Integrated Project on Interaction and Presence in Urban Environments

FP6-2004-IST-4-27571

ipcity.eu

Initial Demonstrators of MR Infrastructure Components

Deliverable D5.1

Doc-Id:	**D 5.1**
Version:	**1.0**
Author(s):	**Bernhard Reitinger, Markus Sareika, Antti Juustila, Jan Ohlenburg, Morten Friesgaard Christensen, Valerie Maquil**
Date:	**2007-02-08**
Status:	**Final**
Availability:	**Public**
Distribution:	**Project Partners / EC / Web**

Table of Content

Abstract

Mixed reality (MR) infrastructure is focusing on basic research for mobile devices and their application for urban environments. Mobile settings in this context can vary in scale between light-weight systems such as smart phones or sub-notebooks, and semi-stationary devices such as high-end equipment in the MR tent. Light-weight devices are used for exploring a larger region in the urban environment. The idea of scouting allows a user, who is equipped with a portable device and several sensors (i.e. GPS receiver, camera), to act as a data collector and to deliver required data (e.g. images, videos, sounds) for the urban planning process.

The idea of the MR tent is to give shelter and to provide high-end equipment to a larger audience. The research of WP5 also includes a communication infrastructure which allows the efficient communication between mobile users and the participants in MR tent. Since mobile users must be located within different environments and settings, the fusion of different sensors such as GPS, interia and vision-based tracking for unconstrained environments must be carried out. The following list summarizes the main research objectives of this workpackage:

- Which mobile light-weight Augmented Reality (AR) devices are suitable for outdoor usage?
- How can we support AR on these devices? Which software infrastructure is required?
- How can the technical advantages of the semi-stationary AR installation such as the MR tent and the mobility of wearable AR be combined?
- How can we achieve a high quality location tracking in all areas of the city? Which precision can a computer vision-based tracking approach deliver. How can current computer vision-based tracking approaches be improved regarding robustness in unconstrained environments (i.e. dynamically changing illumination conditions)?
- How can the data be stored in an efficient way in order to support a wide range of devices?

This document summarizes the scientific and technical achievements in work package 5 during the first year. According to the internal deliverable I5.1 which includes a requirements analysis for WP5, first demonstrators could be developed which have already been tested within several technology probes.

Intended Audience

This document is intended to all partners of the project, and to the reviews for the first project’s phase.

1 Workpackage Objectives

Objectives Phase I	Development of support for deploying mixed reality environments in mobile settings. To address the needs of the various emerging showcase applications, this will encompass the investigation of mobile AR devices on various levels, including handheld, body-worn and semi-stationary devices. Display and tracking devices/technologies will be developed that fulfill the showcase requirements. In addition, a suitable communication infrastructure will be built allowing for efficient communication between a set of users at distributed locations, as well as the management of a distributed and heterogeneous collection of tracking sensors.
Results Phase I	The work in the first year focused on developing demonstrators for the following technologies: • Mobile devices based on ultra-mobile PCs or tablet PC-based setups such as a Sony UX • PDA-based devices such as a Dell Axim • Smartphones based on Symbian such as the Nokia N70 • MR tent infrastructure • Outdoor localization • Data storage of various data formats • 3D reconstruction
Evaluation Results Phase I	The presented demonstrators have been tested in the lab, but have also been partially tested as tech probes during different workshops.
Objectives Phase II	Initial tests with first mobile demonstrators gave promising results concerning hardware and software requirements. The initial software components for augmented and mixed reality applications on different mobile devices will continue according to the requirements of the various emerging showcase applications. In this context, sub-notebook but also PDA-based as well as smart phone-based settings seem to be useful for different showcase scenarios. Therefore, we will continue the core development for these kinds of devices. In addition, a persistent collaborative database seems to be inevitable in order to exchange data between various devices. Although the infrastructure for indoor tracking could be provided by a commercial product, outdoor tracking is still an open research field. We will further work on the localization and tracking of outdoor users by fusing different types of tracking modalities such as GPS, inertia and vision-based systems. Specifically, we will work on software and hardware infrastructure for the following issues: ● **MR projection**: a video-augmentation projection

	system for the MR tent, where augmentation on the real-world can be achieved registered in an interactive way. • **Interaction table**: a centralized interaction table will act as the main user interface in the MR tent. The existing demonstrator, a table-top display with tangible interfaces, will further be developed. • **Master interface**: A control interface will be required where most of the provided services (in the MR tent and mobile setups) can be observed and filtered. • **Mobile setups (handheld mixed reality environments)**: three different mobile devices (scaled in computing performance) will further be developed: a sub-notebook-based (UMPC-based) approach, a PDA-based approach, a smartphone-based approach. • **Tracking and localization**: Vision-based tracking and localization will further be developed in order to get more precise positioning for outdoor MR applications. In addition, a first prototype for ubiquitous tracking will be developed which allows a seamless transition of users moving between different tracking services.

2 State-of-the-Art

State-of-the-art merges related work topics regarding work package 5 “Mixed Reality Infrastructure”. The infrastructure covers both fields hard- and software including physical interfaces to the user and Mixed Reality (MR) software frameworks like *Studierstube* [1] and *Morgan* [4]. The focus of these software frameworks enable features such as scalability, platform independence, support of multiple users, distribution of components, and efficient as well as sophisticated rendering. Mixed- and therefore also Augmented Reality naturally complements mobile and ubiquitous computing on wearable devices by providing an intuitive interface to a three-dimensional information space.

2.1 Mobile Hardware

Mobile AR systems [6][50][54][47][49] are usually based on backpack computer units. The idea of Mobile/Handheld AR has previously been explored by several research groups, but with constraining assumptions. Early work [44][45][52][51] used small tethered screens to display location-based information. Some later attempts used analogue [56][53] or digital [46][48] radio links for image transmission. In all cases, position tracking and image generation was done on a server computer, which limits mobility and/or scalability of these systems. The first fully self-contained mobile AR system on a PDA [55] works with an off-the-shelf PocketPC with a camera and computes real time optical tracking and 3D graphics video.

Figure 1: A typical backpack setup (left) for mobile AR and a lightweight handheld device (right).

2.1.1 HMD-based Mobile AR/MR Setups

AR systems aiming at unconstrained mobility, like the Touring Machine [12], MARS [13], or Tinmith [14] have typically emerged as wearable variants of existing desktop setups. Their creators would commonly package a single (notebook) computer with a variety of sensors and peripheral devices in such a way that users can wear it on their backs, with both arms through shoulder straps. Graphical augmentations were usually shown through an optical see-through head-mounted display (HMD). A typical example of this type of setup is depicted in Figure 1.

The current generation of HMD-based mobile AR systems benefits from the advent of powerful light-weight laptops and wearable computers as well as light-weight monoscopic head-worn displays. Mounted on comfortable back- or belt-packs the computers can even be worn by children and don’t impose a significant weight on the user’s back. Display, head tracking devices on the bearing structure are worn on the head rather in hat- than in helmet-style. Examples are developed in the on-going EU projects CONNECT and IPerG.

2.1.2 Handheld-AR: PDA-based, TabletPC-based

The form factor of Handheld-AR is changing rapidly as well as the available processing power on these mobile devices. The first setup (see Figure 2) is based on a mini tablet PC (Sony Vaio U70, 1GHz Pentium-M, Windows XP). This platform combines a regular PC compatible computer with several peripherals into a very compact form (footprint 15x20cm, 1400g including peripherals). The second setup (see Figure 3) is based on the Pocket PC standard for Personal Digital Assistant (PDA) computers (ARM9 CPU currently attaining maximum speeds of 624MHz, Windows CE). While these devices weigh only around the 180g, they still feature a touch screen and a built-in camera [62].

Figure 2: This handheld AR platform is based on a mini tablet PC that is operated as a video-see through "magic lens" device. It currently supports five different tracking technologies (3 shown) and weights less than 1.5kg.

Figure 3: Mixed Reality PDA Application "The Invisible Train" [3]

PDAs present a system architecture for interactive, infrastructure-independent multi-user AR applications and are off-the-shelf handheld devices. Rekimoto used a tethered analog display and a CCD camera in the NaviCam [7] project to track color-coded fiducial markers on a connected workstation. The Batportal [8] by Newman et al. was driven by a remote VNC display server using ultrasonic outside-in tracking, while the AR-PDA project [9] relied on a digital image streaming mechanism in order to outsource machine vision and graphics rendering tasks to an application server. In summary, the inherent infrastructure dependence of thin-client approaches has confined these projects to restricted working volumes, thereby preventing them from evolving into self-sufficient mobile AR systems. The unconstrained, infrastructure-independent AR Framework Studierstube Light [3] runs on lightweight wearable devices "bridging the gap".

2.1.3 Smartphones

Smartphone hardware efficiency is currently about the same as a typical personal computer's of mid 1990's. Current smartphone platforms also include graphical programming interfaces such as limited OpenGL support (e.g. in Symbian Operating System) and Java 3D support even in some midrange phones. These features of smartphones are used mainly for game application development.

Especially the large amounts of data needed for processing visual data (images, video frames) is a problem in smartphones usually having a range of 4-64 megabytes of RAM in the device. Also the small display size put limitations on which kind of mixed reality applications are usable in the platform. Versions of Symbian OS before 9, for example, have a single threaded kernel, which puts limitations on the efficiency of the high CPU intensive applications that use the various system services for image, video and sound processing.

The strength of the smartphone platform in mixed reality applications is that these devices contain many different functionalities that can be used in concert with each other in the development of mixed reality applications. There is no need to combine several distinct devices to access the functionality for creating distributed, multimedia, location based mixed media applications. The devices are connected through GSM, 3G, WLAN and Bluetooth technologies and they include multimedia features such as still and video cameras, video and sound input and output, and they include messaging features that can be combined with the networking and multimedia features in innovative ways to overcome the technological limitations. The vibration feature in addition to the voice output can be used to augment the displaying of visual data by providing the user with tactile feedback. Smartphones as mixed media devices may also be more suitable as terminals in mobile or mixed reality or multimedia services/systems, comprising many other devices and platforms. MobiLenin [59] is an example of a multimedia system that allows a group of people to interact with a multi-track music video shown on a large public display using their personal mobile phones. As the display size is small, an application can focus on showing only relevant and limited data on screen while providing important audio output for the user. The limitations of both the smartphone (small screens, single user use) and large displays (limited interaction) are surpassed by their strengths – several people can interact with the media using their personal mobile terminals, while the media is then displayed using the large public display.

2.2 Display Technology

Display technology is the key for delivering the visual information of MR to human perception. In Figure 4 Bimber and Raskar depicted image-generation possibilities addressing the human visual sense [42]. In Azuma`s survey from 1997 [43] in the field of AR which is a sub-set of MR, several different applications are described which have been explored so far. Characteristics of Augmented Reality systems are described and tradeoffs regarding display methods discussed in detail.

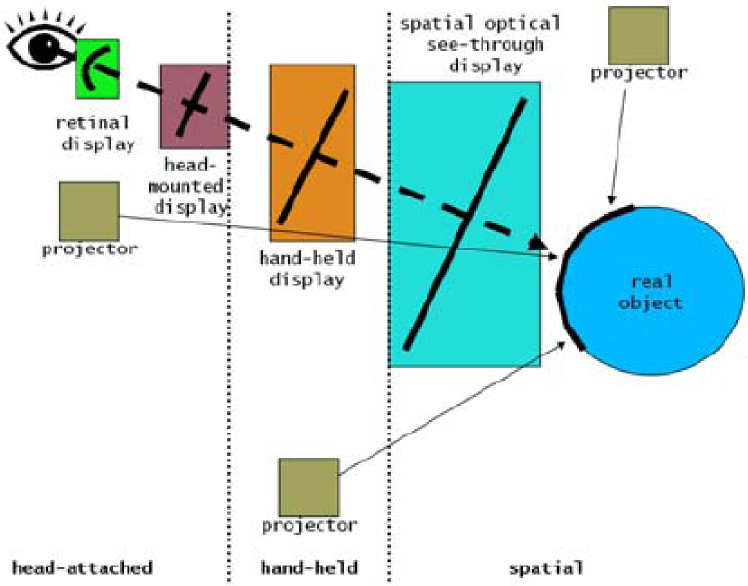

Figure 4: Image-generation for augmented reality displays [50].

2.2.1 Head-Mounted Displays

Head-mounted displays (HMDs) are displays mounted directly to the head of the user (similar to a helmet) displaying the image directly in front of the user's eyes, thus each eye actually has its individual display or screen. This allows for a fixed viewing offset between the displayed image and the user's eye independent of the user's location and head movement. Light-weight HMDs are sometimes called head-worn displays (HWDs) referring to the fact that they are worn similar to glasses rather than mounted like a helmet.

In general we can distinguish four fundamental construction principals for head-mounted displays: mirror-based displays, prism-based displays, holographic displays, and retinal displays. The first HMDs available used CRTs for image generation. Today the actual image is typically displayed using LCD or OLED displays (except for retinal displays).

Mirror-based HMDs represent the classic approach for HMDs: illuminated by a back light the displayed image passes through a diverging lens in order to allow the observer to focus on a display actually too close to the eye. It is then reflected by a semi-transparent mirror and reflected again by a second mirror, passing through the semi-transparent mirror it finally reaches the eye of the observer. Due to the reflection and the passing through the semi-transparent mirror, at least 75% of the original display light does not reach the eye.

Prism-based HMDs use a single curved prism replacing the diverging lens as well as the semi-transparent mirror. The light is reflected at the surface of the prism due to the rather small angle between the light direction and the surface, it is then reflected by a mirror coating at the front of the prism, and finally leaves the prism to the eye of the observer. This approach provides two advantages: the displays are significantly lighter and smaller, and less light is lost (thus the brightness is higher).

The name "holographic displays" for the third type of HMDs is somewhat misleading, as they do not really allow for holographic projections, but rather use holographic optical elements for coupling in and out the image displayed. The only visible optical element of these displays seems to be a flat piece of glass in front of the eye. The LCD is placed on that glass either at the same or the opposite side as the eye, but above or beside the eye. In the glass the holographic optical elements behave like micro-mirrors, reflecting the light in a direction primarily along the main axis of the glass. The light is then reflected by the inner surfaces of the glass until reaching another holographic optical element, which reflects the light that way that it will pass through the surface and thus can reach the observer's eye. This type of HMDs allows for extremely light-weight and unobtrusive displays.

The fourth type of HMDs are retinal displays, which use a modulated laser light to create the image directly on the retina of the observer. Thus they actually are not really “displays”. The laser light is reflected by a mirror or mirroring glass in front of the eye.

Generally displays may be provided for one eye only or for both eyes individually. In the latter case individual images can be displayed to each eye. When using only a single eye display or using the same image for both eyes we talk about monoscopic displays. On the other hand, when providing individual images for each eye we talk about stereoscopic displays. While monoscopic displays are sufficient for annotations, etc. they do not allow for real 3D registration of virtual objects in the environment of the user. Only stereoscopic displays can provide the user with the full information required to register a virtual object in a particular 3D location.

Two general approaches exist to realize Augmented Reality based on HMDs: Optical see-through augmentation and video-see-through augmentation. When using optical see-through augmentation, the displays needs to be able to let the light from the environment of the user pass through the display into his eye, thus at any time the environment is seen similar to looking through sun-glasses. The virtual images displayed by the HMD will then superimpose the real environment view. However, the virtual images will always appear transparent to a certain degree, depending on the relation of the brightness between the (real) background and the virtual image. Thus, it is e.g. not possible to display rather dark virtual objects on a rather bright real background. Those objects will simply appear transparent and by that vanish. This especially applies to virtual shadows, which cannot be displayed using this technique.

When using video-see-through augmentation, the displays must not provide any direct sight of the environment to the user. The environment of the user is captured by a camera mounted to the display. The camera image is then augmented by the virtual objects and displayed in the HMD. Typically the camera image is distorted and adapted to the actual view of the user thus the impression of actually looking through the displays rather than just seeing the displayed image is generated. While this technology allows for displaying dark objects as well as completely blocking real world objects by virtual ones, it also decouples the user from the real world.

Thus the combination of two-eye displays and non-see-through augmentation does not seem to be appropriate for general use in unconstrained outdoor city environments due to potential hazard exposures.

2.2.2 Handheld Displays

Instead of delivering graphical overlays by means of a head-mounted displays, lightweight systems convey information using the magic lens [10], [11] property of camera-equipped portable displays, such as Tablet PCs or keyboard-less personal digital assistants (PDAs) built for pen-computing [3] (also compare previous section). The average display size of smartphones is limited compared to PDAs although the borders of the two seem to decrease in the flow of time. Current graphical capabilities of smartphones are also described in the previous Mobile Hardware section.

2.2.3 Projection-based Displays

New display paradigms exploit large spatially aligned optical elements, such as mirror beam-splitters, transparent screens or holograms, as well as video-projectors in [42]. In the book of Bimber and Raskar especially spatial AR in the context of projection is outlined but also table and handheld projections are discussed.

2.3 Tracking Technology

Tracking is an indispensable requirement for every MR system and is responsible for registering the virtual in the real world. It is also required for the temporal stability of this synchronized space creating mixed reality.

An introduction to motion tracking interesting to experts and novices is given in [40]. Basic techniques of magnetic, ultra sound, radar and GPS based systems for location sensing which define the field of both commerce and research have been discussed in [41]. The exiting era and the potential of sensor fusion as well as ad-hoc location sensing are outlined as future motivation of the field.

2.3.1 Marker-based Tracking

A method for tracking fiducial markers is introduced in [57] and commonly known as ARToolkit. It computes the location which is position and orientation of a square marker in physical space out of an image captured by a camera. A more recent implementation is ARToolkitPlus [58] which is optimized for hardware characteristics of mobile devices and supports the recognition of id encoded markers allowing to distinguish between them.

2.3.2 Computer vision-based Tracking

Tracking of camera motion based on computer vision shows some promising results with robot navigation [16], a principal that seems plausible to fuse with other technologies like GPS [17]. The motion of a camera, so-called ego-motion or visual odometry, can be determined by features, which are extracted from images with use of computer vision. These features may be given by either natural landmarks or fiducial markers, but can be extracted with several methods with each their strength. The procedure of determining ego-motion from visual features can be seen in Figure 5 below. This setup is given with a fixed pair of stereo cameras to be able to determine distances. Alternatively it should be possible to use a single camera and a different approach like uncalibrated stereo [63].

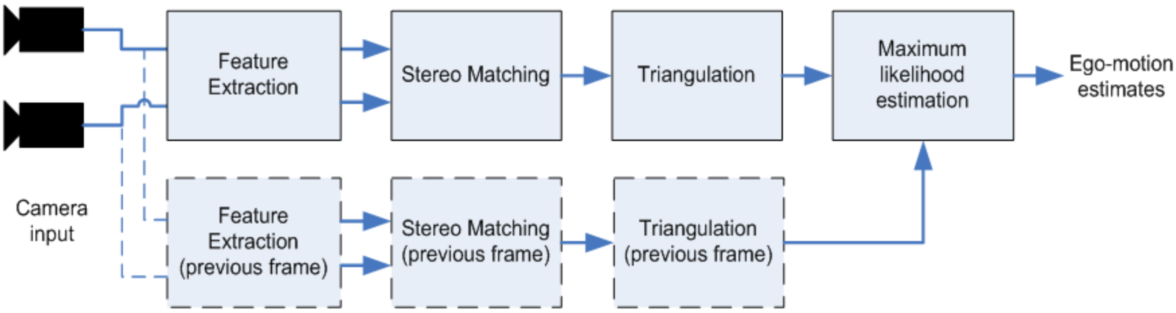

Figure 5: Features are extracted from a stereo camera setup, which are matched in between the stereo cameras. All these matching points can be triangulated into a 3D point-cloud. With use of information from the previous frame, it is possible to do a maximum likelihood estimate of the cameras ego-motion.

2.3.3 Feature-based Tracking

Feature-based tracking algorithms make use of natural landmarks, which are extracted automatically by the computer and can be located anywhere. Some common feature-based methods extract features very differently; SIFT [18] builds a scale space from Difference of Gaussian (DoG) where blobs are extracted and each of these blobs are described with a scale invariant feature transformation. Another method, which is used for image stitching [20], uses Harris Corner detection [23] and makes use of the SIFT descriptor. The KLT-tracker [21] is another method that uses the same principals with image gradients as in Harris corner detection [23].

Extracted features must be described in an invariant way to make these recognisable, e.g. when the camera position, illumination, weather, etc. change. The majority of methods describe features with a certain window size around the feature and calculate an invariant description [18][19][20][21]. A further review of feature descriptors can be found in [22], which concludes that the SIFT descriptor is the best.

Matching or recognition of features can be really slow process, but it is possible to increase speed by indexing features on the behalf of their Haar-wavelet descriptor. By using a 10x10x10 indexing space, matching speed can be increased by a factor of 125 [20].

Classification of many objects (each described by many features) can be obtained using a K-Means clustering technique, spreading features into a tree and using an image/object score to classify images. This method is extremely fast and proved to be useful when many images of the same object used [69].

2.3.4 Mobile Phone-based Tracking

As users are mobile, many applications feature locationing that can then be used to enhance the user experience. Aalto et al. [61] implemented a locationing service based on Bluetooth to experiment mobile advertising systems, as Rashid et al. [60] investigate the different locationing technologies (cell ID, time difference of arrival, enhanced observed time difference of arrival, GPS and assisted GPS) that can additionally be used with cellular phones in gaming. Most recent smartphones also contain wireless LAN, enabling the WLAN based locationing technologies to be used in smartphones too. Furthermore, locationing with mobile phones can be done using RFID tags as well as barcodes and two dimensional barcodes (semacodes). For descriptions of location based data, XML based languages such as GML, POIX and NVML can be used.

2.3.5 Ubiquitous Tracking

The range of AR applications is limited by the scope, range and cost of the used tracking systems. *Ubiquitous Tracking* is an approach to fuse tracking data from diverse heterogeneous tracking systems and to provide transparently tracking data to AR applications. The main problems for realizing ubiquitous tracking are:

Scalability
The tracking system network may consist of several hundred sensors. The ubiquitous tracking middleware has to handle the spatial relationship of the diverse systems.

Migration
Tracking systems are not always stationary setups, e.g. a mobile ARToolkit setup. Both stationary and mobile tracking setups have to been integrated into the ubiquitous tracking middleware.

Bootstrapping
When a user or tracked object enters a new tracking system within the network, an identification process has to be initiated. The changeover to another tracking system has to be transparent for the AR application.

OpenTracker [2] as well as the Morgan framework [4] defines an XML configuration syntax to describe the tracking system setup (see below) and implement a middleware to provide transparent tracking data from various systems.

2.4 Software Infrastructure

2.4.1 The Studierstube Framework

Studierstube [1] extends OpenInventor [64], a scene-graph rendering library, and uses OpenTracker [2], a modular dataflow middleware for pose tracking. Both, Studierstube and OpenTracker make use of ACE [65] for network communication abstraction. Studierstube allows application development in C++ or, more rapidly, via OpenInventor scripts. Alternatively, application developers can choose to use APRIL [15], a high-level descriptive language for authoring story-driven AR presentations. The tracking data and events in Studierstube originate either from its scene-graph or from OpenTracker which serves as easily reconfigurable middleware for interfacing a very broad range of hardware devices and as an abstraction layer for other tracking frameworks. Studierstube supports two different mechanisms for data and event distribution. The acyclic graph traversal builds upon Inventor's event system which uses the "visitor pattern" to distribute events to potentially interested nodes in the scene graph. A visitor traverses the scene graph and calls a previously specified method on each node it encounters, which contains its specific event handling behaviour. This mechanism is important in cases when nodes need to know something about their spatial relations to other nodes (typically along a path), e.g. widgets. The drawback of this mechanism is the temporal responsiveness when traversing the scene-graph. This is suboptimal for near real time tracking which is essential for MR. To account for the drawbacks of this mechanism a second method for the distribution of data and events is available. The generic event bus works with a publish and subscribe mechanism which delivers data and events directly to the subscribers and makes the whole system very responsive. Similarities of the two distribution methods are the same source of events, same event data (reuse event representation) and the use of the same meta-data which is the description of information about events or event sources. The two different subscription and transport mechanisms profit from each other creating a win-win situation in terms of a contemporary MR-framework.

2.4.2 Muddleware

The idea of Muddleware is to provide a general framework for distributed off-line communication of different participants [5]. Data is stored in an XML database which allows the usage of XPath queries. Especially for a multi-user system with multiple clients (e.g. PDAs), data synchronization can be carried out in an efficient way. It also allows the synchronization of heterogeneous systems (e.g. PDAs with a desktop-based system). One main feature is persistency of data. In combination with Studierstube, multiple instances of applications can be synchronized via Muddleware in an effective way.

2.4.3 Morgan Framework

Morgan [4] is a component-based framework that relies on the CORBA middleware for network communication. It currently supports many devices, including mouse and keyboard as well as haptic input devices, object tracking systems and speech recognition libraries. Thus, multi-modal user interfaces can be rapidly developed and evaluated. Additionally, Morgan provides a distributed render engine with automatic scene graph synchronization capabilities. All components are accessible from remote computers.

A broad range of free or commercially available devices and systems have already been integrated into the framework, including object tracking systems from Intersense (IS 600, IS 900, IS 1200, InertiaCube2, InertiaCube3, ARToolkit, IBM ViaVoice, NMEA and Garmin GPS receiver, Stereo HMDs).

All components of the framework can easily be located within the distributed system by a centralized service managing the components. This service is also responsible for remote instantiation of components. The central component of the framework is a render engine that provides high performance rendering capability for the application. Beyond standard functionalities like collision detection, picking and real-time CSG, the render engine has some unique features. The internal scene graph used to store the objects of the scene is designed for efficient rendering, holding only information needed for rendering. All other data of 3D graphics formats, e.g., semantic information, is kept in external scene graphs that provide the mapping of the data onto the internal scene graph. This data can later be mapped back into the source format without loss of semantic information. Initially, the ISO standard VRML'97 is supported by this mechanism.

Distributed multi-user applications are supported by the built-in functionality of the scene graph to automatically synchronize itself with all other scene graphs within the distributed system. An abstraction layer for the frame buffer, e.g., OpenGL or Direct3D, simplifies support for additional frame buffers such as the upcoming Direct3D Mobile.

A viewer realized on top of the framework provides 3D render output for the render engine. Besides keyboard and mouse and head-tracking navigation and interaction capabilities, it offers different output modes, e.g., mono, quad buffered stereo and dual head stereo. They can all be used full screen or in a window, making it possible to display the result on a wide variety of displays, e.g., desktop monitors, stereo projectors, stereo head-mounted displays and virtual glasses. Non-see-through displays, like VR glasses, can be used for AR environments by augmenting a video background provided by a USB or Firewire camera.

2.4.4 OSGART

The Human Interface Technology Lab in New Zealand developed a rapid application prototyping framework – called OSGART - which can be used for generating Mixed Reality applications in a very fast way [67]. This toolkit is based on the OpenSceneGraph [68] libraray and provides a valuable test environment by using ARToolkit [66]. OSGART is implemented in C++ and is portable across multiple platforms.

2.4.5 Alice

Among the tools targeted towards beginners, the Alice system [70] is particularly noteworthy. It was designed as a tool to introduce novice programmers to 3D graphics programming. Alice comes with its own scene editor and an extensive set of scripting commands, but is clearly targeted at an educational setting. For creating “real world” applications, the reusability and modularity of Alice is insufficient. Also, Alice focuses on animation and behavior control of individual objects and does not offer any high-level concepts for application control.

2.4.6 Geist

The Geist project [71] aims at the presentation of historical and cultural information for mobile AR users roaming a city. The Geist engine builds on a detailed analysis of drama theory and interactive storytelling and provides several runtime modules to support applications based on these concepts. Using Prolog, authors can create semiotic functions that drive the story. Virtual characters that are controlled by an expert system demonstrate compelling conversational and emotional behaviour. While this approach is very general and powerful, it can only reveal its full potential in fairly complex applications, incorporating dynamic behaviour of multiple real and virtual actors, and hence requires a correspondingly high effort in content creation. The authors do not provide details of their application examples, making it difficult to assess the final results.

2.4.7 alVRed - Avango

An initial inspiration for the work presented in this paper is alVRed [72], developed at Fraunhofer IMK. The alVRed project is an authoring solution which uses a hierarchical state machine to model the temporal structure of VR applications. In their model, a state represents a scene of the application, while the transitions between states represent changes in the application triggered by user interaction. alVRed provides a runtime engine built on top of the Avango [73] environment for the executing the state machines, as well as a number of editors for supporting various stages of content creation. Particularly interesting is an editor for fine-tuning graphics and animation parameters from within an immersive projection environment. The one area not adequately addressed by alVRed encompasses key AR requirements such as interaction abstraction and multi-user operation.

2.4.8 DART

The Designers Augmented Reality Toolkit (DART) [74] developed at GeorgiaTech is built on commercial software: DART extends Macromedia Director, the premier authoring tool for creating classical screen based multimedia applications. DART allows design students who are already familiar with Director to quickly create compelling AR applications, often using sketches and video based content rather than 3D models as a starting point. Director is an extremely versatile platform used by an extensive community of multimedia developers for a large variety of applications, and these properties are inherited by the DART plug-in. However, DART is ultimately limited by the technical constraints of Director, such as inadequate support for 3D models, stereoscopic rendering, optical see-through displays or multi-user applications.

2.4.9 DWARF

Finally, the Distributed Wearable Augmented Reality Framework (DWARF), developed at Technische Universität München, deserves mentioning. DWARF is a strongly component-oriented middleware, composed of communicating objects. In [75], the authors report on interactive development in “jam sessions”. This expression describes incremental prototyping of a running system by multiple programmers working concurrently. A graphical monitor program allows convenient inspection and remote control of the components. DWARF’s dynamic reconfiguration capabilities allow the developers to replace software components and restart parts of an application on the fly without re-starting the whole system. This is exceptional insofar as it pertains to a distributed multi-user system with full hardware abstraction capabilities rather than a single computer authoring workplace.

2.4.10 AMIRE

The AMIRE framework [76] is component-oriented and designed for creating MR applications and authoring tools. AMIRE components are configurable and communicate via in- and out-slots, comparable to the signal/slot mechanism of the graphics library Qt.

Authoring tools developed with the AMIRE framework are an Authoring Wizard for furniture assembly [77] an Oil Refinery demonstrator and an authoring application for augmenting museum exhibitions [78]. Typically, the result of the authoring process is a XML based application description that is interpreted at runtime. As the same components are used during the authoring process and at runtime, AMIRE allows for desktop based authoring as well as authoring within the MR environment. Typically, the authoring process includes the selection of MR components, the adaptation of the selected components, the specification of the connections between the components and the calibration within the MR environment.

Similar to the Graphics Gems collection that includes useful algorithms for graphics programming MR Gems have been developed within AMIRE that provide solutions to common programming problems in MR applications such as pattern-based object recognition, and path animation among others.

2.4.11 MARS

The MARS (Mobile Augmented Reality System) Authoring Tool [79] uses a timeline to arrange virtual objects and other media objects such as audio, video, images and text. The authoring tool runs on a desktop computer. The author selects media objects and arranges them temporally using the timeline and spatially using a 3D model of the real environment. The virtual content is interconnected with hyperlinks. The author may preview the MR application in a VR mode on the desktop. The content description is stored in a XML-file which is interpreted by the MARS AR presentation tool. The general layout and the timeline-oriented authoring approach are comparable to the multimedia authoring software Macromedia Director.

2.4.12 Atelier Framework

The Atelier infrastructure acts as a mediator between the Atelier components, a component can be simple or a large system by itself. For example, an infrared remote controller device, such as a TV remote controller, with associated component software can be used as a component in the system to control any other component or system in the environment. Components themselves can be combined into applications, larger wholes of functional entities. The infrastructure contains functionality that is needed across components, and it also provides context for requirements that are not necessarily functional (such as the need for location independence).

The infrastructure itself is based on the Microkernel software architecture pattern, and can be expanded by providing new internal or external services. The services are then available to all components, that are connected to the Atelier environment. Examples of Atelier external services are the Hypermedia Database service – for storing hyperlinked multimedia information – and the Email Entrance service – for entering new media into the hypermedia database service by sending e-mail attachments from any kind of internet enabled device.

The main advantage of the infrastructure is flexibility and configurability; it is possible, for example, to replace a positioning (tagging) technology, display or a mobile device with another kind, without losing the interoperability of the Atelier environment. This is feasible as long as the new technology is able to communicate with the Atelier environment using Internet technologies and XML messages. If the technology per se does not have this ability, it is possible to write adapters to enable the connectivity.

This architecture thus allows to build more than just one implementation usable in a specific context, but an environment that is reconfigurable and also extensible in future experiments utilizing different technologies. Because of the requirements for flexibility and extensibility, the specifications and design of the Atelier Infrastructure software has been based on the principles of expandability and abstract interfaces. The system elements communicate by sending XML messages, that are routed by infrastructure Kernel. This mechanism ensures that the elements are efficiently isolated from each other.

The Atelier infrastructure is under active development. Current development initiatives are the additional JXTA protocol module to support peer-to-peer networking in addition to the TCP/IP connected sockets, and a context service to enable social contextual information to be managed centrally for many simultaneous users.

One essential service in the Atelier infrastructure is the hypermedia database. The hypermedia server is run on an HTTP server and it utilizes Java Servlet –technology in order to provide diverse hypermedia applications. The database solution may be any relational database available on the market. By default, MySQL is used. The servers are connected through a high-speed local area network, which enables a secure and fast transfer of large amounts of data. Special feature of the hypermedia system is the possibility to utilize existing multimedia information, which is accessible trough standard URL –accessing scheme. The communication between the client applications and the hypermedia server uses the industrial standard XML –language. Client applications utilize several easy-to-use classes in order to create requests for the hypermedia server. The requests are wrapped into XML –format by infrastructure services and forwarded to the server, which processes them accordingly. Clients may send for example the following requests: create new hyperobject, create a link between two hypermedia objects and get hypermedia object information. XML files in Figure 6 below illustrates the format of the messages, which are sent from client applications to the server.

```
<?xml version="1.0" encoding="ISO-8859-
1"?>
<request>
<create>

<hyperobject>
<name>Oulu University main</name>
<description>
This picture illustrates the main
building of the university of Oulu.
</description>

<multimediaobject>
<content-type>image/jpeg</content-type>
<url>
http://www.server.com/getobject?id=132
</url>
</multimediaobject>

<meta-information>
<city>Oulu</city>
<target>University</target>
<date>12.08.2002</date>
<time>12:30</time>
</meta-information>

</hyperobject>
</create>
</request>
```

A) Sample create hyperobject –request

```
<?xml version="1.0" encoding="ISO-
8859-1"?>
<request>
<get>
<hyperobject id="23422"/>
<hyperobject id="12357"/>
<hyperobject id="2363"/>
<hyperobject id="3498"/>
<hyperobject id="421"/>
</get>
</request>
```

B) Sample get hyperobject –request

Figure 6: XML sample.

2.4.13 CAPNET Middleware for Smartphones

CAPNET [80] is an example of a middleware infrastructure that supports development of ubiquitous applications and services for mobile phones. A prototype is deployed on a mobile phone and on servers in the environment. The environment offers to the user a number of services with tactile physical interfaces using RFID tags. The Symbian OS native C++ implementation on the smartphone has access to all the published functionality in the Symbian OS and is more lightweight has better performance than Java based applications. The server side of the CAPNET architecture is implemented in Java.

2.4.14 Smartphones and Context Aware Presence

The ContextPhone platform. ContextPhone [39] is a generic context-aware platform for Nokia's Series 60 phones. It can provide information on the phone's current location (either based on GSM positioning or information from an external GPS device, complemented with replies to queries from network operator's place-naming service), usage information (i.e., whether the phone is idle or in use, and what is the currently active application), communications (i.e., when and with whom calls have been placed or received, SMS and MMS messages have been, and whether there are any unread messages or unanswered calls), appointments in the phone's calendar, and the Bluetooth environment (i.e., what other Bluetooth-enabled devices are around). All this data is gathered into an internal Blackboard from which this data can be queried and to which subscriptions for change notifications can be made. ContextPhone is a set of native Symbian components, and they cannot be directly accessed, invoked or called from Java MIDlets. To overcome this, we have implemented a small additional component into ContextPhone, a proxy that enables remote changes to the blackboard. The component runs as a local server on the phone and listens to the TCP/IP socket. A MIDlet can connect to this socket and receives a stream of context change events, encoded in XML. From this point on, it is trivial to access context data from the MIDlet. The proxy has now been released as a standard feature in ContextPhone, and can be used by any Java MIDlet (http://www.cs.helsinki.fi/group/context/latest/).

Awareness applications. Mobile awareness systems provide user-controlled and automatic (sensor-derived) cues of other users' situations, and in that way attempt to facilitate group practices and provide opportunities for social interaction. "Cue" refers to a perceptually separable representational entity at the awareness interface that the perceiver can intentionally use in the mental process of social inference of a remote other, for example where the person is, her current activity, availability for communication etc. Very recently, many mobile awareness systems have been introduced, each with different awareness cues and supporting interaction mechanisms. For example, iCAMS2 [24] shows information about locations and proximities of friends integrated to the phone book (see also [25]). AwarePhone [26] integrates automatically fetched location labels to the contact book. Connexus shows information on the use of different terminals (RIM, desktop PC etc.) and location per a contact and tries to facilitate the selection of suitable channel for communication [27]. Friendzone [28] offers a variety of mobile services, one of which is automatic information on friends' proximity (500-1000 m accuracy) and manually entered profiles relating availability to Instant Messaging (IM). A recent study reports three main uses of cues in social interaction in three groups of teenagers: 1) coordination of mobility (e.g., meetings) and of communication (e.g., conversation); 2) expression and discourse; and 3) companionship (e.g., feeling of being together with a remote other). Of these, the role of cues in coordination of activities is the most promising. An important counterpart for the coordinative action on awareness cues involves the opportunities for more informal interactions they provide [29] [30] [31]. The cues can initiate social interactions instead of being facilitators in interactions that have been initiated by other means (e.g., [32]).

Media sharing application. One of the key functionalities of a mobile phone is that it enables people being in touch with each other. Therefore, media creation with a phone can be turned into instant media sharing. Photo blogging systems with mobile uploading are one way to make this possible. For example, MobShare [38] lets users upload images into web albums that only their friends are able to see and comment. Mobile instant messaging systems (e.g., [33], Agile Messenger) are another category. Their benefit is that all the users can be mobile and participate equally in chatting and media sharing. This is true also for Flipper, a system for lightweight photo sharing [34]. However, MMS is the only technology by far that enables sharing multimedia with off-the-shelf mobile phones. Studies on MMS usage show that picture messages can be used e.g. to communicate moods, exchange greetings and jokes, and that their content is very dependent on the context of creation and preceding message exchanges [35]. However, media sharing with MMS is limited essentially to one-to-one messaging, because in the absence of a full recipient list, the recipient can reply to the message's sender only. Therefore, SMS and MMS do not support such patterns of social action that have made instant messaging popular in desktop computing. As was raised in the discussion of collective use, media can be shared also without a wireless network by displaying content from the phone screen to collocated people. This phenomenon of collocated use has been reported as observational accounts on SMS-based text messaging [36] and explained as gift-giving that strengthens mutual trust [37]. The field trial described above showed its importance also in group messaging. There are also media sharing systems that annotate media with contextual metadata (such as picture-taking location and nearby people). In MMM, this was approached by asking the user manually tag the picture with keywords, with an aim to share this effort among the user community [38], whereas in Aware [39] and Merkitys/Meaning (meaning.3xi.org) this is automated and includes e.g. GSM-based locationing and collecting nearby Bluetooth devices. MMM was designed for simplifying management of digital image galleries whereas Aware and Merkitys/Meaning resemble blogging.

2.5 Summary of the State-of-the-art Report

As we can observe, there is a lot of related work in the specific fields of MR and mobile devices, but also smartphone-based devices and multi-media databases. Since part of these technologies is not relevant for the project itself, we summarize the technologies which are interesting for our goals, and will be further developed according to the needs of the showcases. The following table lists the core technologies which are interesting for the project partners.

Hardware Infrastructure

Technology	Related to WP	Relevance (1 low, 5 high)
HMD-based setups	HMD-based setups are mainly addressed by FIT	4
PDA-based setups	PDA-based setups are exploit by FIT, but also partly by TUG	5
UMPC-based and tablet PC-based setups	UMPC-based setups and tablet PC-based devices are used by TUG and FIT as outdoor handheld AR prototypes. First hardware setups will be tested and evaluated within different showcases	5
Smartphone-based setups	Smartphone-based setups are used by HIIT, Oulu, but also partly addressed by TUG in future time	5

Tracking Technology

Technology	Related to WP	Relevance (1 low, 5 high)
Marker-based tracking	Marker-based tracking is currently not in active development for IPCity. However, it may get relevant for HIIT, SNS, and TUW in the next phase	3
CV-based tracking	CV-based tracking is a hot topic and is interested for outdoor tracking of unconstrained environments. Currently AAU is working on CV-based tracking algorithms	5
Mobile-phone-based tracking	Mobile-phone-based tracking is interesting for rough outdoor position estimation. HIIT is currently working on algorithms for mobile-phone-tracking	4
Ubiquitous tracking	The overall project goal is to have a seamless ubiquitous tracking. However, this topic will be addressed in the next project phase	5

Software Infrastructure

Technology	Related to WP	Relevance (1 low, 5 high)
Studierstube framework	The Studierstube framework is extended by TUG but also by TUW	5
Muddleware	Muddleware is developed/extended	4

	by TUG, and used by TUW for IPCity	
Morgan framework	FIT has developed the Morgan framework which is extended for IPCity. A *light* version called *Morgan Light* was developed during the first project phase	5
Atelier framework	The Atelier framework has been developed by Oulu and TUW in previous projects and is now adapted for IPCity issues	5
ContextPhone	The ContextPhone has been developed by HIIT and was extended during the project	4

3 Requirement Analysis

Mixed Reality enhances a user's perception of the real world and creates a 3D information space that lives around the user. The main technological aim of IPCity is to move high-quality MR a step further from labs to real settings. This requires innovation at several levels and therefore going beyond the state of the art. This is achieved by addressing in distinctive ways the various dimensions of presence that have surfaced in research: spatial presence (e.g., perceptual immersion, sense of being there), sensory presence (perceptual realism), engagement (involvement), social presence (co-presence).

The questions catalogue which has been put together by the two research workpackages WP4 and WP5 contains questions related to both and can be found in the Appendix of the Report on the “Basic Tools Requirements” of the Cross-Reality Interaction Tools workpackage. It gives a first orientation on the particular requirements leading to a basis for achieving the objective to support the development of mixed reality environments which are meant to address the needs of the various emerging showcase applications of WP6 to 9.

In WP6, the presence of citizen participants in a particular urban space is mixed with the presence in alternative possible future versions of the same urban space, thus enabling the exploration of using presence for planning and design purposes. The research draws from a number of human, social and design disciplines to understand and design for different aspects of presence. The special “atmospheric” quality of presence in large events is studied in WP7. It helps the spectators to become more actively involved in the events that are set apart from normal daily life and that can extend over days. WP8 aims to explore temporal and social aspects of presence in Mixed Environments by creating mixed reality game spaces between several interlinked cities. In WP9, the presence connected with history, traces of lives of people, and narratives related to specific locations and aimed at mass audience are studied.

The vertically oriented application scenarios add individual strategies to address specific research questions, while the overall project is based on horizontal research topics, building on top of WP5 the Mixed Reality Infrastructure. Besides the question catalogue and the possible Showcase Scenarios the following common denominators can be extracted which serve as a base for the upcoming MR-applications.

3.1 MR Tent

Semi-stationary MR environments are achieved with the MR tent outdoors as required by the filled in question catalogues. They are portable, configurable on-sites and can be moved in actual urban locations instead of being bound to a laboratory as required by the *Urban Renewal* showcase. Furthermore this also allows exploring approaches to mix spectator created content with professional event content in the context of *Large Scale Events*. The MR-Tent provides shelter and gives access to high-end, projection based mixed reality. It is a carefully designed compromise between quality and mobility.

3.1.1 Model Acquisition

In an architect’s daily routine, models are usually manually generated, which means that a CAD program such as AutoCAD, ArchiCAD, etc. is used for constructing a new building. Although these models are very accurate, it is very time consuming especially if the existing environment must also be modelled. For that reason, it is desirable to find algorithms which allow an automated way of building a coarse model of existing buildings which can then be fused with newly generated models.

3D reconstruction algorithms are necessary for achieving this goal. Therefore, the existing surrounding must be captured using a (calibrated) camera setup and data must be collected centrally (in the MR tent) for initiating the reconstruction process. The outcome of this process should contain a coarse (sketchy) model.

3.1.2 Natural Feature Measurements

In some situations it is necessary to measure certain quantities based on images capturing existing buildings. The height or width of buildings as well as the aspect ratio of windows are important information for architects. A measurement toolkit should be developed which operates on 2D image data.

3.1.3 Virtual Workbench and Optical Tracking

Seamless 2D and 3D interaction requires a tracking system which can be used in daylight conditions and is capable of handling reflections. For precise tracking in the context of collaboration, an optical system is required using multiple cameras for handling occlusions. Furthermore an optical system accounts for responsiveness and can be used for handling a large set of input devices used in the MR tent which allow e.g. adding of additional context information or objects such as new trees, new buildings, etc. This setup as an integrated part of the MR tent allows 3D interaction with mixed reality visualization. It has to be evaluated which type of display seems to be best suitable for the MR tent.

3.1.4 Tracked Viewpoint Display

Since head-mounted displays have a lack of collaboration, which means that it cannot be passed around, a new tracked viewport device should be developed. A semi-transparent and moveable display device acts as a window to reality allowing an augmented view outdoors. In order to register the augmentation correctly according to the viewer's position, a tracked frame should be used which calculates the “correct” viewpoint. However, the accuracy depends on the frame's size, and on the distance of virtual objects. Since offsets can occur for this setup, a prototype should deliver first experiences.

3.1.5 MR-Projection Display and Image acquisition

The major requirement for the Tent is the MR visualization for a group of people which can also simply be extended by public displays to address large groups of people in the case of *Large Scale Events*. For image acquisition in order to be able to mix realities and to account for the showcase demands of scalability as well as for a large range of lighting conditions a steerable HDR zoom camera is needed. In addition, a daylight capable projection system is required for displaying MR like fused multiple input sources such as for example views of the tent's surrounding and/or visual feedback of user interaction (see Figure 7). Interaction could either be direct or indirect by using a so-called *Color Table*. The *Color Table* provides a tangible user interface where colored disks can be used for collaborative interaction. The utilized displays must deliver a good contrast and brightness in order to be used within daylight conditions.

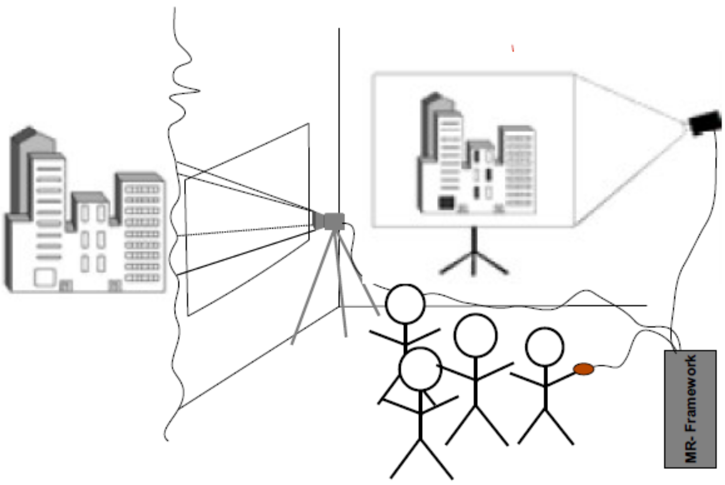

Figure 7: MR-Projection Display and Image acquisition.

3.2 Mobile AR

Mobile AR is typically implemented using wearable computers, head mounted displays, resulting in heavy and complicated equipment. Moreover, the capacity and quality of such systems is limited by the performance of wearable computers and the infrastructure that is available outdoors or in a mobile setting. The demands of the showcases are clear although they concentrate on various device classes and have different categories of end users. A mobile user (“scout”) will either be an expert with high-end mobile equipment providing mobility in the surroundings, or an ordinary citizen, using common devices for the exchange of information in order to communicate or participate in group action. So far mobile MR has only used rudimentary collaboration features for fully mobile users, since it is significantly more difficult to build collaborative applications the requirement in this context is to gain experience with location, size, and other parameters of the different user groups. This comprises the use of network architectures which account for disconnectivity where clients are connected most of the time. Furthermore, best suitable connection types for the individual showcases are required. The use of head mounted devices in a showcase requires carefully selected mini computers optimizing weight, runtime and processing power. This could be either high-end laptops or if feasible PDA class devices. Mobile MR devices require common interfaces and powerful integrated features which depend on the scenario and therefore can be Bluetooth, USB, UMTS, GPRS, GMS, WiFi, Camera, Touch Sensitive Screen, GPS, Compass and other tracking devices. Light-weight Mobile MR requires the dimensioning of the user system optimizing features for application needs while being aware of the form factor. Devices like smartphones would typically be small in size, but it may not be possible yet to integrate all the individual components into a suitable attractive mobile system. That’s why the dimensioning is vital and especially the development and integration of various locating abilities is necessary. As users are mobile, many applications feature locating that enhance user experience.

3.3 Ubiquitous Tracking

The demand for uninterrupted temporally stable tracking with latency below the threshold of perception is on top of the wish list of most researchers in the field of MR. Furthermore the demand also comprises the continuous spatial availability as well as sensor devices which are minimal in physical size allowing them to virtually disappear. The reality is far from achieving this optimum. The requirement which is also reflected by the outcome of the showcase questionnaires in terms of tracking is the fusion of different tracking technologies in order to profit from the merged advantages of the various techniques. Unobstructed ubiquitous tracking is the future demand.

The main problems for realizing ubiquitous tracking are *scalability* where the tracking system network may consist of several hundred sensors and the ubiquitous tracking middleware has to handle the spatial relationship of the diverse systems, *migration* is another issue because tracking systems are not always stationary setups, e.g. a mobile ARToolkit setup. This leads to the demand for integrating both stationary and mobile tracking setups into the ubiquitous tracking middleware. The *bootstrapping* issue is about a user or tracked object entering a new tracking system within the network demanding identification and transparency.

More robust and accurate tracking is achieved by seamlessly fusing several complementary technologies and a further development of computer vision based methods is required.

4 Year 1 Demonstrators

During the first year, different technologies were tested in order to see if a certain technology is suitable for a showcase or not. Based on the initial requirement analysis which is described above, a set of hard- and software demonstrators were developed. After so-called 'technology probes', presented demonstrators were evaluated and a decision was made, which one make sense or not.

We have identified several core technologies (also listed in section 2.4) which should be (further) developed within the IPCity project. There are two different core AR libraries which already exist, but target at different requirements. The Studierstube environment will be used to cover PC-based and UMPC-based AR applications whereas the Morgan framework focuses on PDA-based AR setups (a.k.a. Morgan Light framework). Concerning multimedia data storage, we also use two different technologies. The more elaborate Hypermedia database which allows a rule-based data storage system, and Muddleware which is best used for multi-user prototyping software applications. All of these systems are in use for current demonstrators and will be evaluated according to the requirements of the showcases.

4.1 Overview

We have developed a set of initial demonstrators within the first period of the project. According to the results of phase I, we can group the demonstrator according to the following topics:

- **Mobile devices based on ultra-mobile and tablet PCs**

  We have tested several ultra-mobile and tablet PC-based platforms within possible AR scenarios. Therefore, we headed towards two different kinds of augmentation. On the one hand side a handheld projection and on the other hand side a projection using head-mounted displays. While the latter demonstrators listed below are more hardware demonstrators, the AR scouting expresses the idea of a mobile data collector using outdoor handheld equipment. The three demonstrators concerning mobile AR are called

  - AR scouting, TUG, Section 4.2
  - Mobile AR system, FIT, Section 4.3
  - Tablet-PC-based system, FIT, Section 4.4

- **Interactive 3D reconstruction**

  In combination with the AR scouting idea, we developed a first demonstrator which allows interactive 3D reconstruction based on a sequence of 2D images. Therefore, we used an existing 3D reconstruction engine and developed the interactive pipeline. The developed demonstrator is called

  - Interactive 3D reconstruction, TUG, Section 4.5

- **PDA-based devices**

  In addition to the ultra-mobile devices, we also developed an initial demonstrator for a PDA-based solution. As target platform we used a Dell Axim. The core software framework is provided by the Morgan Light framework. The two demonstrators for this topic are called:

  - Morgan Light infrastructure, FIT, Section 4.11
  - Cal3D XSG, FIT, Section 4.10

- **Symbian-based devices**

As another platform, we did some initial work concerning Symbian-based mobile phones. A very light weight solution for outdoor usage. As a first test device, we are using the S60 mobile phone. The demonstrator for this topic is called:

- Mobile presence scanner, HIIT, Section 4.9

- **MR tent infrastructure**

  In contrast to these three mobile devices, a semi-transparent setup should build an integrated part of the project. Therefore we started the initial concept for the so-called MR tent which should give shelter and can contain a larger variety of technical equipment. The demonstrator for this topic is called:

  - MR tent infrastructure, TUG/TUW, Section 4.12

- **Outdoor localization**

  A challenging goal of our project is to have seamless outdoor tracking for unconstrained environments. Therefore we started with a first prototype which implements fast localization based on 2D image data. The demonstrator for this topic is called:

  - Localization, AAU, Section 4.6

- **Data storage of various data formats**

  Since data storage and data handling plays an important role within collaborative and heterogeneous environments, we developed two different demonstrators which allow the storage and management of multi-media data. The demonstrator for this topic is called:

  - Bluetooth media dispatcher, Oulu, Section 4.7
  - Distributed media entrance and management, Oulu, Section 4.8

4.2 AR Scouting

4.2.1 Description

AR scouting is a newly introduced term in Mixed and Augmented Reality. A scout is a person who is equipped with mobile devices such as an ultra-mobile PC typically with a mounted camera, a GPS receiver, and an optional intertia tracker. Mobile scouting is the idea of going out and collecting necessary data required by a bigger audience. Since the scout is connected wirelessly to a network, the data are delivered on-the-fly and can be processed interactively. Due to an online communication channel, the audience which typically has a birds eye view on the exploring environment can guide the scout to desired locations. The data delivered by the scout can be of various types: images, sounds, videos. By using a sequence of 2D images, even 3D models of the real environment can be captured. For storing all collected data, the multi-media database called *Muddleware* is used where data is stored according to the GPS location (see Section 2.4.2). The AR scout will also be the target platform for the robust outdoor vision-based tracking and/or localization.

Figure 8: AR scouting infrastructure. The left image shows a back-pack system with a Sony Vaio U70. The middle and right image shows a Samsung Q1 with a camera and a GPS receiver.

The first outdoor mobile prototype was tested with three different devices:

- Sony Vaio U70
- Sony Vaio UX90/280
- Samsung Q1

Figure 8 shows the U70 and the Q1 with sensors attached. Both devices have a camera attached and a GPS receiver. The U70 setup consists of an additional Plexiglas back-pack which mounts all required devices. The size of the back-pack is adjusted to the device and is therefore not disturbing. The Q1 mounts the GPS receiver and the camera directly on the device. The Sony UX has a built-in camera and must only attach the GPS receiver.

The user interface is based on the Studierstube framework (see Section 2.4.1). It includes a video background where the live camera stream is displayed. Additionally, a simple system control interface allows simple interaction with the device (e.g. capture the current video stream and send it to a server, or adjusting different parameters).

4.2.2 Specification

Hardware and OS	*The current prototype was tested on the following devices:* • Sony Vaio U70 • Sony UX90/280 • Samsung Q1 *Sensors*: uBlox ANTARIS GPS receiver, Socket BT GPS receiver, HSDPA/UMTS modem The scout application runs under WindowsXP on these devices
Software	Studierstube, OpenTracker, OpenVideo
Core Features	Mobile outdoor AR interface Outdoor data gathering by an expert Platform for outdoor tracking prototypes Client for online 3D reconstruction
Status	Initial prototype which will be further developed and refined
Intended users	Expert user which explores the urban environment; this client can closely be coupled with the 3D reconstruction prototype
Showcases	Currently WP6, also interesting for WP7, but also suitable for WP8 and WP9 as a mobile outdoor client platform
Relevance beyond project	This prototype contributes to the outdoor handheld AR research community. It also contributes to related projects such as WikiVienna or Vidente, both Austrian funded projects.

4.2.3 Testing / Evaluation

We tested different setups for the AR scout technology mainly with the above mentioned handheld devices. For outdoor usage it turned out that the Sony U70 has the best display quality. Since the UX has a glossy display, reflections can occur with direct sunlight. The display of the Q1 is even worse. Hardly anything can be seen if the sun is shining. The UX seems to the target platform, since it comes with an already built-in camera, and a foldable keyboard. The form factor of the UX is also suitable for outdoor applications. In addition, the UX is available with EDGE support for wireless network support.

The user interface is currently redesigned and adjusted to requested outdoor tasks. Similar to the early prototype, the client application will run with the Studierstube environment. Evaluation and first tests within showcases is planned for the next workshops, where experts have to collect different data from the environment.

4.3 Mobile AR System

4.3.1 Description

Our presented mobile AR system consists of a wearable computer, a head-mounted display and tracking devices. This setup allows an overlay of virtual objects onto the physical world. The user perceives the virtual objects as embedded into the real world, leading to a greater immersion due to the worn display.

To avoid heavy and complicated equipment, used hardware has to be carefully selected and optimized according to processing power, weight, runtime, and connectivity.

The first prototypes of mobile outdoor AR systems were tested with different hardware shown in Figure 9.

Figure 9: Shimadzu Data Glass 2 (top left) notebook based mobile AR system (top right), Liteye 750 (bottom left), mobile AR system with ultra mobile PC (bottom right)

No special constructions like a modified baby carrier to carry the hardware is necessary. A special screen was attached to the HMDs to diminish the effect of display transparency.

4.3.2 Specification

Hardware and OS	*Processing units:* • Dell Latitude D420 • Sony Vaio VGN-UX180/280 *HMDs:* • Shimadzu DataGlass 2 • Liteye 750 *Sensors*: Holux GPSlim 240, Holux GPSlim 236 with external antenna, Intersens InertiaCube3, Xsens MTx The Morgan based applications runs under Windows XP on these devices
Software	Morgan AR/VR Framework, Marvin, Cald3D XSG
Core Features	Mobile outdoor AR interface Platform for outdoor tracking prototypes
Status	Stable prototype
Intended users	User which explores the urban environment augmented with rich 3D content.
Showcases	Currently WP8, also interesting for WP6 as a mobile outdoor client platform
Relevance beyond project	This prototype contributes to the mobile AR research community. It also contributes to related projects such as IPerG.

4.3.3 Testing / Evaluation

A comprehensive wearability and performance test is planned in WP8 in Month14.

4.4 Tablet PC-based AR System

4.4.1 Description

A Tablet PC can be used to as an AR system with several sensors or trackers attached, e.g. position or orientation tracker. With a built-in video camera or an USB-webcam mounted to it, the video stream can be used as the video background for the 3D visualization. Using computer vision-based tracking or a positioning device, this feature can be used for video augmentation. Such a scenario is known as the “magic lens”, since the user holds the tablet PC in front of his face and look “through” it to see the augmented reality (similar to the device used for AR scouting). Due to the weight (Fujitsu Siemens 1,5 kg, Panasonic 2 kg) and the display size (10,4’’), regular tablet PCs facilitate a comfortable usage of such an AR system. Furthermore, the CPU (1,2 GHz) and the available on-board graphics card provide enough power to realize an attractive video and 3D content to enable an AR perception and immersion.

Figure 10: Tablet-PC based AR system.

Figure 10 shows a tablet PC-based AR system consisting of a Fujitsu Siemens Tablet PC and a standard commercial webcam for computer vision based tracking.

4.4.2 Specification

Hardware and OS	*The current prototype was tested on the following device:*
	• Tablet PC:
	  o Fujitsu Siemens Stylistic ST5111
	  o Panasonic Toughbook-18
	• Orientation Tracking:
	  o Intersens InertiaCube3
	• GPS:
	  o Holux GPSlim 236 with external antenna
	• Webcams:
	  o Logitech QuickCam for Notebook Pro
	  o Logitech QuickCam for Notebook Deluxe
	The Morgan based applications runs under WindowsXP on this device

Software	Morgan Framework, Marvin, Cald3D XSG
Core Features	Mobile outdoor AR interface Platform for outdoor tracking prototypes
Status	Stable prototype
Intended users	User which explores the urban environment augmented with rich 3D content
Showcases	Perhaps for WP6 as a mobile outdoor client platform
Relevance beyond project	This prototype contributes to the mobile AR research community. It also contributes to related projects such as IPerG.

4.4.3 Testing / Evaluation

With the Tablet PC-based AR system functional tests has been conducted with internal and external users. Further integration testing will be done during February 2007.

4.5 Interactive 3D Reconstruction

4.5.1 Description

During discussions with our project partners (especially with showcase 'urban planning') it turned out that interactive image-based modeling can be an interesting tool for creating 3D content on-the-fly. By using a sequence of 2D images, a 3D representation of the captured object should be calculated. Although for image-based modeling, a large research community exists, only few of them specialized in an automated workflow for generating 3D models. Those groups who provide an automated pipeline for 3D reconstruction, only focus on high-quality off-line reconstruction rather than aiming at on-line reconstruction.

Therefore our idea was to use an existing 3D reconstruction pipeline, which was designed for delivering high-quality models, and embed it into an interactive modeling process. Our goal for 3D reconstruction was to provide an initial 3D representation of the captured scene within a couple of seconds. The input for this modeling pipeline can be delivered by the AR scout's equipment, which consists of a mobile setup with a mounted low-end camera. The following interesting research questions were to find out

- is a low-end camera sufficient for getting an initial 3D model
- what is the quality of the resulting 3D model
- how long it will take to get a 3D model using a low-end camera
- how much user interaction is required
- can a feedback loop be established between the scout and the reconstruction
- what are the limitations of existing vision-based 3D reconstruction algorithms

Since image-based multi-view 3D reconstruction is not the focus of any of our project partners, we got the agreement of the VRVis research group in Graz to use their 3D reconstruction libraries for a first demonstrator (since they are not partner in IPCity, the reconstruction library will not be available for public nor for all other partners in the project). The VRVis group in Graz is working for some years in this research field and is an expert in multi-view 3D reconstruction. By using their software we were able to find answers for the above defined research questions and can decide if the initial demonstrator should further be developed.

This initial demonstrator aims at using the AR scout (presented in the previous section) who is equipped with standard low-cost components. The scout explores the environment and takes several images of a target object. This can either be outdoor buildings, or other objects target for reconstruction. These images are transmitted on-line to a reconstruction engine, where the 3D model is calculated on-the-fly. Therefore, the resulting 3D model can immediately be used, for instance, by another group of urban planners in the MR tent.

The 3D interactive reconstruction system consists of three different components (see Figure 11):

- The AR scout who delivers input data
- A centralized persistent XML database
- The 3D reconstruction engine calculating 3D models

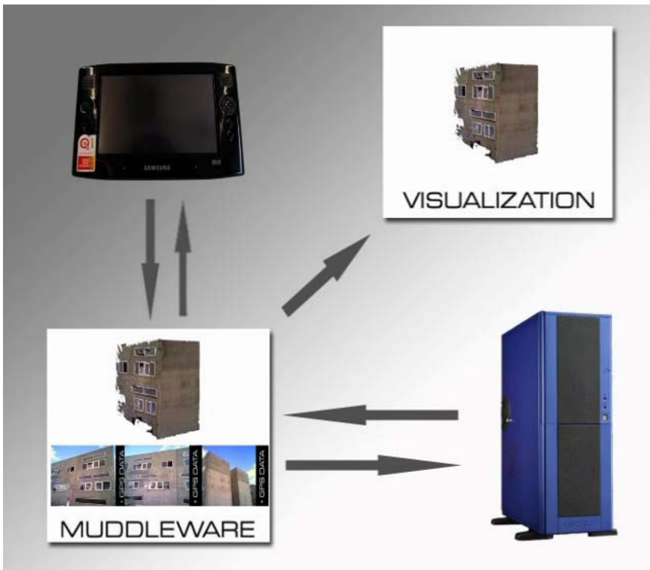

Figure 11: An overview of the online 3D reconstruction system.

The AR scout delivers a sequence of 2D images on-the-fly which are sent to a persistent XML-based database called *Muddleware* which distributes the data to the reconstruction pipeline (see Figure 12). This pipeline processes each image and performs required computer vision algorithms such as feature extraction, correspondence calculation, camera pose estimation and finally dense point cloud generation. All of these steps are performed by the provided VRLib developed by VRVis Graz. The resulting model is again stored in the Muddleware database which can then be visualized by a different application (i.e. showing the result in the MR tent). The details of this process are described in our accepted IEEE VR 2007 application sketch paper (see Section Dissemination).

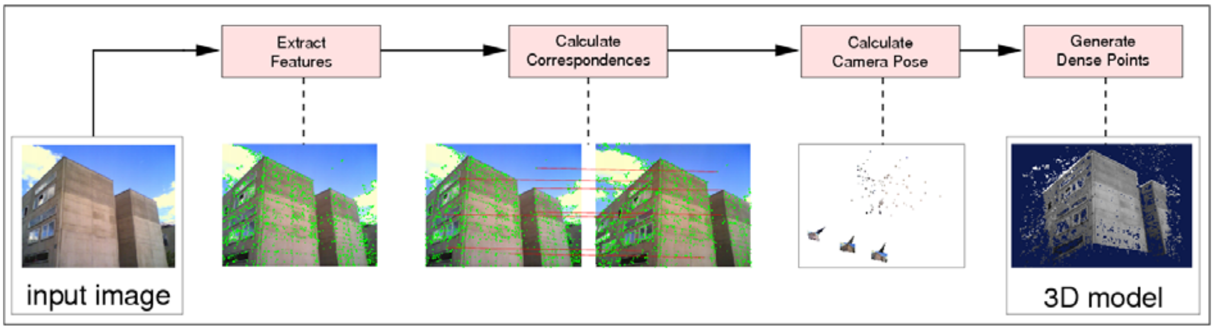

Figure 12: The reconstruction pipeline consists of four main components: feature extraction, correspondence search, camera pose estimation, and dense matching.

4.5.2 Specification

Hardware and OS	The client is the same as for the AR scout (see above). The server is running on a PC with an NVIDIA GeForce 7900
Software	The core vision library uses the VRLib software provided by the VRVis group in Graz. The online pipeline is written from scratch in C++
Core Features	Reconstruct a 3D object from a sequence of 2D images on-the-fly. The output is a textured 3D point cloud.
Status	Initial demonstrator
Intended users	Expert user, similar to AR scouting
Showcases	Mainly WP6, but may also be used for model-based tracking algorithms which is then also relevant for all other showcases
Relevance beyond project	Initial tests for computer vision based 3D reconstruction with low-end cameras

4.5.3 Testing / Evaluation

The online 3D reconstruction has been tested several times for indoor and outdoor scenarios. The first prototype was shown at the Vienna GB16 workshop in October where we have shown the idea by capturing 3D hand-made wooden models (see Figure 13).

Figure 13: The first prototype was shown at the GB16 workshop in Vienna, October 2006.

The local architects and urban renewal team gave promising feedback and stated that this technology can be very beneficial for online 3D content creation for various types of objects. Examples are existing 3D wooden models, 3D outdoor scenes (e.g. statues), or other interior objects.

Figure 14: A demo of the online 3D reconstruction framework reconstructing a face by using a sequence of 2D images.

The next venue was the OpenLab night at our Institute where we tried to capture faces of visitors (see Figure 14). Since faces are not static, it was even harder to get 3D models of the capture image sequence. However, the reconstruction engine was able to reconstruct individual faces and calculated a 3D point cloud.

We also tested the reconstruction framework for outdoor buildings (i.e. our office building at the University). By taking a sequence of several images, we succeeded to reconstruct the building's structure (see Figure 15).

Figure 15: Using the reconstruction engine for calculating outdoor buildings.

Currently, the output of the reconstruction engine is a textured point cloud which is stored in the Inventor file format. The visible clutter in the models has various reasons. Since we use a low-end camera (1.3mpix or even less), the input images are not of the same quality as from a high-end SLR camera. Another issue is that we did not yet tweak the parameters for the reconstruction engine. However, some ideas already exist how to remove the clutter in the final model.

According to the initial research goals/questions, we can draw the following conclusions:

Can a low-end camera be sufficient for getting an initial 3D model?

Yes, according to our initial tests, 3d models of indoor and outdoor objects could be calculated.

What is the quality of the resulting 3D model?

Due to the low resolution and low quality of the camera, some clutter was generated which has to be addressed in the next prototype.

How long it will take to get a 3D model using a low-end camera?

We have not yet performed a formal technical evaluation with exact timings. However, a reconstruction based on 4 images takes about 1 minute overall time depending on the image resolution and feature points used for the reconstruction. However, since it also takes some time to capture the images and the pipeline starts working with the first image, the timing is not critical.

How much user interaction is required?

Currently, the pipeline works without any interaction nor any parameter adjustments. However, by easy interaction on the AR scout application (e.g. setting a rectangle shape in the image masking the region to be reconstructed), the quality of the 3D model may increase. In addition, several required parameters such as the threshold for feature points may also improve the runtime of the reconstruction pipeline.

Can a feedback loop be established between the scout and the reconstruction?

Due to the *Muddleware* server, a direct communication between the reconstruction server and the client can be established. This also allows online parameter setting or visual feedback to the user about the status of the reconstruction.

What are the limitations of existing vision-based 3D reconstruction algorithms?

Since the core reconstruction algorithms are research software, there is a need for overall improvements for all algorithms. Therefore, future collaboration with the VRVis group in Graz may bring improved better results also in the computer vision community.

The 3D reconstruction may also contribute to model-based tracking, where position and orientation is calculated according to a model of the environment. Usually, these models are manually modeled which is a very tedious task. By using the AR scout and the 3D reconstruction engine, one may used this technology to build up an initial coarse model of the environment which can then be used for tracking.

4.6 Localization

4.6.1 Description

Stable tracking for augmented reality in outdoor scenarios is a joint venture between GPS information and computer vision, more specifically feature-based tracking. Feature-based tracking gives ego-motion estimates needed to perform persuasive augmented reality. GPS information on the other hand gives a precise localization until a few centimeters with differential GPS devices, and precision is broadened to a few/couple of meters in an urban environment with buildings. GPS information can therefore be used in ego-motion estimates in collaboration with computer vision tracking; this is "two-fold" where natural landmarks in form of features can be used

- For determining ego-motion; rotated angle and translation of movement in-between images captured from a single or multiple cameras (e.g. mounted on a person's head). This is intended to use fast tracking and no particular recognition.
- For localization or initializing the camera starting location, which could be an initial phase for the tracking describe above.

This demo will focus on the localization, which can be used in a broad context, below is listed a few possible uses:

1. Localization in AR scouting.
2. Virtual tourist guide; gives information about sculptures/buildings throughout an urban environment.
3. Vision-based orienteering; a game to find sculptures/buildings or other significant marks to complete an orienteering.
4. On-site history tutoring; kids could be motivated to walk through a city to find "objects" and get some historical facts, animations and movies about the location they are at.

This first localization demo will demonstrate the principal behind the application with an offline demo, which therefore does not require a person to walk around with a fully equipped setup including a GPS in the city.

Instead a set of images all tagged with GPS information will be used to:

1. Extract features with use of Multi-Scale Oriented Patches (MOPS).
2. These features will coherently be used to form a classification tree.

New images are then processed to extract features, which is successively matched with use of the classification tree. The idea is to narrow the search space with use of GPS, as illustrated in Figure 16. The uncertainty of GPS makes the include images taken within 40-50 meters.

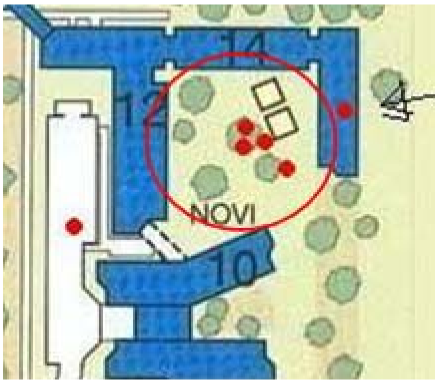

Figure 16: Classification can be narrowed on the basis of GPS information.

The purpose with the first demonstrator shows

- that robust natural landmarks can be extracted from images.
- that new images of know objects can be recognized.
- the advantage of the classification tree.
- that GPS and computer vision can be used in collaboration.

4.6.2 Specification

Hardware and OS	Windows (possibly also Linux)
Software	OpenCV
Core Features	Feature extraction/matching. Vocabulary tree, for classification.
Status	Stable prototype
Intended users	Software is "core" and intended to be used by IPCity members in development of applications.
Showcases	Relevant for all mobile clients
Relevance beyond project	

4.6.3 Testing / Evaluation

Initial test results show a promising potential. Just by using the classification tree, it is possible to obtain classification correctness just above 75 %. Initially the tree was meant to replace a database and eliminate any query time by the database, but since data is arranged in clusters it is possible to obtain a much faster classification (less than 1 second). Results also show that increasing the number of images of an object, increases the rate of success – further testing of the classification will help specify the minimum number of images.

Using GPS information with classification has not been tested yet. However, using GPS should improved classification significantly.

4.7 Bluetooth Media Dispatcher

4.7.1 Description

The idea of the *Bluetooth Media Dispatcher* is to enable entering media into the IPCity systems, to be used in any other context and application. As the demonstrator “Distributed Media Entrance and Management” is used to gather media by using email in a centralized manner, this demonstrator is more directed to using mobile devices in distributed settings. The media is usually gathered using Bluetooth™ enabled mobile devices (for example, using MMS as the WP4 MMS Media Extractor does) and then sent over Bluetooth™ using the OBEX protocol to the Bluetooth Media Dispatcher. This dispatcher can then be located in many places and users can send the collected media to the dispatcher where ever they are located.

The sent media (text, images, videos, sound files, etc) can contain additional metadata, e.g. user, date/time, location etc. Received media can then be used for various purposes. The concrete plan is to use the Bluetooth Media Dispatcher with the WP7 scenarios Forage, and generally in any other showcase in IPCity where generation and entry of user generated media with mobile or smart phones is desirable.

Technically, the Bluetooth Media Dispatcher is using OBEX protocol over Bluetooth™ to listen to connections and to receive the metadata and associated media files. These files are then saved into a Virtual File System. The benefit of using a virtual file system is that we can create our own virtual file system extensions that are highly configurable. For example a virtual file system implementation can actually post the files (as they are “saved”) into a web server. Also, we can implement a virtual file system that actually is an interface to our Hypermedia Database system already in use in IPCity, thus reusing already existing implementations of handling and managing media in IPCity. In WP7, technologies related to a scenario called Alloy will include a “media hub” which will be one of the components where the Bluetooth Media Dispatcher will store the data into. The workflow is sketched in Figure 17 (with an example on the left side where the media is provided by the WP4 MMS Media Extractor component).

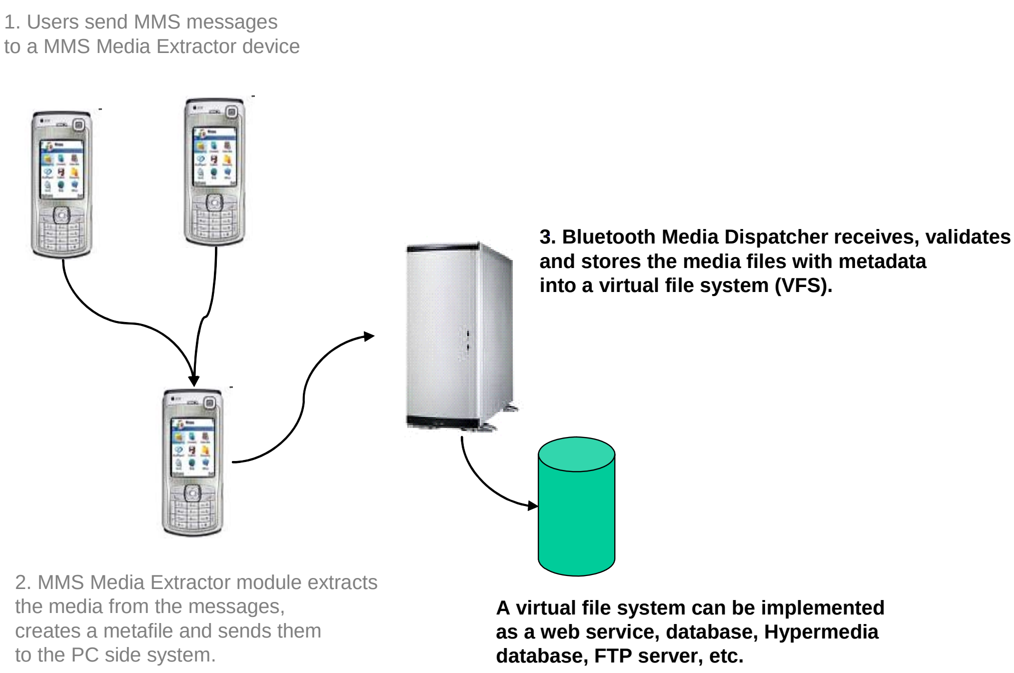

Figure 17: Bluetooth media dispatcher elements (right side).

4.7.2 Specification

Hardware and OS	PC hardware, Linux
Software	• Modified Avetana open source Bluetooth™ OBEX implementation • BlueZ Bluetooth™ stack • Component implemented in Java
Core Features	• Imports media into any storage supporting the VFS interface using mobile devices
Status	• Beta prototype
Intended users	• Any number of users
Showcases	• WP7, others
Relevance beyond project	• Is reusable to any similar purpose (entering media using Bluetooth)

4.7.3 Testing / Evaluation

Component has been tested in laboratory settings, and the core features (receiving the media over BT/OBEX, saving files to VFS) work already. System needs more integration testing and stress testing under heavy loads.

4.8 Distributed Media Entrance and Management

4.8.1 Description

This demonstration gathers together a framework and several components which all relate to an important feature in the IPCity project – how to enable users to enter media (images, sound, video, text, etc) and use the entered media. Where this kind of functionality could be needed are, for example,

- The City tales showcase where inhabitants of the city can leave their tales of a certain place in the city to other to listen.
- The Urban renewal showcase, where people participating in the design process can view and comment media created by designers and other stakeholders of the design process.
- The large scale events showcase, where participants of the event can create their own representation of their experience in the event, and view content created by other people.

Technically, the demonstration consists of

- A framework for enabling creation of distributed software and hardware components (the Atelier infrastructure)
- A service built on top of this infrastructure (the eMailEntrance)
- Hypermedia Database (HMDB) for managing hyperlinked media enhanced with metadata, and a component
- Components for importing media into the system – PT-6 remote automatic cameras, camera phones and email clients (on various devices).

Atelier infrastructure is a distributed framework implemented with Java for building networked systems of components that communicate by sending XML based messages. The framework consists of a Kernel, where components register themselves and send the messages to, and a set of components and services. Kernel has a registry of components in the system, and routes messages between the components – the components themselves do not need to know where the other components are and how to communicate with them. As the components register to the system, they indicate which kind of messages they produce and consume. A component can then receive and handle any kind of messages by indicating to the Kernel which kind of messages it is interested of.

Using this framework it is possible to build systems where the allocation of services and functionality is no longer static and fixed. Applications (consisting of these components and services) can then be distributed over the network, even globally. User interface of an application can be taken apart of the logic and data storage; even the input and output of an application can be separated. Currently the frameworks supports communication over TCP/IP, and in the future support for technologies such as JXTA and Rendezvous can be added.

EMailEntrance is an Atelier Infrastructure External service (which extends the core of the system, the Kernel) for entering media into the IPCity systems. The first prototype of the system was build during the IST funded Atelier project. During fall 2006, the prototype has been further tested and developed, improving the functionality and reliability of the system. The service periodically reads an email inbox and extracts the text and attachments (media files) of a message and then stores these as elements in the Hypermedia database. The message headers are parsed for metadata (date, time, sender) which is stored with the elements into the database. The message can also contain tags which indicate user written metadata (e.g. weather, mood) which also is stored.

Hypermedia database (HMDB) is build on a relational database and enhances the concept of relational database with metadata and hyperlinks. Metadata is stored in relation to the stored hyper-elements as key/value pairs. These key/value pairs enable the HMDB to support storage of contextual data which cannot be in beforehand designed into the database system. Elements in the HMDB database can further be hyperlinked – a certain part of a picture can be linked to a certain set of frames in a video. When user clicks the anchor in the picture, the certain part of the video clip is then played – a timed hyperlink. HMDB saves the actual media files in a web server using WebDAV, so the saved media files can be accessed either using the HMDB API or via the normal web server using HTTP.

The system has been used with input components like a usual email client on a laptop and PDA, a smartphone email application and Nokia PT-6 surveillance cameras to test the reliability of the long term functionality of the system, and with high end digital cameras to test how the system copes with large image files. Users are able to send messages to a plain email address, whereas then the eMailEntrance reads the mail contents and builds hype-elements of these in the HMDB. Atelier infrastructure is used as a framework for the eMailEntrance and uses the infrastructure for service discovery (finding the HMDB in the network). Also a GUI slideshow component has been built during Fall 2006 for viewing the image files entered through the EMail Entrance.

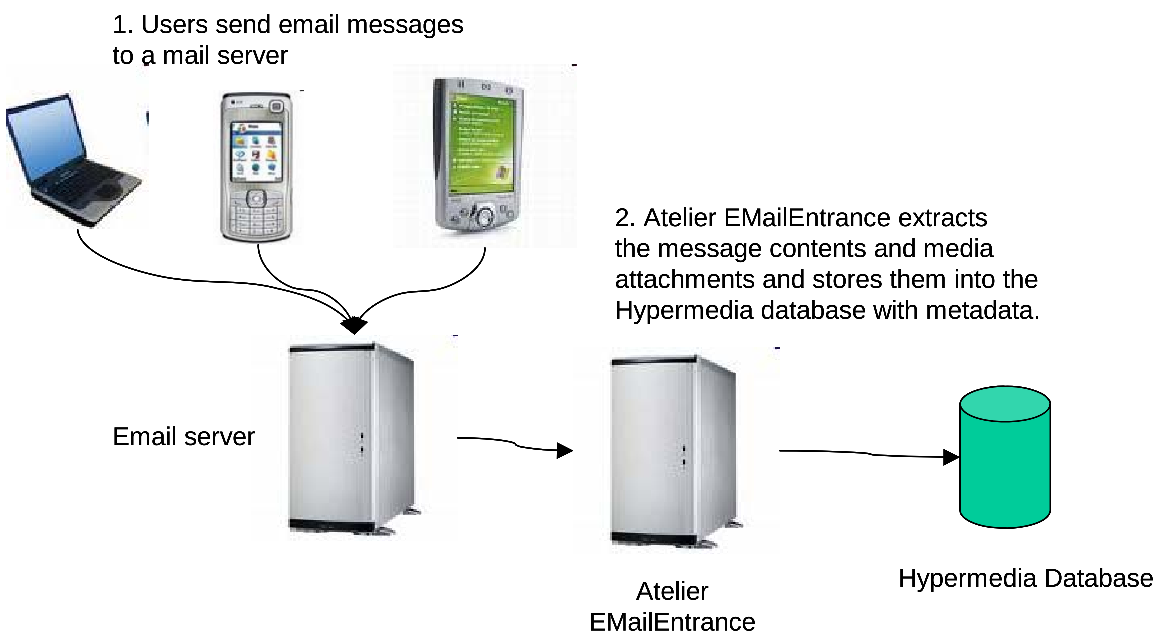

Figure 18: Media Entrance and Management overview.

The EMail Entrance has similar motivation to the Bluetooth media dispatcher – to enable users entering media to be used in other contexts and applications later (see Figure 18). The main differences are:

- Bluetooth media dispatcher works only with MMS messaging (mobile phones), as EMail entrance can be used with any device (mobile or not) capable of sending EMail.
- MMS gateways and phones shrink the size and quality of sent media (videos and images), which is not acceptable in all contexts. With EMail entrance, this does not happen.
- Bluetooth media dispatcher does not currently use the Atelier infrastructure nor the HMDB. If feasible, both integration tasks can be done in the future.

4.8.2 Specification

Hardware and OS	PC hardware, Linux, MySQL Camera phones, Surveillance cameras (with smtp support)
Software	• Atelier infrastructure • Atelier Hypermedia database for storing the media and related metadata • EMailEntrance Atelier External service All these are implemented with Java
Core Features	• Imports media into the Atelier Hypermedia database through email messages
Status	• Stable prototype
Intended users	• Any number of users
Showcases	• WP6/7/9
Relevance beyond project	• Is reusable to any similar purpose (entering media using email)

4.8.3 Testing / Evaluation

System has been tested in fall 2006 for reliability and bug fixing, also some new functionalities have been developed (a GUI for showing a slideshow of the images).

4.9 Mobile Presence Scanner

4.9.1 Description

Mobile presence scanner is a Symbian/C++ component for gathering presence information using a S60 mobile phone. The purpose of this component is to simplify the task of writing presence-enabled applications in the IPCity showcases. It is implemented as a stand-alone component to enable easier software re-use.

The current features of this component are:

- *Bluetooth scan*: Gets the Bluetooth MAC addresses and -names from nearby devices.
- *Cell tracker*: Tracks the GSM cell the phone is in.
- *Profile tracker*: Monitors which phone (normal, silent, outdoors, etc.) profile is being used.
- *File system monitor*: Notifies when new media files are captured with the mobile phone.

The planned features that will be implemented as needed are:

- *GPS location*: Monitor the GPS longitude/latitude information, using an auxiliary device
- *Calendar tracking*: Check the information in the phone calendar.

Figure 19: Overview of the mobile presence scanner.

The application layer can use this information to support presence-awareness. The collected information can provide presence cues about the person using the phone and about the environment of the phone. For example in the Forage scenario (WP7) some users could be given special phone software for media capture. The media capture application can record the presence information which can be used in the Forage wall as the media is displayed.

The technology used in this component is partially taken from the ContextPhone-application, adapting the source code (and writing new code) into a more polished package. The communication of the Presence Scanner is sketched in Figure 19.

4.9.2 Specification

Hardware and OS	• Symbian/S60 2nd edition mobile phones. The current main model is Nokia N70.
Software	• Symbian C++
Core Features	• Scan presence information using the phone: Nearby Bluetooth devices, phone profile, file system changes, GSM cell information • Planned features: GPS locationing, Calendar monitoring
Status	• Beta software
Intended users	• Any number of users, depends on the application
Showcases	• WP7
Relevance beyond project	• The component is useful for smart-phone applications that need to access presence information.

4.9.3 Testing / Evaluation

All code is tested as it built. To guarantee stability more thorough testing is needed before using the components in field trials.

4.10 Cal3D XSG

4.10.1 Description

Cal3D is an open-source animation library for 3D characters. The Cal3D XSG is an integration of this animation library into the Morgan AR/VR Framework through an external scene graph (XSG). Since it is implemented using the plugin mechanism of Morgan, it can extend the functionality of each Morgan based application demanding character animations. Due to available exporters for Blender, 3D Studio MAX and Milkshape 3D character animations can be created using those modeling tools and later loaded into the Morgan Viewer. Morgan XSGs are able to inline other XSG formats and therefore Cal3D characters can be inlined (and transformed) by X3D files.

The Cal3D XSG loads the skeleton file together with a list of meshes, animations and materials and maps them onto a special render scene graph node called Cal3DInstance. This node takes care of updating and the rendering of the character (see Figure 20).

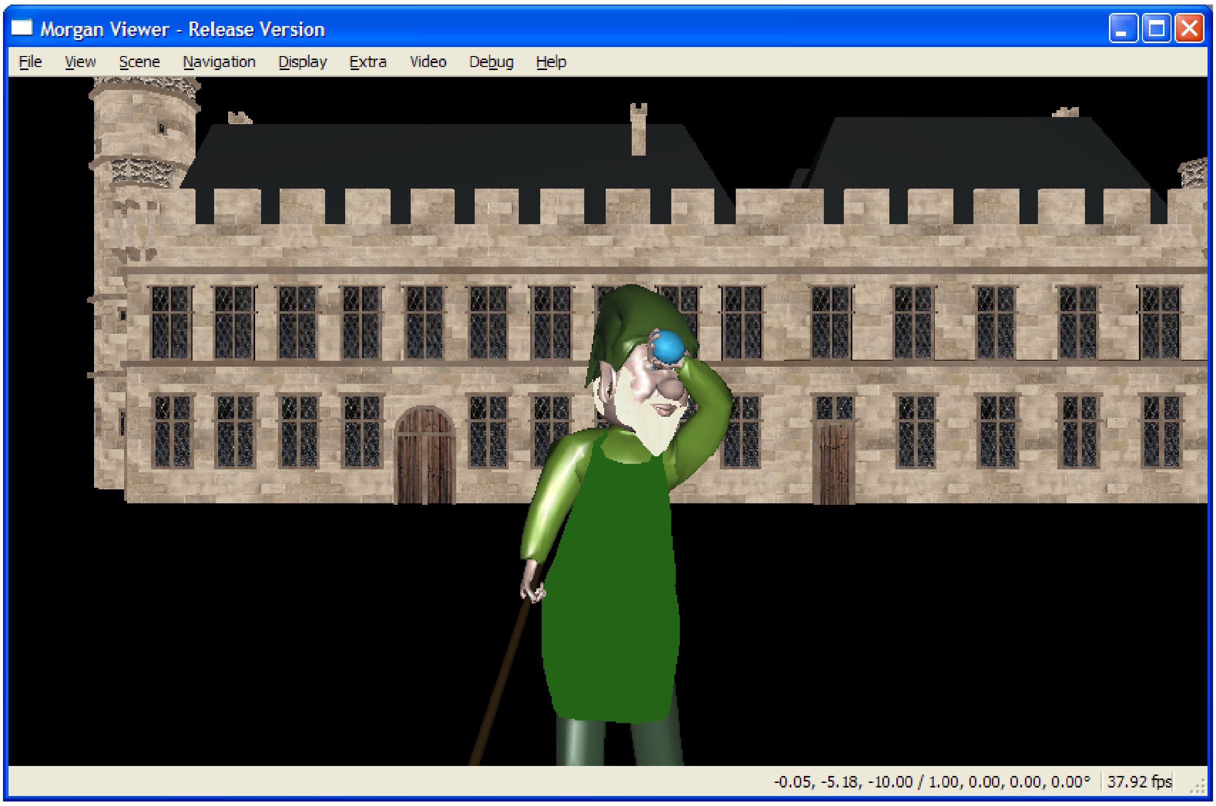

Figure 20: Screenshot of a Cal3D character. WP 8 tech probe: The heinzelman in front of the Cologne Stapelhaus.

The Cal3D XSG is currently available for the platforms Windows XP and Windows Mobile 5.0.

4.10.2 Specification

Hardware and OS	Windows XP and Windows Mobile 5.0
Software	Cal3D library, Morgan
Core Features	• Loading Cal3D characters including meshes, animations and materials. • Blending of animations • Rendering of characters
Status	stable prototype
Intended users	Showcase developer
Showcases	WP 8, possibly WP 9
Relevance beyond project	Character animations are a core feature of all virtual environments and therefore will be used outside of the project as well.

4.10.3 Testing / Evaluation

The implementation has been tested with a number of different characters and WP 8 is already using Cal3D character animations for their tech probes.

4.11 Morgan Light Infrastructure

4.11.1 Description

Morgan Light is a lightweight version of the Morgan AR/VR Framework especially designed for handheld devices, such as a PDA or a Smartphone. Handheld devices have in comparison to sub-notebooks, notebooks, and desktop PC much less processing power and main memory available and therefore are not able to provide the same high quality 3D content to their users. On the other hand, there is a need for 3D content on a handheld device for every day use and users participating in distributed mixed reality environments with them.

Figure 21 - Screenshot of Morgan Light running on a Dell Axim x51v

Figure 22 - Screenshot of an animated IndexedFaceSet

The current version of Morgan Light is a first prototype and provides initial mixed reality support on handheld devices, e.g. the Dell Axim x51v. The abstract framebuffer of the Morgan AR/VR Framework has been implemented for OpenGL ES and some components have been ported to Windows Mobile 5.0 for the realization of this prototype. X3D files (see Figure 22), OBJ Wavefront files (see Figure 21) and Cal3D characters are already supported by Morgan Light. Since on Windows Mobile 5.0 handheld devices no useful Open File dialog is provided by the operating system, effort has also been put into writing one for Morgan Light. A pen-based arc-ball rotation navigation mode is provided for the user and by using the cursor keys, the user can pan in the x-y plane.

4.11.2 Specification

Hardware and OS	Dell Axim x51v, Windows Mobile 5.0
Software	Morgan Light, libpng, Cal3D
Core Features	• PNG image support • X3D XSG • Cal3D XSG • Wavefront OBJ file support • OpenGL ES Framebuffer • Open File dialog • Pen-based arc-ball navigation • Fullscreen display mode
Status	beta prototype
Intended users	Showcase developer, End-users
Showcases	WP8, WP9
Relevance beyond project	Mixed reality environments in mobile settings is a current research question and will be used in other application areas as well.

4.11.3 Testing / Evaluation

Due to the early stage of the prototype, there only some functional testing and some performance measurements have been performed.

4.12 MR Tent Infrastructure

4.12.1 Description

The design of the MR tent was discussed several times in the first year. Mathis Osterhage prepared first design studies according to the requirements of the WP6 partners. The key requirements can be summarized as follows:

- various degrees of openness and flexibility of enclosure
- numerous possibilities for spatial arrangement of enclosure possibility for adaptations to local context)
- space for movement within the spatial set-up
- strong relationship to surrounding environment
- one should feel outside rather than inside an enclosure
- optional extendibility of enclosure
- multiple surfaces for projection
- multiple possibilities for projection setups
- installation of projection screens detached from base-enclosure
- different material possibilities for projection screens/ frame material
- shading possibilities to avoid glare on projection- and computer screens experiments don’t necessarily need to be conducted in mid-day sun)
- structure to withstand light winds and rains, experiments to be normally conducted during “good” weather
- space with a prolific working atmosphere
- space for about 5 people to conduct the experiment
- space for visitors/ audience/ other participants
- space for visitors and spectators outside the tent
- setup duration of enclosure 2 days max.
- electronic equipment to be dismantled at the end of the day
- easy transport and quick installation of tent
- pre-assembled projection frames to shorten duration of installation

According to these requirements, Mathis Osterhage came up with a first sketch shown in Figure 23. The main tent consists of a frame, but the side panels are foldable. This means, that according to the requirements, the openness and closing can be steered.

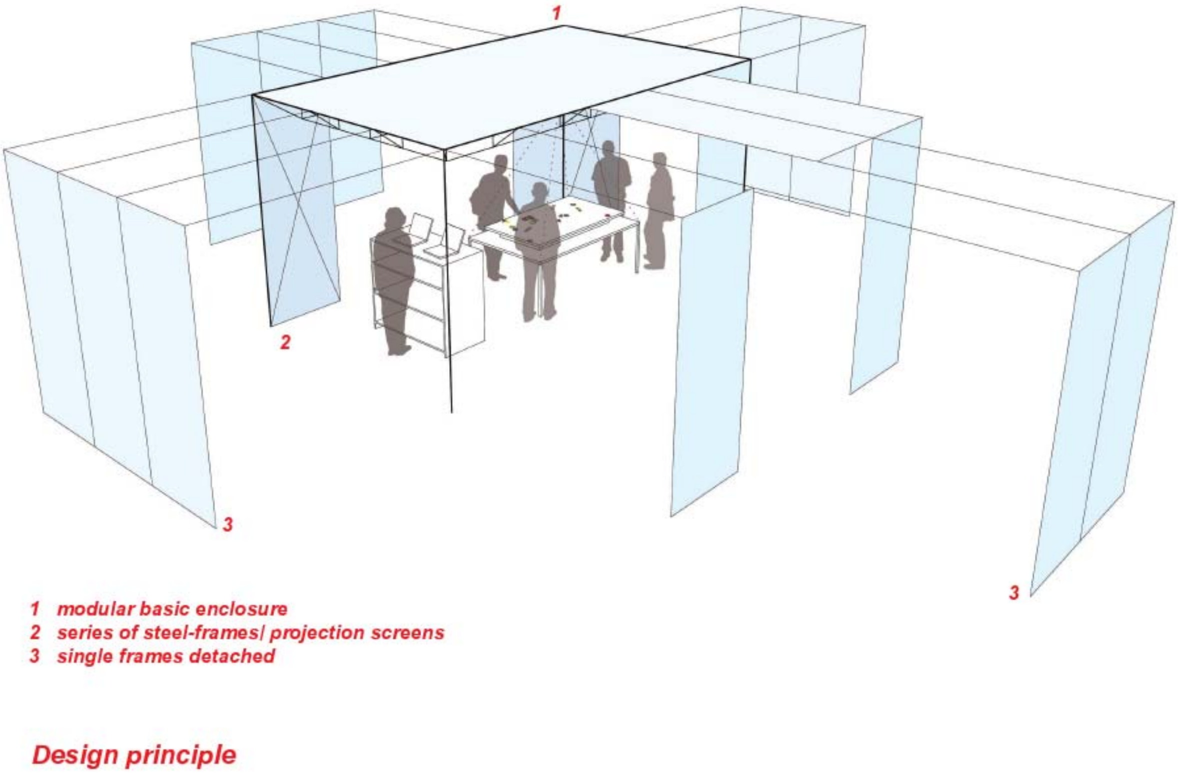

Figure 23: Sketch of the MR tent design.

4.12.2 Hardware for MR tent

The tent hardware was carefully selected going beyond the state of the art in order to meet the requirements and to allow the technological aim of IPCity to move high-quality MR a step further from labs to real settings.

Controllable HDR Zoom Camera

The Pan Tilt Unit (PTU) in combination with the controllable HDR Zoom Camera (HDRCam) was acquired to meet the demand for a "Controllable HDR Zoom Camera" and is the hardware basis for video-augmentation. The implementation as an OpenTracker node integrates this device into the infrastructure and therefore provides an interface for interaction and in combination with other devices forms a key component for interactive video augmentation applications like the Urban-Sketcher (WP6). The setup of the steerable camera is shown in Figure 24.

Figure 24: HD-zoom camera mounted on the pan tilt unit.

High Brightness Projector

The projector intended for video augmentation has a high light intensity so it can be used in an outdoor environment where varying daylight situations occur. A first daylight critical setup shows the projector in Figure 25. In this indoor situation the screen was situated right between two windows and proved to give satisfactory results in terms of video augmentation.

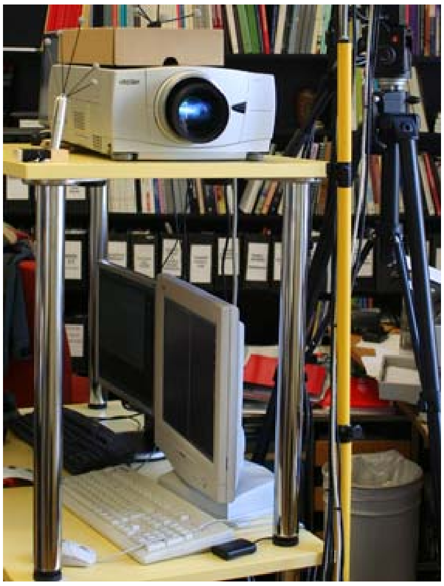

Figure 25: High ANSI lumens projector.

ART Tracking

Optical daylight capable tracking is provided by the ART tracking system, as required for precise and collaborative interaction. This tracking system is integrated into OpenTracker and therefore allows interfacing multiple devices to MR frameworks.

Figure 26: ART tracking system (two camera setup) for precise optical indoor tracking.

The system for the MR tent consists of four cameras for tracking data acquisition but can also be run with two cameras in various configurations giving space for a wide range of scenarios and setups. Figure 26 shows a typical mock-up in the context of urban planning where a pointing device is tracked. The ART tracking system supports the simultaneous tracking of multiple interaction devices.

Interaction Table

The interaction table (also called *ColorTable*) is a tangible user interface that enables users to interact with different mixed reality applications in a collaborative manner.

The system consists of a white surface placed on a conventional table and a large amount of colored objects of different shape and size that may be placed and manipulated on the table. The users may thus stand around the table and interact within the scenario simultaneously from different positions.

A video camera is placed above the table and captures the current state of the system (projector-camera system). It tracks the positions, colors and sizes of the different colored shapes placed on the table.

Two projections allow to reflect the current state of the system. A horizontal one is situated directly on the table and serves therefore as augmentation of the tangible table. It is mainly used to provide additional information such as feedback to the user. A vertical projection is placed in front of the users and shows a visualization of the situation the users are creating. This situation represents a 2D or 3D mixed reality scene with visual and audio elements. The different components of the *ColorTable* are shown in Figure 27.

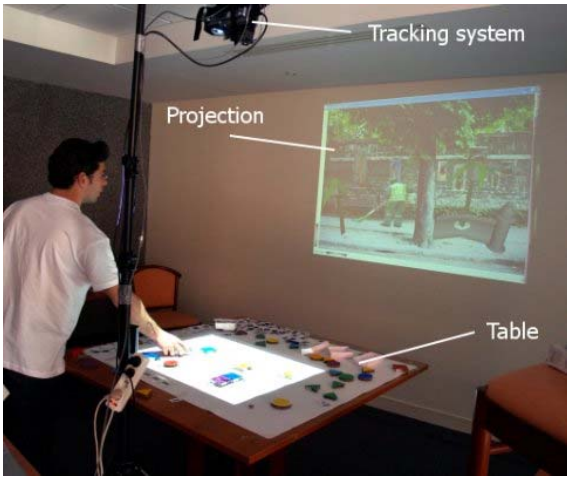

Figure 27: Different components of the ColorTable. Left an overview of the installation. Right the vision-based tracking system.

The coloured shapes are linked to various types of digital content and may be manipulated by changing absolute and relative attributes of the physical objects. The positions of the coloured shapes on the table are directly mapped to the positions of digital elements in the MR scene. Small, green triangles can be attached to cylinders in order to manipulate the orientation of 3D models.

An extension to the ColorTable is the rotating ColorTable which enables users to change progressively the orientation of the viewpoint in a three dimensional mixed reality scene. The coloured objects each represent a certain visual digital content as a 3D model or a 2D billboard. All these virtual elements are positioned in a real environment, captured on a 360 degrees panorama image.

Figure 28 shows the construction and setup of the rotating ColorTable. The rotating infrastructure needs to be placed on a conventional table and is composed of a part that remains fix, and a part that may be rotated. The surface of the table is enriched by a white, rotating disk serving as underground for the coloured objects. The viewpoint is positioned in the centre of the disk and oriented into the direction of the vertical projection. To change the orientation of the viewpoint, the user rotates the disk, and provokes the rotation of the whole scene around the viewpoint.

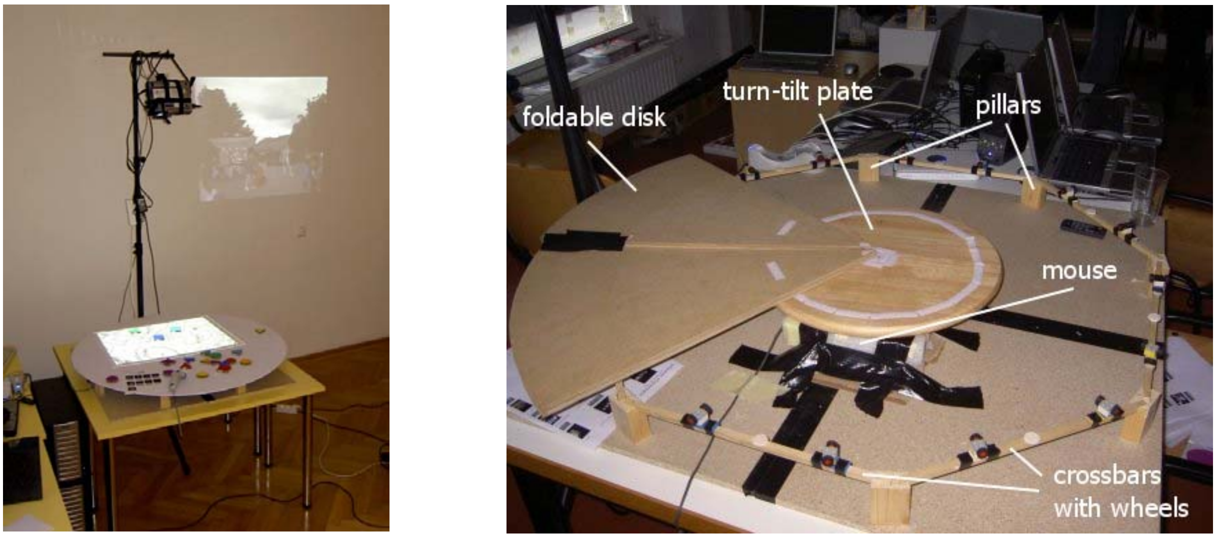

Figure 28: An extension to the ColorTable: the rotating ColorTable. The left image shows the mock-up for its components.

The part that remains fix serves as underground for the rotating disk. Eight pillars and matching crossbars with wheels are mounted on a board lying on the table. A turn-tilt plate is fixed in the centre of the board, ensuring the crossover from the fix part to the rotating part. An optical computer mouse is placed upside down under this plate, in order to track the relative angle of rotation. To finish, a foldable, circular disk, covered with white, opaque cloth is placed upon the turn-tilt plate and the pillars. Figure 28 (right) shows the different elements of this rotating infrastructure.

Projection Displays

The tent provides different projection walls each one for specific applications. We have performed several tests and investigations concerning projection technologies and how each one is applicable for the MR tent installation. We can categorize the display types into two different technologies:

- Optical see-through display
- Video see-through display

Optical See-Through Display

TUW is working on different versions of optical see-through displays. They tried different materials for projecting the virtual objects onto a semi-transparent material mounted on a window, in order to augment the environment. In experiments they tried transparent cloth but also a metal grid in order to augment virtual objects into the environment. Details about these experiments can be obtained in the WP6 report. The main problem they are facing is the parallax. Another issue is focus. In an environment with large depth (e.g. a street), physically it is not possible to focus on the projected object and on the environment at the same time. However, the optical see-through installation may deliver interesting presence research results. Figure 29 shows a first test installation of an MR tent with very first experiments of optical see-through displays.

Figure 29: A test prototype of the MR tent with optical see-through experiments.

Video See-Through Display

TUG is working on a video see-through installation, where the controllable camera delivers the live video image. This video stream is used in the Studierstube application as a video background and can be overlaid with 3D objects, correctly registered. The advantage is, that the augmentation can be viewed correctly registered by a group of people. However, we have to find out if the video see-through has influence on the presence aspect. The video see-through installation was also tested several times within the WP6 showcase. Figure 30 shows an installation of the video see-through display where the camera delivers a live video stream of the environment which is then augmented using the Urban Sketcher application.

Figure 30: Video see-through augmentation using the camera which deliverers live video and with the urban sketcher application.

4.12.3 Specification

Hardware and OS	• Sanyo PLC-XP57L projector, 5500 ANSI lumens • PTU (pan-tilt unit) by RoboSoft • Sony HDV 1080i camera • A.R.T. optical tracking system • Interaction table with a projector-camera setup • Different kinds of projection walls
Software	OpenTracker extensions, and wrappers for different 3rd party libraries
Core Features	Hardware equipment for the MR tent
Status	beta/stable prototype
Intended users	MR tent users
Showcases	WP6 and WP7
Relevance beyond project	

4.12.4 Testing / Evaluation

The setup for the MR tent was tested within different WP6 workshops.

5 Dissemination

Besides the technology probes within several showcases, we also succeeded in showing the core technology at different conferences and events:

- The AR scout with the online 3D reconstruction prototype was shown at ISMAR 2006, Santa Barbara, CA as a technical demo.
- The *online 3D reconstruction* and the *Urban Sketcher* were shown at the OpenLab night at the Institute for Computer Graphics and Vision, Graz University of Technology.

Additional information to activities concerning WP5 at TUG can be observed at the following pages:

- http://www.studierstube.org/ipcity
- http://www.studierstube.org/outdoor
- http://www.ipcity.eu

The following publications related to WP5 were submitted/accepted/published:

[1] B. Reitinger, D. Schmalstieg, Ch. Zach, K. Karner
Automated Model Acquisition using 3D Reconstruction for Urban Planning
CD-ROM proceedings of ISMAR 2007
Demo description

[2] B. Reitinger, Ch. Zach, D. Schmalstieg
AR Scouting for Interactive 3D Reconstruction
Accepted for IEEE Virtual Reality 2007

References

[1] Schmalstieg, D., Fuhrmann, A., Hesina, G., Szalavari, Zs., Encarnacao, L. M., Gervautz, M., & Purgathofer, W.¸ The Studierstube augmented reality project. PRESENCE - Teleoperators and Virtual Environments, 11(1)

[2] Reitmayr, G., & Schmalstieg, D., OpenTracker - an open software architecture for reconfigurable tracking based on XML. Proceedings of IEEE Virtual Reality 2001, pages 285-286, Yokohama, Japan, 2001.

[3] Wagner, D., Pintaric, T., Ledermann, F., Schmalstieg, D., Towards Massively Multi-User Augmented Reality on Handheld Devices. Proceedings of Third International Conference on Pervasive Computing, Pervasive 2005

[4] Ohlenburg, J.,Herbst, I., Lindt, I., Fröhlich, T., Broll, W., *„The MORGAN Framework: Enabling Dynamic Multi-User AR and VR Projects“*, Proc. ACM Symp. Virtual Reality Software and Technology (VRST 2004), *pp. 166-169*, 2004

[5] Wagner, D., Schmalstieg, D., Muddleware for Prototyping Mixed Reality Multiuser Games, Proc. of IEEE Virtual Reality 2006, to appear. 2006.

[6] Höllerer, T., Feiner, S., Pavlik, J., Situated Documentaries: Embedding Multimedia Presentations in the Real World. Proc. ISWC 1999.

[7] Rekimoto, J., Nagao, K.: The world through the computer: Computer augmented interaction with real world environments. In: UIST ’95: Proceedings of the 8th annual ACM symposium on User interface and software technology, Pittsburgh, Pennsylvania, USA (1995) 29–36.

[8] Newman, J., Ingram, D., Hopper, A.: Augmented reality in a wide area sentient environment. In: Proceedings of the 4th IEEE and ACM International Symposium on Augmented Reality (ISAR’01), New York, NY, USA (2001) 77–86.

[9] Gausemeier, J., Fründ, J., Matysczok, C., Bruederlin, B., Beier, D.: Development of a real time image based object recognition method for mobile ar-devices. In: AFRIGRAPH ’03: Proceedings of the 2nd International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa, Cape Town, South Africa (2003) 133–139.

[10] Bier, E.A., Stone, M.C., Pier, K., Buxton, W., DeRose, T.D.: Toolglass and magic lenses: The see-through interface. In: SIGGRAPH ’93: Proceedings of the 20st Annual Conference on Computer Graphics and Interactive Techniques, New York, NY, USA (1993) 73–80.

[11] Viega, J., Conway, M.J., Williams, G.H., Pausch, R.F.: 3d magic lenses. In: UIST '96: Proceedings of the 9th annual ACM symposium on User interface software and technology, Seattle, Washington, USA, ACM Press (1996) 51–58.

[12] Feiner, S., MacIntyre, B., Höllerer, T., Webster, A.: A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. In: Proceedings of the First International Symposium on Wearable Computers (ISWC), Cambridge, Massachusetts, USA (1997) 74–81

[13] Höllerer, T., Feiner, S., Terauchi, T., Rashid, G., Hallaway, D.: Exploring mars: developing indoor and outdoor user interfaces to a mobile augmented reality system. Computers & Graphics 23 (1999) 779–785

[14] Piekarski, W., Thomas, B.H.: Tinmith-evo5 a software architecture for supporting research into outdoor augmented reality environments. Technical report, Wearable Computer Laboratory, University of South Australia (2001).

[15] Ledermann, F., Schmalstieg, D.: APRIL: A high-level framework for creating augmented reality presentations. In: Proceedings of the 2005 IEEE Virtual Reality Conference (to appear), Bonn, Germany, IEEE Computer Society (2005).

[16] Olson, C. F., Matthies, L. H., Schoppers, M., Maimone, M. W.: Rover navigation using stereo ego-motion. Robotics and Autonomous Systems, 43(4):215-229, June 2003.

[17] Andersen, H. J., Didireksen, T.L., Madsen, C., Holte, M.B.: Investigating the potential combination of GPS and scale invariant visual landmarks for robust outdoor cross-country navigation. VISAPP 2006, Volume II, ISBN 972-8865-40-6, Pages 349-356, February 2006.

[18] Lowe, D. G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, 60, 2 (2004), pp. 91-110.

[19] Ke, Y., Sukthankar, R.: PCA-SIFT: A more distinctive representation for local image descriptors. Computer Vision and Pattern Recognition, 2004.

[20] Brown, M., Szeliski, R., Winder, S.: Multi-Image Matching Using Multi-Scale Oriented Patches. Technical Report: MSR-TR-2004-133, December 2004.

[21] Shi, J., Tomasi, C.: Good features to track. IEEE Conference on Computer Vision and Pattern Recognition, pages 593-600, 1994.

[22] Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. CVPR (2) 2003, 257-263.

[23] Harris, C., Stephens, M.: A combined corner and edge detector. In Proceedings of The Fourth Alvey Vision Conference, 1988, 147-151.

[24] Nakanishi, Y., Kumazawa, S., Tsuji, T., Hakozaki, K.: iCAMS2: Developing a Mobile Communication Tool Using Location Information and Schedule Information with J2ME. In Proceedings of the 5th International Symposium on Human-Computer Interaction with Mobile Devices and Services (MobileHCI 2003). Lecture Notes in Computer Science, Vol. 2795. Springer-Verlag, Berlin Heidelberg New York (2003) 400—404

[25] Milewski, A.E., Smith, T.M.: Providing Presence Cues to Telephone Users. In Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work (CSCW 2000). ACM Press, New York (2000) 89—96.

[26] Bardram, J.E., Hansen, T.R.: The AWARE Architecture: Supporting Context-Mediated Social Awareness in Mobile Cooperation. In Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work (CSCW 2004). ACM Press, New York (2004) 192—201

[27] Tang, J.C., Yankelovich, N., Begole, J., Van Kleek, M., Li, F., Bhalodia, J.: ConNexus to Awarenex: Extending Awareness to Mobile Users. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2001). ACM Press, New York (2001) 221—228

[28] Burak, A., Sharon, T.: Analyzing Usage of Location Based Services. In CHI 203 Extended Abstracts on Human Factors in Computing Systems Conference. ACM Press, New York (2003) 970—971

[29] Dourish, P., Bly, S.: Portholes: Supporting awareness in a distributed work group. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 1992). ACM Press, New York (1992) 541—547

[30] Grudin, J.: Computer Supported Cooperative Work: History and Focus. IEEE Computer 27/5 (1994) 19-26. 1994.

[31] Kraut, R.E., Streeter, L.A.: Coordination in Software Development. Communications of the ACM 38/3 (1995) 69-81. 1995.

[32] Isaacs, E.A., Tang, J.C. Morris, T.: Piazza: A Desktop Environment Supporting Impromptu and Planned Interactions. In Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work (CSCW 1996). ACM Press, New York (1996) 315-324.

[33] Isaacs, E., Walendowski, A., Ranganthan, D.: Hubbub: A Sound-Enhanced Mobile Instant Messenger that Supports Awareness and Opportunistic Interactions. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2002), ACM Press, New York (2002) 179-186.

[34] Counts, S.J., Fellheimer, E.: Supporting Social Presence through Lightweight Photo Sharing on and off the Desktop. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2004). ACM Press, New York (2004) 599—606.

[35] Koskinen, I., Kurvinen, E., Lehtonen, T.-K.: Mobile Image. IT Press, Helsinki (2002)

[36] Weilenmann, A.: Local Use and Sharing of Mobile Phones. In: Brown, B., Green, N., Harper, R. (eds.). Wireless World: Social and Interactional Aspects of the Mobile Age. Springer-Verlag, London (2001) 92—107.

[37] Taylor, A.S., Harper, R.: The Gift of the Gab?: A Design Oriented Sociology of Young People's Use of Mobiles. Computer Supported Cooperative Work 12/3 (2003) 267—296.

[38] Sarvas, R., Herrarte, E., Wilhelm, A., Davis, M.: Metadata Creation System for Mobile Images. Proceedings of the International Conference on Mobile Systems, Applications and Services (MobiSys 2004). ACM Press, New York (2004) 36—48.

[39] Raento, M., Oulasvirta, A., Petit, R., & Toivonen,H.: ContextPhone, a prototyping platform for context-aware mobile applications,. IEEE Pervasive Computing, 4(2), apr-jun 2005. ISSN 1536-1268.

[40] Welch, G., Foxlin, E., (2002). “Motion Tracking: No Silver Bullet, but a Respectable Arsenal,” IEEE Computer Graphics and Applications, special issue on “Tracking,” November/December 2002, 22(6): 24–38.

[41] Hightower, J., Borriello, G., "A Survey and Taxonomy of Location Sensing Systems for Ubiquitous Computing," UW CSE 01-08-03, University of Washington, Department of Computer Science and Engineering, Seattle, WA, Aug. 2001.

[42] Bimber,O. and Raskar, R. “Modern Approaches to Augmented Reality”, Siggraph 2005,http://www.spatialar.com/Siggraph05.htm (06.04.2006)

[43] Azuma, R.T., ”A Survey of Augmented Reality”, In Presence: Teleoperators and Virtual Environments 6, 4 (August 1997), 355-385.

[44] Amselem, D. A window on shared virtual environments. *Presence, Vol. 4, No. 2*, pp. 130-145, 1995.

[45] Fitzmaurice, G. W. Situated Information Spaces and Spatially Aware Palmtop Computers. *Communications of the ACM*, Vol.36, Nr.7, pp 38-49, July 1993.

[46] Gausemeier, J., Fruend, J., Matysczok, C., Bruederlin, B., and Beier, D., Development of a real time image based object recognition method for mobile AR-

devices. *Proc. 2nd international conference on Computer graphics, Virtual Reality, visualisation and interaction in Africa*, pp. 133-139, 2003.

[47] Hall, T. et. al., The visitor as virtual archaeologist: explorations in mixed reality technology to enhance educational and social interaction in the museum, Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage, pp. 141-152, 2001.

[48] Ingram, D., Newman, J., Augmented Reality in a Wide Area Sentient Environment. *Proceedings of the 2nd IEEE and ACM International Symposium on Augmented Reality (ISAR 2001)*, October 2001, New York

[49] Kretschmer, U. et. al., "Meeting the spirit of history", Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage, pp. 141-152, 2001.

[50] MacIntyre, Blair, Bolter, Jay David, Moreno, Emmanuel, and Hannigan, Brendan. Augmented Reality as a New Media Experience. International Symposium on Augmented Reality (ISAR 2001), New York, NY, October 29-30, 2001.

[51] Mogilev, D., Kiyokawa, K., Billinghurst, M., Pair, J., AR Pad: an interface for face-to-face AR collaboration. *CHI '02 extended abstracts on Human factors in computer systems*, pp. 654-655, April 2002.

[52] Rekimoto, J., and Nagao, K. The World through the Computer: Computer Augmented Interaction with Real World Environments, *User Interface Software and Technology (UIST '95)*, pp. 29-38, November 14-17, 1995.

[53] Sumi, Y., Etani,T., Fels,S., Simonet,N., Kobayashi,K. and Mase,K.: C-MAP: Building a context-aware mobile assistant for exhibition tours, *1st Kyoto Meeting on Social Interaction and Communityware*, June 1998.

[54] Vlahakis V., N. Ioannidis, J. Karigiannis, M. Tsotros, M. Gounaris, D. Stricker, T. Gleue, P. Daehne, L. Almeida, "ARCHEOGUIDE: Challenges and Solutions of a Personalized Augmented Reality Guide for Archaeological sites", "Computer Graphics in Art, History and Archaeology" Special Issue of the IEEE Computer Graphics and Applications Magazine, Sept.-Oct. 2002.

[55] Wagner, D., Schmalstieg, D. “First Steps Towards Handheld Augmented Reality”, To appear in: Proceedings of the 7th International Conference on Wearable Computers, White Plains, NY, USA, Oct. 21-23, 2003.

[56] Experience Music Project Homepage. http://www.emplive.com (06.04.2005).

[57] Kato, H., Billinghurst, M. (1999) Marker Tracking and HMD Calibration for a video-based Augmented Reality Conferencing System. In Proceedings of the 2nd

International Workshop on Augmented Reality (IWAR 99). October, San Francisco, USA, http://www.hitl.washington.edu/artoolkit/ (06.04.2006).

[58] http://studierstube.icg.tu-graz.ac.at/handheld_ar/artoolkitplus.php (06.04.2005).

[59] Jürgen Scheible and Timo Ojala. 2005. MobiLenin – Combining A Multi-Track Music Video, Personal Mobile Phones and A Public Display into Multi-User Interactive Entertainment. Proc. ACM Multimedia 2005, Singapore, 199-208.

[60] Rashid O.,Mullins I.,Coulton P., and Edwards R. 2006. Extending Cyberspace:Location Based Games Using Cellular Phones. ACM Computers in Entertainment,Vol 4, Issue 1, January, 2006.

[61] Aalto, L., Gothlin, N., Korhonen, J., and Ojala, T. 2004. Bluetooth and WAP push based location-aware mobile advertisement system. MobiSys (2004).

[62] Schmalstieg, D., Reitmayr, G., The World as a User Interface: Augmented Reality for Ubiquitous Computing, Central European Multimedia and Virtual Reality Conference 2005 (CEMVRC 2005), Prague, Czech Republic, June 8-10, 2005.

[63] Pilu, M.: Uncalibrated Stereo Correspondence by Singular Value Decomposition. Digital Media Department HP Laboratories Bristol, Technical report: HPL-97-96, August, 1997.

[64] SGI OpenInventor, http://oss.sgi.com/projects/inventor, http://doc.coin3d.org/

[65] Adaptive Communication Environment http://www.cs.wustl.edu/ schmidt/ACE.html

[66] Billinghurst, M., Campbell, S., Chinthammit, W., Hendrickson, D., Poupyrev, I., Takahashi, K., and Kato, H., "Magic book: Exploring transitions in collaborative AR interfaces". In *Emerging Technologies Proposal*, ACM SIGGRAPH, 2000.

[67] Looser, J., Grasset, R., Seichter, H., Billinghurst, M., "OSGART – A Pragmatic Approach to MR". In *Industrial Workshop at ISMAR 2006*, 22nd October 2006, Santa Barbara, California, 2006.

[68] OpenSceneGraph, http://openscenegraph.org

[69] Nistér, D., and Stewénius H. "*Scalable Recognition with a Vocabulary Tree*". Proceedings of Computer Vision and Pattern Recognition (CVPR), 2161-2168, 2006.

[70] Conway, M., Pausch, R., Gossweiler, R., and Burnette, T. *"Alice: A rapid prototyping system for building virtual environments"*. Proceedings of ACM CHI'94 Conference on Human Factors in Computing Systems, volume 2, 295-296, 1994.

[71] Kretschmer, U., Coors, V., Spierling, U., Grasbon, D., Schneider, K., Rojas, I., and Malaka, R. *"Meeting the spirit of history"*. Proceedings of VAST 2001, Athens, Greece, 2001.

[72] Beckhaus, S., Lechner, A., Mostafawy, S., Trogemann, G., and Wages, R. *"alVRed - Tools for storytelling in virtual environments"*. Internationale Statustagung Virtuelle und Erweiterete Realität, Leipzig, Germany, 2002.

[73] Tramberend H. *"Avocado: A distributed virtual reality framework"*. Proceedings of IEEE Virtual Reality 1999, 1999.

[74] MacIntyre, B., Gandy, M., Dow, S., and Bolter, J. *"DART: A Toolkit for Rapid Design Exploration of Augmented Reality Experiences"*. Proceedings of User Interface Software and Technology (UIST'04), Sante Fe, New Mexico, 2004.

[75] MacWilliams, A., Sandor, C., Wagner, M., Bauer, M., Klinker, G., and Bruegge, B. *"Herding Sheep: Live System Development for Distributed Augmented Reality"*. Proceedings ISMAR 2003, Tokyo, Japan, 2003.

[76] Grimm, P., Haller, M., Paelke, V., Reinhold, S., Reimann, C., and Zauner, J., *"AMIRE – Authoring Mixed Reality"*. The First IEEE International Augmented Reality Toolkit Workshop, Darmstadt, Germany, September, 2002.

[77] Zauner, J., Haller, M., Brandl, A., and Hartmann, W. *"Authoring of a Mixed Reality Assembly Instructor for Hierarchical Structures"*, ISMAR'03, Second International Symposium on Mixed and Augmented Reality, 2003.

[78] Abawi, D., Dörner, R., Haller, M., and Zauner, J. *"Efficient Mixed Reality Application Development"*. Proceedings of the 1st European Conference on Visual Media Production. London, March, 2004.

[79] Höllerer, T., Feiner, S., and Pavlik, J. *"Situated Documentaries: Embedding Multimedia in the Real World"*. Proceedings of the ISWC'99 (International Symposium on Wearable Computers", San Francisco, CA, October 18-19, 1999.

[80] Salminen T., Riekki J., "Lightweight middleware architecture for mobile phones". Proc. 2005 International Conference on Pervasive Systems and Computing, Las Vegas, NE, 147-153. 2005.

Acknowledgements and Further Information

IPCity is partially funded by the European Commission as part of the sixth framework (FP6-2004-IST-4-27571

We would like to thank the VRVis for providing the 3D reconstruction library which is used for the online 3D reconstruction framework.

For further information regarding the IPCity project please visit the project web site at:

www.ipcity.eu