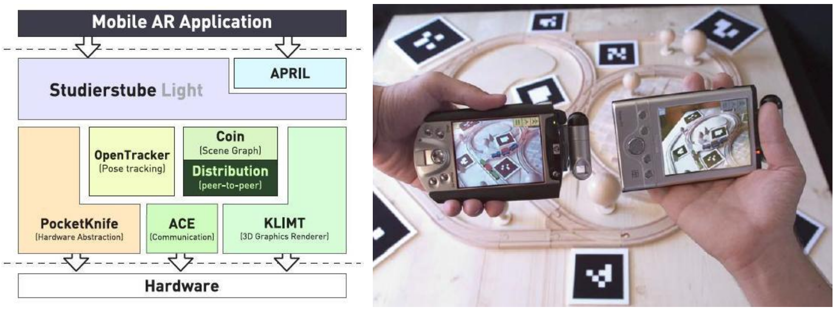


Figure 6.2: Studierstube mobile AR framework (left) and the Invisible Train app developed with it (right). [Wagner et al., 2005, 2004]

graphics rendering. Since it is a porting of C language based desktop version of ARToolkit, ARToolkit for Mobile needs a native development kit to use it on Android platform. Alternatively, the AndAR library[12] fills this gap by providing Java based wrapping classes of ARToolkit library, making the development environment friendlier to Android and Java developers.

For Android, DroidAR[13] also provides vision-based marker tracking, which is based on the OpenCV computer vision library and uses square markers similar to that of ARToolkit. One of the most popular low level libraries for mobile AR applications is the Vuforia library from Qualcomm that specializes in tracking natural features of arbitrary images, and also supports tracking 3D objects with multiple planar image. Vuforia supports both iOS and Android platforms.

Finally, the Metaio SDK supports both fiducial markers and natural feature based image tracking, and recently added a feature for tracking an arbitrary 3D structure in the real world scene. This allows users to track camera position from 3D models that provides more robust tracking (see Figure 6.3). It also offers its own 3D rendering engine that allows developers to work in high-level in terms of 3D visualization. Mobile SDK is also available both for free and under commercial license.

The low level libraries mentioned so far use computer vision techniques for AR tracking. However there are also libraries that use sen-

---

[12] `http://code.google.com/p/andar`

[13] `https://github.com/bitstars/droidar`

Image courtesy of Metaio.

Figure 6.3: Metaio model based tracking. `http://www.metaio.com`

sors for outdoor location based AR. In addition to computer vision based tracking, DroidAR also supports developing location-based mobile AR applications using sensors (e.g., GPS, intertial) on the mobile device. The Metaio Mobile SDK also provides a solution for developing location-based mobile AR applications. In addition to sensor based outdoor location tracking, it provides a partial solution for indoor environment by coding location information in a square fiducial marker.

While these frameworks focus on providing low-level functionalities, the HIT Lab NZ OutdoorAR[14] library [Lee and Billinghurst, 2013] provides high-level abstraction of geo-located scenes. Designed in a component based structure, the library allows developers to focus on high-level design of AR applications such as domain content and logic and user interface design, while it also provides direct access to low-level functionalities.

In addition to the low level libraries mentioned there are many other SDKs that can be used for low-level AR application development. The Technical University of Graz has a list of all available libraries[15].

6.2 Rapid Prototyping/Development Tools

Using a low level software library and framework like Studierstube or OsgART provides the maximum flexibility in development of the AR

---

[14] `http://www.hitlabnz.org/mobileAR`

[15] `http://www.icg.tugraz.at/Members/gerhard/augmented-reality-sdks`

application, but can also be time consuming and requires considerable programming skill. In some cases it can be helpful to rapidly prototype an idea to show to end users or clients before undertaking a major development exercise. In this section we briefly overview a range of different tools that could be used for rapid prototyping of AR experiences.

Adobe Flash[16] is one of the most popular authoring tools for creating multimedia content for the web and desktop platform, and can also be used to prototype a variety of desktop and mobile experiences. Flash developers can easily create AR experience by using one of the AR plug-ins available, such as FLARToolKit[17] or FLARManager[18]. FLARToolKit is a version of ARToolKit ported over to Flash that allows people to have AR experiences from within their web browser. Just by writing a few lines of code users can activate their camera, look for AR markers in the camera view, and load and overlay virtual content onto the tracked image. FLARToolKit makes it possible for anyone with a Flash-enabled web browser and camera connected to their computer to have an AR experience, and also makes it easier to develop those experiences. These attributes make it a perfect platform for prototyping AR applications.

The development of FLARToolKit has lead to an explosion of web-based AR applications, especially for marketing. For example, when GE were seeking a way to explain about Smart Grid power technology they created a website with a simple AR marketing experience that would allow people to see a variety of different clean energy power generation operations superimposed over a printed marker on paper. Users would hold up the paper and see a virtual windfarm appear, and when they blew, the blades of the windmills would turn around faster. Users found this very engaging and uploaded videos to YouTube of themselves playing with the virtual objects, which attracted millions of views. Since this early campaign there have been hundreds of websites that have provided other similar AR marketing experiences.

---

[16] `http://www.adobe.com/flash`

[17] `http://www.libspark.org/wiki/saqoosha/FLARToolKit/en`

[18] `http://words.transmote.com/wp/flarmanager`

FLARManager is the equivalent of osgART for Flash. Just like ARToolKit or ARTag, FLARToolKit just provides AR tracking services with little support for 3d graphics or interaction. FLARManager provides an easy way to add more interactivity to Flash based AR applications and also combine a variety of different AR tracking and graphics libraries together. Table 6.2 shows the different Flash based AR tracking and 3D graphics libraries supported by FLARManager. Using the FLARManager library it is very easy for users to build Flash based AR applications with support for complex 3D model and animation loading, setting different camera parameters, loading of tracking images and many other features.

Table 6.2: FLARManager support for different 3d graphics and AR tracking libraries.

AR Tracking Library	**3D Graphics Library**
FLARToolkit	Alternativa3D
flare*tracker	Away3D
flare*NFT	Away3D Lite
	Papervision3D
	Sandy3D

There are also similar Flash based commercial solutions available from companies such as Metaio and Total Immersion. As an alternative to Flash, there are also AR plug-ins for other authoring tools, such as SLARToolKit[19] for Microsoft Silverlight. Recently researchers have also begun exploring how to using HTML5 to create web-based AR experiences that don't require any third-party browser plug-in. For example, Oberhofer et al. [2012] has shown how Javascript, WebGL and HTML5 can be used to implement AR natural feature tracking viewable on desktop or mobile web browsers. Similarly Ahn et al. [2013] have developed a complete mobile AR framework using just normal HTML and other web component technologies.

In addition to web-based AR prototyping tools, there are a number of desktop and mobile tools for rapidly creating AR applications. One of the most popular tools to use is OpenFrameworks[20], an opensource

---

[19] `http://slartoolkit.codeplex.com`

[20] `http://openframeworks.cc`

C++ middleware library that makes It very easy to build interactive experiences. It provides a framework for combining a wide range of different interactive technologies, and is designed to work as a general purpose glue that wraps together several commonly used libraries[21]. For example, an AR application with gesture-based input could be developed by combining a tracking library such as ARToolKit, with hand tracking libraries, 3D scene graph libraries and perhaps audio playback libraries. Previously this would have required using very different programming APIs for each library and sometime incompatible event mechanisms. Openframeworks provides a common API for all these different libraries and centrally manages application messaging and events.

For mobile applications, one of the most useful prototyping tools are AR browser applications such as Junaio, Layar or Wikitude. These are free applications that run on the mobile device, but connect back to servers for providing content and specifying the user interface elements on the mobile. Using these tools developers rapidly prototype application ideas by providing their own content information. For example, Junaio server architecture allows content developers to provide content on their own server which could be accessed by the mobile AR browser through the Junaio server. Junaio provides support for either image based tracking or location based tracking using GPS and compass sensor input. So with a small amount of code on the server developers can rapidly test out their mobile AR application ideas.

6.3 Plug-ins to Existing Developer Tools

The software libraries mentioned in the previous section require considerable programming ability to use them for creating AR applications. However there are other tools that are plug-ins to existing software packages that add AR functionality to existing interactive 2D and 3D content authoring tools. AR plug-ins for non-AR authoring tools are useful when one already knows how to use the authoring tool that the plug-in supports. These types of tools simply add AR visualization

---

[21] `http://ofxaddons.com`

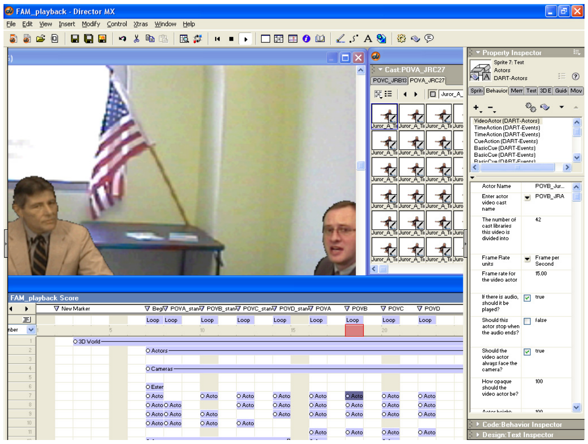


Figure 6.4: DART plug-in interface inside Macromedia Director. [MacIntyre et al., 2004]

and tracking function to existing authoring tool, and rely on content authoring functions provided by the main authoring tool.

One of the good examples of plug-in based AR authoring tool is the Designers AR Toolkit (DART)[22] [MacIntyre et al., 2004]. This was designed as a plug-in to the popular Macromedia Director software for multimedia development. DART was created specifically for designers to enable them to rapidly develop AR applications with a tool that they were already familiar with. Using a the Director graphical interface (see Figure 6.4) the user was able to mix drag and drop visual input with lingo scripting to create interactive AR experiences. Using the plug-in architecture of Director, DART provided AR specific extensions such as support for ARToolKit marker based tracking, physics, 3D camera viewing, and live video display. The authors report that designers found DART very easy to use and were able to successfully create a wide range of different AR applications, such as AR tours of historic sites, and AR theatrical experiences.

There are also various AR plug-ins available for 3D content authoring tools, as well. For example, AR-Media[23] provides a range of

---

[22] `http://ael.gatech.edu/dart/`

[23] `http://www.inglobetechnologies.com/en/products.php`

plug-ins for the Trimble SketchUp 3D modelling software and other tools such as 3ds Max, Maya and Cinema 4D. Using AR-Media plug-ins, content creators can model in SketchUp, then immediately see their model in an AR scene. There is also a free AR-Media player that can be distributed to allow other people to see the AR experience. However, while 3D modeling and animation software can be useful for creating 3D scenes, they have limitations in providing interactive features.

To create highly interactive 3D applications, there are also various interactive 3D graphic authoring tools such as Unity[24], Vizard[25], and Quest3D[26]. While most of these tools have their own plug-ins for AR visualization, Unity has widest range of solutions available for AR visualization. For example, ARToolkit for Unity from ARToolworks[27], provides both marker and image based tracking and AR visualization. Similarly, the Vuforia tracking library has an AR plug-in that works with Unity3D to create mobile AR applications for the iOS and Android platform. In this case developers can use the normal Unity3D visual programming and scripting interface to create rich interactive experiences that can then be

While AR plug-ins provide a easy and quick method for developing AR applications, they have certain limitations. User interface and interaction design could be limited to those provided by the authoring tool, and the software and hardware configuration also depends on what is provided by the tool. In addition, the content produced with an authoring tool usually becomes dependent on the proprietary player software. In order to create an AR application with highly customized interface and interaction, it is unavoidable to use low-level software development tools such as software libraries.

6.4 Stand Alone AR Authoring Tools

A third type of AR authoring tool is a stand alone application that allows users to create their own AR experiences will no-programming

---

[24] `http://unity3d.com`

[25] `http://www.worldviz.com`

[26] `http://quest3d.com`

[27] `https://www.artoolworks.com/products/integrated/artoolkit-for-unity`

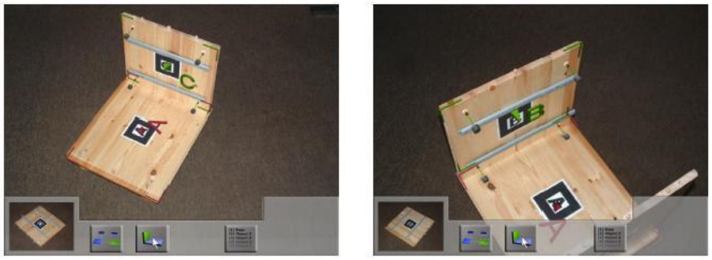


Figure 6.5: AR furniture assembly application created with the AMIRE framework. [Zauner et al., 2003]

experience. Unlike the plug-ins mentioned in the previous section, these tools don't require any additional software to work. A wide range of stand alone authoring tools provides different set of functions. Scene construction, animation, and adding interactive behaviours are the common function sets that many AR authoring tools provide. Some tools only provide scene construction functions, while others have full capability of building complex interactive behaviours. Most of the AR authoring tools provide graphical user interfaces so that a user who has basic personal computer skills could easily learn and use them, and they are useful for non-programmers who want to build AR application.

Early examples of AR authoring tools include AMIRE [Dörner et al., 2003] and CATOMIR [Zauner and Haller, 2004]. AMIRE was designed as an object oriented application framework that could be used to build an AR application out of various available components. AMIRE had a number of features including (1) a generic configuration mechanism of components by so-called properties, (2) slot based communication between components, (3) components for 2D and 3D interaction, and (4) support for prototyping new components. Zauner [Zauner et al., 2003] shows how AMIRE was used to create author an AR application for step by step furniture assembly (Figure 6.5). Träskbäack and Haller [2004] used AMIRE to develop a tablet based Mixed Reality training application for an oil refinery.

However, AMIRE was a component framework that required some programming to use. Zauner and Haller [Zauner and Haller, 2004, Haller et al., 2005] extended this by creating CATOMIR (Component

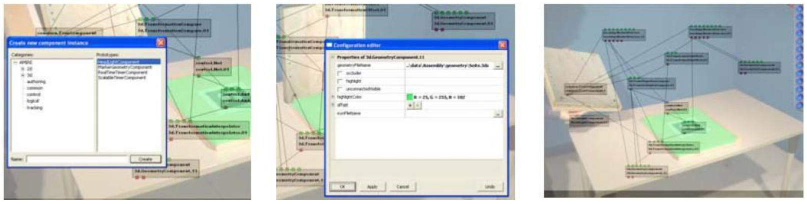


Figure 6.6: Catomir visual interface showing visual links between components. [Haller et al., 2005]

oriented Authoring Tool for Mixed Reality) a visual programming interface based on top of AMIRE. Using CATOMIR a user could drag-and-drop components from a pool and connect them with visual links to define logical behaviours in the AR application. The visual interface was shown overtop of the live camera view and AR scene, allowing the user to switch rapid between authoring and viewing the AR experiences. CATOMIR supported a three-step authoring approach, (1) finding the right components for the application, (2) tweaking the components parameters, and (3) connecting components to define the logical behaviour of the AR application (see Figure 6.6). In their paper they describe how CATOMIR was used to develop a furniture assembly application. However one of the disadvantages of this approach is that AR applications can only have the functionality supported by the components, and it is difficult to add new components.

BuildAR[28] is a standalone AR authoring tool for quickly building AR scenes. It provides computer vision based tracking of both square markers and arbitrary images, and users can add 3D models, images, text, video and sound to the AR scene (see Figure 6.7). While it can play animations built into the 3D model, BuildAR lacks tools for creating animations, and also does not support adding interactive behaviours to the AR scene. The content built by the user is saved into a proprietary file format, and can be viewed using the BuildAR viewer software that is freely available for download. More recently Choi et al. [2010] developed a similar desktop authoring tool that added interactive behaviours to overcome the limitations of BuildAR.

---

[28] `http://www.buildar.co.nz`

Figure 6.7: The BuildAR authoring tool. `http://www.buildar.co.nz`

In addition to basic AR scene construction functions mentioned in BuildAR, D'Fusion Studio[29] from Total Immersion provides a GUI based scene authoring environment using Lua scripting language for describing scenario and behavior of the AR scene. The content is exported into proprietary file format, and it can be shown through its player software.

Some of the AR authoring tools for desktop applications are also capable of publishing content onto mobile platforms. For instance, D'Fusion Studio provides a solution for viewing the content on a mobile platform using its own viewer software. With these AR plug-ins and standalone AR authoring tools, users can use the same authoring environment to create the content and export it to run on different platforms including mobile environment.

On the other hand, there is also AR authoring tools that specifically targets for developing mobile AR applications. Metaio Creator[30] is a simple tool to create AR scenes targeted for the Junaio mobile AR browser. Users can easily include their own tracking images and add content (e.g., pictures, 3D models, video and audio clips) onto it. It also provides simple behaviours that triggers actions (e.g., showing or playing the content, simple animation) based on simple events (e.g., user tapping on the content).

---

[29] `http://www.t-immersion.com/products/dfusion-suite/dfusion-studio`

[30] `http://www.metaio.com/products/creator`

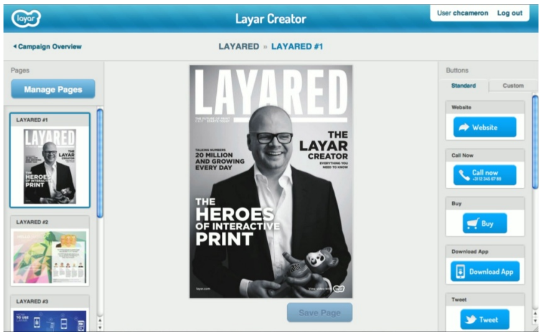


Figure 6.8: Layar Creator, a web-based AR authoring tool. `http://www.layar.com/creator`

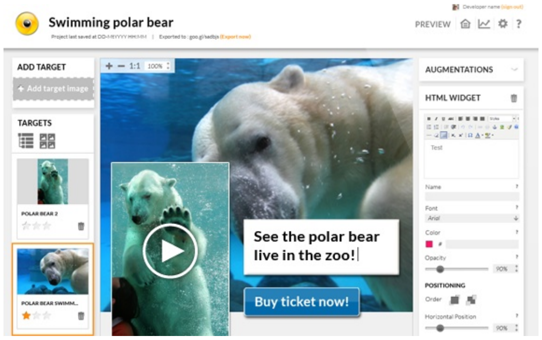
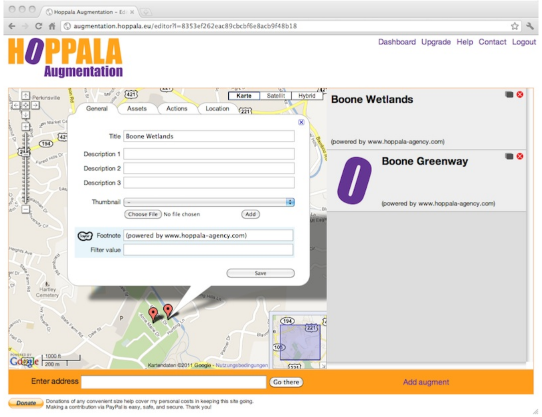


Figure 6.9: Wikitude Studio, a web-based AR authoring tool. `http://studio.wikitude.com`

Recently, web-based authoring tools that can deploy AR content directly onto mobile AR browsers became popular. The Layar Creator[31] is a web based authoring tool that specializes for creating AR links on printed materials (see 6.8). At the Layar Creator website, users upload images of the printed pages to track, and then they can add virtual buttons that have links to various services available on the mobile device, such as opening a webpage or a youtube video, calling to a specific number, or sharing the information on social network service. The created content is published as a layer in the Layar mobile AR browser.

---

[31] `http://www.layar.com/creator`



Title: Boone Wetlands
Description 1
Description 2
Description 3
Thumbnail: -
No file chosen
Footnote: (powered by www.hoppala-agency.com)
Filter value

Boone Wetlands
(powered by www.hoppala-agency.com)

Boone Greenway
(powered by www.hoppala-agency.com)

Kartendaten ©2011 Google - Nutzungsbedingungen

Donations of any convenient size help cover my personal costs in keeping this site going.
Making a contribution via PayPal is easy, safe, and secure. Thank you!

Figure 6.10: Hoppala web-based geo-tagging service. `http://www.hoppala-agency.com`

Another web-based authoring tool Wikitude Studio[32] allows users to create rich mobile AR content and deploy either onto the Wikitude AR browser app or even create a custom mobile app (see Figure 6.9). It supports various types of media including 3D models and animations which can be registered to the real world with computer vision-based tracking technology.

There are also web-based authoring tools focusing on location based mobile AR applications. For example, Hoppala Augmentaion[33] is a web-based geo-tagging tool for mobile AR browsers (see Figure 6.10). Instead of asking developers for a list of geo-coordinates of custom points of interest, it provides an interactive web-based map interface for adding custom geo-tags. The geo-tags created by the user can be browsed on a number of AR browsers including Wikitude, Layar, and Junaio.

6.5 Summary

In this section we reviewed various types of AR development and authoring tools. Low level software libraries and frameworks provide high-

---

[32] `http://studio.wikitude.com`

[33] `http://www.hoppala-agency.com`

est level of flexibility while they require programming skills. On the other hand, standalone authoring tools enable end users to create their own AR content easily with minimal computer skills, although the content created are mostly simple. While commercial authoring tools and AR plug-ins for game engines are widely adopted, the main interface of these authoring tools still remain to be based on 2D GUI which has certain limitations for manipulating 3D content. As the technology advances, it is expected that the authoring and development tools will adopt 3D and AR interfaces more actively as shown in early experiments [Lee et al., 2004b, 2005, Ha et al., 2010, 2012]. In the next section we review interaction technologies for AR systems.

7

AR Input and Interaction Technologies

In this section we review common input methods and interaction technologies for Augmented Reality. AR systems can incorporate various types of input methods, ranging from traditional 2D user interfaces (UI), such as keyboard, mouse and touch screen input, to 3D and multimodal interfaces such as handheld wands, speech, gesture, etc. Different input methods have been used for different types of AR application depending on the user interaction tasks required in the application, and so the type of interface used has evolved over time. There have been a number of AR interface types developed since the 1960's, including:

1) *Information Browsers*: Interfaces for showing AR information on the real world

2) *3D User Interfaces*: Using 3D interaction techniques to manipulate content in space

3) *Tangible User Interfaces*: Using real objects to interact with AR virtual content

4) *Natural User Interfaces*: Using natural body input such as free hand gestures

Figure 7.1: NaviCam AR browser. [Rekimoto and Nagao, 1995]

5) *Multimodal Interfaces*: Using combined speech and gesture input

We will review each of these interface types in turn and describe some interaction technologies used to implement them.

7.1 AR Information Browsers

AR information browsers are one of the representative types of AR applications where AR displays are considered as a window into an information space, and the main task of the user is to manipulate this window to browse the information. One of the earliest AR systems, NaviCam [Rekimoto and Nagao, 1995] is a typical example of AR information browser (see Figure 7.1), as well as many of the AR browser apps on smartphones such as Wikitude, Junaio, and Layar. This type of AR application requires the most basic interaction tasks: viewing the visualized AR scene and browsing the information provided.

Compared to VR systems where a navigational interface is usually needed to manoeuvre in the 3D virtual space, AR systems visualize virtual objects registered to the real world space. So users do not need additional interface object for navigation, but they can simply use their natural skills to move through the physical environment. For instance, when using head mounted AR displays users simply have to turn their head to change their view, while with handheld AR displays holding and moving the device and pointing at the direction to view is sufficient.

Other types of interactivity common to AR information browsers includes choosing different information (e.g. channels) to view, filtering information shown, navigating into details of the information provided, and changing visualization style, etc. Most of these interactions can be accomplished using traditional 2D graphical user interfaces (GUI) and screen input. Users can use traditional input devices, such as keyboard, mouse, joystick, touch pad or touch screen, to interact with the GUI provided with the AR content.

AR information browsers are one of the important classes of AR interface that is widely used in AR navigation with wearable computers, AR simulation and training. The interaction method provided is simple and easy to learn as the users can use their knowledge of traditional mobile user interfaces. On the other hand, these types of applications have limitations in providing direct interaction with virtual objects.

7.2 3D User Interfaces in AR: Interacting with Virtual Objects through Controllers

One of the straightforward approaches for supporting direct interaction with virtual objects is to adopt traditional 3D user interfaces techniques. Various 3D UI technologies and interaction methods have been investigated to provide direct interaction with 3D objects in virtual environments [Bowman et al., 2004], and many of these techniques can be applied to AR as well.

Bowman [Bowman et al., 2004] summarizes various types of 3D interactions into three categories: (1) navigation, (2) selection, and (3) manipulation. While the 3D navigation techniques might not be directly applicable to AR applications, as navigation in AR environments is naturally achieved by the users' moving their body, selection and manipulation interaction techniques can be easily adopted in AR applications [Schmalstieg et al., 2000]. For example in Kiyokawa's VLego application users have their hands tracked in 3d space and can reach

Figure 7.2: Using 3D user interface in a collaborative AR system. [Kiyokawa et al., 2000]

out and directly pick up virtual blocks in the collaborative AR interface, in much the same way that they could interact in an immersive VR application (see Figure 7.2) [Kiyokawa et al., 1999].

In many cases, 3D UIs use input devices that can be used for 6 degree of freedom (DOF) manipulation (translation and rotation in 3D) of virtual objects. There are various types of devices invented and used in VR and 3D UI field, including 3D mouse or wand type pointing devices, 6 DOF joysticks, spaceballs, etc. Among them, 3D motion tracking sensors are one of the most widely used technologies. These allow tracking of various physical objects including the user's body motion, and let users to point at or manipulate virtual objects. VLego used 3D motion tracking magnetic sensors for manipulating virtual objects registered in the physical environment.

Haptic devices (e.g. the Phantom[1]) are another type of traditional 3D UI that researchers investigated using in AR environments [Vallino and Brown, 1999, Adcock et al., 2003] (see Figure 7.3). Haptic interfaces not only work as a 3D pointing device but also provide force and tactile

---

[1] `http://geomagic.com/en/products-landing-pages/haptic`

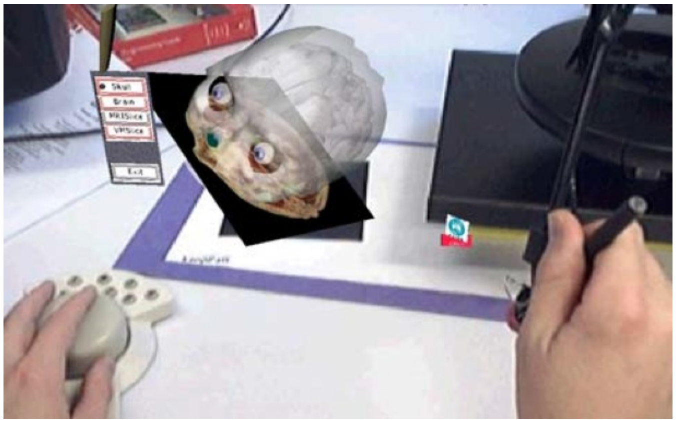


Figure 7.3: Using haptic interface in AR. [Adcock et al., 2003]

feedback which complements the visual experience by creating illusion of physical existence of a virtual object.

Traditional 3D UI can provide good interactivity in AR applications for entertainment, design, and training. With 3D UI, users can interact with 3D virtual objects everywhere in space in a natural and familiar way. While using traditional 3D UI for AR interactions benefit from previous work in VR and 3DUI fields, there are certain limitations with this approach as well. One of the most prominent problems is that the methods used for interacting with virtual objects are different from interacting with physical objects. With most of the traditional 3D UI, users will have to hold a control device and use it to point and manipulate virtual objects, while with physical objects, they mainly use their hands for direct manipulation.

7.3 Tangible User Interface in AR

Using physical objects as a medium for interacting with computers [Ishii and Ullmer, 1997] has been one of the main themes in modern human-computer interface technology research. The concept is referred to as Tangible User Interfaces (TUI). Tangible UI use physical objects for representing virtual entities and information, and to bridge between the physical and digital worlds.

Figure 7.4: Tangible AR interface with a physical book with virtual object overlaid on it.

While Tangible UI provide natural and intuitive interaction with digital information through manipulating physical objects, they can have limitations with display capabilities, either showing very limited information with different status of physical objects. Even when using a visual display in combination with Tangible UI, it still has problem of separation between the physical interaction space and the display where the virtual information is shown. For example, in a TUI digital content may be projected on a physical surface or shown on a screen while the user manipulates physical controllers some distance away creating a gap between the interaction space and presentation space.

To reduce the gap between the interaction methods for virtual and physical objects, Kato et al. [2000] proposed the concept of Tangible AR (TAR). TAR uses Tangible UI as input interaction metaphor while using AR for visualizing virtual information overlaid on the physical object used for interaction. Figure 7.4 shows a typical example of Tangible AR interface that uses a physical book as a Tangible UI while visualizing virtual objects overlaid on its pages. As can be seen in this case the interaction space and display space are seamlessly merged together.

Billinghurst et al. [2005] defines the characteristics of Tangible AR interfaces as 1) each virtual object being registered to a physical object, and 2) the user interacting with virtual objects by manipulating physical objects. To use physical objects as input devices for interaction requires accurate tracking of the objects, and for this purpose many Tangible AR applications use computer vision based tracking software, such as ARToolkit[2]. With accurate tracking, the computer system not only can recognize and identify different physical objects, but also can estimate the 3D motion of the objects so that the pose of the object or the motion gesture can be used for various interactions.

The basic goal of designing a Tangible AR interface is to map physical objects (input) with virtual objects (output) using an appropriate interaction metaphor. Lee et al. [2007] summarizes common interaction metaphors and methods used in various Tangible AR applications, and provides guidelines for designing and implementing interaction methods for Tangible AR applications. Providing a static mapping between the two is a very basic yet useful way for Tangible AR applications where the main purpose is to view 3D objects. However, in order to provide more interactivity, many Tangible AR applications also dynamically map them using a metaphor of tools for performing certain tasks.

Similar to how physical tools are designed, Tangible AR interfaces use two different design approaches, or a combination or both methods. One approach is space multiplexed interfaces where each physical tool is dedicated for one function, and the other is time multiplexed interfaces where a tool is used for many functions and purposes depending on the status and context. Space multiplexed interfaces are considered to be more intuitive to learn as each function is mapped to a single tool, while time multiplexed interfaces could have different function depending on the status and context hence requiring further steps of understanding and learning the functions. Figure 7.5 shows examples of Tangible AR applications that use time multiplexed and space multiplexed interfaces for interaction.

While Tangible AR interfaces provide an intuitive, natural and seamless way to interact with both physical and virtual objects in AR applications there is also a drawback of requiring physical objects for

---

[2] `http://www.hitl.washington.edu/artoolkit`

Figure 7.5: Examples of time multiplexed (left: VOMAR [Kato et al., 2000]) and space multiplexed (right: Tiles [Poupyrev et al., 2002]) Tangible AR interfaces.

interaction which might be inappropriate for mobile or wearable AR applications.

7.4 Natural User Interfaces in AR: Body Motion and Gesture

With traditional 3D user interfaces, body motion can be tracked and recognized using various types of motion tracking sensors that user's wear. Various motion sensors of different size and shape can also be applied in AR applications, ranging from a glove type device [Lee et al., 2010b] used for tracking hand gestures to full body motion tracking systems. An overview of various types of tracking technologies can be found in section 4.

With advance in computer vision technology, AR systems became capable of recognizing user's body motion and gesture in real time without requiring the user to wear any sensors. For example, Lee and Hollerer [2007] developed the HandyAR system capable of bare hand interaction using a standard web camera, although the supported gestures were limited (see Figure 7.6). Another work [Lee et al., 2008b] used a stereo camera to detect natural hand interaction. Most of this research focused on recognizing the human body through detecting features or skin colour Lee and Hollerer [2007], McDonald and Roth [2003] from video image that had limitations of accuracy of recognized hand postures yet requiring high speed processing power for running computer vision algorithms. Some research [Lee et al., 2004a] tried to

Figure 7.6: HandyAR free hand interaction with AR content. [Lee and Hollerer, 2007]

overcome this through reversing the focus from human body to objects of interest and used occlusions by the user's body as an input for interaction.

With recent technical advances and commercialization of depth cameras (e.g. Microsoft Kinect[3]), more accurate motion and gesture-based interaction became widely available for VR and AR applications (see Figure 7.7). Use of depth camera in AR applications enabled tracking dexterous hand motions for physical interaction with virtual objects using bare hands. Microsoft HoloDesk [Hilliges et al., 2012] demonstrated use of the Kinect camera to recognize the user's hands and other physical objects and allow them to interact with virtual objects shown on an optical see-through AR workbench. ARET [Corbett-Davies et al., 2013] shows a similar technique applied in AR based exposure treatment application. With growing interest in using hand gestures for interaction in AR applications, Piumsomboon et al. [2013] categorized a user-defined gesture set that can be applied to different tasks in AR applications.

Integrating hand motion and gesture based interaction with mobile or wearable AR systems is one of the topics actively investigated. Mistry and Maes [2009] demonstrated a wearable camera and projector system that recognizes user's hand gestures for interaction by detecting fingertips with color markers (see Figure 7.8). Using the camera on Google Glass, On the Go Platforms[4] developed hand gesture recognition software though supported gestures are limited to simple ones. Along the efforts for improvement, mobile version of depth sensing

---

[3] `http://www.microsoft.com/en-us/kinectforwindows`

[4] `http://www.otgplatforms.com`

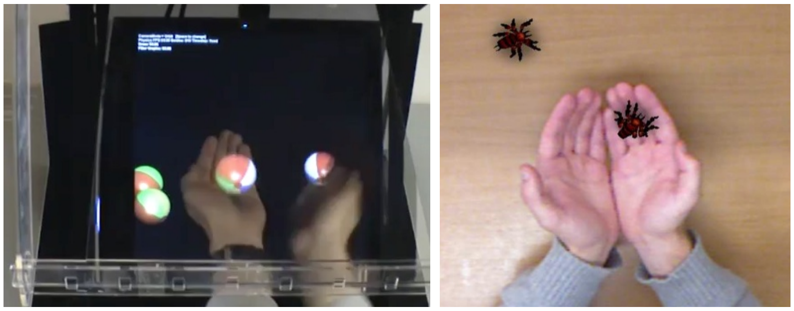


Figure 7.7: Hand gesture based physical interaction with virtual objects (left: HoloDesk [Hilliges et al., 2012], right: ARET [Corbett-Davies et al., 2013]).

Figure 7.8: Tracking fingers positions in the Sixth Sense projection based AR interface. [Mistry and Maes, 2009]

cameras are also under development, such as SoftKinetic[5], that would benefit from implementing more accurate hand gesture interaction in mobile and wearable AR systems. There are also other approaches to use biometric sensors for gesture interaction in mobile and wearable environment, such as the Myo[6] gesture armband controller.

7.5 Multimodal Interaction in AR

To provide richer interactivity in AR applications, there have been efforts to combine different modalities of input. Among different combination of input modalities, speech and gesture recognition combined is

---

[5] `http://www.softkinetic.com`

[6] `https://www.thalmic.com/myo`

one of the most widely and actively researched combinations. Previous research has shown that multimodal interfaces (MMI) that combine speech and gesture input can be an intuitive way to interact with 2D interfaces and 3D graphics on the desktop [Bolt, 1980, Hauptmann, 1989, Cohen et al., 1997]. This is because the input modalities are complimentary, with speech being good for quantitative input, while gesture is ideal for qualitative input [Cohen et al., 1989].

A number of researchers have used speech and gesture input in Virtual Reality interfaces. For example, the Naval Research Laboratory's Dragon 3D VR system [Cohen et al., 1999] used a multimodal system to create digital content in a 3D topographical scene, allowing users to create and position objects by speaking while gesturing in 3D space. Ciger et al. [2003] presented a multimodal user interface that combined a magic wand with spell casting. The user could navigate in the virtual environment, grab and manipulate objects using a combination of speech and pointing with a physical wand. Similarly, LaViola Jr. [1999] developed an immersive VR interface that used whole-hand input captured with a data glove and speech input for interior design. The user could create virtual objects with speech and then perform object manipulation using hand gestures. These examples are just a few of many and show that MMI are also a natural way to interact in immersive virtual environments.

However, despite this earlier research there has relatively little research on the use of multimodal interaction for AR. One of the first systems was the SenseShapes work of Olwal et al. [2003] which used volumetric regions of interest that can be attached to the user, allowing the user to point at virtual objects in the real world and manipulate them with speech commands. Kaiser et al. [2003] extended Olwal's SenseShapes work by focusing on mutual disambiguation between input channels (speech and gesture) to improve interpretation robustness. Heidemann et al. [2004] presented an AR system designed for online acquisition of visual knowledge and retrieval of memorized objects. More recently, Irawati et al. [2006] added speech input to the earlier VOMAR [Kato et al., 2000] interface allowing people to create AR scenes using a combination of speech and tangible paddle gestures. In a user study

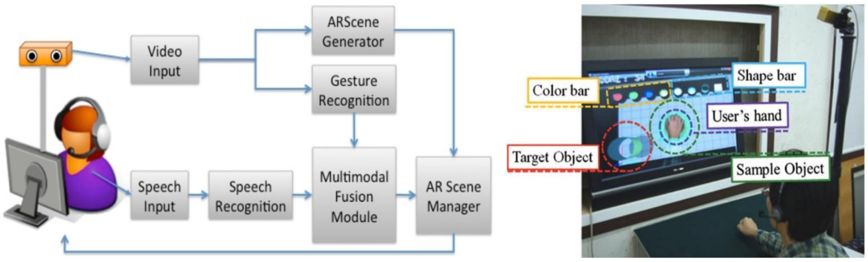


Figure 7.9: Natural hand gesture based multimodal AR system. [Lee et al., 2013b]

with her system, she found that people were able to complete a scene creation task over 35% faster with the interface using combined speech and paddle gesture input, than using gesture input on its own. The multimodal interface was also more accurate and was preferred by the users.

These multimodal systems all required the users to either wear gloves or hold a special input tool in their hands. More recently systems have been developed that support natural bare hand input. For example, Kolsch et al. [2006] create a multimodal interface for outdoor AR that used a wearable computer to perform hand tracking and recognize command gestures, enabling users to interact in a very intuitive manner with AR cues superimposed the real world. Piumsomboon et al. [2014] used a Wizard of Oz technique to classify the types of gesture that users would like to use in an AR multimodal interface, helping to inform the design of such systems. Finally, Lee et al. [2013b] developed a multimodal system that used a stereo camera to tracking users hand gestures and allow them to issue speech and gesture commands to manipulate virtual content in a desktop AR interface (see Figure 7.9. A user study with the system found that the MMI was 25% faster that using gesture only interaction and that users felt that the MMI was the most efficient, fastest, and accurate interface.

7.6 Other Interaction Methods

While speech recognition is the most actively investigate as audio input, there are also attempts to detect other types of sound for interaction

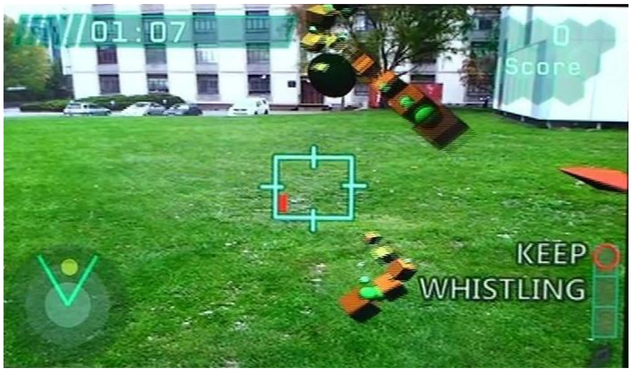


Figure 7.10: Whistle recognition in GeoBoids mobile AR game. [Lindeman et al., 2012]

in AR applications. For instance, Lindeman et al. [2012] used whistle recognition in a mobile AR game where users have to whistle for the right pitch and length to scare virtual creatures in the game (see Figure 7.10).

Another type of input modality gaining interest of AR researchers is Brain Computer Interfaces (BCI). While BCI is at the very early stage of investigation, studies show BCI and AR can be a good combination where AR can be efficient and powerful tool for enhancing and studying BCI technology [Kansaku et al., 2010, Lenhardt and Ritter, 2010, Scherer et al., 2010]. While traditional electroencephalogram (EEG) sensing equipments required cumbersome setup of wearing the sensors, recent development of the technology made a headset style sensor (e.g. NeuroSky[7]) that can be easily integrated with AR applications (See Figure 7.11).

7.7 Summary

In this section we have reviewed some of the common input techniques and interaction methods used in AR systems. As we have seen they have evolved from simple information browsing which support very little interaction with the virtual content, to complex interfaces that can

---

[7] `http://neurosky.com`

Figure 7.11: A mobile BCI headset from NeuroSky. `http://neurosky.com`

recognise and respond to natural gesture input and speech. However, even though there are now a wide range of different input methods available for AR interfaces, they still need to be carefully designed. In the next section we review research into design guideline and interface patterns for effective AR experiences.

8

Design Guidelines and Interface Patterns

As reviewed in the previous section there have been a wide range of different, but there needs to be more research on how to design AR experiences. In this section we review the design guidelines and interface patterns that have been developed.

When a new interface medium, like Augmented Reality, 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 modelling user interactions

For example the well-known desktop metaphor of Windows, Icons, Menus and Pointers (WIMP) has been through all of these stages. Formal theoretical models have been developed that can predict exactly

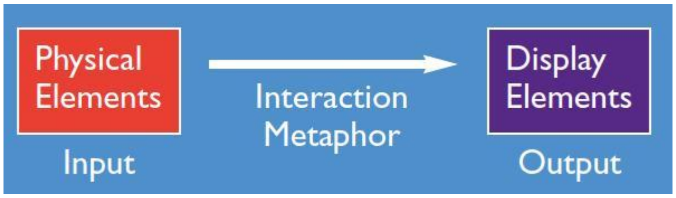


Figure 8.1: The three elements to be designed in an AR interface.

how long it will take a mouse with certain characteristics to select an icon of a given size. Models such as Fitts law [Fitts, 1954] can be used to calculate expected pointing time in user interfaces. Virtual Reality interfaces are at the third stage with a number of interface techniques such as the Go-Go Interaction method [Poupyrev et al., 1996] that is designed specifically for object selection and manipulation in immersive Virtual Worlds.

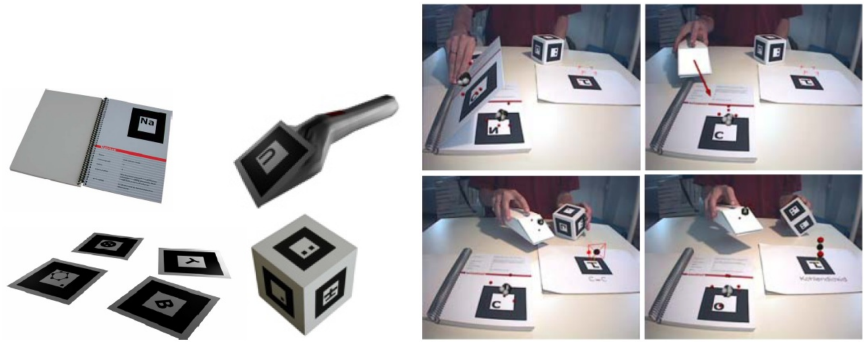
As can be seen from the previous section, in many ways AR interfaces have barely moved beyond the first two stages. There are many different AR interaction methods and input devices used, but these are largely versions of existing 2D and 3D techniques from other desktop, mobile or VR interfaces. For example, most handheld AR applications use familiar touch screen input and gesture applications from other mobile phone experiences. There is a need to develop interface metaphors and interaction techniques specific to AR [Billinghurst et al., 2005].

Developing an appropriate interface metaphor can be achieved by having a deeper understanding of AR interfaces. Unlike most other interfaces, Augmented Reality experiences have a close relationship between real and virtual objects. MacIntyre points out that AR design is driven by the need to define and fuse the relationship between entities in the physical world and virtual world [MacIntyre, 2002]. The basic goal is the map user input with physical objects onto computer generated virtual output using an appropriate interaction metaphor. Thus there are three components that must be designed in an AR application (see Figure 8.1): (1) the real physical objects, (2) the virtual elements to be displayed, and (3) the interaction metaphor that links the real and virtual elements together.

Design of the physical and virtual elements can be guided by the concept of affordances developed by Gibson [1979] and first applied to interface design by Norman [1988]. Norman defines affordance as " .. the perceived and actual properties of the thing, primarily those fundamental properties that determine just how the thing could possibly be used" [Norman, 1988]. For example a chair affords sitting on, and a hammer affords holding by the handle. Even though they are not real objects, computer graphics contact can also have visual affordances showing how they should be manipulated. So a virtual box could have a handle on the top showing that it could be picked up.

In the context of an AR interface, the designer should make sure that the shape of the physical input objects and the virtual content shown on the both provide good affordances showing how they should be used. A good example of this is the Augmented Chemistry application developed by Fjeld and Voegtli [2002]. In this application they wanted to give the user the experience of being able to assemble their own chemical molecules, using an AR display and a number of physical input devices. The application was developed using ARToolKit for tracking, but the researchers very carefully designed the shapes of the physical props that the tracking symbols were attached to, shown in Figure 8.2(a). There was a book that had one atomic element per page, and users could brown through the virtual atoms simply by turning the page. When the user found an element they wanted the could point a shovel shaped selection tool at it, 'scoop' it up and place is into the assembly area (see Figure 8.2(b)). Once the virtual molecule starts to come into shape the user can use a cube shaped rotation device in their other hand that causes the virtual model to rotate. As can be seen the physical shape of the input devices provide a strong clue as to how they are supposed to be used.

In addition to the design of the physical and virtual elements of the AR application, the interaction metaphor connecting the two must also be developed. In the case of the Augmented Chemistry the metaphor was that the user was physically scooping atoms to create molecules. As described in §7.3 this metaphor is an example of a Tangible AR approach that combines Tangible User Interface (TUI) methods for in-



(a) Input elements **(b)** Using the scoop to place an atom

Figure 8.2: The Augmented Chemistry interface. [Fjeld and Voegtli, 2002]

put and AR techniques for display. This means that the interaction metaphor can be created following design principles learned from tangible user interfaces. The basic principles of TUI include:

- The use of physical controllers for manipulating virtual content
- Support for spatial 3D interaction techniques
- Support for both time-multiplexed and space-multiplexed interaction
- Support for multi-handed interaction
- Matching the interface object affordances to the task requirements
- Supporting parallel activity where multiple objects are being manipulated
- Collaboration between multiple participants

Papers from Antle and Wise [2013] and Waldner et al. [2006] among others provide additional design guidelines for TUI applications. Following these guidelines will ensure that the AR application will be very intuitive to use because of seamless interaction between the physical and virtual elements.

8.1 Case Study: levelHead

The award winning AR art application levelHead[1] provides a good example of how to follow this design approach. It is a compelling application that involves real cubes with AR tracking symbols on their sides. The cubes are places on a table in front of a camera and a projected image on a screen from the camera view shows that the cubes contain virtual rooms (see Figure 8.3). A human figure appears in one of the rooms, and when the cube is titled the figure will walk to the opposite corner of the room (see Figure 8.4). Rotating the cube to show a different face to the camera will cause a different room to be loaded. The aim of the installation is to tilt, turn and rotate the cubes until the figure can escape from the connected rooms.

Figure 8.3: The levelHead installation. `http://julianoliver.com/levelhead`

As described above, creating an AR application involves designing three components; (1) the real physical objects, (2) the virtual elements to be displayed, and (3) the interaction metaphor. Thus, Level Head has the following:

1) *Real Object*: physical cube that the user can easily manipulate in their hands

---

[1] `http://julianoliver.com/levelhead`

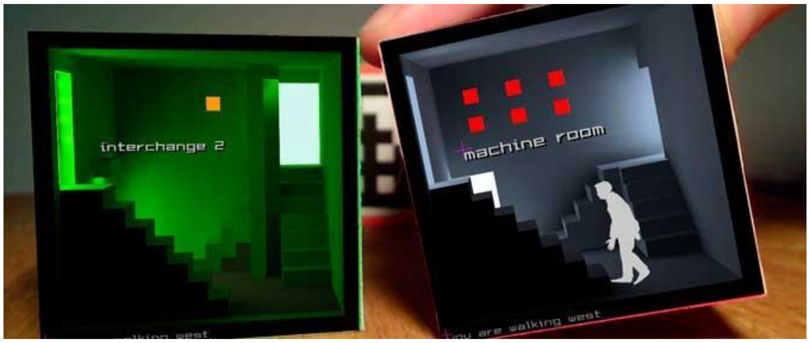


Figure 8.4: Cube manipulation in levelHead, moving the virtual figure.
`http://julianoliver.com/levelhead`

2) *Virtual Content*: Virtual rooms inside the cubes, human figure
3) *Metaphor*: The user is physically tilting the virtual human's world,

The combination of easily understood metaphor, intuitive tangible user interface and effective AR content, makes this a very well designed AR experience that is extremely enjoyable to use.

8.2 Using Design Patterns

Another approach that can be used to help with AR interface design is to use Design Patterns. Introducing the concept of Design Patterns, Alexander et al. say that "Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem in such a way that you can use this solution a million times over, without ever doing it the same way twice." [Alexander et al., 1977] Design Patterns are commonly used in computer science and interface design. For example, Borchers has written the book "A Pattern Approach to Interaction Design" [Borchers, 2001] that provides an introduction to the use of Design Patterns in user interface design, and Yahoo provides an online Design Patterns library for web developers[2].

---

[2] `https://developer.yahoo.com/ypatterns`

Reicher et al. [2003] explore how Augmented Reality development can be improved by following a Pattern Language approach. This is built on earlier work by MacWilliams et al. [2004]. Their approach is mostly focused on the underlying software engineering and showing how complex AR systems can be build by combining modules based on Design Patterns. In this case each Pattern is described by name, goal, motivation, a description, usability, consequences, and known project usage. The DWARF framework [Bauer et al., 2001] is an example of a component based AR framework based on this Design Pattern approach.

In contrast to Reicher's approach, Xu et al. [2011] describe Design Patterns that can be used for the user experience design of Handheld AR interfaces. Their focus is on pre-patterns that bridge the gap between Interaction Design and Game Design. In their paper they list a number of Design Patterns that can be used, described in terms of a title, definition, description, examples, and description of using the pre-patterns. Table 8.1 shows a list of some of the Design Patterns developed, expressed as a Title, Interaction Metaphor and the Embodied Skills that are used. The Embodied Skills are those skills that users have by virtue of having a body. For example. Body Awareness and Skills from the Device Metaphors row, refers to the user's ability to be aware of their body and how it is able to move real objects.

As an example of how a Design Pattern can be used we consider the case of Seamful Design. As Table 8.1 shows, the Seamful Design pattern is that designers need to integrate technology seams into the Handheld AR game design. The term "technology seam" refers to the limitations of the technology used in the application. For example, most handheld AR systems use computer vision based tracking, so one of the technology seams are the conditions under which the tracking fails (such as bad lighting or fast camera movement etc).

The handheld AR game Paparazzi[3] shows how the Design Pattern can be followed and the technology seam integrated into the application. In this game the user plays the role of a Paparazzi photographer and gets points for taking pictures of a character. The character ap-

---

[3] `http://pixel-punch.com/project.php?project=Paparazzi`

Table 8.1: Design Patterns for Handheld AR games. [Xu et al., 2011]

Title	**Meaning**	**Embodied Skills**
Device Metaphors	Using metaphor to suggest available player actions.	Body A&S* Naive physics
Control Mapping	Intuitive mapping between physical and digital objects.	Body A&S Naive physics
Seamful Design	Making sense of and integrating the technological seams through game design.	Body A&S
World Consistency	Whether the laws and rules in the physical world hold in digital world.	Naive physics Environmental A&S
Landmarks	Reinforcing the connection between digital-physical space through landmarks.	Environmental A&S
Personal Presence	The way that a player is represented in the game decides how much they feel like living in the digital game world.	Environmental A&S Naive physics
Living Creatures	Game characters that are responsive to physical, social events that mimic behaviours of living beings.	Social A&S Body A&S
Body constraints	Movement of one's body position, constrains another player's action.	Body A&S Social A&S

∗ A&S: short for 'Awareness and Skills'

pears as a virtual model superimposed on a real piece of paper. Once the user takes a few pictures of the character it will get angry and appear to jump on the camera screen (see Figure 8.5). The user will then need to shake their phone as hard as possible to dislodge the character and continue the photo taking. Thus in this game there are two distinct types of tracking with a technology seam between them; (1) computer vision based tracking when the user is pointing the phone at the AR tracking image, (2) inertial compass tracking when the virtual character is attached to the phone and the user needs to shake it loose. Quick shaking of the phone will cause the computer vision tracking to fail, but in this case the game has designed for that as part of the game story, transitioning to the compass based tracking. Due to this design the technology seam becomes transparent and the game play is not disrupted.

Xu et al. [2011] have many other examples of how handheld AR games use the Design Patterns listed in Table 8.1 to improve their game design and so make the game more enjoyable and intuitive.

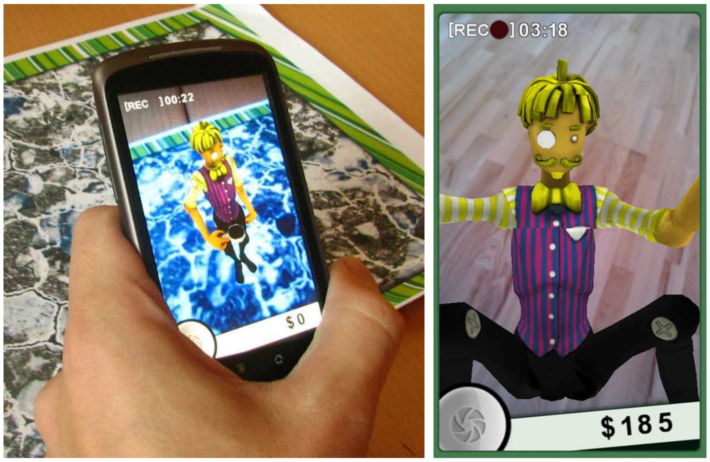


Figure 8.5: Two different game states in Paparazzi tracking with computer vision (left) and inertial sensor (right). `http://pixel-punch.com`

8.3 Designing for Special Needs

So far we have described several techniques that can be used to design effective AR experiences. However there are some categories of users that may need extra attention when designing for them. In particular Radu and MacIntyre [2012] point out that designing AR systems for children aged 6 to 9 years old should take into account their developmental stages. Previous research has shown that Augmented Reality can be used to create powerful educational experiences [Kerawalla et al., 2006, Billinghurst, 2002]. At the same time, research outside the AR field has shown that when designing technology for children it is very important to take into account the children's developmental abilities [Bruckman and Bandlow, 2003, Wyeth and Purchase, 2003]. However, there has been little research presenting guidelines for developing AR experiences for children.

In Radu's paper they provide a list of elements that must considered from a developmental perspective, in four broad areas: (1) Motor Abilities, (2) Spatial Abilities, (3) Attention Abilities, and (4) Logic and Memory Abilities. For example, many AR interfaces support tangible interaction with two-handed input, but studies have shown that bi-manual input abilities doesn't reach adult levels until around 9 years old [Hourcade, 2008, Fagard, 1990], and before age children may have trouble using both hands for input.

Radu makes the comment in their paper that young children tend to hold handheld AR input devices with both hands and it is not until they are older than 7 that they can hold the device with one hand, freeing up the other hand to interact with the AR content. They suggest that AR games could be created for training motor coordination skills in children, particularly for two-handed input. In their paper they also explore other aspects of motor ability such as hand-eye coordination, fine motor skills, and gross motor skills.

8.4 Summary

In summary, in this section we have provided an overview of design principles for AR experiences. In general designers should begin by considering the three core elements of physical objects, virtual content and the interaction metaphor than connects them. They can also use well-known Design Patterns to assist with the interaction techniques, and in some cases they may need to consider special needs of people using the system. In the next section we review methods than can be used to evaluate the AR experiences once they have been developed.

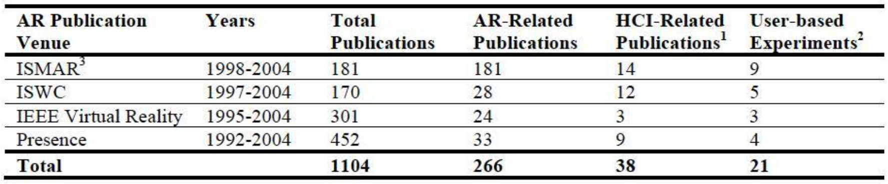
9

Evaluation of AR Systems

In the previous sections we have reviewed AR technology and tools for building AR applications. However, a key activity in developing AR experiences is evaluating the usability of the AR application. Gabbard and Swan [2008] argue that user-based experiments are critical for driving design activities, usability, and discovery early in an emerging technology's development, such as in the case of Augmented Reality. They point out that lessons learned from user studies provide value to the field as a whole in terms of insight into the user interface design space. In this section we describe factors that should be taken into account when developing AR usability studies, and review the major types of user studies that are typically conducted with AR systems.

Although Augmented Reality has been researched for nearly fifty years, it is only recently that significant numbers of AR user studies have begun to appear in the research literature. In 2005, Swan and Gabbard [2005] presented a survey of all of the user-based experimentation in AR they could find in the research literature. They reviewed papers from the IWAR, ISAR and ISMAR conferences, the International Symposium on Wearable Computers, the IEEE Virtual Reality conference and the journal Presence: Teleoperators and Virtual Envi-

Table 9.1: The results of Swan and Gabbard's survey of AR publications. [Swan and Gabbard, 2005]



AR Publication Venue	**Years**	**Total Publications**	**AR-Related Publications**	**HCI-Related Publications[1]**	**User-based Experiments[2]**
ISMAR[3]	1998-2004	181	181	14	9
ISWC	1997-2004	170	28	12	5
IEEE Virtual Reality	1995-2004	301	24	3	3
Presence	1992-2004	452	33	9	4
Total		**1104**	**266**	**38**	**21**

ronments, from 1992 until 2004. From these four sources they identified 266 AR-related papers, of which only 38 addressed some aspect of Human Computer Interaction, and only 21 described a formal user study. This meant that less than 8% of all the AR related papers published in these conferences and journal had any user-based experimentation. Table 9.1 shows the summary of this breakdown.

More recently Dünser et al. [2008] conducted a broader survey of AR research published between 1993 and 2007, looking at 28 outlets such as the ACM Digital Library, IEEE Xplore Journals, SpringerLink, and others. They were able to identify a total of 557 AR related publications, and of these a total of 161 publications that included a user evaluation of one type or another. Figure 9.1 shows the number of AR papers collected and the papers in this collection that had a user evaluation. As can be seen the proportion of AR papers with a user evaluation is increasing over time. They reported that overall their survey showed that an estimated 10% of the AR papers published in ACM and IEEE included some user evaluation, agreeing closely with the figures that Swan and Gabbard [2005] found from their smaller survey. These results show that there is still room for significant improvement in the number of formal user evaluations conducted in AR research.

One reason for the lack of user evaluations in AR could be a lack of education on how to evaluate AR experiences, how to properly design experiments, choose the appropriate methods, apply empirical methods, and analyse the results. There also seems to be a lack of understanding of the need of doing studies or sometimes the incorrect motivation for doing them. If user evaluations are conducted out of incorrect motivation or if empirical methods are not properly applied,

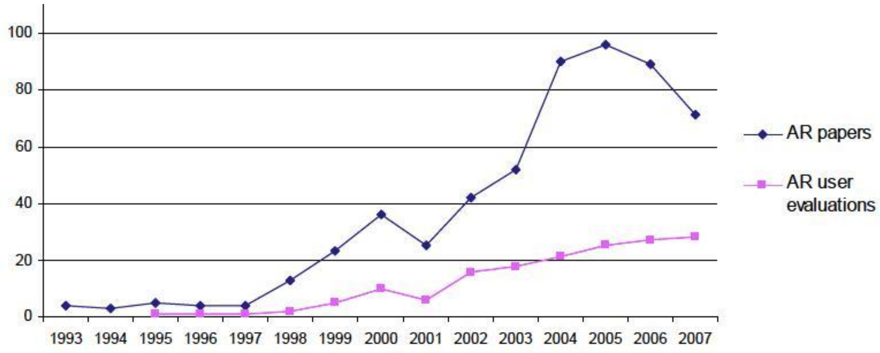


Figure 9.1: Total number of AR papers published and papers containing a user evaluation. [Dünser et al., 2008]

the reported results and findings are of limited value or can even be misleading.

9.1 Types of User Studies

Swan and Gabbard [2005] report that the first user-based experimentation in AR was published in 1995, and since then usability studies have been reported in three related areas:

- *Perception*: How do users perceive virtual information overlaid on the real world? What perceptual cues can be used to distinguish between real and virtual content?
- *Performance/Interaction*: How do users interact with virtual information overlaid on the real world? How can real world objects be used to interact with augmented content?
- *Collaboration*: How can MR / AR interfaces be used to enhance face-to-face and remote collaboration?

Dünser et al. [2008] provided a breakdown of the papers they reviewed into these three categories and found that the majority of the evaluations conducted were Performance Studies with 75 examples, followed by Perception and Cognition based user studies with 35 papers, and finally Collaboration user studies with 10 examples. This shows that user evaluation of collaborative AR systems is an area that needs to be studied more.

In the remainder of this section we describe typical user evaluations conducted in the areas of Perceptual Studies, Performance/Interface Studies and Collaborative Studies. This should provide readers with a good representative set of AR user studies that they base their own user evaluations on. For a more exhaustive survey readers should refer back to the work of Swan and Gabbard [2005] and Dünser et al. [2008].

9.1.1 Perceptual Studies

In an Augmented Reality interface the blending of Reality and Virtuality is a perceptual task in which the interface designer tries to convince the human perceptual system that virtual information is as real as the surrounding physical environment. It is difficult, if not impossible to control all the possible perceptual cues, so perceptual biases can occur that affect task performance. Drascic and Milgram [1996] provide an overview of eighteen perceptual issues that relate to Augmented Reality, including miss-matches in clarity and luminance between real and virtual imagery, accommodation and vergence conflict, and occluding virtual imagery by real objects. As they point out, in a task in which different depth cues conflict, results may be wildly inaccurate or inconsistent. Thus experiment/interface designers should be aware of the cues they are introducing that may affect user perception.

Many researchers have conducted studies to quantify some of the perceptual issues associated with AR interfaces. These studies often relate to the use of optical or video see-through head mounted displays to merge real and virtual images (see [Rolland and Fuchs, 2000] for a description of head-mounted display issues).

Since the 1960's studies have been conducted on size and distance judgments of virtual imagery presented in AR displays. Rolland and Gibson summarize these results in [Rolland et al., 1995b] as well as reporting new results for the perceived size and depth of virtual objects when presented in a see-through display. Their experimental design is typical of perception studies. A bench-mounted optical see-through display was built and a careful calibration technique used to adjust the display elements for each individual subject. The subjects were then shown a pair of objects (a cube and a cylinder) side by side and asked to judge which was closet. Three object conditions were used; both objects were real, one object is real and the other is virtual, both objects are virtual. Objects were shown at a variety of depths and they report finding that that the virtual objects are perceived systematically farther away than real objects.

A similar design is used by Ellis et al. in their distance judgment studies [Ellis and Menges, 1997]. Their work explored how accurately subjects could place a physical cursor under a virtual object in 3 different conditions; monocular display, binocular display and stereoscopic display. In each case a subject looked through a bench-mounted see-through display and moved a real pointer until it appeared until a virtual tetrahedron displayed at different depths. The main experimental measure was the actual distance between the real and virtual objects. Depth judgment in a stereoscopic display was found to be almost perfect, while the monocular display produced large overestimates in depth position. Ellis and Menges report on a suite of related experiments in [Ellis and Menges, 2001].

9.1.2 Interaction

Section 7 described a range of different interaction methods and interface metaphors for Augmented Reality. However, before new interaction metaphors can be , experiments must be conducted to understand how effective the techniques are. In general, AR interaction studies can use established techniques from immersive VR experiments although a particularly important area of study is the effect of physical objects on virtual object manipulation. For example, Lindeman finds that physical constraints provided by a real object can significantly improve performance in an immersive virtual manipulation task [Lindeman et al., 1999]. Similarly Hoffman finds adding real objects that can be touched to immersive Virtual Environments enhances the feeling of Presence in those environments [Hoffmann, 1998]. While in Poupyrev's virtual tablet work, the presence of a real tablet and pen enable users to easily enter virtual handwritten commands and annotations [Poupyrev et al., 1998].

Mason et al. [2001] provide a good example of an interaction experiment designed for an AR interface. Their work explored the role of visual and haptic feedback in reaching and grasping for objects in a table-top AR environment. They were particularly interested to find out if Fitt's law held in an AR setting. Fitt's law is a basic interaction law that relates movement time to index of difficulty [Fitts and Peterson, 1964]. In the experiment subjects reached for and grasped a cube in the presence or absence of visual feedback of seeing their own limb. In half the conditions the cube was purely virtual, while in the other half the virtual cube was superimposed over a real cube. Finally, four different sizes of cubes were used.

An optical tracking system was used to measure hand motion and a set of kinematic measures used, including Movement Time, Peak Velocity of the Wrist, Time to Peak Velocity of the Wrist, and Percent Time from Peak Velocity of the Wrist. Using these measures Mason et al. found that Fitt's Law was followed when a real cube was present, but not when the target was entirely virtual. This result implies that some form of haptic feedback is essential for effective task performance in augmented and virtual environments. This work also illustrates that kinematic variables can be a powerful tool for interaction experiments. Similar AR interaction and object positioning experiments have been reported by Wang and MacKenzie [2000] and Drascic and Milgram [1991].

An important difference between AR and VR interfaces is that in an AR interface physical object manipulations can be mapped one-to-one to virtual object operations, and so follow a space-multiplexed input design [Fitzmaurice and Buxton, 1997]. 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. Conversely, 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 [Fitzmaurice and Buxton, 1997]. In most manual tasks space-multiplexed devices are used to interact with the surrounding physical environment. In contrast, the limited number of tracking devices in an immersive VR system makes it difficult to use a space-multiplexed interface.

The use of a space-multiplexed interface makes it possible to explore interaction metaphors that are difficult in immersive Virtual Environments. One promising area for new metaphors is Tangible Augmented Reality. Tangible AR interfaces are AR interfaces based on Tangible User Interface design principles. The Shared Space [Billinghurst et al., 2000, Kato et al., 2000] interface (see Figure 9.2) is an early example of a Tangible AR interface. In this case the goal of the interface was to create a compelling AR experience that could be used by complete novices. In this interface several people stand around a table wearing HMDs. 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. The goal of the game was to collaboratively match objects that logically belonged together. When cards containing correct matches are placed side by side an animation is triggered. Tangible User Interface design principles are followed in the use of physically based interaction and a form-factor that matches the task requirements.

The SharedSpace interface was shown at the Siggraph 1999 conference where there was little time for formal evaluation. However an informal user study was conducted by observing user interactions, asking people to fill out a short post-experience survey, and conducting a limited number of interviews. From these observations it was found that 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

Figure 9.2: The Shared Space interface. [Billinghurst et al., 2000]

children could play and enjoy the game. 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. Users felt that they could very easily play with the other people and interact with the virtual objects. 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. This feedback shows the usefulness of informal experimental observation, particularly for new exploratory interfaces.

9.1.3 Collaboration

A particularly promising area for AR user studies is in the development and evaluation of collaborative AR interfaces. The value of immersive VR interfaces for supporting remote collaboration has been shown by the DIVE [Carlsson and Hagsand, 1993] and GreenSpace [Mandeville et al., 1996] projects among others. However, most current multi-user VR systems are fully immersive, separating the user from the real world and their traditional tools. While this may be appropriate for some applications, there are many situations where a user requires collaboration on a real world task. Other researchers have explored the use of augmented reality to support face-to-face collaboration and remote collaboration. Projects such as Studierstube [Schmalstieg et al., 1996, Szalavári et al., 1998], Transvision [Rekimoto, 1996b], and AR2 Hockey [Ohshima et al., 1998] allow users to see each other as well as 3D virtual objects in the space between them. Users can interact with the real world at the same time as the virtual images, supporting spatial cues and facilitating very natural collaboration. Although these projects have successfully demonstrated collaborative AR interfaces there have been few formal user studies conducted. In contrast there have been many decades of studies into various aspects of audio and video conferencing. We can draw on the lessons from these experiments when evaluating collaborative AR interfaces.

In the telecommunications literature, there have been many experiments conducted comparing face-to-face, audio and video, and audio only communication. Sellen [1995] provides a good summary. While people generally do not prefer audio only, they are often able to perform tasks as effectively as in the video conditions. Both the audio and video, and audio only cases typically produce poorer communication than face-to-face collaboration, so Sellen reports that the main effect on collaborative performance is due to whether the collaboration was technologically mediated or not, not on the type of technology mediation used. Naturally this varies somewhat according to task. While face-to-face interaction is no better than speech only communication for cognitive problem solving tasks [Williams, 1977], visual cues can be important in tasks requiring negotiation [Chapanis, 1975].

Although the outcome may be the same, the process of communication can be affected by the presence or absence of visual cues [O'Malley et al., 1996] because video can transmits social cues and affective information, although not as effectively as face-to-face interaction [Heath and Luff, 1991]. However, the usefulness of video for transmitting nonverbal cues may be overestimated and video may be better used to show the communication availability of others or views of shared workspaces [Whittaker and O'Conaill, 1997]. So even when users attempt nonverbal communication in a video conferencing environment, their gestures must be wildly exaggerated to be recognized as the equivalent face-to-face gestures [Heath and Luff, 1991].

These results imply that in collaborative AR experiments process measures and subjective measures may be more important than quantitative outcome measures. Process measures are typically gathered by transcribing the speech and gesture interaction between the subjects and performing a conversational analysis. Measures that are often collected include the number of words spoken, average number of works per phrase, number and type of gestures, number of interruptions, number of questions and the total speaking time. Although time consuming, this type of fine-grained analysis often reveals differences in communication patterns between experimental conditions.

One of the difficulties with collecting process measures is that of deciding which metrics to use in developing a data coding technique. Transcribing audio and video tapes is a very time-consuming process and can be unfruitful if the wrong metrics are used. Nyerges et al. [1998] provide a good introduction to the art of coding groupware interactions and give guidance on good metrics. Measures that have been found to be significantly different include:

- Frequency of conversational turns [Daly-Jones et al., 1998, O'Conaill and Whittaker, 1997]
- Conversational Handovers [O'Conaill and Whittaker, 1997]
- Incidence/duration of overlapping speech [Daly-Jones et al., 1998, Sellen, 1995]
- Use of pronouns [McCarthy and Monk, 1994]
- Number of interruptions [Boyle et al., 1994, O'Conaill and Whittaker, 1997]
- Turn Completions [Tang and Isaacs, 1993]
- Dialogue length [Boyle et al., 1994, O'Conaill and Whittaker, 1997, O'Malley et al., 1996, Anderson et al., 1996]

- Dialogue structure [Boyle et al., 1994, O'Malley et al., 1996, Anderson et al., 1996]

- Backchannels [O'Conaill and Whittaker, 1997]

Gesture and non-verbal behaviors can also be analyzed for characteristic features. Generally these behaviors are first classified according to type and the occurrences of each type and then counted. Bekker et al. [1995] describe an observational study they performed on groups of subjects engaged in a face-to-face design task. From video of the subject groups four categories of gesture, kinetic, spatial, pointing and other, were identified. They were then able to calculate the average numbers of gestures per minute for each of the different stages in the design task. These four categories were based on the more complex coding categories used by McNeill [1992] and Ekman and Friesen [1981].

In contrast to this work, there have been very few user studies with collaborative AR environments and almost none that examined communication process measures. Kiyokawa et al. [2000] conducted an experiment to compare gaze and gesture awareness when the same task was performed in an AR interface and an immersive virtual environment. In his SeamlessDesign interface, users were seated across a table from one another and use a collaborative AR design application (see Figure 7.2). A simple shared pointing task was used to compare gaze and gesture awareness and the influence of a virtual body and gaze directed viewing lines. The experimental measures were the time to perform the task and a subjective survey on the ease of use. Subjects performed significantly faster in the AR interface than in the immersive condition and felt that this was the easiest condition to work together.

There have been several studies performed using wearable computers and displays for supporting remote collaboration. In this case the remote users can typically manipulate a virtual pointer in the users wearable display or share their view of the real world. An early example was the SharedView system of Kuzuoka [1992]. This was a video see-through head mounted display with a camera attached. A machine operator would wear the HMD enabling a remote expert to see what he was seeing and make gestures in the display to show him how to operate the machinery. In a simple evaluation study, SharedView was found to have better performance than collaboration with a remote fixed camera, but worse than the face-to-face collaboration.

Kraut et al. [1996] provides another example of collaboration using a wearable interface. They were interested in how the presence or absence of a remote expert might help a subject repair a bicycle and what differences in communication patterns may result with and without shared video. Subjects wore a head mounted display that allowed them to see video of the remote expert, or images of a repair manual. Subjects could complete the repairs in half the time with a remote expert and produced significantly higher-quality work. When video was used they found that the experts were more proactive with help and that subjects did not need to be as explicit in describing their tasks. In a follow-up experiment, Fussell et al. [2000] add a condition where the expert is in the same room as the subject. The same metrics are used (performance time and quality and conversational analysis of speech), and they find that the task is completed significantly faster in the face-to-face condition. This time they find that speech patterns were significantly different between face-to-face and mediated conditions; experts in the face-to-face condition used significantly more deictic references, used shorter phrases, and were more efficient in their utterances.

Several collaborative experiments and the measures used are summarized in Table 9.2.

The MR conferencing experiment [Billinghurst and Kato, 2000], provides an example of how to conduct a collaborative AR experiment with conversational analysis. The MR conferencing interface supports conferencing between a desktop user and a person wearing a lightweight head mounted display. The person in the HMD sees their remote collaborator as a live video texture superimposed over a real world object (a name card) (see Figure 9.3). This configuration has a number of possible advantages over normal video conferencing, so the goal of this experiment was to compare MR conferencing to normal video and audio conferencing.

Each pair of subjects talked with each other for 10 minutes in each of audio only, video and MR conferencing conditions. Each of these

Table 9.2: Collaborative AR experiments.

Interface	Task	Conditions	Measures Used	Outcome
SeamlessDesign (Kiyokawa 00)	Pointing, Co-located collaboration	AR vs. VR	Performance time, Subjective ease of use survey	AR faster. AR rated as easier to use.
AR2 Hockey (Ohshima 98)	Collaborative game, Face-to-face	AR vs. VR	Subjective survey, Game scores	Mixed
SharedView (Kuzuoka 92)	Machine operation, Remote viewing, Remote pointing	AR vs. FtF vs. Fixed Camera	Performance time, Speech classification	FtF faster than SharedView which is faster than fixed Camera. Fixed camera and Shared View have similar speech patterns.
NetMan (Bauer 99)	Remote pointing, Remote collaboration	AR pointing vs. no pointing	Gesture count, Speech classification, Subjective survey	Gestures used more than speech. Deictic speech most common type of speech act.
Bike Repair I (Kraut 96)	Bike repair, Remote collaboration	Single user vs. remote expert, Video vs. no-video	Performance time, Performance quality, Coding of speech acts	Subjects faster and produce better quality work with a remote expert. With video experts were more proactive and subjects did not need to be as explicit in describing the problem.
Bike Repair II (Fussell 00)	Bike repair, Remote collaboration	Co-located expert vs. Audio-video vs. Audio only	Performance time, Performance quality, Conversational analysis	Performance and quality best in FtF condition. Significant differences in conversational coding.

Figure 9.3: The MR conferencing collaborative AR interface. [Billinghurst and Kato, 2000]

sessions were video taped and after each condition subjects filled in a survey about how present they felt the remote person was and how easily they could communicate with them. After the experiment was over the video taped were transcribed and a simple speech analysis performed, including counting the number of works/minute each of the users uttered, the number of interruptions and the back-channels spoken. This analysis revealed that not only did the user feel that the remote collaborators were more present in the MR conferencing condition, but that they used less words and interruptions per minute than in the two other conditions. These results imply that MR conferencing is indeed more similar to face to face conversation than Audio or Video conferencing.

An alternative to running a full collaborative experiment is to simulate the experience. This may be particularly for early pilot studies for multi-user experiments where it may be difficult to gather the number of subjects. The WearCom project [Billinghurst et al., 1998] is an example of a pilot study that uses simulation to evaluate the interface. The WearCom interface is a wearable communication space that uses spatial audio and visual cues to help disambiguate between multiple speakers. To evaluate the interface a simulated conferencing space was created where 1,3 or 5 recorded voices could be played back and spatialized in real time. The voices were played at the same time and said almost the same thing, expect for a key phrase in the middle. This simulates the most difficult case for understanding in a multi-party conferencing experience. The goal of the subject was to listen for a specific speaker and key phrase and record the phrase. This was repeated for 1,3 and 5 speakers with both spatial and non-spatial audio, so each user generated a score out of 6 possible correct phrases. In addition, subjects were asked to rank each of the conditions on how understandable they were.

When the results were analyzed from this experiment, users scored significantly better results on the spatial audio conditions than with the non-spatial audio. They also subjectively felt that the spatial audio conditions were far more understandable. Although these results were found using a simulated conferencing space, they were so significant that it is expected that the same results would occur in a conferencing space with real collaborators.

9.2 Evaluation Methods

In their survey Dünser et al. [2008] also categorizes AR papers containing a user evaluation according to the type of user study method and approaches. They identified the following five main types of evaluation techniques used:

1) *Objective measurements*: measures such as task completion times and accuracy / error rates, scores, positions, movements, number of actions, etc.

2) *Subjective measurements*: measures such as user questionnaires, subjective user ratings, or judgements.

3) *Qualitative analysis*: measures such as formal user observations, formal interviews, or classification or coding of user behaviour (e.g. speech or gesture coding).

4) *Usability evaluation techniques*: evaluation techniques that are often used in interface usability evaluations such as heuristic evaluation, expert based evaluation, task analysis, think aloud methods, or Wizard of OZ methods.

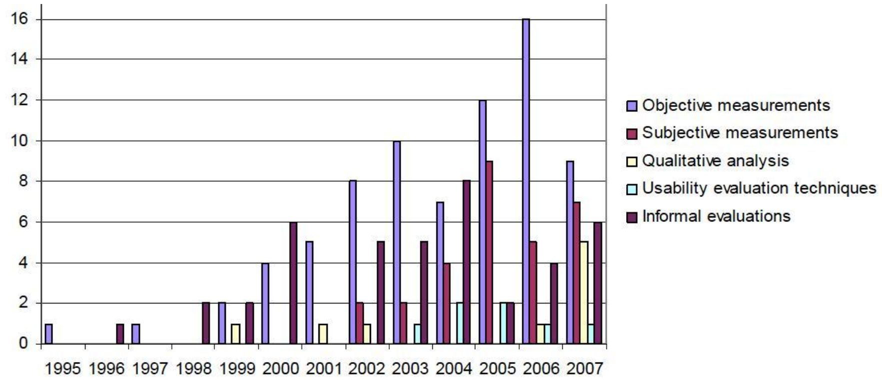


Figure 9.4: Classification of AR publications by evaluation method. [Dünser et al., 2008]

5) *Informal evaluations*: evaluations such as informal user observations or informal collection of user feedback.

Figure 9.4 shows a breakdown of the AR papers reviewed by Dünser et al. [2008] according to the main type of usability method used in the paper. As can be seen the majority of the papers reviewed used objective and subjective measures and very few of them used usability evaluation techniques.

In their work they classify AR papers according to the main type of user evaluation performed. However is it common for researchers to combine several evaluation methods together to explore different aspects of the user experience. Gabbard et al. [1999] provides a design methodology for combining together several different evaluation techniques in Virtual Environment system design. Their model iteratively moves through the stages of (1) User Task Analysis, (2) Expert Guidelines-Based Evaluation, (3) Formative User-Centered Evaluation, and (4) Summative Comparative Evaluation (see Figure 9.5). Outputs at each of these stages help progress the system design to a usable prototype.

Hix et al. [2004] demonstrate how this same methodology can be applied to the design of AR systems. In this case they use the example of designing the Battlefield Augmented Reality System (BARS), an outdoor AR interface for information presentation and navigation for

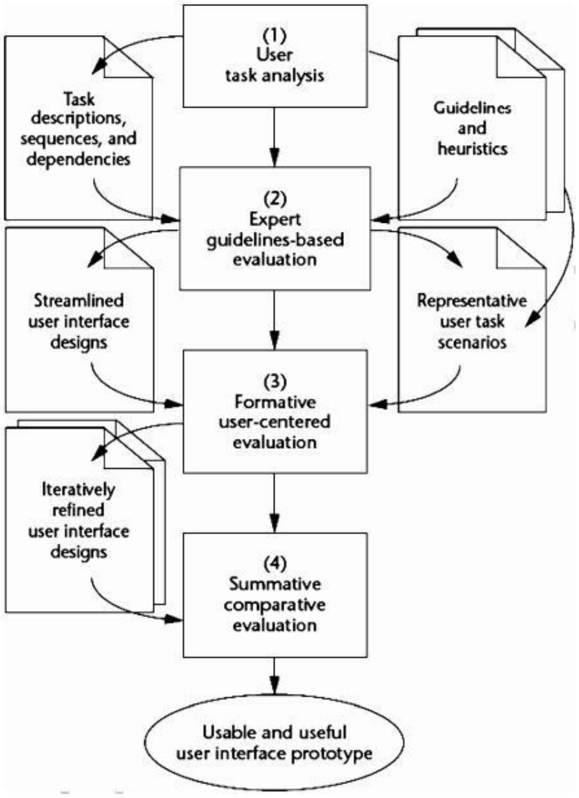


Figure 9.5: Virtual Environment system design methodology. [Gabbard et al., 1999]

soldiers. In this case the design of the system was begun with a user needs and domain analysis to understand the requirements of soldiers in the field. This was followed by an expert evaluation, and user-centered formative and summative evaluations. In all there were six evaluation cycles conducted over a two month period with nearly 100 mockup prototypes created. Figure 9.6 shows one of these prototypes and the AR view provided by the system to assist the user in outdoor navigation.

Hix et al. [2004] reported that applying Gabbard's methodology and progressing from expert evaluation to user-based summative evaluation was an efficient and cost-effective strategy for assessing and improving a user interface design. In particular the expert evaluations of BARS before conducting any user evaluations enabled the identification of any

Figure 9.6: The BARS outdoor AR system. [Hix et al., 2004]

obvious usability problems or missing functionality early in the BARS development life cycle. This allowed improvements to be made to the user interface prior to performing user-based statistical and formative evaluations.

9.3 Summary

In this section we have provided an overview of techniques for AR evaluation, and a review of typical methods that have been used in the past. We began by summarizing the meta-reviews conducted by Swan and Gabbard [2005] and Dünser et al. [2008], showing that in the AR papers that they collected only a small percentage (around 8-10%) contained any formal evaluation. This demonstrated the need for more user studies and evaluations in AR research and development going forward. The report identified three main areas for types of user studies; (1) Perception, (2) Performance/Interaction and (3) Collaboration. We then reviewed example AR systems and user evaluations in each of these areas. Finally we discussed an AR system design methodology that combines different types of evaluation techniques, and show how this was applied in the design of the BARS system. In the next section we will review areas for ongoing research in Augmented Reality.

10

AR Applications Today

In section 3 we presented a short history of Augmented Reality and discussed how the rise of mobile devices and advances in web technologies means that AR experiences are available to more people than ever before. In this section we give some examples of typical modern day AR applications and identify some of the features of successful products. AR technology can be used in many different domains, but we focus on the areas of Marketing, Medicine, Education, Entertainment, and Architecture, discussing applications in each of these areas in turn.

10.1 Education

When new technologies are developed, attempts are often made to try and use them in an educational setting. Augmented Reality is no exception and for over ten years AR technology has been tested in a number of different educational applications. These trials have shown that in some situations AR can help students learn more effectively and have increased knowledge retention relative to traditional 2D desktop interfaces.

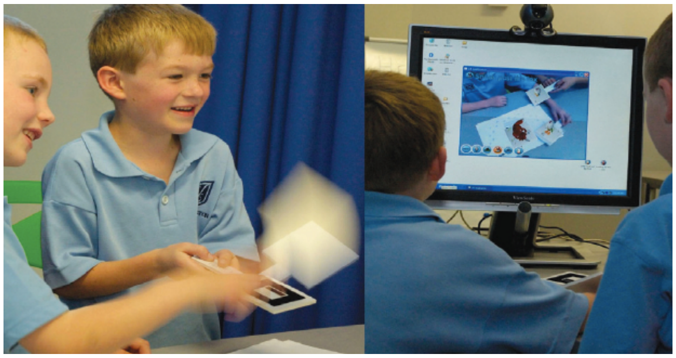


Figure 10.1: Children playing with an AR enhanced book together. [Dünser and Hornecker, 2007]

This is particularly the case for reading and book centric learning, where AR can be used to overlay interactive 3D digital content on the real book pages. In §11.2 we describe the MagicBook application [Billinghurst et al., 2001] and how AR can be used to transition from reading a real book to exploring an immersive virtual reality space. Since the development of the first MagicBook prototype a wide range of other AR enhanced books have been developed by academics and companies, some of which have been commercially available since 2008.

The use of AR enhanced books in an educational setting has been found to improve story recall and increase reading comprehension. One study paired children together and observed them while the interacted with an AR book using physical paddles to manipulate the virtual content (see Figure 10.1) [Dünser and Hornecker, 2007]. They found that children were able to easily interact with the books and AR content. The use of physical objects to interact with the AR objects and characters was very natural. However some children were confused by the mirrored video view shown on the screen, and also had a tendency to interact with the virtual content in the same way as they would with real objects, which didn't always work. Nevertheless, this research shows that AR books could be introduced into an educational setting relatively easily.

A second study explored how AR enhanced books could help enhance story recall [Dünser, 2008]. In this case two groups of children read either traditional books, or books with animated AR scenes as well as the printed text. The children were divided into those that had a high reading ability and those with low reading ability (as determined by their teacher). From the text-based story conditions, those in the high-ability group could retell significantly more key story points than the low-ability group. However, the two groups showed no significant difference in recalling key story points from the interactive AR condition. The researcher concludes that AR books could benefit students who are less able to comprehend text-based learning materials, and so could be very helpful in learning environments.

Another example of the power of Augmented Reality for education is the colAR mix application[1] developed by the company, Puteko. This is an AR colouring experience available on iOS and Android mobile devices. Using the application, users can point their mobile device to a colouring book page and see an animated 3D virtual scene come to life. However unlike other AR applications, with colAR mix users can colour the pages and see the virtual objects textured with their own colours (see Figure 10.2). Any colour that they add to the colouring page will be textured mapped onto the 3D model and seen in the AR scene. Some of the models also have simple interactivity. For example, when the user views the animated dragon model, they can touch a virtual button on the screen to cause it to breath fire. Similarly, in the scene with a dancing girl they can turn on the virtual radio to see her dancing to the music.

The application works by using the Vuforia library to track the black lines on the book page and calculate the phone or tablet camera position from markerless image tracking. Once the camera position has been determined then the pixels from the live video that are inside the different parts of the coloured page can be found. The colours of these pixels are then used to create a texture map that is applied to the animated AR 3D model. The end result is that the AR model appears coloured with the same colours that the user has applied to the colouring book page. A more detailed technical explanation of how the system works can be found in [Clark and Dünser, 2012].

---

[1] `http://colarapp.com`

(a) Uncoloured page **(b)** Coloured AR model

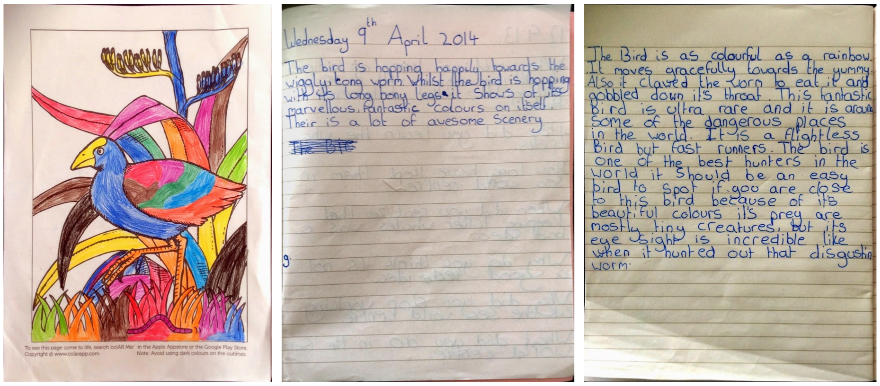
Figure 10.2: ColAR Mix AR colouring book application. `http://colarapp.com`

From a technical perspective the colAR mix application is very simple, however the ability for users to see their own colouring appear in the AR scene creates a very compelling experience. Since it's release in the middle of 2013 the application has become very popular with more than a million downloads across the Android Play store and iOS App store.

Teachers have been using the application in a number of interesting ways in the classroom. For example, one teacher has been using the technology to inspire creative writing[2]. He showed the children the uncoloured page and then asked them to write a short story about it. Then he had them colour the page and used colAR mix to bring it to life. Once students have had a chance to use the AR application and view the content, he asked them to write a story describing what they could see. He found that after trying the AR application, the quality of the writing improved, and children used more precise language in their descriptions. For some of the more reluctant writers, it was especially useful, giving them a lot more ideas to work with. Figure 10.3 shows an

---

[2] `http://mrparkinsonict.blogspot.com.au/2014/04/can-augmented-reality-improve-writing.html`



Wednesday 9th April 2014

The bird is hopping happily towards the wiggly, long worm. Whilst the bird is hopping with its long bony legs it Shows of its marvellous, fantastic colours on itself. Their is a lot of awesome Scenery

~~The Bi~~

The Bird is as colourful as a rainbow. It moves gracefully towards the yummy. Also it clawed the worm to eat it and gobbled down it's throat. This fantastic Bird is ultra rare and it is arou Some of the dangerous places in the world. It is a flightless Bird but fast runners. The bird is one of the best hunters in the world it Should be an easy bird to spot if you are close to this bird because of it's beautiful colours it's prey are mostly tiny creatures, but its eye sight is incredible like when it hunted out that disgustin worm.

(a) Coloured page **(b)** Before trying AR **(c)** After trying AR

Figure 10.3: Using ColAR mix to teach creative writing. `http://mrparkinsonict.blogspot.com.au/2014/04/can-augmented-reality-improve-writing.html`

example of a page coloured in by one of the children in his class, and the writing sample before and after using colAR mix. As can be seen, the student writes over twice as much after using the AR application.

As can be seen, these examples show the potential educational benefit of AR technology. Even simple AR scenes can be used to motivate children in a classroom. Results show that AR's high level of interactivity enhances learning, particularly for students who learn through kinesthetic, visual, and other non-text-based methods. After trying the technology one teacher said that there is "no question that AR will prove to be a highly effective medium both for entertainment and education." [Billinghurst and Dünser, 2012].

10.2 Architecture

Augmented Reality is an ideal technology for showing virtual information superimposed over the real world, and as such it can be used to solve one of the most important problems in architecture; how can you see a building that hasn't been built yet? Traditionally architects used a range of different tools to show their clients what their buildings will be like, including 2D plans, physical models, 3D renderings, interactive walkthroughs and animated fly throughs. However there are a number of shortcomings with these methods. For example, 2D plans can show in great detail the layout of a building, but sometimes the client has trouble imagining what the building would look like from the 2D drawing. Physical models, 3D renderings and interactive fly-throughs help the client understand what the building looks like, but they are a reduced scale version of the final space. These may not convey a clear understanding of what the building will look like full-size on the final location.

Mobile AR can be used to view full-sized 3D virtual models of future buildings on the local where they will be built. This allows those interested in the building to clearly understand what it will look like. The application could also be used as a planning tool, placing several versions of the same building on site and allowing the client to provide feedback about the different design options.

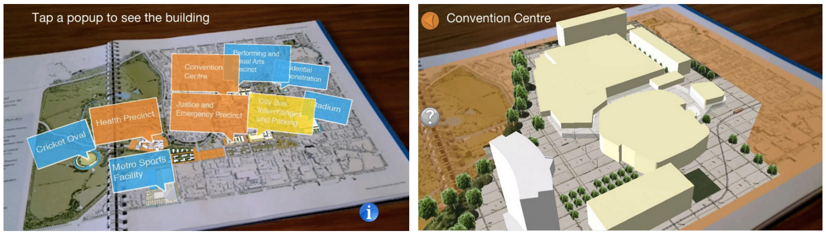
There are many examples of AR applications that can provide this functionality, but in this section we describe two in more detail. These applications were developed by the HIT lab NZ in response to a terrible natural disaster. In September 2010 and again in February 2011, the city of Christchurch, New Zealand was hit by several very strong earthquakes, magnitude 7.1 and 6.9. The earthquakes caused large scale damage with the result that over 70% of the buildings in the inner city needed to be demolished in the rebuild process. The city council had a problem how to communicate to the residents what the city would look like once the rebuild was finished. There was also considerable interest in communicating to visitors what the city used to look like before the earthquakes destroyed most of the historical buildings.

In response to this, the HIT Lab NZ developed two mobile AR applications CCDU AR and CityViewAR [Lee et al., 2012], both of which were made available on the Android Play store and iOS app store. CCDU AR was an application designed to show architectural information superimposed over the printed pages of a book the city council had produced explaining the city rebuild and showing concept images of what the future landmark buildings would look like. The application was developed for the iOS and Android mobile devices and used the Vuforia library for image based tracking of the real book pages.

The Unity game engine was used to show the 3D virtual building models and add simple interactivity to the application.

Figure 10.4 shows the application being used. The user can take their tablet or smart phone and when they run the application and point the camera at the printed map of the city they will see 3D virtual labels appearing showing them the key building redevelopments in the city (Figure 10.4(b)). The labels appear fixed in space and turn to face the user as they move around the map, so that they are always readable. The user can then tap one of the labels to see a larger 3D model of the building appearing fixed to the printed map (Figure 10.4(c)). Some of the buildings also had simple interactivity added to them. For example if the user moved their device closer to the virtual stadium model they would hear the sound of the crowd growing louder, or they can tap on the police station model to see virtual police cars exit the building and hear siren sounds.

(a) Using the application



(b) Label view **(c)** Building view

Figure 10.4: CCDU AR mobile AR application.

The application was used by the Christchurch City Council at a number of public events, and was also made freely available so that any interested people could try it out. Overall the feedback was very positive and users felt that they could more clearly understand the key redevelopments that were going to take place, and also how the final buildings could look.

The CCDU AR application enabled people to better understand what key buildings in the rebuilt city would look like, but it had the limitation that users weren't able to see the buildings on-site. The CityViewAR application addressed this limitation and allowed users to see full-sized virtual buildings on the real site. This was achieved using the HIT Lab NZ's Outdoor AR framework [Lee and Billinghurst, 2013], a software library developed specifically for creating outdoor AR applications for mobile devices. Unlike the CCDU AR application, it was designed to using GPS and compass sensor input and so enabled outdoor tracking.

The Outdoor AR framework consists of two layers of components (see Figure 10.5). The lower layer provides abstraction of functional modules that are essential for building outdoor AR applications. These components include basic data structures, tracking sensors, network communication, 3D graphics and sound rendering, and tools for managing UI elements, file parsing and data loading. The framework includes a custom 3D graphics engine based on a scenegraph data structure designed and optimized for mobile rendering of geo-located 3D scenes. The Scene and Tracking data manager components play the role of models, and the Map, AR, and List View components are higher-level abstractions of the AR scene visualization function. These model and view components are ready-to-use components and developers can simply choose which components they want to use. Though developers are free to modify these components, the controller components are where most customization typically happens, mixing and matching which model and view components that the application is going to use.

Using the Outdoor AR framework, the CityViewAR application was developed to help people explore destroyed historical sites and buildings after the major earthquakes in Christchurch. Using the applica-

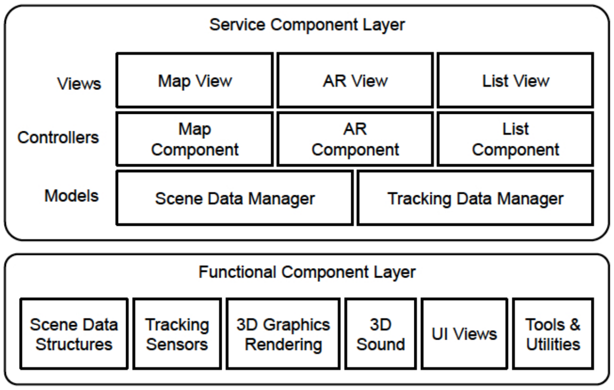
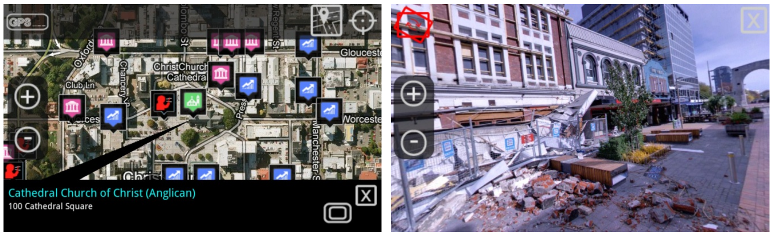


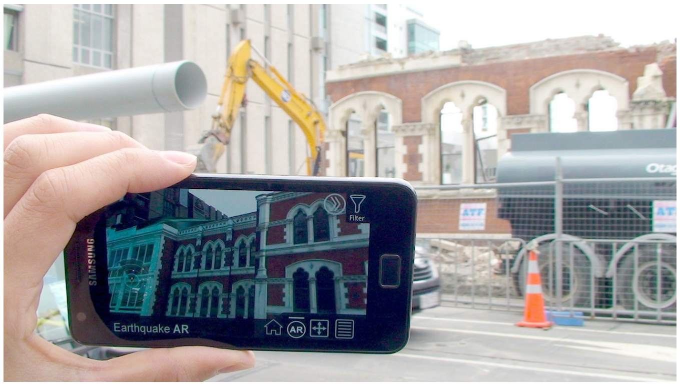
Figure 10.5: Outdoor AR Library architecture. [Lee and Billinghurst, 2013]

tions people can walk on site and see information about the buildings that used to be around them. The geo-located content is provided in a number of formats including 2D map views, AR visualization of 3D models of buildings on-site, immersive panorama photographs, and list views of all the content available. Figure 10.6(a) shows the map view with points of interest icons shown on it. Users can select icons they are interested in and switch to a panorama view (Figure 10.6(b)) or AR view (Figure 10.6(c)). The panorama view allows users to view immersive 360 photospheres take at their location several weeks after the earthquake, showing the destruction at the time. The users can rotate their mobile device around to see different portions of the panorama, using the compass in the device to set the viewing orientation. The AR view shows virtual images superimposed over a live video background, transforming the empty lots of Christchurch into the buildings that existed before the earthquake.

A user study was conducted to see if the AR component enhanced the user experience (AR condition), compared to using the application without AR viewing (non AR condition) [Lee et al., 2012]. When the AR feature was available users used it over half of the time that they were using the application, far more than the map or panorama views, and they judged their overall experience to be better than in the non-AR condition. Participants were also asked which features they liked



(a) Map view **(b)** Panorama view



(c) AR view

Figure 10.6: Using the CityViewAR application to see a historical building on-site. [Lee et al., 2012]

the most, and in the AR condition 42% of the users picked the AR view, while in the non AR condition 62% of people picked the panorama view. People with the AR condition also moved around the street more, trying to explore the different elements of the 3D buildings they were able to view. Overall users enjoyed having a rich interface that used several different ways to show the building information, but they felt that the AR element definitely enhanced the user experience.

These applications show that even with current technology, AR experiences can be developed that help solve the key problem of client communication for architects. Wearable devices will even accelerate the use of AR experiences in architectural applications in outdoor environment (see Figure 5.10 showing CityViewAR running on Google Glass). CityViewAR and CCDU AR are just two of dozens of mobile AR applications available that enable users to view architectural content ways that are not possible with more traditional tools.

10.3 Marketing

One of the greatest opportunities for AR is in the use of the technology for product marketing. Marketing material is typically designed to capture the attention of a person and provide them with motivation to learn more about the product. AR technology can be used to create experiences that are more memorable than other more traditional forms of marketing.

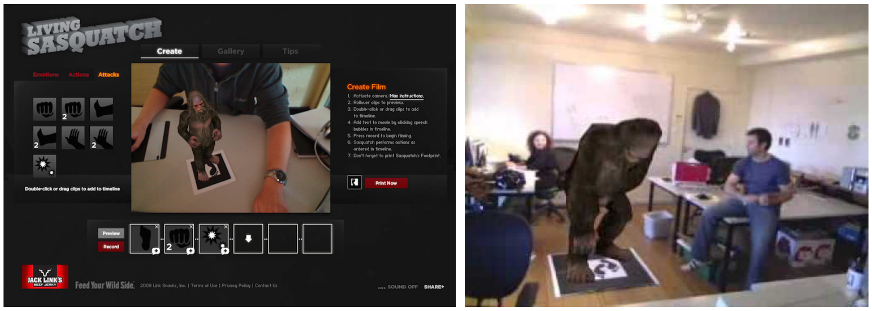
AR can be also be used to develop experiences that are impossible with other technology. For example, in section 3, we already mentioned the Living Sasquatch marketing campaign developed created by Boffswana to market Jack Links beef jerky. This used a simple web interface (see Figure 10.7(a)) to allow users to create their own animated AR scene with a Sasquatch monster. This was developed through using a FLARToolKit based Flash application installed in a web page that didn't require any complex software installation by the end user. The application allowed people to arrange sasquatch animations simply by dragging and dropping icons on the screen. Once the animation was completed they could view the output in the AR window in the centre of the screen.

The public response was very strong with hundreds of people creating images and short videos of their interaction with the sasquatch and posting them online. People discovered that the size of the sasquatch depended on the size of the AR tracking marker, and so were able to create a number of humorous situations[3] (see Figure 10.7(b)). In this way the technology enables users to have a new experience, but also for the end users to provide a word of mouth campaign for the brand being promoted.

A second way that AR can be used in marketing is to enhance traditional printed media advertising. Many companies have developed AR experiences that allow people to point their phones at magazine

---

[3] See for example, `https://www.youtube.com/watch?v=eA1X0vfJGs0`



1. Activate camera. Mac instructions.
2. Rollover clips to preview.
3. Double-click or drag clips to add to timeline.
4. Add text to movie by clicking speech bubbles in timeline.
5. Press record to begin filming.
6. Sasquatch performs actions as ordered in timeline.
7. Don't forget to print Sasquatch's Footprint.

(a) Web interface **(b)** User created image

Figure 10.7: Living Sasquatch interface and user experience.

advertisements or posters and see virtual 3D content or 2D movies appear. For example, in 2012 company Explore Engage developed an AR marketing experience for the Transformers 3 movie that allowed people to point their phone at the movie poster and see 3D content from the movie pop out[4]. This turned the static poster into an interactive experience that showed a wide range of different clips and content from the movie. It also allowed users to unlock an immersive first-person AR game, playing as Optimus Prime defending Earth against the evil forces of the Decepticons. The marketing campaign was a big success with over 400,000 app downloads in the first few weeks of the movie opening.

From these examples we can see that Augmented Reality is a technology that has proven itself in several different application areas. However the technology is just being applied in commercial settings and it is obvious that even more real world applications will emerge in the coming years. In the next section we discuss research areas that are being explored that will increase the number of application areas where the technology could be used.

---

[4] `http://www.tf3ar.com`

11

Research Directions

Augmented Reality have been researched for nearly 50 years and is beginning to become more mainstream through the commercialization efforts of many companies. Despite this there are still a number of research areas that need to be explored further before the technology reaches its full potential. In Azuma et al. [2001] review paper they identify the major obstacles limiting the wider use of AR as falling into three themes: (1) technological limitations, (2) user interface limitations, and (3) social acceptance issues. Martínez et al. [2014] used the Technology Acceptance Model [Davis et al., 1989] and Roger's innovation diffusion theory [Rogers, 2010] to identify the following bottlenecks to its widespread adoption; (1) No AR standards, (2) Limited computational power in AR devices, (3) Tracking Inaccuracy, (4) Social acceptance, (5) Information overload.

In 2008, Zhou et al. [2008] reviewed ten years of ISMAR papers and identified the main research trends over that time as being in the areas of Tracking, User Interaction and Display Technology. So it is evident that research has been conducted in most of the areas identified by Azuma and Martinez, although not in Social Acceptance. In this section we review ongoing research and future opportunities in the four areas of (1) Tracking, (2) Interaction, (3) Displays, and (4) Social Acceptance. In addition to these topics, Zhou et al. [2008] identified other AR research as being conducted on topics such as rendering and visualization methods, evaluation techniques, authoring and content creation tools, applications, and other areas. However space limitations prevent a deeper review of these areas.

11.1 Tracking

Tracking is one of the most popular areas for AR research because it is one of the most important of the AR enabling technologies. As we have seen from section 4, AR tracking systems have evolved from simple fiducial marker based systems, to natural feature tracking, and hybrid sensor based methods. However there is still significant work that needs to be done before the vision of "anywhere augmentation" [Höllerer et al., 2007] is achieved, where users will be able to have a compelling AR experience in any environment.

In section 4 we showed how that there are a number of reliable systems for images based tracking from printed images, or outdoor tracking using GPS and other sensors. However there are still a number of important research problems that need to be solved, including (1) Wide area tracking, (2) Reliable indoor tracking, and (3) Indoor/Outdoor ubiquitous tracking, among others.

For reliable outdoor Augmented Reality, tracking methods are needed that can provide accurate location over a wide area. GPS hardware can be used for position sensing, but consumer grade GPS in mobile devices typically provides an average accuracy of 5 Ð 10 meters depending on satellite signal strength. One alternative method is to use computer vision techniques to estimate the camera pose relative to known visual features, often in conjunction with other GPS and inertial sensors [Reitmayr and Drummond, 2006]. However this approach can be difficult to scale to wide area tracking because of the difficulty of creating a large scale feature set [Castle et al., 2008] or capturing thousands of pictures for image matching [Arth et al., 2009]. In contrast Ventura and Hollerer [2012] have developed a system that

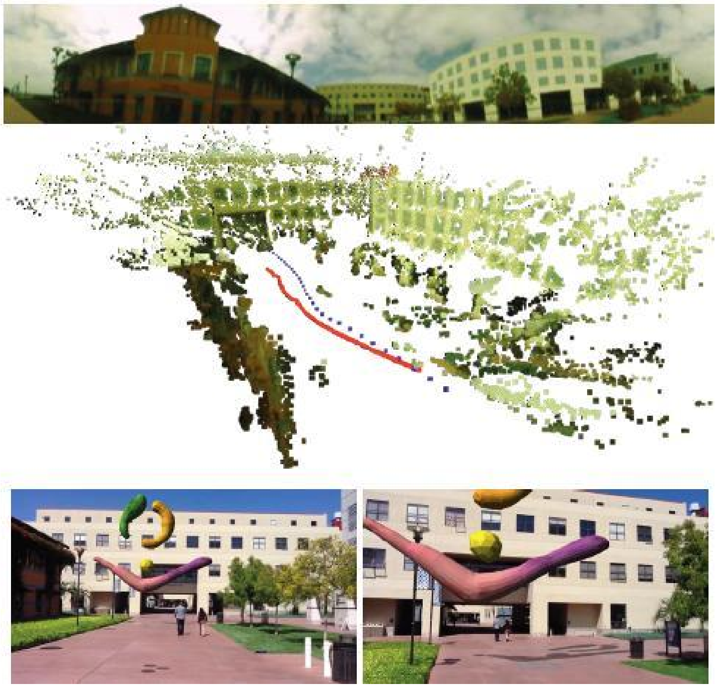


Figure 11.1: Wide area AR tracking using panorama imagery (top) processed into a point cloud dataset (middle) and used for AR localisation (bottom). [Ventura and Hollerer, 2012].

combines panoramas captured from several positions to create a point cloud model which can then be used for feature based localization using a remote server, and real time tracking on a mobile device. In this way they were able to track over of 1000 m2 with a positional accuracy of less than 25cm error, and rotational accuracy of under 0.5 degrees. Figure 11.1 shows the data captured by the system and the final tracking results. This approach looks promising for wide area tracking in an unprepared outdoor environment, although more work will need to be done to verify this.

In favourable conditions, GPS technology can be used to provide accurate outdoor tracking however lack of satellite coverage means that GPS systems fail indoors and other methods must be used. A range of different methods have been explored for providing robust indoor tracking. For example, Dissanayake et al. [2001] showed how ultra-sound could be used for position localisation inside buildings, and demonstrated several applications that this could enable, such as navigation, asset localisation, and creating virtual interactive buttons in space. However this system required the installation of a set of ceiling mounted ultra-sonic receivers and so become more expensive the larger the space tracked over, the tracking accuracy is also limited and not precise enough for exact overlay of virtual content on real objects. Other methods for precise indoor tracking use a camera combined with a 2D laser tracker for range finding [Scheer and Müller, 2012], computer vision tracking from markers pasted over the walls [Wagner, 2002, Schmalstieg and Wagner, 2007], and hybrid systems with computer vision, inertial sensors and ultra-wide-band tracking [Newman et al., 2006], among others.

One particularly promising direction for research is the use of handheld depth sensors for indoor tracking. The Google Tango project[1] is a handheld device that uses customized depth sensors with motion tracking cameras and inertial sensors to update the position and orientation of the device in real time, while at the same time creating a map of the users environment. This has the advantage that it is completely self contained and doesn't require any image markers or electronic sensors to be placed in the tracked environment. Figure 5.13 shows a typical depth image of the real world captured by Tango. Due to its precise indoor positioning, Tango provides an ideal platform for indoor AR. Other technologies, such as Intel's RealSense[2], will also provide depth sensing on handheld tablets, and so there are opportunities for further research in this area.

One of the existing challenges is seamless switching from outdoor to indoor tracking environments. This is necessary for the development of ubiquitous tracking systems that work continuously as the user walks in and out of buildings, or other locations. Piekarski et al. [2003] developed an early prototype that combined GPS outdoor tracking (accurate to

---

[1] `https://www.google.com/atap/projecttango`

[2] `http://www.intel.com/content/www/us/en/architecture-and-technology/realsense-overview.html`

50cm) with indoor marker based tracking (accurate to 20 cm), and automatically switching between the two based on GPS reception. More recently, Behzadan et al. [2008] developed a system that combined indoor tracking using Wireless Local Area Networks with GPS tracking for outdoor position sensing, while Akula et al. [2011] combined GPS with an inertial tracking system to provide continuous location sensing. Huber et al. [2007] describe a system architecture for ubiquitous tracking environments. There needs to be more research conducted in this area before the goal of ubiquitous tracking will be achieved.

11.2 Interaction

Section 7 provided a review of different interaction methods for Augmented Reality. As was shown, early AR interfaces used input techniques inspired from desktop interfaces or Virtual Reality, but over time more innovative methods have been used, such as Tangible AR, or natural gesture interaction. There is still significant opportunity for on-going research in new interaction methods, especially in the areas of (1) Intelligent systems, (2) Hybrid user interfaces, and (5) Collaborative systems, among others. In this section we provide a brief overview of some opportunities in each of those areas.

Previous research has shown the AR is a very natural way to interact with virtual content, but in many cases the interfaces themselves have not been very intelligent with simple interactions that don't respond in different ways to user input. Over the last twenty years the field of Intelligent User Interface (IU) has emerged [Sullivan, 1991] which explores how Artificial Intelligence can be combined with Human Computer Interaction methods to produce more responsive interfaces. Unfortunately there has been little research in AR on how to include IUI methods. Some researcher have begun exploring the user of virtual characters that display limited intelligence, for example the Welbo interface was a character seen in the real world that responded to simple speech commands [Anabuki et al., 2000], and Mr Virtuoso was a handheld AR interface that used an AR virtual character to teach about art [Wagner et al., 2006].

One topic that has a lot of potential in this area is Intelligent Training Systems (ITS). Earlier research has shown that both AR and ITS applications can significantly improve training. For example AR technology allows virtual cues to be overlaid on the workers equipment and help with performance on spatial tasks. In one case an AR-based training system improved performance on a vehicle maintenance tasks by 50% [Henderson and Feiner, 2009]. Similarly, ITS applications allow people to have a educational experience tailored to their own learning style, providing intelligent, responsive feedback, and have been shown to improve learning results of one letter grade or higher, produce significantly faster learning and impressive results in learning transfer [Mitrovic, 2012, VanLehn, 2011]. However there has been little research that explores how both technologies can be combined together.

Overall, current AR training systems are not intelligent and current ITS do not use AR for their front end interface. Current AR training systems typically provide checklists of actions that must be done to achieve a task, but don't provide any feedback as to how well the user has performed that task. One exception to this is the work of Westerfield et al. [2013] who combined the ASPIRE constraint based ITS [Mitrovic et al., 2009] with an AR interface for training people on computer assembly. Using this interface the system would monitor user performance and automatically provide virtual cues to help them perform the tasks they were training on (see Figure 11.2). In trials will this system it was found that the AR with ITS enabled people to complete the training task faster and score almost 30% higher on a retention of knowledge test compared to training with the same AR interface without intelligent support. This is an encouraging result and shows significant opportunity for more research in the area.

A second promising area for research is hybrid AR interfaces and interaction. The first AR systems were stand-alone applications in which the user focused entirely on using the AR interface and interacting with the virtual content. However, as we saw in Milgram's Mixed Reality continuum and the Metaverse taxonomy, AR technology is complimentary to other interface technologies and so hybrid interfaces can be built which combine AR with other interaction methods. For exam-

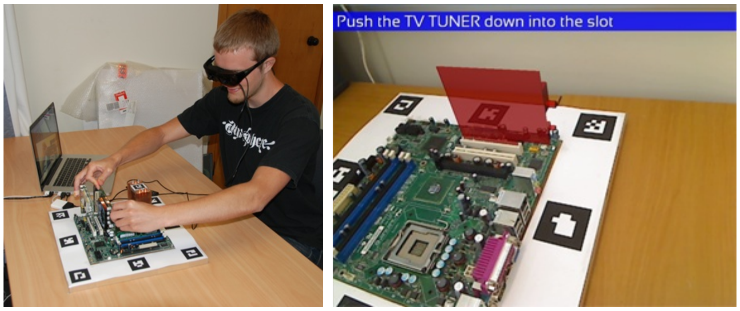
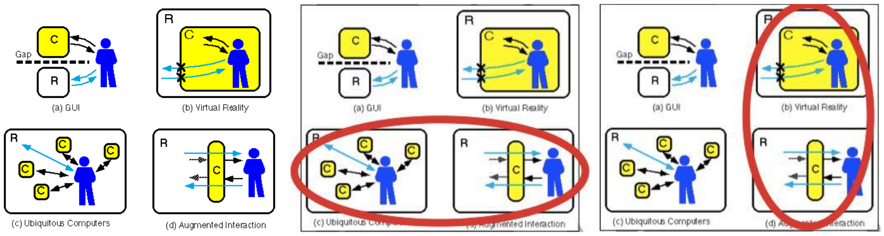


Figure 11.2: Using AR and ITS for training computer assembly. [Westerfield et al., 2013]

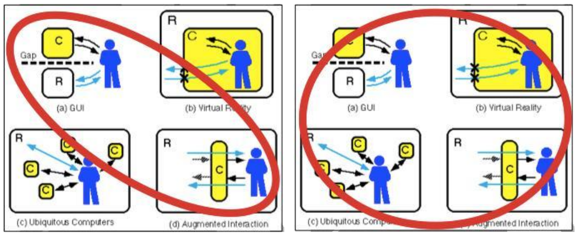
ple, if we consider Rekimoto's comparison of HCI styles from section 2, there are artificial boundaries between the four interface types shown. As Figure 11.3 shows, interesting research can be explored by crossing these boundaries.

Considering each of Figures 11.3(b)~(d) separately, we can see some interesting opportunities for further research. For example, Figures 11.3(b) shows that the boundary between AR and Ubiquitous Computing can be crossed to create hybrid AR/Ubicomp interfaces. This could be accomplished by using AR to make visible what is normally invisible in Ubiquitous Computing applications. Rauhala et al. [2006] developed a handheld AR interface that used virtual cue to show the data from humidity sensors hidden behind a wall in a ubiquitous computer interface (see Figure 11.4). This allows the used to see superimposed over the real world data that is usually only visible on a desktop interface remote from the location that it is captured in. In this case a physical marker placed on the wall was used for mobile phone tracking and the sensor information is collected by a computer and wirelessly transmitted to the mobile device for AR visualization. There are also other examples of how AR can be used to make visible data captured by ubiquitous computers and sensing systems [Goldsmith et al., 2008, Veas, 2012, van der Vlist et al., 2013].

Figure 11.3(c) shows boundary crossing between AR and VR interfaces. One example of this is the MagicBook interface [Billinghurst et al., 2001] that uses AR to explore the full length of Milgram's Mixed



(a) Original classification **(b)** AR and Ubicomp **(c)** AR and VR



(d) AR and GUI **(e)** AR, VR, GUI, and Ubicomp

Figure 11.3: Research opportunities in hybrid AR interfaces.

Reality continuum. The system consists of a real book and handheld AR display. The user can read the book without the use of any technology (Figure 11.5(a)). However when they look at the book pages through the handheld display they can see animated AR content overlaid on the printed page (Figure 11.5(b)). When they see an AR scene that they like they can touch a button on the handheld display and transition into a fully immersive VR viewing mode (Figure 11.5(c)). In the VR view users are free to turn their heads to look around the virtual environment and also to travel through the space. In this way the MagicBook uses AR to support seamless transition from Reality to Virtual Reality. Other interfaces that also allow users to move from AR to VR include [Nilsen and Looser, 2005], [Grasset et al., 2011], and [Kiyokawa et al., 1999].

AR interfaces can also be combined with more traditional desktop interfaces (Figure 11.3(d)). The Emmie system of Butz et al. [1999] is a system that mixes several display and interaction types together,

Figure 11.4: Using mobile AR for visualization ubiquitous computing sensor data. [Rauhala et al., 2006]

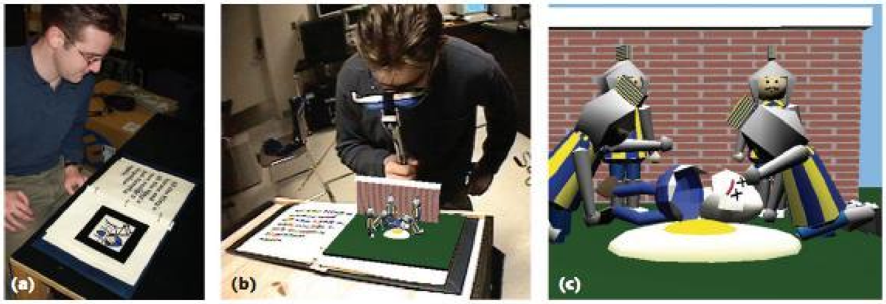


Figure 11.5: The MagicBook interface showing transitioning into a VR experience. [Billinghurst et al., 2001]

allowing users to view AR content at the same time as using 2D GUI interfaces. Wearing see-through HMDs, users can look at normal desktop screens and view and interact with the content on the screen using a normal mouse-based interface. However, using a 3D wand they can also pull the content off the screen and see it as a 3D AR object floating in space in front of them. In this way they can drag and drop content between different types of interfaces and seamlessly mix GUI and AR interaction. More recently Benko et al. [2005] developed a hybrid AR and table top interface that combine gesture interaction in space with AR objects and touch input on the table surface.

Finally, Figure 11.3(e) shows how AR can be combined with GUI, VR and Ubiquitous Computing interfaces. In 2006 Kim et al. [2006]

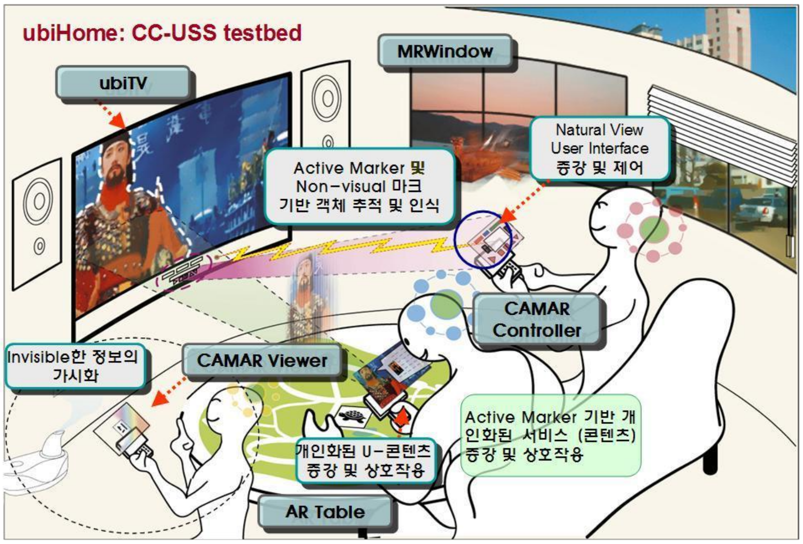


Figure 11.6: Ubiquitous-VR interface concept. [Kim et al., 2006]

proposed the idea of Ubiquitous Virtual Reality (U-VR) which they defined as "..creating ubiquitous VR environments which make VR pervasive into our daily lives and ubiquitous by allowing VR to meet .. ubiquitous computing". Their view is that the technologies of Ubiquitous computing, Virtual Reality and Mixed Reality can be seamlessly blended together to provide people will constant access to information, anywhere and at any time using a variety of interface presentations. To demonstrate this concept they developed the ubiHome [Oh and Woo, 2005] testbed (see Figure 11.6) which showed a number of concepts such as the handheld AR CAMAR Viewer [Suh et al., 2007], an AR table, immersive Mixed Reality Window, and smart ubiTV [Oh et al., 2005], among other objects. These are all connected together with a smart services infrastructure, so for example, the CAMAR Viewer can be used to control the ubiTV and content could be presented over both the AR table and MRWindow.

11.3 Displays

Although AR display technology has significantly improved from Ivan Sutherland's original system, current displays are still very far from Sutherland's vision of the "Ultimate Display" [Sutherland, 1965]. There are important research opportunities available in the design of head mounted displays, in projection technology, in contact-lens displays, and other areas.

In section 5 we provided an overview of the use of optical-see through head mounted displays for Augmented Reality. However traditional optical-see through displays have a number of disadvantages, such not being able to support a wide field of view, or provide true occlusion of the real world. The ideal display would be one that provides a wide field of view, supports occlusion of the real world, and provides images at a variety of focal planes, all in a small unencumbering form factor. Researchers have been exploring each of these areas although there is still more work to be done.

A number of different designs for optical see-through displays have been developed that address these shortcomings. For the occlusion problem, Kiyokawa et al. [2001] and others [Cakmakci et al., 2004, Santos et al., 2008, Gao et al., 2012] have been researching how electronic masking elements can be added to optical see-through displays. For example, Cakmakci et al. [2004] have designed an optical see through display with a spatial light modulator and polarization-based optics that can block parts of the scene that should be occluded. To address the challenged of small field of view (FOV), Cheng et al. [2011] uses tiled optical elements to create an optical see-through display with 55 degree diagonal FOV. Liu et al. [2008] uses a liquid lens to vary the focal length of an optical see-through head mounted display, enabling users to see virtual content overlaid on real objects at a wide range of distances. Hu and Hua [2012] address the same problem by using a wedge-shaped eyepiece and a compensator. However, each of these display prototypes just addresses one of the shortcomings of optical see-through display.

The recent work of Maimone and Fuchs [2013] is one of the first optical see-through AR displays that tries to address all of these problems

Figure 11.7: Computational AR eyeglasses and view through the display. [Maimone and Fuchs, 2013]

in one design. They have developed a prototype that uses a computational displays approach with stacked spatial light modulator (SLM) layers placed directly in front of the eyes. These SLM are transparent elements that can provide a per-pixel occlusion of the view of the real world and also support for multiple focal planes, achieving image overlays at different focal depths. The first version of the display is shown in Figure 11.7. However it has relatively poor image quality, requires calibration of the stacked layers, and has an occlusion mask with blurry edges. Thus there is still room for significant improvement.

A second area of display research that will offer interesting research opportunities is in head mounted projection displays. The first head mounted projection systems were developed by Fisher [1996] and Fergason [1997]], although it was Rolland [Rolland et al., 1998], Kijima [Kijima and Ojika, 1997] and Inami [Inami et al., 2000] that provided many of the first examples of head mounted projection display (HMPD) used for AR experiences. Rolland et al. [Rolland 05] provides a summary of early HMPD research. These early projector systems were bulky and lacked portability, however recent advances in pico-projector technology has meant that these limitations can be overcome. For example, the REFLCT system [Krum et al., 2012] uses a small HMPD and retroreflective material to allow users to see projected AR content without the need for placing an optical combiner in front of the eyes. The castAR[3] system is smaller still, with two micro-projectors directly over the eyes creating a stereo 3D projected image. In this case the user has to wear lightweight polarized glasses and look at a retro-reflective surface to see the 3D imagery.

---

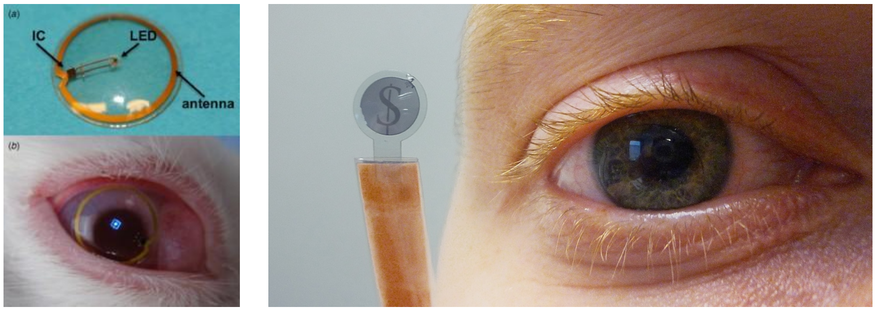
[3] `http://technicalillusions.com`

Now that HMPD are as small and lightweight as head mounted displays, there is still considerable research to be conducted. For example, a number of people have begun to explore interaction methods should be used for projected AR content. Hua et al. [2003] explore the use of datagloves for gesture based interaction, Brown and Hua [2006] use physical objects as tangible interface widgets, and researchers have developed projected AR interfaces that use freehand input [Benko et al., 2012]. Another important problem is exploring how to overcome some of the limitations of projecting on the retro-reflective surface, such as real objects always occluding the projected AR content, even when the content is supposed to be in-front of the real object. This may be solved by creating hybrid HMD and HMPD systems.

One of the most interesting types of displays being currently researched is contact lens based displays. One of the long-term goals of AR researchers is to have a head worn display that is imperceptible to people surrounding the user. This might be achievable through in-eye contact lens displays [Parviz, 2009]. Using MEMS technology and wireless power and data transfer a contact lens could be fabricated that contains active pixels. Pandey et al. [2010] developed a single pixel contact lens display that could be wirelessly powered. This was then tested in the eye of a rabbit with no adverse effects [Lingley et al., 2011] (see Figure 11.8(a)). Since that time researchers have developed a curved liquid crystal cell that can be integrated into a contact lens [De Smet et al., 2013] (see Figure 11.8(b)). However there are a number of significant problems that will need be to solved before such displays become readily usable, such as adding optics to allow the eye to focus on the display, ensuring sufficient permeability to oxygenate the cornea, and providing continuous power and data.

11.4 Social Acceptance

As Augmented Reality technology has moved from the lab to the living room, some of the obstacles that prevent it from being more widely used are Social rather than Technological, particularly for wearable or mobile systems.



(a) Single pixel contact lens in rabbit eye [Lingley et al., 2011]

(b) Curved liquid crystal display embedded in contact lens [De Smet, 2014]

Figure 11.8: Prototype contact lens displays.

For example wearable AR systems have shrunk in size from bulky backpacks to small head worn displays like Google Glass[4]. Despite this, there is still considerable social pushback that is preventing devices like Google Glass from being accepted in some social settings. For example, in a recent survey of 4,600 US adults, only 12 percent said that they would be willing to wear "augmented reality glasses" that came from a brand they trusted[5]. This reluctance could be for several reasons such concerns about privacy [Hong, 2013], or fear that using the technology will make a user look silly, or make the user a target for thieves.

These concerns are not just confined to head worn wearable systems. For example, if a person is walking through a city with a mobile phone or tablet and using an AR browser to help them navigate, or viewing AR content, they may need to hold the phone at eye level in front of them while walking (see Figure 11.9). This is an unnatural pose that may make them feel silly, look foolish, or make other people feel that they are being filmed.

A number of researchers have raised social acceptance as an important issue for AR [Azuma et al., 2001, Martínez et al., 2014, Carmigniani et al., 2011, Narzt et al., 2006], but initially there was been little

---

[4] `http://www.google.com/glass`

[5] `http://www.newsfactor.com/story.xhtml?story_id=12100BWEZRVS`

Figure 11.9: Walking while looking at AR virtual tour content on a tablet. [Lee et al., 2013a]

research in the field. For example, in Zhou's ISMAR research review paper, none of the papers published up to 2007 had focused on Social Acceptance. More recently there has been some work addressing this issue. For example Nilsson and Johansson [2008] describes an AR training interface for a medical device in a hospital and evaluation that was performed in terms of technology usefulness and social acceptance. The study participants were very positive about the use of AR technology as a tool for clinical instruction and there were no issues raised in terms of social acceptance. Similarly Rasimah et al. [2011] report on a study of the perceived usefulness and acceptance of AR in a university setting with more than 60 biomedical student subjects. They found that nearly 95% of the students felt that AR would useful in teaching and learning, and over 85% said they would use the technology frequently if they had access to it.

However both of these studies we conducted in professional or educational settings. In a hospital is it common for medical practitioners to wear devices to help them perform their tasks, while in a university students often use a variety of pieces of equipment in labs and classes for short periods of time. It is very different to use a wearable or handheld AR device in a social setting or in public in front of strangers.

There has been little research dedicated to the topic. Olsson points out ".. despite a large body of AR demonstrators and related research, the AR research community lacks understanding of what the user experience with mobile AR services could be, .. especially regarding the emotional and hedonic elements of user experience." [Olsson et al., 2013]. The closest research to this is the work of Grubert et al. [2011] who conducted an online survey of 77 people who have used mobile AR browser applications. They asked them about the social aspects of mobile AR use and around 20% of users said that they experienced social issues with using an AR browser application either often or very often. A similar percentage also said that they had experienced situations in which in which they refrained from using the application because they were worried about social issues. This is a relatively small percentage, but users were using mobile AR applications for just a short period of a time each task (70% of users reported using the application for less than 5 minutes at a time), and so were unlikely to draw significant attention.

It would be expected that social acceptance problems will be significantly higher with a person that is wearing a head mounted display all the time while interacting with AR content. Unfortunately this is an area that has not been well studied in the AR community, but as most social acceptance issues seem to arise from the use of mobile or wearable AR systems, then there is related from the wearable computing community that could be beneficial. For example, Serrano et al. [2014] explored the social acceptability of making hand to face gestures for controlling a head mounted display, finding that they could be acceptable in different social contexts, but care would needed to taken around the design of the gestures. However their studies were in a controlled laboratory environment and not in a public setting where the wearable computer would be typically used. Rico and Brewster [2010] have previously evaluated the social acceptability of gestures for mobile user (although not wearable computers), finding that location and audience have a significant impact on the type of gestures people will be willing to perform. They found that gestures that required the participant to perform large or noticeable actions were the most commonly disliked gestures. Other research has explored the social acceptability of shoe mounted input devices [Bailly et al., 2012], or minimal displays [Costanza et al., 2006].

This research shows that social acceptability is an important issue in handheld or wearable AR experiences, especially those that are expected to be used in a public setting. This is an area that has had little research to date from the AR community and so could form an important area for work in the future. Of course as the technology becomes more unobtrusive with displays and input devices vanishing into clothing then some social concerns may go away.

11.5 Summary

In this section we have reviewed four areas that will provide significant opportunities for further research in AR in the coming years. However, there are many other topics that can also be studied in the field of AR. Zhou et al. [2008] identified a total of 11 categories of research papers that had been published in the ten years of the ISMAR conference until 2008, and categories such as Social Acceptance or Collaborative interfaces were not included amongst these. So it should be accepted that there will be at least as many research topics going forward. As the field matures, the challenge for AR researchers will be to continue to identify the most significant obstacles that prevent the development of compelling AR experiences and to find ways to overcome them.

12

Conclusion

Nearly fifty years ago Ivan Sutherland imagined an "Ultimate Display" that would allow people to see digital content become part of their real worlds [Sutherland, 1965]. In a famous quote he says: "The ultimate display would, of course, be a room within which the computer can control the existence of matter. A chair displayed in such a room would be good enough to sit in. Handcuffs displayed in such a room would be confining, and a bullet displayed in such a room would be fatal. With appropriate programming such a display could literally be the Wonderland into which Alice walked."

Although complete Science Fiction at the time, this article has shown how Augmented Reality is beginning to achieve Sutherland's vision. What was once usable only in research labs is now available to hundreds of millions of people through the technology they have in their pockets, homes and workplaces. It is easier than ever before to create content and applications for AR enabled devices, meaning that even non-programmers can create rich AR experiences. Adoption is only expected to increase with the availability of inexpensive wearable computers (e.g. Google Glass) and head mounted displays (e.g. Oculus Rift), and the growth in AR applications available.

Significant progress has been made in solving many of the fundamental problems in AR. For example, in section 4 we showed that there are a number of computer vision libraries that provide robust tracking from printed images, and there are sensor based systems that can provide relatively accurate outdoor tracking. A variety of AR display options were reviewed in section 5. Similarly, section 6 presented a wide range of authoring tools that can be used to create AR experience. This means that the research field is evolving away from being solely focused on low-level enabling technology, to exploring AR experiences, using interaction methods reviewed in section 7, and design guidelines contained in section 8. section 9 presented a variety of methods that can be used to evaluate the quality of user experience and showed that more and more AR papers with evaluation studies in them are being published. Finally, section 10 presented some case study examples of AR applications being used today.

Researchers in the field have made tremendous advances since the 1960's, but in many ways the full potential of Augmented Reality is still to be realized. Section 11 touched on several of the promising areas for future research such as social acceptance and hybrid AR interfaces, although there are many other directions people have begun to explore. The pace of research is dramatically increasing and the developments to date will seem trivial when we look back in another 50 years time. It is certain that AR will dramatically change how humans interact with digital content in the years to come.

Acknowledgements

We would like to thank all those who have kindly granted us to use figures from their original work.

References

Kyriacos Achilleos. Evolution of lifelogging, 2003. URL `http://mms.ecs.soton.ac.uk/2010/papers/Evolution_of_Lifelogging.pdf`.

Matt Adcock, Matthew Hutchins, and Chris Gunn. Augmented reality haptics: Using artoolkit for display of haptic applications. In *Proceedings of 2nd IEEE International Augmented Reality Toolkit Workshop*, pages 1–2, 2003.

Kusuma Agusanto, Li Li, Zhu Chuangui, and Ng Wan Sing. Photorealistic rendering for augmented reality using environment illumination. In *Mixed and Augmented Reality, 2003. Proceedings. The Second IEEE and ACM International Symposium on*, pages 208–216. IEEE, 2003.

Sangchul Ahn, Heedong Ko, and Steven Feiner. Webizing mobile ar contents. In *Proceedings of Virtual Reality 2013 (VR 2013)*, pages 131–132. IEEE, 2013.

Manu Akula, Suyang Dong, Vineet R Kamat, Lauro Ojeda, Adam Borrell, and Johann Borenstein. Integration of infrastructure based positioning systems and inertial navigation for ubiquitous context-aware engineering applications. *Advanced Engineering Informatics*, 25(4):640–655, 2011.

Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst. Freak: Fast retina keypoint. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 510–517. IEEE, 2012.

Christopher Alexander, Sara Ishikawa, and Murray Silverstein. *A Pattern Language*. Oxford Univ. Press, 1977.

Mahoro Anabuki, Hiroyuki Kakuta, Hiroyuki Yamamoto, and Hideyuki Tamura. Welbo: An embodied conversational agent living in mixed reality space. In *CHI'00 extended abstracts on Human factors in computing systems*, pages 10–11. ACM, 2000.

Anne H. Anderson, Alison Newlands, Jim Mullin, Anne Marie Fleming, Gwyneth Doherty-Sneddon, and Jeroen Van der Velden. Impact of video-mediated communication on simulated service encounters. *Interacting with computers*, 8(2):193–206, 1996.

Alissa N. Antle and Alyssa F. Wise. Getting down to details: Using theories of cognition and learning to inform tangible user interface design. *Interacting with Computers*, 25(1):1–20, 2013.

Clemens Arth, Daniel Wagner, Manfred Klopschitz, Arnold Irschara, and Dieter Schmalstieg. Wide area localization on mobile phones. In *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on*, pages 73–82. IEEE, 2009.

Mark Assad, David J. Carmichael, Daniel Cutting, and Adam Hudson. Ar phone: Accessible augmented reality in the intelligent environment. In *Proceedings of OZCHI*, pages 232–237, 2003.

Thomas Auer and Axel Pinz. Building a hybrid tracking system: Integration of optical and magnetic tracking. In *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*, pages 13–22. IEEE, 1999.

Ronald Azuma. Tracking requirements for augmented reality. *Communications of the ACM*, 36(7):50–51, 1993.

Ronald Azuma and Gary Bishop. Improving static and dynamic registration in an optical see-through hmd. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 197–204. ACM, 1994.

Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, and Blair MacIntyre. Recent advances in augmented reality. *Computer Graphics and Applications, IEEE*, 21(6):34–47, 2001.

Ronald T. Azuma. A survey of augmented reality. *Presence*, 6(4):355–385, 1997.

Ronald T. Azuma, Bruce R. Hoff, Howard E. Neely III, Ronald Sarfaty, Michael J Daily, Gary Bishop, Vern Chi, Greg Welch, Ulrich Neumann, Suya You, et al. Making augmented reality work outdoors requires hybrid tracking. In *Proceedings of the First International Workshop on Augmented Reality*, volume 1, 1998.

Gilles Bailly, Jörg Müller, Michael Rohs, Daniel Wigdor, and Sven Kratz. Shoesense: a new perspective on gestural interaction and wearable applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1239–1248. ACM, 2012.

Michael Bajura and Ulrich Neumann. Dynamic registration correction in augmented-reality systems. In *Virtual Reality Annual International Symposium, 1995. Proceedings.*, pages 189–196. IEEE, 1995.

Michael Bajura, Henry Fuchs, and Ryutarou Ohbuchi. Merging virtual objects with the real world: Seeing ultrasound imagery within the patient. *ACM SIGGRAPH Computer Graphics*, 26(2):203–210, 1992.

Martin Bauer, Bernd Bruegge, Gudrun Klinker, Asa MacWilliams, Thomas Reicher, Stefan Riss, Christian Sandor, and Martin Wagner. Design of a component-based augmented reality framework. In *Augmented Reality, 2001. Proceedings. IEEE and ACM International Symposium on*, pages 45–54. IEEE, 2001.

Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Computer vision–ECCV 2006*, pages 404–417. Springer, 2006.

Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3): 346–359, 2008.

Paul Beardsley, Jeroen Van Baar, Ramesh Raskar, and Clifton Forlines. Interaction using a handheld projector. *Computer Graphics and Applications, IEEE*, 25(1):39–43, 2005.

Amir H. Behzadan, Zeeshan Aziz, Chimay J. Anumba, and Vineet R. Kamat. Ubiquitous location tracking for context-specific information delivery on construction sites. *Automation in Construction*, 17(6):737–748, 2008.

Jeffrey S. Beis and David G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 1000–1006. IEEE, 1997.

Mathilde M. Bekker, Judith S. Olson, and Gary M. Olson. Analysis of gestures in face-to-face design teams provides guidance for how to use groupware in design. In *Proceedings of the 1st conference on Designing interactive systems: processes, practices, methods, & techniques*, pages 157–166. ACM, 1995.

Hrvoje Benko, Edward W. Ishak, and Steven Feiner. Cross-dimensional gestural interaction techniques for hybrid immersive environments. In *Proceedings of Virtual Reality 2005 (VR 2005)*, pages 209–216. IEEE, 2005.

Hrvoje Benko, Ricardo Jota, and Andrew Wilson. Miragetable: freehand interaction on a projected augmented reality tabletop. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 199–208. ACM, 2012.

Devesh Kumar Bhatnagar. Position trackers for head mounted display systems: A survey. Technical Report TR93-10, University of North Carolina, Chapel Hill, 1993.

Mark Billinghurst. Augmented reality in education. *New Horizons for Learning*, 12, 2002.

Mark Billinghurst and Andreas Dünser. Augmented reality in the classroom. *Computer*, 45(7):56–63, 2012. ISSN 0018-9162.

Mark Billinghurst and Hirokazu Kato. Out and about—real world teleconferencing. *BT technology journal*, 18(1):80–82, 2000.

Mark Billinghurst, Suzanne Weghorst, and Thomas Furness. Shared space: Collaborative augmented reality. In *Proceedings of CVE' 96 Workshop*, 1996.

Mark Billinghurst, Suzane Weghorst, and Tom Furness III. Wearable computers for three dimensional cscw. In *2012 16th International Symposium on Wearable Computers*, pages 39–39. IEEE Computer Society, 1997.

Mark Billinghurst, Jerry Bowskill, and Jason Morphett. Wearcom: A wearable communication space. In *Proceedings of CVE*, volume 98, 1998.

Mark Billinghurst, Ivan Poupyrev, Hirokazu Kato, and Richard May. Mixing realities in shared space: An augmented reality interface for collaborative computing. In *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, volume 3, pages 1641–1644. IEEE, 2000.

Mark Billinghurst, Hirokazu Kato, and Ivan Poupyrev. The magicbook: a transitional ar interface. *Computers & Graphics*, 25(5):745–753, 2001.

Mark Billinghurst, Raphael Grasset, and Julian Looser. Designing augmented reality interfaces. *ACM Siggraph Computer Graphics*, 39(1):17–22, 2005.

Oliver Bimber and Ramesh Raskar. Spatial augmented reality. In *International Conference on Computer Graphics and Interactive Techniques: ACM SIGGRAPH 2006 Courses*, volume 2006, 2006.

Oliver Bimber, L Miguel Encarnação, and Pedro Branco. The extended virtual table: An optical extension for table-like projection systems. *Presence: Teleoperators and Virtual Environments*, 10(6):613–631, 2001a.

Oliver Bimber, Bernd Frohlich, Dieter Schmalsteig, and L. Miguel Encarnacao. The virtual showcase. *Computer Graphics and Applications, IEEE*, 21(6):48–55, 2001b.

Gabor Blasko, Franz Coriand, and Steven Feiner. Exploring interaction with a simulated wrist-worn projection display. In *Wearable Computers, 2005. Proceedings. Ninth IEEE International Symposium on*, pages 2–9. IEEE, 2005.

Richard A. Bolt. "put-that-there": Voice and gesture at the graphics interface. *SIGGRAPH Computer Graphics*, 14(3):262–270, 1980.

Jan Borchers. *A Pattern Approach to Interaction Design.* Wiley, 2001.

Doug A Bowman, Ernst Kruijff, Joseph J. LaViola Jr, and Ivan Poupyrev. *3D user interfaces: theory and practice.* Addison-Wesley, 2004.

Elizabeth A. Boyle, Anne H. Anderson, and Alison Newlands. The effects of visibility on dialogue and performance in a cooperative problem solving task. *Language and speech*, 37(1):1–20, 1994.

José M. Braz and João M. Pereira. Tarcast: Taxonomy for augmented reality casting with web support. *The International Journal of Virtual Reality*, 7 (4):47–56, 2008.

David E. Breen, Ross T. Whitaker, Eric Rose, and Mihran Tuceryan. Interactive occlusion and automatic object placement for augmented reality. *Computer Graphics Forum*, 15(3):11–22, 1996.

Jeremy Brooker. The polytechnic ghost: Pepper's ghost, metempsychosis and the magic lantern at the royal polytechnic institution. *Early Popular Visual Culture*, 5(2):189–206, 2007.

Leonard D. Brown and Hong Hua. Magic lenses for augmented virtual environments. *Computer Graphics and Applications*, 26(4):64–73, 2006.

Amy Bruckman and Alisa Bandlow. Human-computer interaction for kids. In Julie A. Jacko and Andrew Sears, editors, *The Human-computer Interaction Handbook*, pages 428–440. L. Erlbaum Associates Inc., 2003.

Grigore Burdea and Philippe Coiffet. Virtual reality technology. *Presence: Teleoperators and virtual environments*, 12(6):663–664, 2003.

Andreas Butz, Tobias Hollerer, Steven Feiner, Blair MacIntyre, and Clifford Beshers. Enveloping users and computers in a collaborative 3d augmented reality. In *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*, pages 35–44. IEEE, 1999.

Ozan Cakmakci, Yonggang Ha, and Jannick P. Rolland. A compact optical see-through head-worn display with occlusion support. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 16–25. IEEE Computer Society, 2004.

Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *Computer Vision–ECCV 2010*, pages 778–792. Springer, 2010.

Christer Carlsson and Olof Hagsand. Dive—a platform for multi-user virtual environments. *Computers & graphics*, 17(6):663–669, 1993.

Julie Carmigniani, Borko Furht, Marco Anisetti, Paolo Ceravolo, Ernesto Damiani, and Misa Ivkovic. Augmented reality technologies, systems and applications. *Multimedia Tools and Applications*, 51(1):341–377, 2011.

Robert Castle, Georg Klein, and David W. Murray. Video-rate localization in multiple maps for wearable augmented reality. In *Wearable Computers, 2008. ISWC 2008. 12th IEEE International Symposium on*, pages 15–22. IEEE, 2008.

Thomas P. Caudell and David W. Mizell. Augmented reality: An application of heads-up display technology to manual manufacturing processes. In *System Sciences, 1992. Proceedings of the Twenty-Fifth Hawaii International Conference on*, volume 2, pages 659–669. IEEE, 1992.

Alphonse Chapanis. Interactive human communication. *Scientific American*, 232:36–42, 1975.

Dewen Cheng, Yongtian Wang, Hong Hua, and Jose Sasian. Design of a wide-angle, lightweight head-mounted display using free-form optics tiling. *Optics letters*, 36(11):2098–2100, 2011.

Youngkwan Cho, Jun Park, and Ulrich Neumann. Fast color fiducial detection and dynamic workspace extension in video see-through self-tracking augmented reality. In *Computer Graphics and Applications, 1997. Proceedings., The Fifth Pacific Conference on*, pages 168–177. IEEE, 1997.

Youngkwan Cho, Jongweon Lee, and Ulrich Neumann. A multi-ring color fiducial system and an intensity-invariant detection method for scalable fiducial-tracking augmented reality. In *Proceedings of International Workshop on Augmented Reality '98*, 1998.

Jinhyuk Choi, Youngsun Kim, Myonghee Lee, Gerard J. Kim, Yanghee Nam, and Yongmoo Kwon. k-mart: Authoring tool for mixed reality contents. In *Mixed and Augmented Reality (ISMAR), 2010 9th IEEE International Symposium on*, pages 219–220. IEEE, 2010.

Jan Ciger, Mario Gutierrez, Frederic Vexo, and Daniel Thalmann. The magic wand. In *Proceedings of the 19th spring conference on Computer graphics*, pages 119–124. ACM, 2003.

Adrian Clark and Andreas Dünser. An interactive augmented reality coloring book. In *3D User Interfaces (3DUI), 2012 IEEE Symposium on*, pages 7–10. IEEE, 2012.

Adrian Clark and Thammathip Piumsomboon. A realistic augmented reality racing game using a depth-sensing camera. In *Proceedings of the 10th International Conference on Virtual Reality Continuum and Its Applications in Industry*, pages 499–502. ACM, 2011.

Philip Cohen, David McGee, Sharon Oviatt, Lizhong Wu, Joshua Clow, Robert King, Simon Julier, and Lawrence Rosenblum. Multimodal interaction for 2d and 3d environments. *IEEE Computer Graphics and Applications*, 19(4):10–13, 1999.

Philip R. Cohen, Mary Dalrymple, Douglas B. Moran, F. C. Pereira, and Joseph W. Sullivan. Synergistic use of direct manipulation and natural language. *ACM SIGCHI Bulletin*, 20(SI):227–233, 1989.

Philip R. Cohen, Michael Johnston, David McGee, Sharon Oviatt, Jay Pittman, Ira Smith, Liang Chen, and Josh Clow. Quickset: Multimodal interaction for distributed applications. In *Proceedings of the fifth ACM international conference on Multimedia*, pages 31–40. ACM, 1997.

Andrew I. Comport, Éric Marchand, and François Chaumette. A real-time tracker for markerless augmented reality. In *Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality*, page 36. IEEE Computer Society, 2003.

Andrew I. Comport, Eric Marchand, Muriel Pressigout, and Francois Chaumette. Real-time markerless tracking for augmented reality: the virtual visual servoing framework. *Visualization and Computer Graphics, IEEE Transactions on*, 12(4):615–628, 2006.

Sam Corbett-Davies, Andreas Dunser, Richard Green, and Adrian Clark. An advanced interaction framework for augmented reality based exposure treatment. In *Virtual Reality (VR), 2013 IEEE*, pages 19–22. IEEE, 2013.

Maria Francesca Costabile and Maristella Matera. Evaluating wimp interfaces through the sue approach. In *Image Analysis and Processing, 1999. Proceedings. International Conference on*, pages 1192–1197. IEEE, 1999.

Enrico Costanza, Samuel A. Inverso, Elan Pavlov, Rebecca Allen, and Pattie Maes. eye-q: Eyeglass peripheral display for subtle intimate notifications. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, pages 211–218. ACM, 2006.

Owen Daly-Jones, Andrew Monk, and Leon Watts. Some advantages of video conferencing over high-quality audio conferencing: fluency and awareness of attentional focus. *International Journal of Human-Computer Studies*, 49(1):21–58, 1998.

Fred D. Davis, Richard P. Bagozzi, and Paul R. Warshaw. User acceptance of computer technology: a comparison of two theoretical models. *Management science*, 35(8):982–1003, 1989.

Andrew J. Davison, Ian D. Reid, Nicholas D. Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):1052–1067, 2007.

Jelle De Smet. *The Smart Contact Lens: From an Artificial Iris to a Contact Lens Display.* PhD thesis, Faculty of Engineering and Architecture, Ghent University, Ghent, Belgium, 2014.

Jelle De Smet, Aykut Avci, Pankaj Joshi, David Schaubroeck, Dieter Cuypers, and Herbert De Smet. Progress toward a liquid crystal contact lens display. *Journal of the Society for Information Display*, 21(9):399–406, 2013.

MWM Gamini Dissanayake, Paul Newman, Steve Clark, Hugh F. Durrant-Whyte, and Michael Csorba. A solution to the simultaneous localization and map building (slam) problem. *Robotics and Automation, IEEE Transactions on*, 17(3):229–241, 2001.

Neil A. Dodgson. Autostereoscopic 3d displays. *Computer*, 38(8):31–36, 2005.

Klaus Dorfmüller. Robust tracking for augmented reality using retroreflective markers. *Computers & Graphics*, 23(6):795–800, 1999.

Ralf Dörner, Christian Geiger, Michael Haller, and Volker Paelke. Authoring mixed reality—a component and framework-based approach. In *Entertainment Computing*, pages 405–413. Springer, 2003.

David Drascic and Paul Milgram. Positioning accuracy of a virtual stereographic pointer in a real stereoscopic video world. In *Electronic Imaging'91, San Jose, CA*, pages 302–313. International Society for Optics and Photonics, 1991.

David Drascic and Paul Milgram. Perceptual issues in augmented reality. In *Electronic Imaging: Science & Technology*, pages 123–134. International Society for Optics and Photonics, 1996.

David Drascic, Julius J Grodski, Paul Milgram, Ken Ruffo, Peter Wong, and Shumin Zhai. Argos: A display system for augmenting reality. In *Proceedings of the INTERACT'93 and CHI'93 Conference on Human Factors in Computing Systems*, page 521. ACM, 1993.

Andreas Dünser. Supporting low ability readers with interactive augmented reality. *Annual Review of CyberTherapy and Telemedicine*, page 39, 2008.

Andreas Dünser and Eva Hornecker. Lessons from an ar book study. In *Proceedings of the 1st international conference on Tangible and embedded interaction*, pages 179–182. ACM, 2007.

Andreas Dünser, Raphaël Grasset, and Mark Billinghurst. A survey of evaluation techniques used in augmented reality studies. Technical Report TR2008-02, The Human Interface Technology Laboratory New Zealand, University of Canterbury, 2008.

Paul Ekman and Wallace V. Friesen. The repertoire of nonverbal behavior: Categories, origins, usage, and coding. In Adam Kendon, Thomas A. Sebeok, and Jean Umiker-Sebeok, editors, *Nonverbal communication, interaction, and gesture*, pages 57–106. Mouton Publishers, 1981.

Stephen R. Ellis and Brian M. Menges. Judgments of the distance to nearby virtual objects: interaction of viewing conditions and accommodative demand. *Presence (Cambridge, Mass.)*, 6(4):452–460, 1997.

Stephen R. Ellis and Brian M. Menges. *Studies of the localization of virtual objects in the near visual field*. Lawrence Erlbaum Associates, 2001.

Frédéric Evennou and François Marx. Advanced integration of wifi and inertial navigation systems for indoor mobile positioning. *Eurasip journal on applied signal processing*, 2006:164–164, 2006.

J. Fagard. The development of bimanual coordination, 1990.

Gregg E. Favalora. Volumetric 3d displays and application infrastructure. *IEEE Computer Society*, 8:37–44, 2005.

Steven Feiner, Blair MacIntyre, Marcus Haupt, and Eliot Solomon. Windows on the world: 2d windows for 3d augmented reality. In *Proceedings of the 6th annual ACM symposium on User interface software and technology*, pages 145–155. ACM, 1993a.

Steven Feiner, Blair Macintyre, and Dorée Seligmann. Knowledge-based augmented reality. *Communications of the ACM*, 36(7):53–62, 1993b.

Steven Feiner, Blair MacIntyre, Tobias Höllerer, and Anthony Webster. A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. *Personal Technologies*, 1(4):208–217, 1997.

James L. Fergason. Optical system for a head mounted display using a retroreflector and method of displaying an image, April 15 1997. US Patent 5,621,572.

Mark Fiala. Artag revision 1, a fiducial marker system using digital techniques. Technical report, National Research Council Canada, 2004.

Mark Fiala. Artag, a fiducial marker system using digital techniques. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 590–596. IEEE, 2005a.

Mark Fiala. Comparing artag and artoolkit plus fiducial marker systems. In *Haptic Audio Visual Environments and their Applications, 2005. IEEE International Workshop on*, pages 6–pp. IEEE, 2005b.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

Ralph W. Fisher. Head-mounted projection display system featuring beam splitter and method of making same, November 5 1996. US Patent 5,572,229.

Scott S. Fisher, Micheal McGreevy, James Humphries, and Warren Robinett. Virtual environment display system. In *Proceedings of the 1986 workshop on Interactive 3D graphics*, pages 77–87. ACM, 1987.

Paul M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6):381, 1954.

Paul M. Fitts and James R. Peterson. Information capacity of discrete motor responses. *Journal of experimental psychology*, 67(2):103, 1964.

George W. Fitzmaurice. Situated information spaces and spatially aware palmtop computers. *Communications of the ACM*, 36(7):39–49, 1993.

George W. Fitzmaurice and William Buxton. An empirical evaluation of graspable user interfaces: towards specialized, space-multiplexed input. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, pages 43–50. ACM, 1997.

Morten Fjeld and Benedikt M. Voegtli. Augmented chemistry: An interactive educational workbench. In *Mixed and Augmented Reality, 2002. ISMAR 2002. Proceedings. International Symposium on*, pages 259–321. IEEE, 2002.

Eric Foxlin. Inertial head-tracker sensor fusion by a complementary separate-bias kalman filter. In *Virtual Reality Annual International Symposium, 1996., Proceedings of the IEEE 1996*, pages 185–194. IEEE, 1996.

Eric Foxlin, Michael Harrington, and George Pfeifer. Constellation: a wide-range wireless motion-tracking system for augmented reality and virtual set applications. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 371–378. ACM, 1998.

Eric Foxlin, Yury Altshuler, Leonid Naimark, and Mike Harrington. Flight-tracker: a novel optical/inertial tracker for cockpit enhanced vision. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 212–221. IEEE Computer Society, 2004.

Eric M. Foxlin. Head tracking relative to a moving vehicle or simulator platform using differential inertial sensors. In *AeroSense 2000*, pages 133–144. International Society for Optics and Photonics, 2000.

Henry Fuchs, Andrei State, Etta D. Pisano, William F. Garrett, Gentaro Hirota, Mark Livingston, Mary C. Whitton, and Stephen M. Pizer. Towards performing ultrasound-guided needle biopsies from within a head-mounted display. In *Visualization in Biomedical Computing*, pages 591–600. Springer, 1996.

Henry Fuchs, Mark A. Livingston, Ramesh Raskar, Kurtis Keller, Jessica R. Crawford, Paul Rademacher, Samuel H. Drake, Anthony A. Meyer, et al. *Augmented reality visualization for laparoscopic surgery.* Springer, 1998.

Anton Fuhrmann, Dieter Schmalstieg, and Werner Purgathofer. Fast calibration for augmented reality. In *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 166–167. ACM, 1999.

Thomas A. Furness III. The application of head-mounted displays to airborne reconnaissance and weapon delivery. Technical Report TR-69-241, Wright-Patterson Air Force Bose, OH: U.S. Air Force Avionics Laboratory, 1969.

Thomas A. Furness III. The super cockpit and its human factors challenges. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 30(1):48–52, 1986.

Susan R. Fussell, Robert E. Kraut, and Jane Siegel. Coordination of communication: Effects of shared visual context on collaborative work. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, pages 21–30. ACM, 2000.

Joseph L. Gabbard and J. Edward Swan. Usability engineering for augmented reality: Employing user-based studies to inform design. *Visualization and Computer Graphics, IEEE Transactions on*, 14(3):513–525, 2008.

Joseph L. Gabbard, Deborah Hix, and J. Edward Swan. User-centered design and evaluation of virtual environments. *Computer Graphics and Applications, IEEE*, 19(6):51–59, 1999.

Chunyu Gao, Yuxiang Lin, and Hong Hua. Occlusion capable optical see-through head-mounted display using freeform optics. In *Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on*, pages 281–282. IEEE, 2012.

Christian Geiger, Bernd Kleinnjohann, Christian Reimann, and Dirk Stichling. Mobile ar4all. In *Augmented Reality, 2001. Proceedings. IEEE and ACM International Symposium on*, pages 181–182. IEEE, 2001.

David Gelernter. *Mirror worlds, or, The day software puts the universe in a shoebox–: how it will happen and what it will mean.* Oxford University Press New York, 1991.

James J. Gibson. The theory of affordances. In Jen Jack Gieseking, William Mangold, Cindi Katz, Setha Low, and Susan Saegert, editors, *The People, Place, and Space Reader*, chapter 9, pages 56–60. Routledge, 1979.

S. Burak Gokturk, Hakan Yalcin, and Cyrus Bamji. A time-of-flight depth sensor-system description, issues and solutions. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*, pages 35–35. IEEE, 2004.

Daniel Goldsmith, Fotis Liarokapis, Garry Malone, and John Kemp. Augmented reality environmental monitoring using wireless sensor networks. In *Information Visualisation, 2008. IV'08. 12th International Conference*, pages 539–544. IEEE, 2008.

Stefan Gottschalk and John F. Hughes. Autocalibration for virtual environments tracking hardware. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 65–72. ACM, 1993.

Michael Grabner, Helmut Grabner, and Horst Bischof. Fast approximated sift. In *Computer Vision–ACCV 2006*, pages 918–927. Springer, 2006.

Raphael Grasset, Alessandro Mulloni, Mark Billinghurst, and Dieter Schmalstieg. Navigation techniques in augmented and mixed reality: Crossing the virtuality continuum. In *Handbook of Augmented Reality*, pages 379–407. Springer, 2011.

Jens Grubert, Tobias Langlotz, and Raphaël Grasset. Augmented reality browser survey. Technical Report 1101, Institute for Computer Graphics and Vision, University of Technology Graz, Technical Report, 2011.

Chris Gunn, Matthew Hutchins, Matt Adcock, and Rhys Hawkins. Trans-world haptic collaboration. In *ACM SIGGRAPH 2003 Sketches & Applications*, pages 1–1. ACM, 2003.

Taejin Ha, Woontack Woo, Youngho Lee, Junhun Lee, Jeha Ryu, Hankyun Choi, and Kwanheng Lee. Artalet: tangible user interface based immersive augmented reality authoring tool for digilog book. In *Ubiquitous Virtual Reality (ISUVR), 2010 International Symposium on*, pages 40–43. IEEE, 2010.

Taejin Ha, Mark Billinghurst, and Woontack Woo. An interactive 3d movement path manipulation method in an augmented reality environment. *Interacting with Computers*, 24(1):10–24, 2012.

Michael Haller, Erwin Stauder, and Juergen Zauner. Amire-es: Authoring mixed reality once, run it anywhere. In *Proceedings of the 11th International Conference on Human-Computer Interaction (HCII)*, volume 2005, 2005.

Chris Harrison, Hrvoje Benko, and Andrew D. Wilson. Omnitouch: wearable multitouch interaction everywhere. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 441–450. ACM, 2011.

Alexander G. Hauptmann. Speech and gestures for graphic image manipulation. *ACM SIGCHI Bulletin*, 20(SI):241–245, 1989.

Christian Heath and Paul Luff. Disembodied conduct: communication through video in a multi-media office environment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 99–103. ACM, 1991.

Gunther Heidemann, Ingo Bax, and Holger Bekel. Multimodal interaction in an augmented reality scenario. In *Proceedings of the 6th international conference on Multimodal interfaces*, pages 53–60. ACM, 2004.

Steven J. Henderson and Steven Feiner. Evaluating the benefits of augmented reality for task localization in maintenance of an armored personnel carrier turret. In *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on*, pages 135–144. IEEE, 2009.

Peter Henry, Michael Krainin, Evan Herbst, Xiaofeng Ren, and Dieter Fox. Rgb-d mapping: Using depth cameras for dense 3d modeling of indoor environments. In *In the 12th International Symposium on Experimental Robotics (ISER*, 2010.

Anders Henrysson and Mark Ollila. Umar: Ubiquitous mobile augmented reality. In *Proceedings of the 3rd international conference on Mobile and ubiquitous multimedia*, pages 41–45. ACM, 2004.

Anders Henrysson, Mark Ollila, and Mark Billinghurst. Mobile phone based ar scene assembly. In *Proceedings of the 4th international conference on Mobile and ubiquitous multimedia*, pages 95–102. ACM, 2005.

Otmar Hilliges, David Kim, Shahram Izadi, Malte Weiss, and Andrew Wilson. Holodesk: direct 3d interactions with a situated see-through display. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2421–2430. ACM, 2012.

Deborah Hix, Joseph L. Gabbard, J. Edward Swan, Mark A. Livingston, T. H. Hollerer, Simon J. Julier, Yohan Baillot, Dennis Brown, et al. A cost-effective usability evaluation progression for novel interactive systems. In *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, pages 10–pp. IEEE, 2004.

Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin Smyth, Narinder Kapur, and Ken Wood. Sensecam: A retrospective memory aid. In *UbiComp 2006: Ubiquitous Computing*, pages 177–193. Springer, 2006.

H. G. Hoffmann. Physically touching virtual objects using tactile augmentation enhances the realism of virtual environments. In *Virtual Reality Annual International Symposium, 1998. Proceedings., IEEE 1998*, pages 59–63. IEEE, 1998.

Tobias Höllerer, Steven Feiner, Tachio Terauchi, Gus Rashid, and Drexel Hallaway. Exploring mars: developing indoor and outdoor user interfaces to a mobile augmented reality system. *Computers & Graphics*, 23(6):779–785, 1999.

Tobias Höllerer, Jason Wither, and Stephen DiVerdi. "anywhere augmentation": Towards mobile augmented reality in unprepared environments. In *Location Based Services and TeleCartography*, pages 393–416. Springer, 2007.

Jason Hong. Considering privacy issues in the context of google glass. *Commun. ACM*, 56(11):10–11, 2013.

Juan Pablo Hourcade. Interaction design and children. *Foundations and Trends in Human-Computer Interaction*, 1(4):277–392, 2008.

Xinda Hu and Hong Hua. Design of an optical see-through multi-focal-plane stereoscopic 3d display using freeform prisms. In *Frontiers in Optics*, pages FTh1F–2. Optical Society of America, 2012.

Hong Hua, Leonard D Brown, Chunyu Gao, and Narendra Ahuja. A new collaborative infrastructure: Scape. In *Virtual Reality, 2003. Proceedings. IEEE*, pages 171–179. IEEE, 2003.

Yetao Huang, Dongdong Weng, Yue Liu, and Yongtian Wang. Key issues of wide-area tracking system for multi-user augmented reality adventure game. In *Image and Graphics, 2009. ICIG'09. Fifth International Conference on*, pages 646–651. IEEE, 2009.

Manuel Huber, Daniel Pustka, Peter Keitler, Florian Echtler, and Gudrun Klinker. A system architecture for ubiquitous tracking environments. In *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 1–4. IEEE Computer Society, 2007.

Olivier Hugues, Philippe Fuchs, and Olivier Nannipieri. New augmented reality taxonomy: Technologies and features of augmented environment. In *Handbook of augmented reality*, pages 47–63. Springer, 2011.

Masahiko Inami, Naoki Kawakami, Dairoku Sekiguchi, Yasuyuki Yanagida, Taro Maeda, and Susumu Tachi. Visuo-haptic display using head-mounted projector. In *Virtual Reality, 2000. Proceedings. IEEE*, pages 233–240. IEEE, 2000.

Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 604–613. ACM, 1998.

Sylvia Irawati, Scott Green, Mark Billinghurst, Andreas Duenser, and Heedong Ko. An evaluation of an augmented reality multimodal interface using speech and paddle gestures. In *Advances in Artificial Reality and Tele-Existence*, pages 272–283. Springer, 2006.

Hiroshi Ishii and Brygg Ullmer. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, pages 234–241. ACM, 1997.

Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568. ACM, 2011.

Seokhee Jeon and Seungmoon Choi. Haptic augmented reality: Taxonomy and an example of stiffness modulation. *Presence: Teleoperators and Virtual Environments*, 18(5):387–408, 2009.

Simon Julier, Marco Lanzagorta, Yohan Baillot, Lawrence Rosenblum, Steven Feiner, Tobias Hollerer, and Sabrina Sestito. Information filtering for mobile augmented reality. In *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*, pages 3–11. IEEE, 2000.

Ed Kaiser, Alex Olwal, David McGee, Hrvoje Benko, Andrea Corradini, Xiaoguang Li, Phil Cohen, and Steven Feiner. Mutual disambiguation of 3d multimodal interaction in augmented and virtual reality. In *Proceedings of the 5th international conference on Multimodal interfaces*, pages 12–19. ACM, 2003.

Roy Kalawsky. *The science of virtual reality and virtual environments.* Addison-Wesley Longman Publishing Co., Inc., 1993.

Peter Kán and Hannes Kaufmann. Differential irradiance caching for fast high-quality light transport between virtual and real worlds. In *Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on*, pages 133–141. IEEE, 2013.

Anantha Kancherla, Mark Singer, and Jannick Rolland. Calibrating see-through head-mounted displays. Technical Report TR95-034, Department of Computer Science, University of North Carolina at Chapel Hill, 1996.

Kenji Kansaku, Naoki Hata, and Kouji Takano. My thoughts through a robot's eyes: An augmented reality-brain–machine interface. *Neuroscience research*, 66(2):219–222, 2010.

Hirokazu Kato and Mark Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*, pages 85–94. IEEE, 1999.

Hirokazu Kato, Mark Billinghurst, Ivan Poupyrev, Kenji Imamoto, and Keihachiro Tachibana. Virtual object manipulation on a table-top ar environment. In *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*, pages 111–119. Ieee, 2000.

Yan Ke and Rahul Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–506. IEEE, 2004.

Lucinda Kerawalla, Rosemary Luckin, Simon Seljeflot, and Adrian Woolard. "making it real": exploring the potential of augmented reality for teaching primary school science. *Virtual Reality*, 10(3-4):163–174, 2006.

Ryugo Kijima and Takeo Ojika. Transition between virtual environment and workstation environment with projective head mounted display. In *Virtual Reality Annual International Symposium, 1997., IEEE 1997*, pages 130–137. IEEE, 1997.

Hyemi Kim, Gun A. Lee, Ungyeon Yang, Taejin Kwak, and Ki-Hong Kim. Dual autostereoscopic display platform for multi-user collaboration with natural interaction. *Etri Journal*, 34(3):466–469, 2012.

Sehwan Kim, Youngjung Suh, Youngho Lee, and Woontack Woo. Toward ubiquitous vr: When vr meets ubicomp. *Proc of ISUVR*, pages 1–4, 2006.

Kiyoshi Kiyokawa, Haruo Takemura, and Naokazu Yokoya. A collaboration support technique by integrating a shared virtual reality and a shared augmented reality. In *Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on*, volume 6, pages 48–53. IEEE, 1999.

Kiyoshi Kiyokawa, Haruo Takemura, and Naokazu Yokoya. Seamlessdesign for 3d object creation. *IEEE multimedia*, 7(1):22–33, 2000.

Kiyoshi Kiyokawa, Yoshinori Kurata, and Hiroyuki Ohno. An optical see-through display for mutual occlusion with a real-time stereovision system. *Computers & Graphics*, 25(5):765–779, 2001.

Georg Klein and Tom Drummond. Sensor fusion and occlusion refinement for tablet-based ar. In *Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on*, pages 38–47. IEEE, 2004.

Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pages 225–234. IEEE, 2007.

Georg Klein and David W. Murray. Simulating low-cost cameras for augmented reality compositing. *Visualization and Computer Graphics, IEEE Transactions on*, 16(3):369–380, 2010.

Mathias Kolsch, Ryan Bane, Tobias Hollerer, and Matthew Turk. Multimodal interaction with a wearable augmented reality system. *Computer Graphics and Applications, IEEE*, 26(3):62–71, 2006.

Robert E. Kraut, Mark D. Miller, and Jane Siegel. Collaboration in performance of physical tasks: Effects on outcomes and communication. In *Proceedings of the 1996 ACM conference on Computer supported cooperative work*, pages 57–66. ACM, 1996.

David M. Krum, Evan A. Suma, and Mark Bolas. Augmented reality using personal projection and retroreflection. *Personal and Ubiquitous Computing*, 16(1):17–26, 2012.

Oliver Kutter, André Aichert, Christoph Bichlmeier, Jörg Traub, SM Heining, B Ockert, Ekkehard Euler, and Nassir Navab. Real-time volume rendering for high quality visualization in augmented reality. In *International Workshop on Augmented environments for Medical Imaging including Augmented Reality in Computer-aided Surgery (AMI-ARCS 2008), New York, USA*, 2008.

Hideaki Kuzuoka. Spatial workspace collaboration: a sharedview video support system for remote collaboration capability. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 533–540. ACM, 1992.

Peter Lang, Albert Kusej, Axel Pinz, and Georg Brasseur. Inertial tracking for mobile augmented reality. In *Instrumentation and Measurement Technology Conference, 2002. IMTC/2002. Proceedings of the 19th IEEE*, volume 2, pages 1583–1587. IEEE, 2002.

Joseph J. LaViola Jr. Whole-hand and speech input in virtual environments. Master's thesis, Brown University, 1999.

Gun Lee, Gerard Kim, and Mark Billinghurst. Interaction design for tangible augmented reality applications. *Emerging Technologies of Augmented Reality: Interfaces and Design*, pages 261–282, 2007.

Gun A. Lee and Mark Billinghurst. A component based framework for mobile outdoor ar applications. In *Proceedings of the 12th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, pages 207–210. ACM, 2013.

Gun A. Lee, Gerard Jounghyun Kim, and Chan-Mo Park. Modeling virtual object behavior within virtual environment. In *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 41–48. ACM, 2002.

Gun A. Lee, Mark Billinghurst, and Gerard Jounghyun Kim. Occlusion based interaction methods for tangible augmented reality environments. In *Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, pages 419–426. ACM, 2004a.

Gun A. Lee, Claudia Nelles, Mark Billinghurst, and Gerard Jounghyun Kim. Immersive authoring of tangible augmented reality applications. In *Proceedings of the 3rd IEEE/ACM international Symposium on Mixed and Augmented Reality*, pages 172–181. IEEE Computer Society, 2004b.

Gun A. Lee, Gerard J. Kim, and Mark Billinghurst. Immersive authoring: What you experience is what you get (wyxiwyg). *Communications of the ACM*, 48(7):76–81, 2005.

Gun A. Lee, Ungyeon Yang, and Wookho Son. Layered multiple displays for immersive and interactive digital contents. In *Entertainment Computing-ICEC 2006*, pages 123–134. Springer, 2006.

Gun A. Lee, Hyun Kang, and Wookho Son. Mirage: A touch screen based mixed reality interface for space planning applications. In *Virtual Reality Conference, 2008. VR'08. IEEE*, pages 273–274. IEEE, 2008a.

Gun A. Lee, Ungyeon Yang, Wookho Son, Yongwan Kim, Dongsik Jo, Ki-Hong Kim, and Jin Sung Choi. Virtual reality content-based training for spray painting tasks in the shipbuilding industry. *ETRI Journal*, 32(5): 695–703, 2010a.

Gun A. Lee, Andreas Dunser, Seungwon Kim, and Mark Billinghurst. Cityviewar: A mobile outdoor ar application for city visualization. In *Mixed and Augmented Reality (ISMAR-AMH), 2012 IEEE International Symposium on*, pages 57–64. IEEE, 2012.

Gun A. Lee, Andreas Dünser, Alaeddin Nassani, and Mark Billinghurst. Antarcticar: An outdoor ar experience of a virtual tour to antarctica. In *Mixed and Augmented Reality-Arts, Media, and Humanities (ISMAR-AMH), 2013 IEEE International Symposium on*, pages 29–38. IEEE, 2013a.

Jae Yeol Lee, Gue Won Rhee, and Dong Woo Seo. Hand gesture-based tangible interactions for manipulating virtual objects in a mixed reality environment. *The International Journal of Advanced Manufacturing Technology*, 51(9-12):1069–1082, 2010b.

Minkyung Lee, Richard Green, and Mark Billinghurst. 3d natural hand interaction for ar applications. In *Image and Vision Computing New Zealand, 2008. IVCNZ 2008. 23rd International Conference*, pages 1–6. IEEE, 2008b.

Minkyung Lee, Mark Billinghurst, Woonhyuk Baek, Richard Green, and Woontack Woo. A usability study of multimodal input in an augmented reality environment. *Virtual Reality*, 17(4):293–305, 2013b.

Taehee Lee and Tobias Hollerer. Handy ar: Markerless inspection of augmented reality objects using fingertip tracking. In *Wearable Computers, 2007 11th IEEE International Symposium on*, pages 83–90. IEEE, 2007.

Alexander Lenhardt and Helge Ritter. An augmented-reality based brain-computer interface for robot control. In *Neural Information Processing. Models and Applications*, pages 58–65. Springer, 2010.

Vincent Lepetit and Pascal Fua. Keypoint recognition using randomized trees. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(9): 1465–1479, 2006.

Stefan Leutenegger, Margarita Chli, and Roland Yves Siegwart. Brisk: Binary robust invariant scalable keypoints. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2548–2555. IEEE, 2011.

Michael Emmanuel Leventon. *A registration, tracking, and visualization system for image-guided surgery*. PhD thesis, Massachusetts Institute of Technology, 1997.

Robert W. Lindeman, John L Sibert, and James K. Hahn. Towards usable vr: an empirical study of user interfaces for immersive virtual environments. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 64–71. ACM, 1999.

Robert W. Lindeman, Haruo Noma, and Paulo Goncalves de Barros. Hear-through and mic-through augmented reality: Using bone conduction to display spatialized audio. In *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 1–4. IEEE Computer Society, 2007.

Robert W. Lindeman, Gun Lee, Leigh Beattie, Hannes Gamper, Rahul Pathinarupothi, and Aswin Akhilesh. Geoboids: A mobile ar application for exergaming. In *Mixed and Augmented Reality (ISMAR-AMH), 2012 IEEE International Symposium on*, pages 93–94. IEEE, 2012.

Andrew R. Lingley, Muhammad Ali, Y. Liao, R. Mirjalili, M. Klonner, M. Sopanen, S. Suihkonen, T. Shen, B. P. Otis, H. Lipsanen, et al. A single-pixel wireless contact lens display. *Journal of Micromechanics and Microengineering*, 21(12):125014, 2011.

Sheng Liu, Dewen Cheng, and Hong Hua. An optical see-through head mounted display with addressable focal planes. In *Mixed and Augmented Reality, 2008. ISMAR 2008. 7th IEEE/ACM International Symposium on*, pages 33–42. IEEE, 2008.

David G. Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.

David G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

Blair MacIntyre. Authoring 3d mixed reality experiences: Managing the relationship between the physical and virtual worlds. *At ACM SIGGRAPH and Eurographics Campfire: Production Process of 3D Computer Graphics Applications-Structures, Roles and Tools, Snowbird, UT*, pages 1–5, 2002.

Blair MacIntyre, Maribeth Gandy, Steven Dow, and Jay David Bolter. Dart: a toolkit for rapid design exploration of augmented reality experiences. In *Proceedings of the 17th annual ACM symposium on User interface software and technology*, pages 197–206. ACM, 2004.

Asa MacWilliams, Thomas Reicher, Gudrun Klinker, and Bernd Bruegge. Design patterns for augmented reality systems. In *MIXER*, 2004.

Andrew Maimone and Henry Fuchs. Computational augmented reality eyeglasses. In *Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on*, pages 29–38. IEEE, 2013.

Elmar Mair, Gregory D. Hager, Darius Burschka, Michael Suppa, and Gerhard Hirzinger. Adaptive and generic corner detection based on the accelerated segment test. In *Computer Vision–ECCV 2010*, pages 183–196. Springer, 2010.

J. Mandeville, J. Davidson, D. Campbell, A. Dahl, P. Schwartz, and T. Furness. A shared virtual environment for architectural design review. In *CVE '96 Workshop Proceedings*, pages 19–20, 1996.

Steve Mann. Mediated reality. Technical Report MIT-ML Percom TR-260, University of Toronto, 1994.

Steve Mann. Mediated reality with implementations for everyday life. *Presence Connect, August*, 6, 2002.

Héctor Martínez, Danai Skournetou, Jenni Hyppölä, Seppo Laukkanen, and Antti Heikkilä. Drivers and bottlenecks in the adoption of augmented reality applications. *Journal of Multimedia Theory and Application*, 2(1), 2014.

Andrea H. Mason, Masuma A. Walji, Elaine J. Lee, and Christine L. MacKenzie. Reaching movements to augmented and graphic objects in virtual environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 426–433. ACM, 2001.

John C. McCarthy and Andrew F. Monk. Measuring the quality of computer-mediated communication. *Behaviour & Information Technology*, 13(5):311–319, 1994.

Chris McDonald and Gerhard Roth. Replacing a mouse with hand gesture in a plane-based augmented reality system. Technical report, National Research Council Canada, 2003.

Erin McGarrity and Mihran Tuceryan. A method for calibrating see-through head-mounted displays for ar. In *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*, pages 75–84. IEEE, 1999.

David McNeill. *Hand and mind: What gestures reveal about thought.* University of Chicago Press, 1992.

Paul Milgram and Fumio Kishino. A taxonomy of mixed reality visual displays. *IEICE TRANSACTIONS on Information and Systems*, 77(12):1321–1329, 1994.

Paul Milgram, Shumin Zhai, David Drascic, and Julius J. Grodski. Applications of augmented reality for human-robot communication. In *Intelligent Robots and Systems' 93, IROS'93. Proceedings of the 1993 IEEE/RSJ International Conference on*, volume 3, pages 1467–1472. IEEE, 1993.

Pranav Mistry and Pattie Maes. Sixthsense: a wearable gestural interface. In *ACM SIGGRAPH ASIA 2009 Sketches*, page 11. ACM, 2009.

Antonija Mitrovic. Fifteen years of constraint-based tutors: what we have achieved and where we are going. *User Modeling and User-Adapted Interaction*, 22(1-2):39–72, 2012.

Antonija Mitrovic, Brent Martin, Pramuditha Suraweera, Konstantin Zakharov, Nancy Milik, Jay Holland, and Nicholas McGuigan. Aspire: an authoring system and deployment environment for constraint-based tutors. *International Journal of Artificial Intelligence in Education*, 19(2):155–188, 2009.

D. Mogilev, Kiyoshi Kiyokawa, Mark Billinghurst, and J. Pair. Ar pad: An interface for face-to-face ar collaboration. In *CHI'02 Extended Abstracts on Human Factors in Computing Systems*, pages 654–655. ACM, 2002.

Mathias Mohring, Christian Lessig, and Oliver Bimber. Video see-through ar on consumer cell-phones. In *Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 252–253. IEEE Computer Society, 2004.

Takuji Narumi, Shinya Nishizaka, Takashi Kajinami, Tomohiro Tanikawa, and Michitaka Hirose. Meta cookie+: an illusion-based gustatory display. In *Virtual and Mixed Reality-New Trends*, pages 260–269. Springer, 2011.

Wolfgang Narzt, Gustav Pomberger, Alois Ferscha, Dieter Kolb, Reiner Müller, Jan Wieghardt, Horst Hörtner, and Christopher Lindinger. Augmented reality navigation systems. *Universal Access in the Information Society*, 4(3):177–187, 2006.

Nassir Navab, A Bani-Kashemi, and Matthias Mitschke. Merging visible and invisible: Two camera-augmented mobile c-arm (camc) applications. In *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*, pages 134–141. IEEE, 1999.

Nassir Navab, Marco Feuerstein, and Christoph Bichlmeier. Laparoscopic virtual mirror new interaction paradigm for monitor based augmented reality. In *Virtual Reality Conference, 2007. VR'07. IEEE*, pages 43–50. IEEE, 2007.

Ulrich Neumann and Youngkwan Cho. A self-tracking augmented reality system. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 109–115, 1996.

Ulrich Neumann and Jun Park. Extendible object-centric tracking for augmented reality. In *Virtual Reality Annual International Symposium, 1998. Proceedings., IEEE 1998*, pages 148–155. IEEE, 1998.

Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*, pages 127–136. IEEE, 2011.

Joseph Newman, Martin Wagner, Martin Bauer, Asa MacWilliams, Thomas Pintaric, Dagmar Beyer, Daniel Pustka, Franz Strasser, Dieter Schmalstieg, and Gudrun Klinker. Ubiquitous tracking for augmented reality. In *Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on*, pages 192–201. IEEE, 2004.

Joseph Newman, Gerhard Schall, István Barakonyi, Andreas Schürzinger, and Dieter Schmalstieg. *Wide-area tracking tools for augmented reality.* na, 2006.

Trond Nilsen and Julian Looser. Tankwar-tabletop war gaming in augmented reality. In *2nd International Workshop on Pervasive Gaming Applications, PerGames*, volume 5, 2005.

Susanna Nilsson and Björn Johansson. Acceptance of augmented reality instructions in a real work setting. In *CHI'08 extended abstracts on Human factors in computing systems*, pages 2025–2032. ACM, 2008.

Mark A. Nixon, Bruce C. McCallum, W. Richard Fright, and N. Brent Price. The effects of metals and interfering fields on electromagnetic trackers. *Presence: Teleoperators and Virtual Environments*, 7(2):204–218, 1998.

Donald A. Norman. *The psychology of everyday things.* Basic books, 1988.

Jean-Marie Normand, Myriam Servières, and Guillaume Moreau. A new typology of augmented reality applications. In *Proceedings of the 3rd Augmented Human International Conference*, page 18. ACM, 2012.

Timothy Nyerges, T. J. Moore, Robb Montejano, and Marcie Compton. Developing and using interaction coding systems for studying groupware use. *Human-Computer Interaction*, 13(2):127–165, 1998.

Christoph Oberhofer, Jens Grubert, and Gerhard Reitmayr. Natural feature tracking in javascript. In *VR*, pages 113–114, 2012.

Brid O'Conaill and Steve Whittaker. Characterizing, predicting, and measuring video-mediated communication: A conversational approach. *Video-mediated communication*, pages 107–132, 1997.

Ohan Oda, Levi J. Lister, Sean White, and Steven Feiner. Developing an augmented reality racing game. In *Proceedings of the 2nd international conference on INtelligent TEchnologies for interactive enterTAINment*, page 2. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

Yoosoo Oh and Woontack Woo. A unified application service model for ubihome by exploiting intelligent context-awareness. In *Ubiquitous Computing Systems*, pages 192–202. Springer, 2005.

Yoosoo Oh, Choonsung Shin, Woojin Jung, and Woontack Woo. The ubitv application for a family in ubihome. In *2nd Ubiquitous Home workshop*, pages 23–32, 2005.

Toshikazu Ohshima, Kiyohide Satoh, Hiroyuki Yamamoto, and Hideyuki Tamura. Ar 2 hockey: A case study of collaborative augmented reality. In *Virtual Reality Annual International Symposium, 1998. Proceedings., IEEE 1998*, pages 268–275. IEEE, 1998.

Thomas Olsson, Else Lagerstam, Tuula Kärkkäinen, and Kaisa Väänänen-Vainio-Mattila. Expected user experience of mobile augmented reality services: a user study in the context of shopping centres. *Personal and ubiquitous computing*, 17(2):287–304, 2013.

Alex Olwal, Hrvoje Benko, and Steven Feiner. Senseshapes: Using statistical geometry for object selection in a multimodal augmented reality system. In *Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality*, page 300. IEEE Computer Society, 2003.

Claire O'Malley, Steve Langton, Anne Anderson, Gwyneth Doherty-Sneddon, and Vicki Bruce. Comparison of face-to-face and video-mediated interaction. *Interacting with computers*, 8(2):177–192, 1996.

Mustafa Ozuysal, Michael Calonder, Vincent Lepetit, and Pascal Fua. Fast keypoint recognition using random ferns. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(3):448–461, 2010.

Jagdish Pandey, Yu-Te Liao, Andrew Lingley, Ramin Mirjalili, Babak Parviz, and B Otis. A fully integrated rf-powered contact lens with a single element display. *Biomedical Circuits and Systems, IEEE Transactions on*, 4(6):454–461, 2010.

Babak A. Parviz. For your eye only. *Spectrum, IEEE*, 46(9):36–41, 2009.

Wayne Piekarski and Bruce Thomas. Arquake: the outdoor augmented reality gaming system. *Communications of the ACM*, 45(1):36–38, 2002a.

Wayne Piekarski and Bruce H. Thomas. Using artoolkit for 3d hand position tracking in mobile outdoor environments. In *Augmented Reality Toolkit, The first IEEE International Workshop*, pages 2–pp. IEEE, 2002b.

Wayne Piekarski and Bruce H Thomas. Augmented reality user interfaces and techniques for outdoor modelling. In *Proceedings of the 2003 symposium on Interactive 3D graphics*, pages 225–226. ACM, 2003.

Wayne Piekarski, Ben Avery, Bruce H. Thomas, and Pierre Malbezin. Hybrid indoor and outdoor tracking for mobile 3d mixed reality. In *Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality*, page 266. IEEE Computer Society, 2003.

Thammathip Piumsomboon, Adrian Clark, Mark Billinghurst, and Andy Cockburn. User-defined gestures for augmented reality. In *Human-Computer Interaction–INTERACT 2013*, pages 282–299. Springer, 2013.

Thammathip Piumsomboon, David Altimira, Hyungon Kim, Adrian Clark, Gun Lee, and Mark Billinghurst. Grasp-shell vs gesture-speech: A comparison of direct and indirect natural interaction techniques in augmented reality. In *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on*, pages 73–82. IEEE, 2014.

Timothy Poston and Luis Serra. The virtual workbench: dextrous vr. In *Proceedings of the conference on Virtual reality software and technology*, pages 111–121. World Scientific Publishing Co., Inc., 1994.

Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. The go-go interaction technique: non-linear mapping for direct manipulation in vr. In *Proceedings of the 9th annual ACM symposium on User interface software and technology*, pages 79–80. ACM, 1996.

Ivan Poupyrev, Numada Tomokazu, and Suzanne Weghorst. Virtual notepad: handwriting in immersive vr. In *Virtual Reality Annual International Symposium, 1998. Proceedings., IEEE 1998*, pages 126–132. IEEE, 1998.

Ivan Poupyrev, Desney S. Tan, Mark Billinghurst, Hirokazu Kato, Holger Regenbrecht, and Nobuji Tetsutani. Developing a generic augmented-reality interface. *Computer*, 35(3):44–50, 2002.

Lulian Radu and Blair MacIntyre. Using children's developmental psychology to guide augmented-reality design and usability. In *Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on*, pages 227–236. IEEE, 2012.

Che Mohd Yusoff Rasimah, Azlina Ahmad, and Halimah Badioze Zaman. Evaluation of user acceptance of mixed reality technology. *Australasian Journal of Educational Technology*, 27(8):1369–1387, 2011.

Ramesh Raskar, Greg Welch, and Wei-Chao Chen. Table-top spatially-augmented realty: bringing physical models to life with projected imagery. In *Augmented Reality, 1999.(IWAR'99) Proceedings. 2nd IEEE and ACM International Workshop on*, pages 64–71. IEEE, 1999.

Ramesh Raskar, Greg Welch, Kok-Lim Low, and Deepak Bandyopadhyay. Shader lamps: Animating real objects with image-based illumination. In StevenJ. Gortler and Karol Myszkowski, editors, *Rendering Techniques 2001*, Eurographics, pages 89–102. Springer Vienna, 2001. ISBN 978-3-211-83709-2. . URL `http://dx.doi.org/10.1007/978-3-7091-6242-2_9`.

Ramesh Raskar, Jeroen van Baar, Paul Beardsley, Thomas Willwacher, Srinivas Rao, and Clifton Forlines. ilamps: Geometrically aware and self-configuring projectors. *ACM Trans. Graph.*, 22(3):809–818, 2003.

Malinda Rauhala, Ann-Sofie Gunnarsson, and Anders Henrysson. A novel interface to sensor networks using handheld augmented reality. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, pages 145–148. ACM, 2006.

Thomas Reicher, Asa MacWilliams, and Bernd Bruegge. Towards a system of patterns for augmented reality systems. In *International Workshop on Software Technology for Augmented Reality Systems (STARS 2003)*, 2003.

Gerhard Reitmayr and Tom W. Drummond. Going out: robust model-based tracking for outdoor augmented reality. In *Mixed and Augmented Reality, 2006. ISMAR 2006. IEEE/ACM International Symposium on*, pages 109–118. IEEE, 2006.

Jun Rekimoto. Augmented reality using the 2d matrix code. *Interactive Systems and Software IV. Kindaikagaku-sha*, pages 199–208, 1996a.

Jun Rekimoto. Transvision: A hand-held augmented reality system for collaborative design. In *Proc. Virtual Systems and Multimedia*, pages 85–90, 1996b.

Jun Rekimoto. Matrix: A realtime object identification and registration method for augmented reality. In *Computer Human Interaction, 1998. Proceedings. 3rd Asia Pacific*, pages 63–68. IEEE, 1998.

Jun Rekimoto and Yuji Ayatsuka. Cybercode: designing augmented reality environments with visual tags. In *Proceedings of DARE 2000 on Designing augmented reality environments*, pages 1–10. ACM, 2000.

Jun Rekimoto and Katashi Nagao. The world through the computer: Computer augmented interaction with real world environments. In *Proceedings of the 8th annual ACM symposium on User interface and software technology*, pages 29–36. ACM, 1995.

Miguel Ribo, Axel Pinz, and Anton L. Fuhrmann. A new optical tracking system for virtual and augmented reality applications. In *Instrumentation and Measurement Technology Conference, 2001. IMTC 2001. Proceedings of the 18th IEEE*, volume 3, pages 1932–1936. IEEE, 2001.

Julie Rico and Stephen Brewster. Usable gestures for mobile interfaces: evaluating social acceptability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 887–896. ACM, 2010.

Everett M. Rogers. *Diffusion of innovations*. Simon and Schuster, 4 edition, 2010.

Jannick P. Rolland and Henry Fuchs. Optical versus video see-through head-mounted displays in medical visualization. *Presence: Teleoperators and Virtual Environments*, 9(3):287–309, 2000.

Jannick P. Rolland, Richard L Holloway, and Henry Fuchs. Comparison of optical and video see-through, head-mounted displays. In *Photonics for Industrial Applications*, pages 293–307. International Society for Optics and Photonics, 1995a.

Jannick P. Rolland, Jim Parsons, David Poizat, and Dennis Hancock. Conformal optics for 3d visualization. In *International Optical Design Conference*, pages 760–764. International Society for Optics and Photonics, 1998.

Jannick P. Rolland, Larry Davis, and Yohan Baillot. A survey of tracking technology for virtual environments. *Fundamentals of wearable computers and augmented reality*, 1:67–112, 2001.

J.P. Rolland, W. Gibson, and D. Arierly. Towards quantifying depth and size perception as a function of viewing distance. *Presence: Teleoperators and Virtual Environments*, 4:24–49, 1995b.

Paul L. Rosin. Measuring corner properties. *Computer Vision and Image Understanding*, 73(2):291–307, 1999.

Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *Computer Vision–ECCV 2006*, pages 430–443. Springer, 2006.

Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: an efficient alternative to sift or surf. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2564–2571. IEEE, 2011.

Pedro Santos, Thomas Gierlinger, Oliver Machui, and André Stork. The daylight blocking optical stereo see-through hmd. In *Proceedings of the 2008 workshop on Immersive projection technologies/Emerging display technologiges*, page 4. ACM, 2008.

Daniel Scharstein and Richard Szeliski. High-accuracy stereo depth maps using structured light. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–195. IEEE, 2003.

Fabian Scheer and Stefan Müller. Indoor tracking for large area industrial mixed reality. In *ICAT/EGVE/EuroVR*, pages 21–28, 2012.

Reinhold Scherer, Mike Chung, Johnathan Lyon, Willy Cheung, and Rajesh PN Rao. Interaction with virtual and augmented reality environments using non-invasive brain-computer interfacing. In *1st International Conference on Applied Bionics and Biomechanics (October 2010)*, 2010.

Dieter Schmalstieg and Daniel Wagner. Experiences with handheld augmented reality. In *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pages 3–18. IEEE, 2007.

Dieter Schmalstieg and Daniel Wagner. Mobile phones as a platform for augmented reality. In *Proceedings of the IEEE VR 2008 Workshop on Software Engineering and Architectures for Real Time Interactive Systems (SEARIS)*, volume 1, page 3, 2009.

Dieter Schmalstieg, Anton Fuhrmann, Zsolt Szalavári, and Michael Gervautz. "studierstube" - an environment for collaboration in augmented reality. In *CVE '96 Workshop Proceedings*, 1996.

Dieter Schmalstieg, Anton Fuhrmann, and Gerd Hesina. Bridging multiple user interface dimensions with augmented reality. In *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*, pages 20–29. IEEE, 2000.

Dieter Schmalstieg, Anton Fuhrmann, Gerd Hesina, Zsolt Szalavári, L Miguel Encarnaçao, Michael Gervautz, and Werner Purgathofer. The studierstube augmented reality project. *Presence: Teleoperators and Virtual Environments*, 11(1):33–54, 2002.

Johannes Schöning, Michael Rohs, Sven Kratz, Markus Löchtefeld, and Antonio Krüger. Map torchlight: a mobile augmented reality camera projector unit. In *CHI'09 Extended Abstracts on Human Factors in Computing Systems*, pages 3841–3846. ACM, 2009.

Abigail J Sellen. Remote conversations: The effects of mediating talk with technology. *Human-computer interaction*, 10(4):401–444, 1995.

Marcos Serrano, Barrett M Ens, and Pourang P. Irani. Exploring the use of hand-to-face input for interacting with head-worn displays. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, pages 3181–3190. ACM, 2014.

Gilles Simon, Andrew W. Fitzgibbon, and Andrew Zisserman. Markerless tracking using planar structures in the scene. In *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*, pages 120–128. IEEE, 2000.

Asim Smailagic and Daniel P. Siewiorek. The cmu mobile computers: A new generation of computer systems. In *Compcon Spring '94, Digest of Papers.*, pages 467–473. IEEE, 1994.

John M. Smart, Jamasi Cascio, and Jerry Paffendorf. Metaverse roadmap overview, 2007.

Thad Starner. Human-powered wearable computing. *IBM systems Journal*, 35(3.4):618–629, 1996.

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*, 6(4):386–398, 1997.

Neal Stephenson. *Snow crash.* Bantam Books, 1992.

Youngjung Suh, Youngmin Park, Hyoseok Yoon, Yoonje Chang, and Woontack Woo. Context-aware mobile ar system for personalization, selective sharing, and interaction of contents in ubiquitous computing environments. In *Human-Computer Interaction. Interaction Platforms and Techniques*, pages 966–974. Springer, 2007.

Joseph W. Sullivan, editor. *Intelligent User Interfaces.* ACM, New York, NY, USA, 1991. ISBN 0201606410.

Ivan E. Sutherland. Sketch pad a man-machine graphical communication system. In *Proceedings of the SHARE design automation workshop*, pages 6–329. ACM, 1964.

Ivan E. Sutherland. The ultimate display. In *Proceedings of the IFIP Congress*, 1965.

Ivan E. Sutherland. A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, pages 757–764. ACM, 1968.

J. Edward Swan and Joseph L. Gabbard. Survey of user-based experimentation in augmented reality. In *Proceedings of 1st International Conference on Virtual Reality*, pages 1–9, 2005.

Zsolt Szalavári and Michael Gervautz. The personal interaction panel–a two-handed interface for augmented reality. *Computer Graphics Forum*, 16(3): C335–C346, 1997.

Zsolt Szalavári, Dieter Schmalstieg, Anton Fuhrmann, and Michael Gervautz. "studierstube": An environment for collaboration in augmented reality. *Virtual Reality*, 3(1):37–48, 1998.

Akinari Takagi, Shoichi Yamazaki, Yoshihiro Saito, and Naosato Taniguchi. Development of a stereo video see-through hmd for ar systems. In *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*, pages 68–77. IEEE, 2000.

Susumu Takumai. Surround-sound system. US Patent 8,041,060, October 18 2011.

John C. Tang and Ellen Isaacs. Why do users like video? *Computer Supported Cooperative Work (CSCW)*, 1(3):163–196, 1993.

Bruce Thomas, Victor Demczuk, Wayne Piekarski, David Hepworth, and Bernard Gunther. A wearable computer system with augmented reality to support terrestrial navigation. In *Wearable Computers, 1998. Digest of Papers. Second International Symposium on*, pages 168–171. IEEE, 1998.

Bruce Thomas, Wayne Piekarski, and Bernard Gunther. Using augmented reality to visualise architecture designs in an outdoor environment. *International Journal of Design Computing Special Issue on Design Computing on the Net (DCNet)*, 1(4.2), 1999.

Bruce Thomas, Ben Close, John Donoghue, John Squires, Phillip De Bondi, and Wayne Piekarski. First person indoor/outdoor augmented reality application: Arquake. *Personal and Ubiquitous Computing*, 6(1):75–86, 2002.

Michael Tidwell, Richard S Johnston, David Melville, and TA Furness. The virtual retinal display-a retinal scanning imaging system. *Proceedings of virtual reality World*, 95:325–333, 1995.

Engin Tola, Vincent Lepetit, and Pascal Fua. Daisy: An efficient dense descriptor applied to wide-baseline stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(5):815–830, 2010.

Marjaana Träskbäack and Michael Haller. Mixed reality training application for an oil refinery: user requirements. In *Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, pages 324–327. ACM, 2004.

Roger Y Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *Robotics and Automation, IEEE Journal of*, 3(4):323–344, 1987.

Mihran Tuceryan and Nassir Navab. Single point active alignment method (spaam) for optical see-through hmd calibration for ar. In *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*, pages 149–158. IEEE, 2000.

Yuko Uematsu and Hideo Saito. Ar registration by merging multiple planar markers at arbitrary positions and poses via projective space. In *Proceedings of the 2005 international conference on Augmented tele-existence*, pages 48–55. ACM, 2005.

Luca Vacchetti, Vincent Lepetit, and Pascal Fua. Fusing online and offline information for stable 3d tracking in real-time. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–241. IEEE, 2003.

James Vallino and Christopher Brown. Haptics in augmented reality. In *Multimedia Computing and Systems, 1999. IEEE International Conference on*, volume 1, pages 195–200. IEEE, 1999.

Bram van der Vlist, Gerrit Niezen, Stefan Rapp, Jun Hu, and Loe Feijs. Configuring and controlling ubiquitous computing infrastructure with semantic connections: a tangible and an ar approach. *Personal and ubiquitous computing*, 17(4):783–799, 2013.

DWF Van Krevelen and R. Poelman. A survey of augmented reality technologies, applications and limitations. *International Journal of Virtual Reality*, 9(2):1, 2010.

Kurt VanLehn. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist*, 46 (4):197–221, 2011.

Eduardo E Veas. *Augmented reality interfaces for environmental monitoring.* PhD thesis, Graz University of Technology, 2012.

Jonathan Ventura and Tobias Hollerer. Wide-area scene mapping for mobile visual tracking. In *Mixed and Augmented Reality (ISMAR), 2012 IEEE International Symposium on*, pages 3–12. IEEE, 2012.

Daniel Wagner and Dieter Schmalstieg. First steps towards handheld augmented reality. In *2012 16th International Symposium on Wearable Computers*, pages 127–127. IEEE Computer Society, 2003.

Daniel Wagner and Dieter Schmalstieg. Artoolkitplus for pose tracking on mobile devices. In *Proceedings of 12th Computer Vision Winter Workshop*, page 8pp, 2007.

Daniel Wagner, Thomas Pintaric, and Dieter Schmalstieg. The invisible train: a collaborative handheld augmented reality demonstrator. In *ACM SIGGRAPH 2004 Emerging technologies*, page 12. ACM, 2004.

Daniel Wagner, Thomas Pintaric, Florian Ledermann, and Dieter Schmalstieg. Towards massively multi-user augmented reality on handheld devices. In *Pervasive Computing*, volume 3468 of *Lecture Notes in Computer Science*, pages 208–219. Springer Berlin Heidelberg, 2005.

Daniel Wagner, Mark Billinghurst, and Dieter Schmalstieg. How real should virtual characters be? In *Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology*, page 57. ACM, 2006.

Martin Wagner. Building wide-area applications with the artoolkit. In *Augmented Reality Toolkit, The First IEEE International Workshop*, page 7pp. IEEE, 2002.

Manuela Waldner, Jörg Hauber, Jürgen Zauner, Michael Haller, and Mark Billinghurst. Tangible tiles: design and evaluation of a tangible user interface in a collaborative tabletop setup. In *Proceedings of the 18th Australia conference on Computer-Human Interaction: Design: Activities, Artefacts and Environments*, pages 151–158. ACM, 2006.

Jih-fang Wang, Ronald T Azuma, Gary Bishop, Vernon Chi, John Eyles, and Henry Fuchs. Tracking a head-mounted display in a room-sized environment with head-mounted cameras. In *Orlando'90, 16-20 April*, pages 47–57. International Society for Optics and Photonics, 1990.

Yanqing Wang and Christine L. MacKenzie. The role of contextual haptic and visual constraints on object manipulation in virtual environments. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 532–539. ACM, 2000.

Mark Ward, Ronald Azuma, Robert Bennett, Stefan Gottschalk, and Henry Fuchs. A demonstrated optical tracker with scalable work area for head-mounted display systems. In *Proceedings of the 1992 symposium on Interactive 3D graphics*, pages 43–52. ACM, 1992.

Jens Weidenhausen, Christian Knoepfle, and Didier Stricker. Lessons learned on the way to industrial augmented reality applications, a retrospective on arvika. *Computers & Graphics*, 27(6):887–891, 2003.

Mark Weiser. Hot topics-ubiquitous computing. *Computer*, 26(10):71–72, 1993.

Greg Welch, Gary Bishop, Leandra Vicci, Stephen Brumback, Kurtis Keller, et al. The hiball tracker: High-performance wide-area tracking for virtual and augmented environments. In *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 1–10. ACM, 1999.

Dongdong Weng, Yongtian Wang, Yue Liu, Jing Chen, and Dewen Cheng. Display systems and registration methods for mixed reality applications. In *SPIE Optical Engineering+ Applications*, pages 742805–742805. International Society for Optics and Photonics, 2009.

Giles Westerfield, Antonija Mitrovic, and Mark Billinghurst. Intelligent augmented reality training for assembly tasks. In *Artificial Intelligence in Education*, pages 542–551. Springer, 2013.

Steve Whittaker and Brid O'Conaill. The role of vision in face-to-face and mediated communication. In Kathleen E. Finn, Abigail J. Sellen, and Sylvia B. Wilbur, editors, *Video-mediated communication. Computers, cognition, and work.*, pages 23–49. Lawrence Erlbaum Associates Publishers, 1997.

Ederyn Williams. Experimental comparisons of face-to-face and mediated communication: A review. *Psychological Bulletin*, 84(5):963, 1977.

Rafal Wojciechowski, Krzysztof Walczak, Martin White, and Wojciech Cellary. Building virtual and augmented reality museum exhibitions. In *Proceedings of the ninth international conference on 3D Web technology*, pages 135–144. ACM, 2004.

Eric Woods, Mark Billinghurst, Julian Looser, Graham Aldridge, Deidre Brown, Barbara Garrie, and Claudia Nelles. Augmenting the science centre and museum experience. In *Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia (GRAPHITE 2004)*, pages 230–236. ACM, 2004.

Harald Wuest, Florent Vial, and D. Strieker. Adaptive line tracking with multiple hypotheses for augmented reality. In *Mixed and Augmented Reality, 2005. Proceedings. Fourth IEEE and ACM International Symposium on*, pages 62–69. IEEE, 2005.

Peta Wyeth and Helen C. Purchase. Using developmental theories to inform the design of technology for children. In *Proceedings of the 2003 conference on Interaction design and children*, pages 93–100. ACM, 2003.

Yan Xu, Evan Barba, Iulian Radu, Maribeth Gandy, Richard Shemaka, Brian Schrank, Blair MacIntyre, and Tony Tseng. Pre-patterns for designing embodied interactions in handheld augmented reality games. In *Mixed and Augmented Reality-Arts, Media, and Humanities (ISMAR-AMH), 2011 IEEE International Symposium On*, pages 19–28. IEEE, 2011.

Suya You and Ulrich Neumann. Fusion of vision and gyro tracking for robust augmented reality registration. In *Virtual Reality, 2001. Proceedings. IEEE*, pages 71–78. IEEE, 2001.

Suya You, Ulrich Neumann, and Ronald Azuma. Orientation tracking for outdoor augmented reality registration. *Computer Graphics and Applications, IEEE*, 19(6):36–42, 1999.

Jürgen Zauner and Michael Haller. Authoring of mixed reality applications including multi-marker calibration for mobile devices. In *10th Eurographics Symposium on Virtual Environments, EGVE*, pages 87–90, 2004.

Jürgen Zauner, Michael Haller, Alexander Brandl, and W Hartman. Authoring of a mixed reality assembly instructor for hierarchical structures. In *Mixed and Augmented Reality, 2003. Proceedings. The Second IEEE and ACM International Symposium on*, pages 237–246. IEEE, 2003.

Shumin Zhai and Paul Milgram. Telerobotic virtual control system. In *Robotics-DL tentative*, pages 311–322. International Society for Optics and Photonics, 1992.

Zhengyou Zhang. Microsoft kinect sensor and its effect. *MultiMedia, IEEE*, 19(2):4–10, 2012.

Feng Zhou, Henry Been-Lirn Duh, and Mark Billinghurst. Trends in augmented reality tracking, interaction and display: A review of ten years of ismar. In *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 193–202. IEEE Computer Society, 2008.

Dmitry N. Zotkin, Jane Hwang, R. Duraiswaini, and Larry S. Davis. Hrtf personalization using anthropometric measurements. In *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, pages 157–160. IEEE, 2003.