MonkeyBridge: Autonomous Agents in Augmented Reality Games

István Barakonyi[1] Markus Weilguny[2] Thomas Psik[3] Dieter Schmalstieg[1]

[1] Graz University of Technology, Institute of Computer Graphics and Vision, Inffeldgasse 16, Graz, A-8010, Austria

[2] Hagenberg University of Applied Sciences, Media Technology and Design, Hauptstraße 117, Hagenberg, A-4232, Austria

[3] Vienna University of Technology Institute of Design & Assessment of of Technology, Argentinierstr. 8/187, Vienna, A-1040, Austria

{ bara | hammer | tomp | dieter } @ ims.tuwien.ac.at

ABSTRACT
MonkeyBridge is a collaborative Augmented Reality (AR) game employing autonomous animated agents embodied by lifelike, animated virtual characters and "smart" physical objects. The game serves as a pilot application to examine how "smart" software and hardware components capable of observing and reacting to events in the physical and virtual world can be useful in AR applications. We describe the implementation details of our test setups as well as how autonomous agents offer a rich gaming experience in AR games by engaging users in various domains.

Categories and Subject Descriptors
H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems - *Animations, Artificial, Augmented, and Virtual Realities.* I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – *Intelligent Agents*.

General Terms
Design, Experimentation, Human Factors

Keywords
Augmented Reality, Autonomous Agents, Multiplayer Games

1. INTRODUCTION
MonkeyBridge is a multiplayer game, where users place real and virtual objects onto a physical surface, thus influencing the behavior of virtual animated characters and responsive physical objects (see Figure 1). During the development of the game we have been examining the use of embodied autonomous agents in Augmented Reality (AR) applications. We are interested in how "smart" software and hardware components are able to make autonomous decisions based on their observation of AR environments in which they are embedded. In this paper we argue that embodied autonomous agents can be used as intuitive user interfaces in AR environments, and discuss the design and implementation details of our pilot game application that illustrates our concepts.

2. MOTIVATIONS
The instinct for play is one the most fundamental elements of human culture [1]. Humans have not only always been *homo faber*, working, tool-making creatures but *homo ludens,* playful humans as well that create tools not only for survival in everyday life but also for entertainment. This instinct has not weakened within the modern man but rather got amplified with the appearance of technology: new tools exploiting latest computer hardware and software developments and trends emerge to serve recreation.

On the other hand, computer games have always served as perfect testbeds and prototyping tools for novel user interfaces and interaction techniques. When put into a game context, users tend to alleviate their strong expectations and strict requirements otherwise set for industrial applications, and rather concentrate on the application's playful nature while testing new research ideas. This is one major reason why we have chosen a computer game as our pilot application to validate our theories.

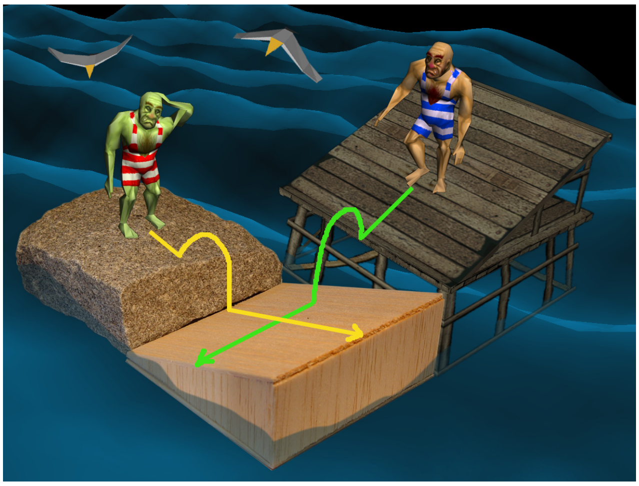


Figure 1. Concept image from the MonkeyBridge game: the agents autonomously plan their path and behavior (walk, jump, climb etc.) on physical and virtual game tiles

2.1 Fantasy and Augmented Reality

Children possess the remarkable skill of being able to quickly assign roles to real-world objects and act out complex scenes with multiple actors. Plastic soldiers are thereby turned into affective, “live” characters with a particular personality, toy cars into F-1 racing cars and the living room carpet into a battle field or a racecourse. Kids typically play all the roles simultaneously by themselves.

A key element in games and stories is fantasy. While playing, a virtual, imaginary world is created within our mind, inhabited by characters obeying our imagination. In classic make-believe games this fantasy world and their characters connect to the real environment through physical game props to which various roles are assigned, thus making heretofore passive objects active players in the game story.

AR applications are aiming at achieving the same effect by superimposing a virtual world on top of the real environment. Since the virtual world is registered with the real one, they appear to coexist. As pointed out by Stapleton and colleagues in their mixed-fantasy framework [2], the combination of the real and virtual help suspend disbelief and enrich the audience’s fantasy experience. AR is able to visually change real world attributes, make passive objects appear animated or play sound effects besides sounds in the real environment to further enhance the atmosphere of the perceived mixed environment.

2.2 Collaborative Playing

The popularity of multiplayer games indicates that people greatly enjoy the social aspects of gaming such as competing against or teaming up with fellow players in various situations such as a battle zone or stock exchange without the responsibility of affecting real life. Developing skills and comparing them with competitors create a challenge that motivates people to play the game again and again, which is a major goal in the commercial game industry.

Computer technology can greatly assist in engaging players in a multi-user game experience. Classic networked multiplayer computer games connect remote users over great distances and enable them to coinhabit, collaboratively explore and interact with virtual worlds. However, most PC-based games are still closer to individual than to true multiplayer games such as board games since the perception of fellow players is fairly limited. By the observation of non-verbal communication or facial expressions players can often deduce their opponents’ emotional state, which is another central element of gaming. This information channel is largely constrained by the computer game’s communication tools and bandwidth, and is often limited to a single chat window. Similarly, input devices such as game pads, joysticks or keyboards rather support individual interaction than collaborative access to game environments.

AR applications preserve natural face-to-face communication since they require that collaborators either share the same physical space or rely on special remote collaboration tools [3] connecting remote AR environments. Thus the use of natural conversation, facial expressions and body gestures is enabled.

2.3 Embodied Autonomous Agents

In virtual game environments co-players are often represented by avatars. The behavior of the avatars is constrained and controlled by the game logic, which maps player actions to the set of capabilities and behavior elements of the avatars. In addition to players there is often a large variety of non-player characters serving as allies, bystanders or competitors.

As games become increasingly complex, it is desirable that some workload is taken off from both game developers and players by adding autonomy to player and non-player characters. An autonomous character does not need constant user guidance or thoroughly scripted behavior prepared for all possible situations. Instead it proactively makes decisions based on events coming from sensors present in its environment. Thus only high-level goals are needed to be set, while the character’s reasoning engine takes care of low-level details to achieve the goals as quickly as possible. Such a character that appears to have a “brain” behind its movements is called an *embodied autonomous agent*.

Besides being efficient problem solvers, game characters need to appear believable as well. Artificial intelligence technology attempts to breathe life into embodied agents by equipping them with human-like capabilities so that they appear lifelike. Porter and Susman of Pixar Animation Studios [4] define the term “lifelike” as a capability of “convincing the audience an animated character has intelligence, personality, and emotion while inhabiting a physical world”. Lifelike agent behavior does not necessarily require the use of complex reasoning engines. Bates [5] makes the point that appropriately timed and clearly expressed emotion is an essential and often sufficient requirement for characters to behave in a believable way. Prominent examples are “tamagotchis”, handheld virtual pets originally created in Japan. These primitive yet addictive digital creatures possess a fairly simple behavior model based on two major goals: keeping the health and the happiness index at a high score by letting their owners manipulate some controls to feed them, play with them etc. Tamagotchis were a big commercial success since they exploited people’s natural affection to pets. Experience has proven that this affection is independent from the fact that the pet is virtual as long as it manages to engage users emotionally.

3. AUTONOMOUS AGENTS IN AR

The work of Nilsen et al. on motivations of AR gaming [6] suggests that designers of AR games consider four relevant domains during development: physical, mental, social and emotional. In the MonkeyBridge project we focus on embedding embodied autonomous agents into collaborative AR games. As we demonstrate in later sections, AR agents engage users in all these domains, thus create a rich gaming experience.

3.1 Embodiment

AR agents are embodied as three-dimensional virtual or *physical* objects. They share users’ physical environment, in which they can freely move using all 6 degrees of freedom. Virtual agents in AR scenarios appear to have a solid, tangible body that can be observed from an arbitrary viewpoint, thus becoming integral parts of the physical environment. Virtual objects are typically animated characters but are not necessarily anthropomorphic.

Figure 2. Virtual repairman explaining the operation of a real LEGO® Mindstorms robot (note the correct occlusion)

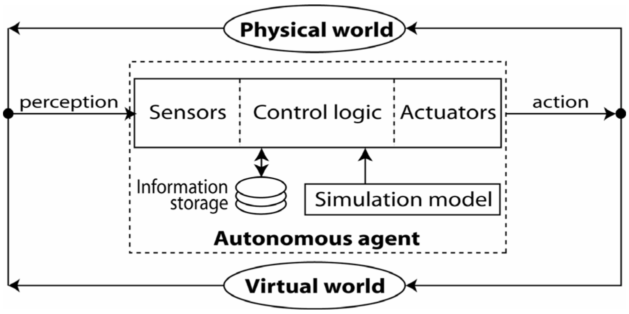


Figure 3. Behavior scheme for autonomous agents in AR applications

A novel and exciting new aspect of AR agents is that physical objects like a printer, a digital piano or an interactive robot can be turned into intelligent, responsive entities that collaborate with virtual characters. If we track and monitor relevant physical attributes and process this data, attribute changes generate events that can be interpreted by other agents and application logic. Using network packets, infrared messages, MIDI code sequences or other means of low-level communication, physical objects can not only be queried for status information but also be controlled by external commands. Therefore physical objects act as input and output devices in AR spaces and become equal and active partners of virtual characters. Figure 2 shows a screenshot of an AR machine maintenance application, where a virtual repairman character explains the structure and operation of a LEGO® Mindstorms robot in accordance with the robot's actions in the physical world. The robot receives commands and sends status information via an infrared communication channel to the application running on a PC.

The combination of the real and virtual embodiment results in the augmented embodiment. This assumes the presence of the physical representation and only superimposes necessary virtual information. Virtual agent bodies may have an associated tracked physical object, which serves as a tangible control interface, while this is a prerequisite for augmented agent representations. The augmented representation also helps overcome the problem of correct visual occlusion. This means that we should ensure that physical objects placed between the user's viewpoint and virtual agents appear to cover parts of the virtual objects behind. This issue can be easily supported by tracking the occluding object's pose and associating it with an augmented representation that only renders an approximate virtual model into the depth buffer at the right location. The physical LEGO® robot shown in Figure 2 is augmented with virtual information indicating the current battery level and a button to be pressed. The robot correctly occludes the virtual repairman character standing behind the robot.

An AR agent may have several representations and always choose the most appropriate one deliberately or upon user command. For instance the robot maintenance application is able to automatically switch from the robot's augmented representation to the virtual one if the "alive" messages cease to arrive from the robot due to a failure in the robot control module or the infrared link. Thus the explanations can be continued with a simulated virtual model. Later the user may wish to switch back to the physical or augmented representation if the real robot becomes available again.

3.2 Agents as User Interfaces

As pointed out in our previous work [7], AR agents are capable of automatically taking care of low-level details such as network communication between components or switching between representations. Because of the highly dynamic and complex nature of AR systems, agents enable users to focus on higher, application-level objects instead of low-level details such as networking or pose tracking, thus improving usability. Therefore AR agents represented by autonomous software and hardware components can serve as key user interface elements in AR applications.

Figure 3 illustrates the behavior scheme of autonomous agents in an AR environment. To add semantics to the visual and aural means of AR agents' augmented representation, our system needs a model of the physical world. Staying with the LEGO® robot example, the system needs to be aware of the mechanical and spatial structure of the robot, the meaning of status messages and possible error reports arriving from the robot control module via the infrared link. The system also needs to maintain its knowledge of the real world, therefore the data received by the agent's sensors is processed by some control logic or reasoning module based on the agent's internal world model. The agent stores and retrieves information to implement memory and persistency. Having interpreted sensor data, the agent can react accordingly by generating actions in the physical world such as triggering some actuators, and in the virtual world by rendering visual and aural content such as 3D models, animations, 2D text or sound effects.

4. GAME CHARACTERS IN AR

Agent communication with the augmented environment includes four information channels to sense and affect the physical and virtual environment. The implementation of these channels is usually not a straightforward task. We provide some examples to demonstrate how the real and the virtual world can serve as input and output communication modality.

4.1 Physical input effecting virtual output

Figure 4 illustrates a simple example how a virtual animated agent is able to respond to changes in attribute changes of the real world. A tangible, physical optical marker acts as a platform for a virtual monster artiste to stand on. The user holds and tilts around the marker in front of a webcam, while the artiste agent appears to be struggling to maintain its balance. If the angle of the marker becomes too steep, the monster falls down with a roar. The application retrieves the current pose of the marker relative to the webcam using the ARToolKit optical marker recognition module [8]. The pose of the marker is directly mapped to the pose of the virtual platform of the artiste in the agent's world model. The webcam and the marker recognition library act as the agent's sensor to perceive changes in the physical marker's attributes. The agent's control logic then checks whether the platform orientation is still within bounds and decides whether to play the "fall down" or the balancing animation. The balancing animation is a blended motion interpolating between the neutral center and four extreme points in the animation space with factors calculated from the platform's pitch and roll rotation angles.

4.2 Virtual input effecting physical output

Sensing events generated by virtual objects is usually not a complicated task since virtual sensors such as vision, hearing, touching can be implemented in software using various algorithms. However, using physical objects as output communication modalities has several constraints. Our entire physical environment cannot be affected by virtual control logics, only by specially prepared objects which require communication channels and actuators to be set up.

The screenshots shown in Figure 5 depict a sample scenario implementing defensive behavior for the aforementioned physical LEGO® robot that tries to avoid collision with a virtual character. The pose of the robot and the character is again tracked using ARToolKit. If the character enters the virtual "safety" area around the robot, a command is sent to the robot from the PC via an infrared link, instructing it to move away.

We are currently implementing responsive physical objects in our game that are able to react to actions of virtual characters. Although other researchers have made some efforts to create standard physical hardware components that are able to communicate with their virtual software counterparts such as the "phidgets" of Greenberg and Fitchett [9], we decided to implement our own physical game elements relying on our own ideas and techniques. Reasons include the need for highly customized real-world props in our game and the hobby electronics skills of one of the authors, who could quickly design and build prototypes matching the project's experimental nature.

4.3 Other scenarios

In the case of physical input affecting the real world (e.g. two physical robots interacting with each other) the hard part of both previous scenarios is taken: sensors need to be installed in the real world again to monitor changes, in addition to communication channels and actuators to produce physical output.

Generating virtual output based on virtual input is a trivial Virtual Reality (VR) problem that we do not discuss here in detail.

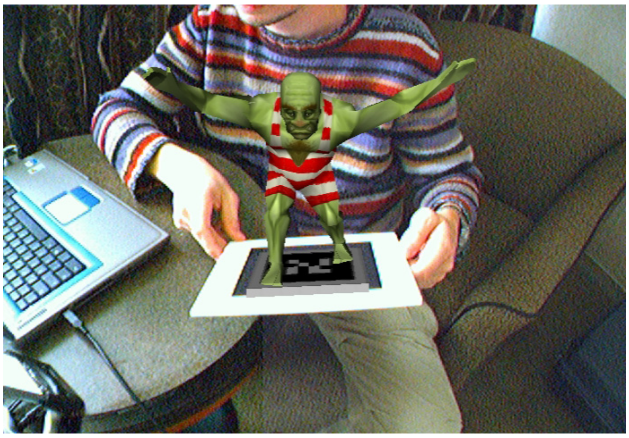


Figure 4. Screenshot of an animated character balancing on a tangible optical marker

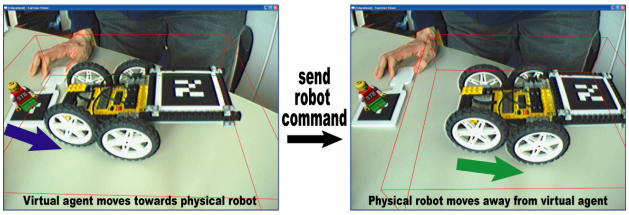


Figure 5. Screenshots of a defensive physical LEGO® robot avoiding collision with a virtual character

5. RELATED WORK

5.1 AR in Entertainment

Bolter and Grusin [10] make the point that a new media, such as AR, starts to gain wider public acceptance once it enters the game, art and entertainment domain. Numerous research projects and conferences in recent years indicate that AR is well on its way to receive more public attention and become a new media form.

Background work for AR games embraces several domains including indoor and outdoor AR, ubiquitous and social computing and recently even mobile computing. Early works such as the Mah-Jongg game from Szalavári et al. [11] and the AR^{2}Hockey [12] and AquaGauntlet applications [13] from Ohshima and colleagues implement collaborative AR games with users wearing see-through HMDs tracked by magnetic sensors. Players interact with the game space using handheld, tangible tracked physical objects that are visually augmented. These props are mostly used as 3D pointers or personal augmented displays. The aforementioned games do not require their users to extensively move around in the physical environment since the game area is fixed to a certain indoor location and slight movements changing user viewpoint are sufficient to fully observe the augmented space.

More physical involvement is required by the PingPongPlus application [14] from Ishii et al., which enhances the physical activity and social interaction in a table tennis game with fun visual effects. To label their application, the authors coined the term Computer-Supported Cooperative Play (CSCP), which may serve as an umbrella category for all applications presented in this section as well as our own game. Another two-player game demanding active body movement from players is MIND-WARPING [15]. An interesting aspect of this game is the use of dual viewpoints: an immersive and an external, world-in-miniature view associated with the two different player roles.

Active physical movement ensures a better sense of presence in the real world, therefore virtual content causes less separation from the physical environment, enabling more natural game interfaces. One important type of bodily involvement is walking around in the physical world to explore an augmented environment, which is typical for outdoor AR applications. While being technically challenging, pioneering projects such as ARQuake [16] or Human Pacman [17] offer great potentials for gaming. ARQuake recreates the famous first-person shooting game in a real campus setting, while the Human Pacman project applies mobile and ubiquitous computing technologies in a wide-area entertainment system. Although an indoor application, the TouchSpace system [18] also lets its users walk around and explore a room-sized augmented world with "magic" windows fixed to tracked, handheld props. The project also examines ways of smoothly traveling between AR and VR within the same application.

The SharedSpace project [19] provides freedom in physical interaction and communication similar to the real world. Here users collaboratively reveal virtual 3D content attached to tangible physical cards. This game as well as experimental games from Magerkurth et al. [20] emphasize and rely on the rich social experience and collaborative interaction that players may be already familiar with in traditional board games.

Mixed Reality games have recently appeared on the commercial market with the EyeToy™ technology [21] on Sony PlayStation® game consoles. EyeToy™ includes a motion sensing technology and a live video background augmented by virtual content. The synthetic objects appear to interact with the physical environment, in particular users' body parts. However, these commercial games do not truly fit the category of AR games yet since they lack the possibility of dynamically changing the viewpoint and ignore depth information in the recorded scenes.

5.2 Agents in Games

As the computing power of commercially available PCs increases year to year, computer games become equipped with increasingly complex "intelligence" engines. One of the earliest implementations of software agents appearing in games are robot players such as the ChatterBot by Mauldin [22] living in the famous MUD (Multiuser Dungeon) world. These "smart" characters are able to autonomously explore the virtual environment, and converse with other players. The SOAR Quakebot [23] developed for the popular shoot'em-up game Quake approximates the game play of a human. It builds its own map while exploring a level, applies various tactics based on its internal world representation, and attempts to anticipate its opponents' actions. The Sims game [24] from Electronic Arts allows players to create virtual human beings and guide them through a virtual suburban life. Several attributes can be set that influence the characters' decisions and actions. The characters are highly reactive and endow users with God-like capabilities, making the game a commercial success.

The physical counterparts of virtual embodied autonomous agents are called "smart toys". A typical example is Tiger Electronics' Furby [25], an interactive animatronic plush toy, which is equipped by several sensors to get feedback from the user, an infrared sensor to communicate with other Furbies and tiny electric motors for movement. This toy is capable of expressing simple emotions, playing child games, and when being stimulated, appears to become progressively more intelligent. A more advanced example is Sony's AIBO® [26], an autonomous robotic dog. The built-in software provides AIBO with emotions and instincts as well as the ability to learn, communicate and mature based on external stimuli from its owner or environment and from other AIBOs.

5.3 Agents in AR

Although extensively researched in VR, agents have only recently appeared in AR environments. An early AR application providing character support is the ALIVE system [27], where a virtual animated character composited into the user's real environment responds to human body gestures on a large projection screen. This type of display separates the user's physical space from the AR environment, which demands carefully coordinated user behavior. The Welbo project [28] features an immersive setup, where an animated virtual robot assists an interior designer wearing an HMD. The character lacks a tangible physical representation and can only interact with virtual objects. Another HMD-based system from MacIntyre et al. [29] creates an interactive theater experience by placing prerecorded video-based actors into an AR environment. The characters do not possess any autonomy, as their behavior is scripted, and interaction is limited to changing viewpoints and roles in the story. Cavazza et al. [30] place a live video avatar of a real person into a Mixed Reality setting, and interact with a digital storytelling system with body gestures and language commands. Balcisoy et al. [31] experiment with interaction techniques with virtual humans in Mixed Reality environments, which play the role of a collaborative game partner and an assistant for prototyping machines. Cassell et al.'s Sam agent [32] is a virtual playmate assisting children in a natural storytelling play with real objects. Access to the real game props is shared between the child and the animated agent.

We believe that the next step in agent-enabled gaming is mixing physical and virtual embodied autonomous agents in a single augmented game space. We are currently not aware of any applications that would fully exploit both the virtual and physical domain as input and output communication modalities except one notable application. The ActiveCube project [33] implements an interactive toy consisting of a set of computerized tangible blocks equipped with several input/output devices, from the player constructs a shape. The physical model is automatically recognized by a 3D shape retrieval module and transformed into a virtual 3D model on a display. Various sensors allow real-time interaction with the model while multimedia content related to the chosen model is being played. The key difference between ActiveCube and our project is the clear border between the real and the virtual world. Even though a special gyroscopic sensor can ensure that the virtual and physical representation move together on the display, the virtual content appears to be separated since it is not overlaid on top the physical environment, which is a key requirement for true AR applications.

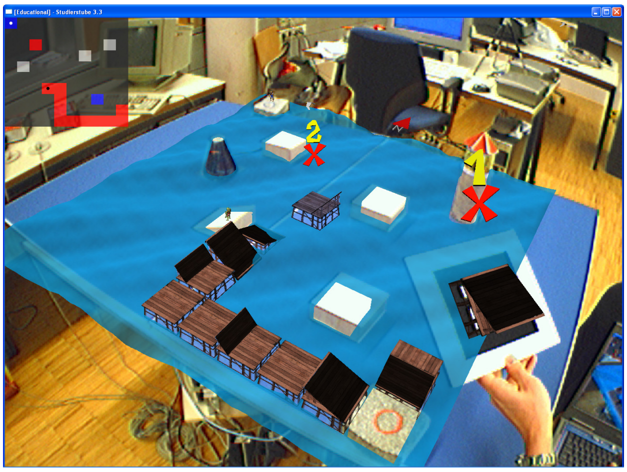


Figure 6. Screenshot of a two-player MonkeyBridge game

6. THE MONKEYBRIDGE GAME

A “monkey bridge” is a fragile wooden construction over a river in South-East Asia [34]. People frequently risk their lives as they try to keep their balance crossing to the other side. In our demo players dynamically build a monkey bridge for their own monster-like characters using virtual and physical pieces of landing stage, which vary in shape. The goal is to reach a dedicated target in a virtual ocean. Figure 6 provides a screenshot of a typical game scene. In this picture one of the players has already built a bridge for his character, which consists of virtual blocks (models with the dark wooden texture) and physical tiles (bright balsa-wood and stone cubes showing through the virtual objects). The monster character of the user standing in the middle has just hopped over from a virtual tile onto a physical platform. The user is holding the next building block in his hand.

6.1 Contributions

The MonkeyBridge game incorporates novel concepts for AR gaming since it exploits the real world in an advanced way. Our project has made the following contributions:

- The game employs embodied autonomous agents that are able to dynamically recognize changes and events in both the physical and virtual game environment and proactively generate real-time behavior without direct user control.
- Unlike in the aforementioned related projects, physical objects act as active partners of virtual characters yielding output information in the augmented game space while appearing to coexist with virtual game objects.
- Our modular software architecture enables the use of various tracking and display technologies without changing the game application. Thus several game setups are possible so that issues such as financial constraints and portability can be considered.

6.2 Building a Bridge

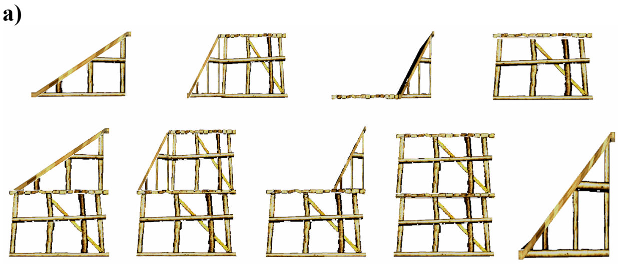
Players do not have direct influence on the game characters’ behavior; instead they indirectly control character movement by providing the agents with building blocks to walk on above the virtual ocean. In a typical setup the ocean is divided into 10x10 cells yielding a 1m x 1m rectangular physical game board, however, the grid and cell size can be customized. Each cell may host a building block functioning either as a bridge element or an obstacle. All possible building blocks can be seen in Figure 7a-c. Bridge elements are either physical or virtual and are composed of simple geometrical shapes that fit together smoothly, while obstacles are physical objects that serve as strategic hindrances to players as well as decoration elements.

The virtual bridge elements are auto-generated in a similar way to Tetris games and are dynamically laid onto the game board by the players. The position and orientation of the blocks are automatically snapped to the cells to make positioning easier. If left unmoved above an unoccupied cell for a given time, the block becomes fixed, and the player occupies the given cell. A player is allowed to place a tile only into cells that are adjacent to other cells already occupied by the player.

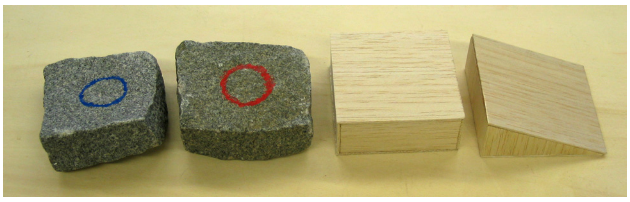
The real building blocks are crafted from stone and wood. Unlike their virtual counterparts, the position and orientation of the physical blocks are fixed during the game, although they can be arbitrarily configured before start-up. The physical blocks represent the start and target platforms for the characters as well as strategic points to reach.

We have built two physical obstacles: a lighthouse with rotating spotlights and a volcano puffing real, illuminated smoke. The casing can be quickly assembled from paper templates which then host electric engines, LEDs and a smoke generator. The responsive feature of the obstacles is still under development. Our test setup currently lets users manually control electric parts functions using a custom-made control box**.**

a)



b)



c)

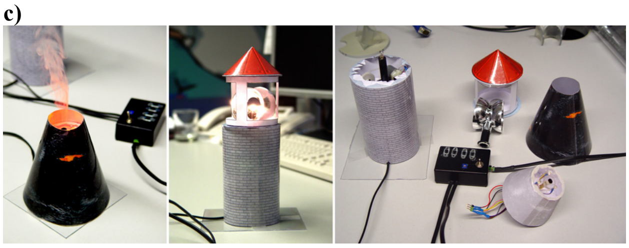


Figure 7. Building blocks: a) Virtual bridge elements, b) Physical bridge elements, c) Physical obstacles

6.3 Autonomous Behavior

The monster-like characters are embodied autonomous agents since their behavior does not require careful scripting. Instead a dedicated control logic or virtual "brain" decides which animations and sound effects to play, which direction to turn or whether the target has been reached. The only factors that directly influence agent behavior are the spatial distribution, pose and shape of the virtual and physical building blocks placed on the game board. Figure 8a-b provide illustration.

The characters autonomously choose: the path they walk on; decide how to get from one platform to the other, e.g. jump up or down when there is a slight difference in height between platform edges; automatically choose the straightest path from several available tiles; and fall into the water if there is no suitable piece of landing stage to walk on. They happily cheer with their hands up when they win, and cry over a lost game.

The responsive physical agents embodied by the volcano and lighthouse object incorporate a rather simple behavior to prove our concept: whenever a virtual agent reaches a cell adjacent to an obstacle, the associated object is turned on. The fun factor of seeing the volcano puffing smoke or the rotating lights of the lighthouse motivates players to lead the path of the monsters towards these objects, imposing influence on play strategy.

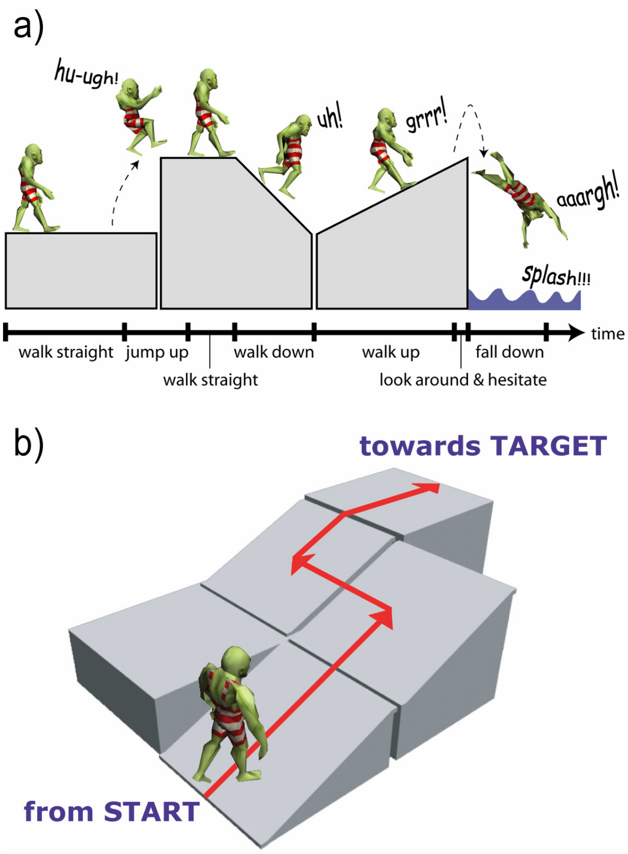


Figure 8. Autonomous agent behavior: a) choosing animation and sound based on platform type, b) path planning depending on the spatial distribution of available blocks

6.4 Domains of Game Experience

Recent researcher papers on AR gaming [6][20][17] have established the convention of discussing how game elements contribute to user stimuli in individual domains of game experience, therefore we also provide our own summary.

Mental domain: We attempt to create the illusion of a real marine landscape by using various visual and aural elements. Visual and audio effects help suspend disbelief, therefore enhance the mental image created by the players' own fantasy. In our game a flock of animated virtual seagulls fly around squawking while realistically animated 3D ocean waves boom below. The seagulls' movement is controlled by a boid algorithm based on the work of Reynolds [35]. The physical bridge blocks are made of stone and wood while their virtual counterparts use similar textures. Additional real world decoration elements such as our lighthouse and volcano offer unlimited opportunities to create a powerful game atmosphere. If registered correctly, physical objects appear to be washed by synthetic ocean waves, further blurring the boundary between the real and virtual.

Emotional domain: The emotional experience is predominantly delivered by the animated monster agents. Animations and sound effects imply that the monsters possess a lazy, dull yet likeable personality. Since virtual characters situated in AR environments give the impression of possessing a real, tangible body that is part of the player's own environment, characters appear more lifelike. In addition, the almighty nature of the character control deciding about life and death attaches players more emotionally to the creatures they are responsible for.

Physical domain: As previously discussed, physical activity greatly enhances the playful nature of games. Playing with MonkeyBridge also involves a considerable amount of body movement because users constantly have to find the best cell to place the next building block while preventing the opponents from reaching their target first. A Head-Up Display (HUD) provides an overview about the currently occupied cells allowing strategic plans. Time pressure causes tension in the game that comes from the suicidal attitude of the characters. Similar to the virtual rodents of the famous Lemmings game [36], the monster agents keep walking forward, even if there is no bridge platform ahead to step on.

Social domain: This domain is strongly connected to the physical domain. In AR opponents are natural parts of the environment preserving such important communicational cues as body and facial gestures. While the players compete against one another to occupy cells, they unwittingly block the other users' hand, camera etc., which may result in debates or jokes.

7. IMPLEMENTATION

7.1 Game Setups

Our game is grounded in the Studierstube AR platform [37], which is a middleware allowing for a wide range of collaborative multi-user AR applications. Its modular structure enables experiments with several game setups using various tracking systems, displays and interaction devices. The game can be configured to be a distributed application or to run on a single computer. Before deciding for a setup, financial and technical factors such as cost, installation time and calibration efforts of the tracking system and display need to be considered.

We realized three demo setups to test MonkeyBridge. The first and simplest one runs on a single computer without video background, and uses a simple keyboard-based tracking simulator. This is not an AR application, only VR. The second prototype relies on a multi-user setup with two computers sharing application data and tracking information provided by the ARToolKit optical marker recognition system. The setup that we demonstrated at the ISMAR'04 conference requires two computers equipped with webcams and three optical markers: a large calibration marker to register the physical game board with the virtual game environment, and two small markers acting as user interaction devices to place the bridge blocks. The live video stream recorded by the cameras is augmented on the user's computer screen residing next to the physical game board. Although this setup is highly portable, requires simple calibration, lacks cables and can built at low cost, it has significant drawbacks such as inferior tracking quality severely affected by lighting conditions and occlusion, camera distortion and static viewpoint. The reason for why the cameras have to remain fixed after an initial calibration phase is the fact that we place several props onto the game board upon start which would cause the tracking algorithm to lose the game board calibration marker during recognition.

Figure 9. Pictures of the optical marker-based setup (upper) and the magnetic tracker-based setup (lower)

The third setup uses the Flock of Birds magnetic tracking system from Ascension to track two Sony Glasstron optical see-through Head-Mounted Displays (HMDs) and two Plexiglas pucs to place the virtual tiles. This setup requires a specially manufactured table lacking any metal parts (screws, bolts etc.) to avoid distortion of the magnetic field. As tracking quality is superior to the previous setup and the HMDs provide dynamical viewpoint change, users have a strong sense of presence. The tracking system and the HMDs need precise calibration every time the game is installed in a new location.

Despite the superior visual quality the magnetic setup offers, it is heavily tethered, which might come as a nuisance to some players. This problem could be overcome by a fourth setup variant which we haven't implemented yet: a natural feature tracking-based solution using a handheld tablet PC serving as a window onto the augmented world.

7.2 AR Agents

The virtual monster agents are skeleton-based animated characters built on the open source library Cal3D [38]. This library is capable of the import and display of high-quality animation exported from 3D Studio MAX's Character Studio, allowing for almost endless artistic possibilities. It also allows direct access to bones, which permits inverse kinematics and the linking of objects to joints, for example to pick up and carry an object.

The planned implementation of the responsive physical objects relies on a microcontroller attached to the PC via serial port. The "firmware" running on the microcontroller is able to perform some basic actions such as "switch on device A for n seconds" or "switch device B off" in addition to commands arriving from the control software running on the PC. Thus autonomous behavior generation resides both on the PC and the microcontroller, thus sharing high- and low-level tasks and reducing the amount of exchanged information.

The virtual and physical agents are integrated into the AR Puppet hierarchical animation framework [7] that is specifically designed to support AR agents and is built on top of the aforementioned Studierstube platform.

8. DISCUSSION

AR has great potentials in nearly all parts of the game content development pipeline. While existing systems focus on story or game environment creation [39][40], we claim that character modelers and animators could greatly benefit from AR techniques. Modelers and animators often rely on real-life references to build and animate 3D characters for games or film production. Observing the real world by photographing or videotaping the subject, or asking someone to pose helps create more believable, precise and expressive character animation. Figure 10 shows our colleague posing for us while creating animation for the balancing demo described in Section 4.1. Professional artists use motion capture techniques or other expensive means of acquiring motion data to create an essential initial data set for the final, refined animation. With AR new possibilities in character animation open up. The animated virtual model and the real-world reference can be merged into one, interactive modeling instrument. We would like to develop a tool using pose-tracked physical objects as input devices to animate skeleton-based virtual 3D models. This requires the mapping real-

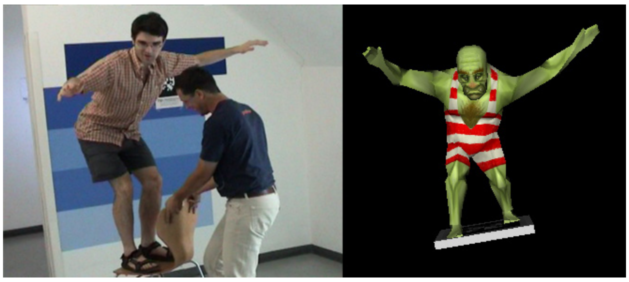


Figure 10. Human actor as a model for the balancing monster animation

time pose information to rotation information for the joints of the character skeleton using inverse kinematics and motion mapping techniques. AR would enable not only a close interaction with the virtual model by using tangible objects but also the creation of complex motions like walking up stairs or lifting a ball, since the animator can use the actual physical models of the stairs or the ball together with the character to create the motion.

9. CONCLUSION AND FUTURE WORK

In this paper we have examined aspects how embodied autonomous agents can be used in AR games. Our pilot game application called MonkeyBridge demonstrates how these agents offer rich game experiences by stimulating users in several domains at the same time, while being embodied as responsive physical objects and lifelike animated virtual characters. We have built three test setups from which we gained experience how to design technically and financially feasible, visually attractive AR games.

An important aspect we would like to improve in the future is playability. Although we wanted to create a real game with rules, not just a simple demo, some people were not willing to invest time into learning the game and just focused on the technical side. We hope that if the game becomes more intuitive, we manage to hide the underlying technology. We also would like to improve our content by integrating shaders for more attractive visual content, experiment with new responsive objects, and add more intelligence to the game logic so that it can assess and adapt to the players' current level.

10. ACKNOWLEDGMENTS

This project was sponsored by the Austrian Science Fund FWF (contract no. Y193). The authors would like to say thank you to Circled Cube for the sound effects and to Gerhard Reitmayr for his useful comments.

11. REFERENCES

[1] Huizinga, J., "Homo Ludens: A Study of the Game Element in Culture", *Beacon Press*, Boston, 1971, ISBN-0-8070-4681-7

[2] Stapleton, C., Hughes, C., Moshell, M., Micikevicius, P., Altman, M., "Applying Mixed Reality to Entertainment", *IEEE Computer* 35(12), 2002, pp. 122-124.

[3] Barakonyi, I., Fahmy, T., Schmalstieg, D., "Remote Collaboration Using Augmented Reality Videoconferencing", *Proc. of Graphics Interface 2004 (GI'04)*, London, ON, Canada, 2004, pp. 89-96.

[4] Porter, T., Susman, G., "On Site: Creating Lifelike Characters in Pixar Movies", *Communications of the ACM* 43(1), 2000, pp. 25-29.

[5] Bates, J., "The Role of Emotion in Believable Agents", *Communications of the ACM* 37(7), Special Issue on Agents, 1994, pp. 122-125.

[6] Nilsen, T., Linton, S., Looser, J., "Motivations for AR Gaming". *Proc. of Fuse 04, New Zealand Game Developers Conference*, Dunedin, New Zealand, 2004, pp. 86-93.

[7] Barakonyi, I., Psik, T., Schmalstieg, D., "Agents That Talk And Hit Back: Animated Agents in Augmented Reality", *Proc. of IEEE and ACM International Symposium on Mixed and Augmented Reality 2004 (ISMAR'04)*, Arlington, VA, USA, 2004, pp. 141-150.

[8] ARToolKit Developer Homepage, http://sourceforge.net/projects/artoolkit/

[9] Greenberg, S., Fitchett, C., "Phidgets: Easy Development of Physical Interfaces through Physical Widgets", *Proc. of the 15th Annual ACM Symposium on User Interface Software and Technology (UIST 2001)*, Orlando, FL, USA, 2001, pp. 209-218.

[10] Bolter, J.D., Grusin, R., "Remediation: Understanding New Media", *MIT Press*, Cambridge, Massachusetts, 2000, ISBN-0-262-52279-9

[11] Szalavári, Zs., Eckstein, E., Gervautz, M., "Collaborative Gaming in Augmented Reality", *Proc. of Virtual Reality Software and Technology Conference (VRST'98)*, Taipei, Taiwan, 1998, pp. 195-204.

[12] Ohshima, T., Satoh, K., Yamamoto, H., Tamura, H., "AR2 Hockey: A Case Study of Collaborative Augmented Reality", *Proc. of the Virtual Reality Annual International Symposium (VRAIS '98)*, Atlanta, GA, USA, 1998, pp. 268-275

[13] Ohshima, T., Satoh, K., Yamamoto, H., Tamura, H., "RV-Border Guards: A Multi-Player Mixed Reality Entertainment", *Transactions of the Virtual Reality Society of Japan* 4(4), 1999, pp. 696-706.

[14] Ishii, H., Wisneski, C., Orbanes, J., Chun, B., Paradiso, J., "PingPongPlus: Design of an Athletic-Tangible Interface for Computer-Supported Cooperative Play", *Proc. of Human Factors in Computing System (CHI'99)*, Pittsburgh, PA, USA, 1999, pp. 394-401.

[15] Starner, T., Leibe B., Singletary, B., Pair, J., "MIND-WARPING: Towards Creating a Compelling Collaborative Augmented Reality Game", *Proc. of the 5th International Conference on Intelligent User Interfaces (IUI 2000)*, New Orleans, LA, USA, 2000, pp. 256-259.

[16] Piekarski, W., Thomas, B., "ARQuake: The Outdoor Augmented Reality Gaming System", *Communications of the ACM* 45(1), 2002, pp. 36-38.

[17] Cheok, A.D., Goh, K., Liu, W., Farbiz, F., Fong, S., Teo, S., Li, Y., and Yang, X., "Human Pacman: a Mobile, Wide-Area Entertainment System Based on Physical, Social, and Ubiquitous Computing", *Personal and Ubiquitous* Computing 8(2), 2004, pp. 71-81.

[18] Cheok, A.D., Yang, X., Ying, Z.Z., Billinghurst, M., Kato, H., "Touch-Space: Mixed Reality Game Space Based on Ubiquitous, Tangible, and Social Computing", *Personal and Ubiquitous Computing* 6(5/6), 2002, pp. 430-442.

[19] Billinghurst, M., Poupyrev, I., Kato, H., May, R., "Mixing Realities in Shared Space: An Augmented Reality Interface for Collaborative Computing", *Proc. of the IEEE International Conference on Multimedia and Expo (ICME2000)*, New York, NY, USA, 2000

[20] Magerkurth, C., Engelke, T., Memisoglu, M., "Augmenting the Virtual Domain with Physical and Social Elements Towards a Paradigm Shift in Computer Entertainment Technology", *Proc. of Advances of Computer Entertainment 2004 (ACE 2004)*, Singapore, 2004, pp. 163-172.

[21] EyeToy webpage, http://www.eyetoy.com/

[22] Mauldin, M.L., "ChatterBots, TinyMuds, and the Turing Test: Entering the Loebner Prize Competition", *Proc. of the 12th National Conference on Artificial Intelligence*, Vol. 1 (AAAI'94), Seattle, WA, USA, 1994, pp. 16-21.

[23] Laird, J., "It Knows What You're Going To Do: Adding Anticipation to a Quakebot", *Proc. of the 5th International Conference on Autonomous Agents*, Montreal, Canada, 2001, pp. 385-392.

[24] The SIMS webpage, http://thesims.ea.com/

[25] Furby webpage, http://www.furby.com/

[26] Fujita, M., Kitano, H., Kageyama, K., "Reconfigurable physical agents", *Proc. of the 2nd International Conference on Autonomous Agents (Agents'98)*, Minneapolis/St.Paul, MN, USA, 1998, pp. 54–61.

[27] Maes, P., Darrell, T., Blumberg, B., Pentland, A., "The ALIVE System: Wireless, Full-body Interaction with Autonomous Agents", *ACM Multimedia Systems* 5(2), 1997, pp. 105-112.

[28] Anabuki, M., Kakuta, H., Yamamoto, H., Tamura, H., "Welbo: An Embodied Conversational Agent Living in Mixed Reality Space", *Proc. of Human Factors in Computing System (CHI 2000)*, Extended Abstracts, The Hague, The Netherlands, 2000, pp. 10-11.

[29] MacIntyre, B., Bolter, J.D., Vaughan, J., Hannigan, B., Moreno, E., Haas, M., Gandy, M., "Three Angry Men: Dramatizing Point-of-View Using Augmented Reality", *SIGGRAPH 2002 Technical Sketches*, San Antonio, TX, USA, 2002.

[30] Cavazza, M., Charles, F., Mead, S.J., Martin, O., Marichal, X., Nandi, A., "Multimodal acting in mixed reality interactive storytelling", *IEEE Multimedia* 11(3), 2004, pp. 30-39.

[31] Balcisoy, S., Kallmann, M., Torre, R., Fua, P., Thalmann, D., "Interaction Techniques with Virtual Humans in Mixed Environments", *Proc. of International Symposium on Mixed Reality (ISMAR 2001)*, Tokyo, Japan, 2001.

[32] Cassell, J., Ananny, M., Basu, A., Bickmore, T., Chong, P., Mellis, D., Ryokai, K., Smith, J., Vilhjálmsson, H., Yan, H., "Shared Reality: Physical Collaboration with a Virtual Peer", *Proc. of Human Factors in Computing System (CHI 2000)*, The Hague, The Netherlands, 2000, pp. 259-260.

[33] Itoh, Y., Akinobu, S., Ichida, H., Watanabe, R., Kitamura, Y., Kishino, F., "TSU.MI.KI: Stimulating Children's Creativity and Imagination with Interactive Blocks", *Proc. of the 2nd International Conference on Creating, Connecting and Collaborating through Computing*, Kyoto, Japan, 2004, pp. 62-70.

[34] Florence, M., Storey, R., "Vietnam", *Lonely Planet Publications*, 2001, p. 530.

[35] Reynolds, C.W., "Flocks, Herds, and Schools: A Distributed Behavioral Model", *Computer Graphics* 21(4) (Proc. of SIGGRAPH '87), Anaheim, CA, USA, 1987, pp. 25-34.

[36] The Lemmings Compendium webpage, http://lemmings.deinonych.com/

[37] Schmalstieg, D., Fuhrmann, A., Hesina, G., Szalavári, Zs., Encarnação, M., Gervautz, M., Purgathofer, W., "The Studierstube Augmented Reality Project", *PRESENCE - Teleoperators and Virtual Environments*, MIT Press, 2002.

[38] Cal3D project website, http://cal3d.sourceforge.net

[39] MacIntyre, B., Gandy, M., "Prototyping Applications with DART, The Designer's Augmented Reality Toolkit", *Proc. of Software Technology for Augmented Reality Systems Workshop (STARS 2003)*, Tokyo, Japan, 2003.

[40] Gun A. Lee, G.A., Nelles, C., Billinghurst, M., Kim, G.J., Immersive Authoring of Tangible Augmented Reality Applications, *Proc. of IEEE and ACM International Symposium on Mixed and Augmented Reality 2004 (ISMAR'04)*, Arlington, VA, USA, 2004, pp. 172-181.