



**Integrated Project on Interaction and Presence  
in Urban Environments**

FP6-2004-IST-4-27571

[ipcity.eu](http://ipcity.eu)

**Improved Prototypes for MR Infrastructure**  
Deliverable D5.3



<b>Doc-Id:</b>	<b>D 5.3</b>
<b>Version:</b>	<b>1.0</b>
<b>Author(s):</b>	<b>Markus Sareika, Ann Morrison, Giang Phuong Nguyen, Gerhard Reitmayr, Alessandro Muloni, Antti Juustila, Andrea Börner, Peter Peltonen, Jan Herling</b>
<b>Date:</b>	<b>2009-02-13</b>
<b>Status:</b>	<b>Final</b>
<b>Availability:</b>	<b>Restricted</b>
<b>Distribution:</b>	<b>Project Partners / EC</b>

## Table of Content

---

1	Introduction .....	1
2	Conceptual Infrastructure Framework for Interaction and Presence Experience.....	3
3	Mixed Reality Infrastructure Building Blocks .....	1
3.1	Tracking.....	1
3.2	Storage/Content .....	1
3.3	Computation .....	1
3.4	Mobile AR/MR .....	1
4	Workpackage Objectives .....	3
	Year 3 Prototypes.....	6
4.1	Tracking.....	7
4.1.1	Mobile Presence Scanner.....	7
4.1.2	Illuminate Technology.....	8
4.1.3	Dotted Map Tracking .....	12
4.1.4	Mobile Map Tracking from natural features .....	15
4.1.5	Color Table Tracking .....	18
4.1.6	Augmented Map Table .....	19
4.1.7	Bluetooth Media Dispatcher.....	21
4.1.8	Vision Based Localization.....	22
4.1.9	Model-based Outdoor Tracking .....	24
4.2	Storage/Content .....	29
4.2.1	Distributed Media Entrance and Management .....	29
4.2.2	HMDB Interfacing .....	31
4.2.3	ContentManager .....	32
4.2.4	Middleware .....	33
4.2.5	Audio/Video Streaming.....	34
4.3	Computation .....	38
4.3.1	Spatial Sound .....	38
4.4	Mobile AR/MR .....	42
4.4.1	MR-Tent.....	42
4.4.2	AR Scouting.....	46
4.4.3	Studierstube ES.....	49
4.4.4	Studierstube - Urban Sketcher and Expression.....	51
4.4.5	Color Compensation for See-Through Displays .....	54
4.4.6	Device Comparison Tests.....	56



## Intended Audience

---

This document is intended to all partners of the project and to the reviews for the third project's phase.



# 1 Introduction

---

Mixed Reality (MR) enhances a user's perception by providing stimuli augmenting the real world. IPCity moves high-quality MR a step further from labs to real settings enabling the experience and the expression of presence in its various dimensions.

The vertically oriented application scenarios of the project add individual strategies to address specific research questions, while the overall project is based on horizontal research topics, building on top of WP5 the Infrastructure.

MR infrastructure is focusing on basic research of mobile devices and their specifics to realize MR applications in 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.

Driven by the showcases, progress towards the project's objectives has been achieved in all different building blocks of the MR infrastructure. For example, the integration inside the further developed MR-Tent where the MR projection is realized and registered in an interactive way with the centralized table top projection. Progress has also been made towards open interfaces as a basis for collaboration supporting user interface in the tent. Furthermore, progress towards mobile AR software frameworks has been made in terms of code optimization for handheld devices as well as in terms of modular extensions supporting a wider range of possible applications. The increasing degree of mobility emerging from the showcases led to improving the integration of storage solutions and extending their content organizing features. Components of the building block tracking within the MR infrastructure were extended towards ubiquity. The core contributing technologies are now ranging from vision based locators and pose estimators over modules using Bluetooth or GPS receivers to various combinations of currently available tracking solutions, leading to a more robust localization for in- and outdoor MR scenarios.

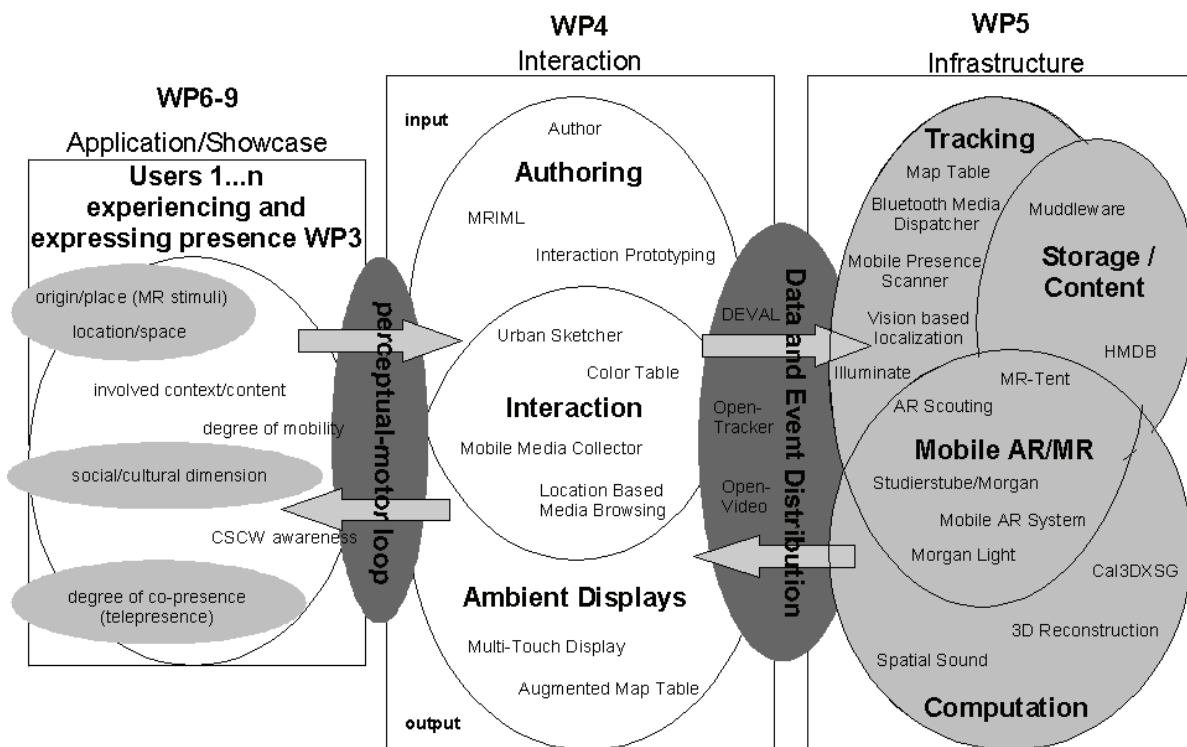
The infrastructure developed in phase II has undergone a re-design process in order to take into account feedback from the showcases and the results from the evaluation. This deliverable states the overall strategy, the decisions made about the infrastructure developed in phase II and outlines the development that took place during phase III. In the third development phase previously developed functionality was improved and new features were implemented based on the redesign decisions.





## 2 Conceptual Infrastructure Framework for Interaction and Presence Experience

From a technical point of view, enabling presence and experience in mixed reality environments requires a multi-layer approach. Firstly, providing the general infrastructure (hardware and software) and services to realize MR systems. Secondly, the provision of higher-level tools for authoring MR environments and supporting the realization of MR user interfaces. Thirdly, the development of the actual MR application including application-specific features and tools. Figure 1. further clarifies the concept between the various building blocks of MR technology and gives an overview of interconnections between the developed tools and senses and sensations of presence.

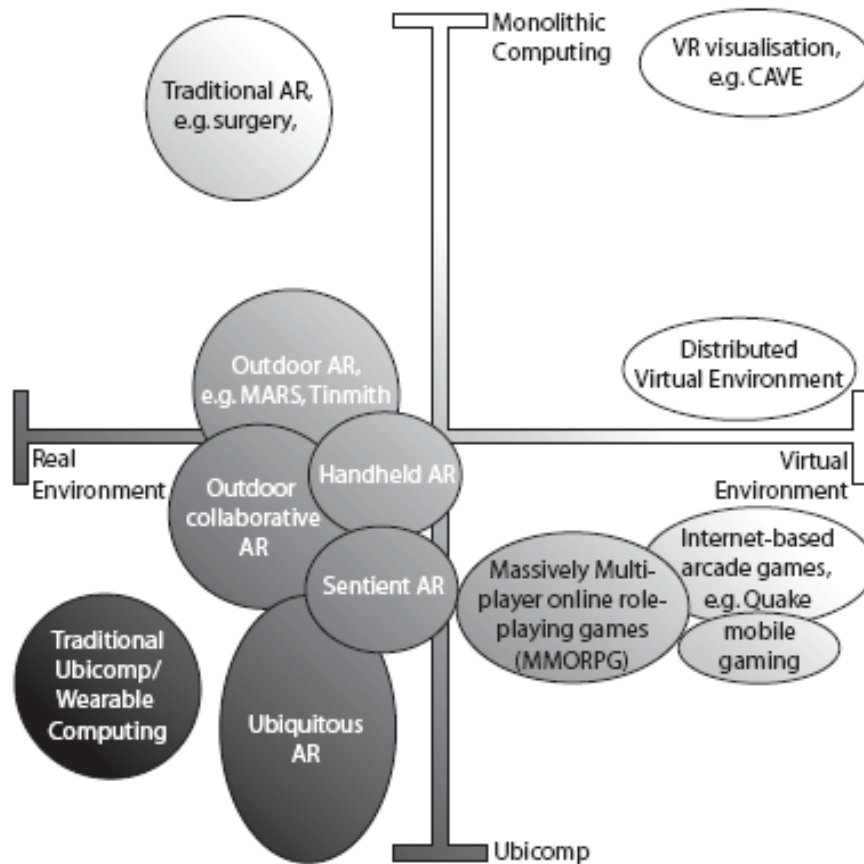


**Figure 1:** Flow of information in the application of IPCity. A conventional human-computer interaction loop demonstrates that users, interaction tools and infrastructure each have very complex aspects that need to be orchestrated in order to support Mixed Reality co-presence

The first section of Figure 1 represents the user in his current context of environment and intention. This is the state of presence or co-presence which is experienced and defines an entry point for mixing reality. The initial situation imposed by a showcase environment implicitly defines suitable characteristics for interface technology. Furthermore the application of software tools and building blocks, which define the infrastructure, allows to dynamically route the exchange of information not only between two users but enables 1-to-n as well as n-to-1 communication, while several modalities supported by the hardware interface can be involved in the communication process.

Interaction between human and machine is possible through these hardware interfaces by connecting senses thus leading to co-presence experiences. Various feedback channels engage individual users and integrate their expressions thus allowing co-construction of presence. The ability to mix the experienced reality at an arbitrary scale is only limited by the capabilities of the underlying infrastructure (the third section of Figure 1). Therefore integration and the application of open interfaces are essential for large scale collaboration. The cross-reality interaction tools research work package (the second section of Figure 1 )

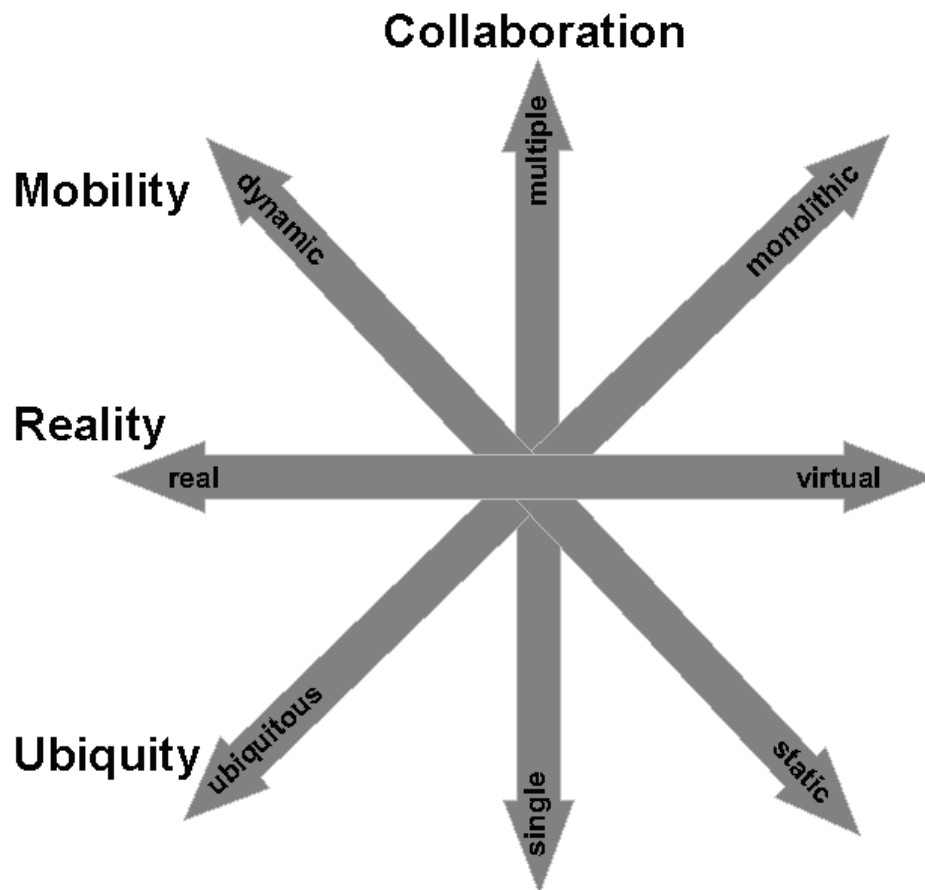
focuses on support of mixed reality user interface creation, development, and execution. In contrast to traditional user interfaces mixed reality user interfaces are typically not limited to one or two particular devices, but rather use a large variety of individual devices supported by the underlying infrastructure.



**Figure 2:** Milgram-Weiser-Newman Continuum

Interaction, presence and mixed reality in urban environments are complex phenomena. In contrast to classical research on presence, the phenomena considered in IPCity have collaboration as an essential property. From a technical point of view, Mixed Reality was initially described as a continuum by Milgram. Independently, Weiser examined ubiquity, which is obviously important for a project operating in urban space, such as IPCity. These considerations were always kept distinct. The recent publication of Newman et. al.<sup>1</sup> suggests to organize ubiquitous MR applications in a two-dimensional Milgram-Weiser continuum (Figure 2) taking the quantity and density of spatial distribution into account. This approach is able to better represent configurations where multiple input and/or output devices are interconnected to contribute to MR-systems blurring the border to ubiquitous computing.

<sup>1</sup> Joseph Newman, Alexander Bornik, Daniel Pustka, Florian Echtler, Manuel Huber, Dieter Schmalstieg, Gudrun Klinker: Tracking for Distributed Mixed Reality Environments Proceedings of IEEE Virtual Reality Workshop on Trends and Issues in Tracking for Virtual Environments, Charlotte NC, USA, March 2007.



**Figure 3:** Multidimensional continuum spanning a design space

We felt that even the Milgram-Weiser continuum is not sufficiently expressive when it comes to the representation of co-presence in MR. Among the multiple dimensions of presence that have surfaced in our and related research are at least spatial presence (e.g., perceptual immersion, sense of being there), sensory presence (perceptual realism), engagement (involvement), social presence (co-presence). Consequently Figure 3 suggests a four dimensional continuum enclosing reality, mobility, collaboration and ubiquity. A similar 3-dimensional taxonomy, covering immersion, collaboration and mobility, has been proposed by Broll<sup>2</sup> (2002).

The technology we are developing in work packages 4 and 5 represent samples or probes at interesting positions in this very complex conceptual design space. Together they represent the necessary building blocks required by Mobile Mixed Reality applications for users engaging in mixed reality co-presence.

The conceptual design space guides the development and leads to decisions made to address specific aspects out of the design space but also to provide solutions at a possibly large scale. For example, handheld devices have the potential to provide a strong mobile interface, where as stationary technologies have their strength in face to face collaboration. The combination of various types of input devices by interfacing their infrastructures closes the gap between different levels of scale and enriches the overall communication process.

<sup>2</sup> Broll, W.: "Collaborating in Mixed Realities". 3D FORUM, The Journal of Three Dimensional Images Vol. 16, No. 4, (Dec. 2002): 135-140. Also in Proceedings of HC 2002 – the Fifth International Conference on Humans and Computers (Sept. 11 – Sept. 14, 2002), Aizu, Japan, 138-143.

The showcases – due to their differences – serve as a good cross section through the design space of Mixed Reality applications. In the spirit of high ubiquity, we favor a building block approach over monolithic solutions. The overall shared vision by the technical work packages WP 4 and WP 5, is to provide all the necessary building blocks required by modern (Mobile) Mixed Reality applications and additionally provide tools to support the design, authoring, direction and evaluation process of its content, user interfaces and interactions. The showcases – due to their differences – also serve as a good cross section through the whole area of Mixed Reality applications. In order to concentrate only on those technologies, which are not unique to only one type of application scenario, we have defined requirements for technologies developed in the technical directed work packages. These requirements for all tools and services are:

- has to be required by at least two showcases,
- must actually be used by at least one showcase,
- must be flexible enough to be used in other showcases and even in other projects.

Since we do not believe that a single tool is able to provide all required aspect of a service, our developed technologies can be overlapping in regards to functionality. The application is hence able to pick the most appropriate set of technologies, which serves its requirements best.

This work package is visualized in the third section of Figure 1. The infrastructure is intrinsically tied close to technology which has been divided into four major building blocks:

- Tracking
- Storage/Content
- Computation
- Mobile AR/MR

They are visualized by four bubbles inside this section providing an overview to the developed infrastructure components which are needed to create an information space that lives around the users cross reality presence and are described in the next chapter in detail.

## 3 Mixed Reality Infrastructure Building Blocks

---

### 3.1 Tracking

A very important building block, which is required by all AR/MR applications, is the tracking building block. While for user tracking (head-tracking) and object tracking a lot of solutions exist for indoor MR applications based on fixed installations, no acceptable solution has yet been proposed for 6-DOF outdoor tracking in unconstrained environments. Even more importantly that new groundbreaking research is conducted in this area. While the first showcase demonstrators had to use low quality user tracking, e.g. GPS tracking combined with inertial tracking, we are currently working on computer vision based tracking technology that allows high quality user, map and tangible object tracking in urban environments.

The recent selection of the University of Cambridge as a new partner adds expertise in computer vision based real-time tracking solutions to the consortium. One task established in the competitive call is the development of a model-based tracking system for urban environments.

In addition to user tracking, which is a basic requirement for all applications, we are also working on map tracking technology. Here natural feature-based techniques allow the move away from fiducial markers towards use of unmodified, every-day objects such as printed paper maps and event brochures as background. Localization of handheld devices over these printed artifacts provides interaction with additional digital information.

Other aspects of tracking technology is that not all tracking devices and technologies are available anytime and anywhere, therefore it is important to abstract from individual tracking devices and provide tracking services to the applications. OpenTracker and DEVAL provide the abstraction layer for data and event distribution and allow applications to be developed independently of the utilized technology. Additionally, some applications also require seamless tracking, although different tracking technologies are used. Ubiquitous tracking deals with these kinds of issues.

### 3.2 Storage/Content

The storage of media and information is essential for continuity and persistence in AR/MR applications. The first main component is the Hyper Media Data Base providing a storage solution for various types of media which can be enriched with meta information. The second component is Muddleware which stores hierarchies and states in a XML structure and provides very flexible access by a memory mapped DOM graph. This component is intended to be used as Middleware for a large number of clients and has a strong affiliation to data and event distribution. Another form of storage is provided by the MR-Tent as it provides shelter for the MR equipment and a small group of urban presence explorers.

### 3.3 Computation

Computation is a general term for any type of information processing that can be represented mathematically. This can be calculations done in a process following a well defined model that is understood and can be expressed in an algorithm, protocol, network topology, etc. The bubble in Computation comprises components which are computation intensive for solving a specific task and are not strongly affiliated with another component.

### 3.4 Mobile AR/MR

The Mobile AR/MR building block comprises the systems of assembled hardware components forming solutions for mobility including software frameworks which combine appropriate components.

Mobile AR/MR 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.

## 4 Workpackage Objectives

<p>Success Criteria Phase III</p>	<ol style="list-style-type: none"> <li>(1) The tools and services developed within the current working period (a)/since project start (b)/based on existing technology (c)</li> <li>(2) Significant progress in the development of technologies of each major building block</li> <li>(3) Each technology requested by at least two showcases and actually used by at least one showcase</li> <li>(4) Each technology developed is contributing to the overall project objectives</li> <li>(5) Technology used or requested by other projects or third parties and to what extend</li> <li>(6) Submission of at least six conference and/or journal papers (at least one for each of the five major building blocks)</li> <li>(7) Identify new technologies developed within the project not available elsewhere</li> </ol>
<p>Objectives Phase III</p>	<p>After a set of tools has been developed and most of them already have been used within the showcases, these tools have to be redesigned based on the experience and the results of the evaluation from the showcases. Additional tools will be designed and developed according to the needs of the showcases. The redesigned and new tools inside the outlined design space will be delivered to the showcases to be included within phase three.</p> <p>The focus of our work will be on the following topics and tools due to demands from the showcases:</p> <p><b>Component Integration:</b> Improve exchange of information between component structures leading to enhanced functionality.</p> <p><b>Interaction table:</b> 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 be further developed and integrated.</p> <p><b>Mobile setups (handheld mixed reality environments):</b> Three different mobile devices (scaled in computing performance) will further be developed: a sub-notebook-based (UMPC-based) approach, a PDA-based approach, a smart phone-based approach.</p> <p><b>Tracking and localization:</b> Map tracking is a new core issue where development will mainly be in the context of mobile devices. Vision-based tracking and localization will further be developed in order to</p>

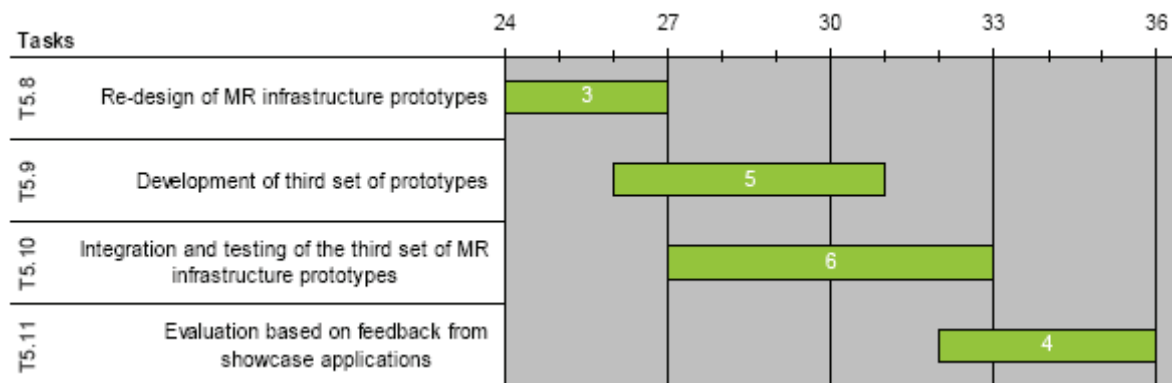
	<p>get more precise positioning for outdoor MR applications. In addition ubiquitous tracking will be improved which should allow a seamless transition of users moving between different tracking services.</p>
<p>Results Phase III</p>	<p>The following tools are new developments within phase III (1a):</p> <ul style="list-style-type: none"> <li>• <b>ContentManager:</b> A software for downloading content from Internet's social media services (e.g. Flickr, YouTube) and to insert into a database (to be used by CityWall in WP7)</li> <li>• <b>Dotted Map Tracking:</b> A tracker for mobile phones allowing to track dotted maps with mobile phones</li> <li>• <b>Mobile Map Tracking from natural features:</b> We implemented a computer vision-based tracking method that works with any image that provides sufficient texture.</li> <li>• <b>Audio/Video Streaming:</b> The MORGAN Media Library has been extended with new video and especially audio streaming features.</li> </ul> <p>The following tools from previous phases have been extended (1b):</p> <ul style="list-style-type: none"> <li>• <b>Bluetooth Media Dispatcher:</b> Enhanced features like thumbnail support.</li> <li>• <b>Color Table Tracking:</b> Added automatic token management.</li> <li>• <b>Vision based localization:</b> Work was done on the communication between a server and a user device.</li> <li>• <b>Distributed media entrance and management system:</b> Integration work with HMDB</li> <li>• <b>HMDB interfacing:</b> Web interface improvements</li> <li>• <b>MR tent:</b> A few corrections concerning the tent construction were done.</li> <li>• <b>AR-Scouting:</b> Live video streaming implementation.</li> <li>• <b>Studierstube ES:</b> Various new feature implementations</li> <li>• <b>Muddleware:</b> Used to realize persistent open inventor fields under coin.</li> <li>• <b>Illuminate:</b> The final state of Illuminate has</li> </ul>



	<p>been reached, as it has been placed on hold as WP7 changed focus to address environmental issues and the focus on lighting was in conflict with this.</p> <ul style="list-style-type: none"> <li>• <b>Urban Sketcher and Expression:</b> The integration process of the functionality for the infrastructure of ColorTable and UrbanSketcher results in the Urban Expression framework.</li> </ul> <p>The following pre-existing tools have been extended (1c):</p> <ul style="list-style-type: none"> <li>• <b>Augmented Map Table:</b> Now also a physical map can be used in combination with the ColorTable.</li> </ul>
<p>Objectives Phase IV</p>	<p>The focus of our work will be on the following topics and tools due to demands from the showcases:</p> <ul style="list-style-type: none"> <li>• <b>ContentManager</b> Extend support for more services if required. Add content moderation tools.</li> <li>• <b>Sketcher usability assessment:</b> Perform assessment of the Sketcher’s usability through a quantitative evaluation.</li> <li>• <b>UrbanSketcher:</b> Introduce GPU Sketching for a better performance.</li> <li>• <b>Performance optimization with Slow-Fast Rendering:</b> Implement a sort last compositing for multiple scenes rendered at different framerates.</li> <li>• <b>Scouting and multi-perspective MR:</b> Extend scouting capabilities by multiple perspective MR.</li> <li>• <b>KML XSG:</b> A prototype or demonstrator implementation of an XSG supporting KML (Google Earth) files or a similar technology.</li> <li>• <b>Mobile Stereo Displays:</b> We will investigate the feasibility of new mobile stereo display system ideas and test hardware prototypes we develop.</li> <li>• <b>Physics Abstraction Layer:</b> A design and prototype implementation of a physics engine abstraction layer for the MORGAN framework.</li> <li>• <b>Model-based Outdoor tracking:</b> Integrating the outdoor tracking with Vision-based localization and further extend to use Geo-referenced images as model.</li> </ul>

## Year 3 Prototypes

The year 3 prototypes have been re-designed, developed, tested and evaluation according to D1.9 *Detail work plan for Phase III Month 24-36* (see Figure 4). Hence the work for the prototypes is described in relation to the different phase, after a general introduction the review and re-design phase is described, followed by the development & research phase and finally the testing and public demonstration phase as well as the evaluation phase based on showcase feedback.



**Figure 4:** Detail work plan for Month 24-36 (Phase III).

The view overlaid on the infrastructure components draws a fuzzy border for a rough classification of the developed components. The already outlined mixed reality infrastructure building blocks are: Tracking, Storage/Content, Mobile AR/MR and Computation.

## 4.1 Tracking

### 4.1.1 Mobile Presence Scanner

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.

The mobile presence scanner was developed during the year 2006. The basic design proved to be robust and consequently the component has not been developed significantly after that time.

The presence scanner has been used in the creating a demo version of Illuminate project (WP7). The demo was developed in early 2007 and displayed in the IPCity project review in Bonn. In the demo any number of mobile phones could be used to share ambient information over the Bluetooth protocol. The phones reacted to the presence of other phones by changing their color over time. This demo used the Bluetooth capabilities of the presence scanner.

In the future the presence scanner will be further developed as the showcases have need for new features. The features we were planning to add to the component were GPS locationing using an auxiliary device and calendar tracking. These have not been implemented as there has been no demand for them from the showcases. Further, some recent phone models have a built-in GPS device, which should be used directly with the native API instead of relying on an external device. Thus the implementation strategy of the GPS component needs to be decided when there is actual need for it.

#### 4.1.1.1 Review and Re-design

Since the workpackage 7, for which this tool was intended, has moved focus from large scale events to environmental awareness research, this scanner as well as the Illuminate project (see 4.1.2) have been halted.

#### 4.1.1.2 Specification

Hardware and OS	<ul style="list-style-type: none"> <li>Symbian/S60 2<sup>nd</sup> edition mobile phones. The current main model is Nokia N70.</li> </ul>
Software	<ul style="list-style-type: none"> <li>Symbian C++</li> </ul>
Core Features	<ul style="list-style-type: none"> <li>Scan presence information using the phone: Nearby Bluetooth devices, phone profile, file system changes, GSM cell information</li> <li>Planned features: GPS locationing, Calendar monitoring</li> </ul>
Status	<ul style="list-style-type: none"> <li>Beta software</li> </ul>
Intended users	<ul style="list-style-type: none"> <li>Any number of users, depends on the application</li> </ul>
Showcases	<ul style="list-style-type: none"> <li>WP7</li> </ul>
Relevance beyond project	<ul style="list-style-type: none"> <li>The component is useful for smart-phone applications that need to access presence information.</li> </ul>

#### 4.1.2 Illuminate Technology

Illuminate technology is now a working prototype that was demonstrated in the Barcelona review 2008. The concept behind the technology (described in more detail in D7.3) is to track the movement of people through distributed event spaces that function as the staging area for social interactions that occur outside a given users peer group. The interactions create a flow of movement where people's paths cross invisibly. Visualizing these flows and social interactions visible enriches a spectator's presence at and awareness of an event.

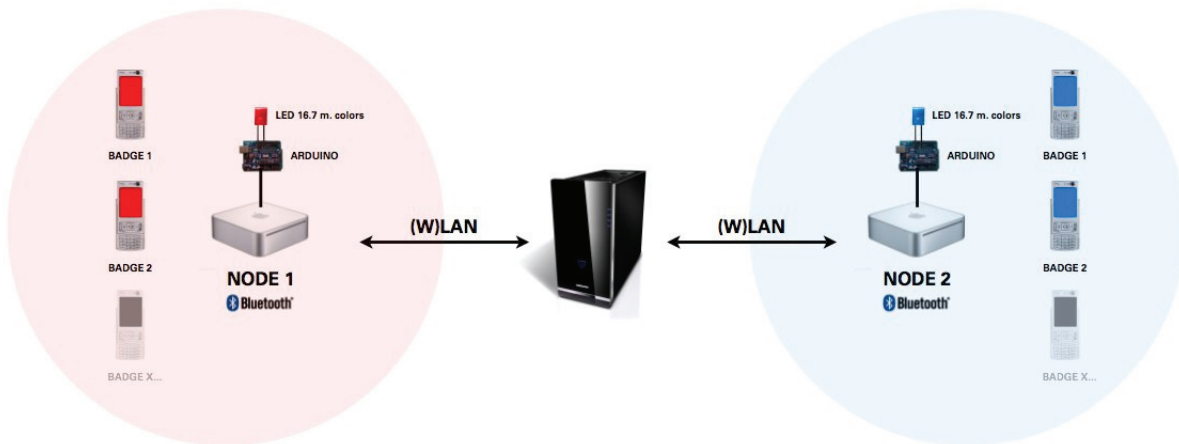
Illuminate development has been placed on hold for most of the third year as WP7 changed its focus from large-scale events to address environmental issues (for more detailed plans for months 24-42, see D1.9) and the focus on lighting was in conflict with this. The focus of WP7 will be in environmental issues rest of the project, so no further development for this technology is to be expected to happen, unless some other workpackage requires it. We will now describe the current state of the work to give an overview of the final stage of the prototype, although most of it has been implemented during the first and second year of the project.

##### 4.1.2.1 Review and Re-design

A re-design to implement a larger scale installation view was planned in the early stages of 2008. Behind the original Illuminate technology is a network of Nodes each equipped with Bluetooth™ sensors and a wireless Internet connection (see Figure 5). Atelier Infrastructure, a Java and XML based distributed framework, is used to establish the communication between the nodes. To control ambient lighting within the space the Nodes are installed in, an Arduino board is used. The wearable electronic badge is prototyped by using mobile phones which include a Bluetooth™ sensor and a light source react to communicate with the Node network.

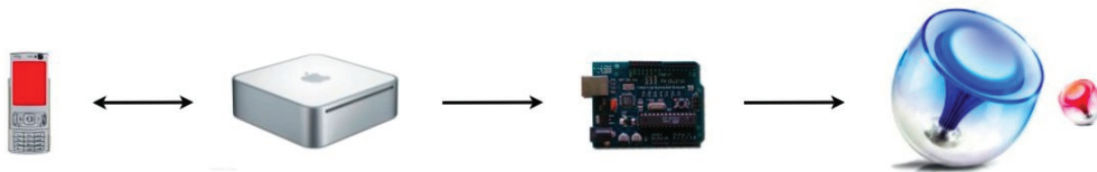
The system would use Bluetooth™ protocol for device and service discovery. To overcome the problem with long Bluetooth™ device discovery times with multiple nearby devices, we decided to use caching of Bluetooth™ device addresses As Bluetooth device pairing is

impossible with long ranges, Bluetooth Service IDs are used as the means to pass color information between the Nodes and the Badges..



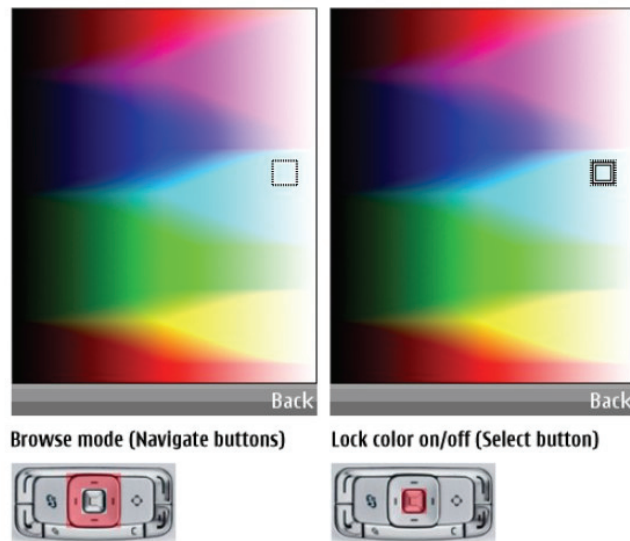
**Figure 5.** Original Illuminate system architecture.

The new idea was to allow for the users direct access to the independent Nodes (e.g Mac Minis), placed in different locations controlling the space's lighting systems via Arduino board (see Figure 6).



**Figure 6.** Simple system for controlling e.g lighting via arduino boards and independent mac minis.

With the envisaged system the user would be able to browse colour scheme with the navigate buttons of her mobile phone and lock a colour on or off with the select (see Figure 7) When the color of the Illuminate node is changed, either manually or by the internal logic component, all registered light control components are notified. In the current version three different components are implemented. The Bluetooth™ light control dynamically updates the light color information (stored as a hexadecimal string) in the Bluetooth™ service attributes, which can in turn be queried by other bluetooth devices to discover the color of the Illuminate node. The Arduino light control transmits the color information to the Arduino board. In the current version, it controls three LEDs, one for each color component. It simply sends the values (0-255) of each of the color components, together with a color identifier. These identifiers correspond to the numbers of the analog pins on the Arduino board, in which the LEDs are inserted. The GUI for manually changing the color is itself also registered as a lightcontroller, and shows the current state of the node's lighting on screen.



**Figure 7.** The selection for colour could occur through the standard phone navigation buttons.

#### 4.1.2.2 Development & Research

The background research and design of the concept was done in TKK/HIIT. Actual implementation and laboratory testing was done by University of Oulu.

#### 4.1.2.3 Related Work

MacIntyre et al (2004) propose the term *aura* to "enrich the current language for designing and analyzing media experiences, especially when using augmented reality, mixed reality and ubiquitous computing technology.". Following the theme of presence and space/space, "Aura provides a necessary complement to the concepts of presence, which is commonly used to evaluate VR applications, and of place, which refers to the more generic significance of places, particularly in CSCW applications."

Ada (Eng et al 2003) is an interactive space developed for the Swiss national exhibition Expo.02 located in Neuchâtel. Ada is being described as a "robot with visual, audio and tactile input, and non-contact light and sound effectors". Unlike the usual robot, Ada is seen as an environment where the visitors of a space are immersed. The sensory stimulation of visitors comes from Ada (and other visitors). The main goal of Ada seems to be the illusion of an intelligent and emotional being that the human visitors are interacting with (Wassermann et al 2006).

Manovich (2006) uses the term "augmented space" to refer to a space which is slowly becoming a reality. He sees that "...this decade of the 2000s will turn out to be about the physical – that is, physical space filled with electronic and visual information". Advances in different technological areas, networking and mobile technologies as enablers, combining video surveillance, cell spaces, public displays, ubiquitous computing, augmented reality, tangible interfaces, wearable computers, intelligent buildings and intelligent spaces we can embed the space with information for the people in the move.

1. Eng, K., Baebler, A., Bernardet, U., Blanchard, M., Costa, A., Delbruck, M., Douglas, R. J., Hepp, K., Klein, D., Manzolli, J., Mintz, M., Roth, F., Rutishauser, U., Wassermann, K., Whatley, A. M., Wittmann, A., Wyss, R. and Verschure, P.F.M.J. (2003) *Ada - Intelligent Space: An artificial creature for the Swiss Expo.02*, presented at IEEE/RSJ International Conference on Robotics and Automation (ICRA 2003), Taipei, Taiwan.

2. MacIntyre, B., Bolter, J.D. and Maribeth Gandy (2004). "Presence and the Aura of Meaningful Places" 7th Annual International Workshop on Presence (PRESENCE 2004), Polytechnic University of Valencia, Valencia, Spain.

3. Manovich, L. (2006). The poetics of augmented space. *Visual Communication*.2006; vol 5. [http://creativetechnology.salford.ac.uk/fuchs/modules/creative\\_technology/architecture/manovich\\_augmented\\_space.pdf](http://creativetechnology.salford.ac.uk/fuchs/modules/creative_technology/architecture/manovich_augmented_space.pdf)

#### 4.1.2.4 Testing and Public Demonstration

The Illuminate has been previously tested in a laboratory environment using four Nodes, one Arduino board (other nodes using flat displays to simulate lights) and eight mobile phones. Considering the limitations of the Bluetooth™ discovery and the fact that the open source Bluetooth™ stack used had to be hacked (in order to enable the usage of the service attributes in conveying the devices' colors between the Nodes and the Badges), the system performed well.

The new implementation having mobile phones controlling the Nodes directly through the color picker application developed (see Figure 7) was also tested in laboratory settings and

The final work was seen as a public space with a pervasive and all encompassing installation atmosphere, where passers-by could affect and adapt their environment (Figure 8), but as stated before, due to the shift of focus from large-scale events to environmental awareness, the development of Illuminate was put on hold and no testing or public demonstration in a real-life urban setting has occurred.



**Figure 8.** As envisaged implemented in an urban environment.

Many alternate ways to light large spaces were considered, but as none fitted with the technology that had been developed to date, and other projects began to consume priority this work was placed on hold.

#### 4.1.2.5 Evaluation

No evaluation of the specifications and research occurred. Has potential to be open source and interest has been shown from art students and designers at University of Art and Design (Taideteollinen korkeakoulu, TAIK) in Helsinki, but to date there has been no real implementation or evaluation.

#### 4.1.2.6 Specification

Hardware and OS	PC hardware, Linux, MacOS X, Arduino board for controlling lightning. J2ME 2.0 enabled mobile phones (with necessary JSRs) as Badges.
Software	<ul style="list-style-type: none"> <li>• Modified Avetana open source Bluetooth™ OBEX implementation</li> <li>• BlueZ Bluetooth™ stack</li> <li>• Component implemented in Java, using the Atelier Infrastructure as a platform for distribution.</li> </ul>
Core Features	<ul style="list-style-type: none"> <li>• Enables control of lightning in a space, as well as controlling the color of the Badges carried by the event participants, in a wireless environment. In new version, active control of the lightning in a space.</li> </ul>
Status	<ul style="list-style-type: none"> <li>• Beta prototype</li> </ul>
Intended users	<ul style="list-style-type: none"> <li>• Any number of users</li> </ul>
Showcases	<ul style="list-style-type: none"> <li>• WP7, others</li> </ul>
Relevance beyond project	<ul style="list-style-type: none"> <li>• Can be used outside the project in similar purposes; uses open source and freely available software and hardware.</li> </ul>

#### 4.1.2.7 Publications

No Publications to date.

### 4.1.3 Dotted Map Tracking

In order to support the augmentation of maps, we have designed an efficient solution for tracking maps on mobile phones in real time. Since state-of-the art tracking solutions for mobile phones are mostly bound to marker tracking, we decided to adopt an approach that merges maps and tracking artifacts unobtrusively (guaranteeing that 99% of the map is not covered by the markers). We overlay a grid of high-contrast dots, easily identifiable in live video images, on standard maps. We believe this to be a reasonable transitioning step from pure marker-based tracking to natural feature tracking of original untouched maps. From now on, we refer to such maps as *dotted maps*.

#### 4.1.3.1 Review and Re-design

*Map tracking (TUG)*. Starting from our existing marker-based tracking solution for mobile phones (StbTracker), we integrated into the tracking pipeline dotted map tracking. The original StbTracker library was historically developed for Windows Mobile devices. In order to cover a broader range of devices, and to favor the integration with WP7, we revised the structure of our runtime software solution to support also the Symbian platform (see Figure 9).





**Figure 9:** Map tracker running on a Symbian device (Nokia N95) displaying a simple augmentation registered in 3D onto a dotted map.

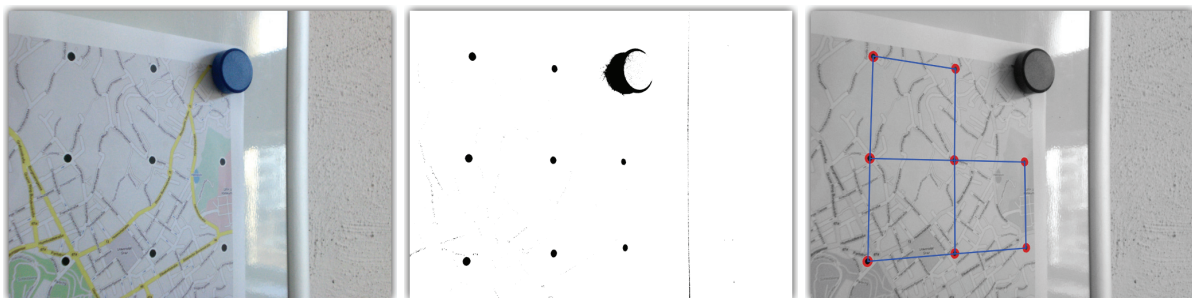
We made our solution available to the project partners as an SDK composed by:

- Static libraries and header files for application development.
- Necessary tools for creating the printable dotted maps and the template databases.

#### 4.1.3.2 Development & Research

*Map tracking (TUG).* We overlay a grid of high-contrast dots on standard maps. The original image patches enclosed by four dots are interpreted as grid cells. These patches are stored as ground truth templates in a database and used at run-time to rapidly and reliably identify particular grid cells from a live video stream.

At runtime, a low threshold value is applied to live video images (typically, the 3% darkest pixels in the whole image). Closed black contours are extracted from the thresholded image (see middle image in Figure 10). The next step identifies groups of 4 dots that form a grid cell (see right image in Figure 10). Since there are usually many possible combinations of four dots, including contours falsely identified as dots, a variety of heuristics is applied to quickly reject incorrect or unlikely combinations.



**Figure 10:** Map with dot markers (left), thresholded camera frame reveals black dots (middle), reconstructed grid cells (right)

The patches contained in all identified grid cells are unwarped one by one and compared to our database using template matching. This process considers the cells with the largest areas first, because they are assumed to provide the best image patch quality. The unwarped image patches are compared against all patterns in the pre-computed database, using normalized cross correlation. The comparison starts with an 8x8 resolution and proceeds to a more expensive test at 32x32 only if the correlation at 8x8 exceeds an arbitrary threshold. Image patches are considered in all their four possible orientations.

If a grid cell is successfully detected, its offset in the grid is determined and its four vertices are used for estimating the 3D pose of the mobile phone with respect to the dotted map, with 6 degrees of freedom. Nonlinear optimization of the camera pose is performed using standard Gauss-Newton iteration.

#### 4.1.3.3 Related Work

The concept of *magic lens* was first introduced in 1993 [Bier, 1993] as a ‘focus & context’ technique for 2D visualisations and was later extended to 3D [Viega, 1996]. AR on handheld devices has been explored with different applications, including peephole applications where the background is used for positioning in virtual space [Henrysson, 2005, Wang, 2006] and multi-user AR games [Wagner, 2005]. A number of papers study target acquisition performance in magic lens pointing [Cao, 2008; Mehra, 2006; Rohs, 2008; Rohs, 2007].

Although carried out in the lab, there are two studies relevant to our work. Henrysson et al. [Henrysson, 2005] piloted positioning and orientation of 3D virtual objects using a mobile phone. They observed that users adopted a bi-manual strategy and sat down rather than stood up to stabilise the phone in hand. Reilly et al. [Ozuysal, 2007] reported an exploratory study where 10 subjects performed pre-defined tasks on an RFID versus non-augmented PDA version. The effectiveness of the technique depended on the size of the map, information tied to it, and the needs of the user. The authors point out that the tasks required little or no spatial knowledge as the trial was conducted in a single location and involved no routes, landmarks, or navigation.

1. Bier, E., Stone, M., Pier, K., Buxton, W., DeRose, T. Toolglass and Magic Lenses: The See-Through Interface. In *Proc. of ACM SIGGRAPH 1993*, ACM Press (1993), 73-80.
2. Cao, X., Jie Li, J., Balakrishnan, R. Peephole pointing: modeling acquisition of dynamically revealed targets. In *Proc. CHI 2008*, ACM Press (2008), 1699-1708.
3. Henrysson, A., Ollila, M., Billinghurst, M. Mobile phone based AR scene assembly. In *Proc. MUM 2005*, ACM Press (2005), 95-102.
4. Mehra, M., Werkhoven, P., Worrying, M. Navigating on handheld displays: Dynamic versus static peephole navigation. *ACM Transactions on Computer-Human Interaction (TOCHI)* 13, 4 (2006), 448-457.
5. Ozuysal, M., Fua, P., Lepetit, V. Fast Keypoint Recognition in Ten Lines of Code. In *Proc. IEEE CVPR 2007*, IEEE Press (2007), 1-8..
6. Rohs, M., Oulasvirta, A. Target acquisition with camera phones when used as magic lenses. In *Proc. CHI 2008*, ACM Press (2008), 1409-1418.
7. Rohs, M., Schöning, J., Raubal, M., Essl, G., Krüger, A. Map Navigation with Mobile Devices: Virtual versus Physical Movement with and without Visual Context. In *Proc. ICMI 2007*, ACM Press (2007), 146-153.
8. Viega, J., Conway, M. J., Williams, G., Pausch, R. 3D magic lenses. In *Proc. of UIST 1996*, ACM Press (1996), 51-58.
9. Wagner, D., Pintaric, T., Ledermann, L., Schmalstieg, D. Towards Massively Multi-User Augmented Reality on Handheld Devices. In *Proc. Pervasive 2005*, Springer (2005), 208-219.
10. Wang, J., Zhai, S., Canny, J. Camera phone based motion sensing: Interaction techniques, applications and performance study. In *Proc. UIST 2006*, ACM Press (2006), 101-110.

#### 4.1.3.4 Testing and Public Demonstration

Mobile map tracking client-server communication analysis and design has been tested in the laboratory. First tests of video streaming from Nokia N95 to PC, were made through WLAN and UI extension for runtime map adaptations was also tested. The MapLens prototype was then extensively tested on the field.

One field trial was held in March 2008 in Helsinki with 5 users and a series of three field trials took place in August 2008 in Helsinki with a total of 37 users. In our trials we enlisted a mix of 37 early-adopters, environmental researchers, scouts and their families to use MapLens, to play an environmental awareness-raising location-based game. A comparative trial was run with a non-AR digital system.

#### 4.1.3.5 Evaluation

Analyses of videos, field notes, interviews, questionnaires and user-created content expose phenomena that arise uniquely when using AR maps in the wild. We report on how augmentation affects the way participants use their body and hands, manipulate the mobile device in tandem with the physical map, walk while using, and collaborate. We found that the MapLens solution facilitates place-making by its constant need for referencing to the physical, and in that it also allows for ease of bodily configurations for the group, encourages establishment of common ground, and thereby invites discussion, negotiation and public problem-solving. Its main potential lays not so much in use for navigation but in use as a collaborative tool.

#### 4.1.3.6 Specification

Hardware and OS	Windows Mobile, Nokia N95 and Symbian OS 6.2
Software	Symbian OS or Windows Mobile, StbMath library and StbCore library
Core Features	Tracking of prepared maps with less visual clutter compared to existing markers
Status	Prototype
Intended users	Any number of users
Showcases	WP7 - Maplens
Relevance beyond project	Demonstrated a the use of an advanced marker with less visual clutter which runs in real-time on consumer smart phones

#### 4.1.3.7 Publications

1. Wagner, Daniel; Langlotz, Tobias; Schmalstieg, Dieter, "Robust and unobtrusive marker tracking on mobile phones," Mixed and Augmented Reality, 2008. ISMAR 2008. 7th IEEE/ACM International Symposium on , vol., no., pp.121-124, 15-18 Sept. 2008
2. Morrison, A., Jacucci, G, Peltonen, P., Juustila, A., & Reitmayr, G. Using locative games to evaluate hybrid technology. British HCI workshop: Using locative games to evaluate hybrid technology, Liverpool, September 1-5, 2008. Accepted as workshop paper and presented.
3. Morrison, A, Oulasvirta, A, Peltonen, P, Jacucci, G, Lemella, S, Juustila, A., & Reitmayr, G. "Like Bees Around the Hive": Interaction In the Wild with a Mobile AR Map. CHI 2009. Submitted.

#### 4.1.4 Mobile Map Tracking from natural features

To accurately overlay information of the image of the map in the mobile phone's display, the 3D pose - translation and rotation - of the phone's camera with respect to the map must be known. To this end we implemented a computer vision-based tracking method that works with any image that provides sufficient texture such as the colored maps as used in the MapLens application.

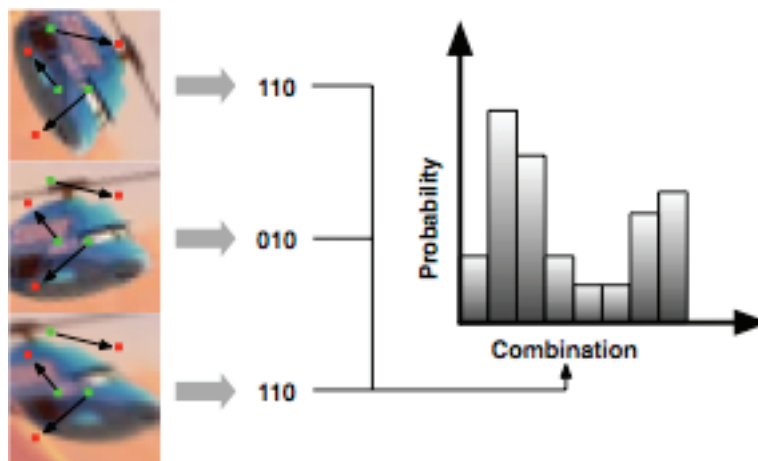
##### 4.1.4.1 Review and Re-design

While the first prototype of the MapLens system described in section 5.1.4 used marked up maps (*dotted map*), a response from the initial user trials in March 2008 was that unmarked maps would be preferable. Thus we investigated the possibility of using *natural feature-based* tracking methods that use visually salient areas of the original printed maps themselves.

#### 4.1.4.2 Development & Research

To track an image, we select distinct feature points in a representative template image and try to find these feature points again in the live image produced by the phone's camera. Because we do not modify the template image and do not require special fiducial markers to be applied, this is a so-called natural feature tracking method. Recent work in computer vision has given rise to a number of methods to accomplish this, however, none of which has been optimized to perform well on platforms with limited processing power such as mobile phones.

Identification of feature points uses a fast classification technique based on intensity differences between pixels in the neighborhood of the feature point (Figure 11). In an offline training stage, the distribution of the sign of these differences (is pixel a brighter then pixel b) under multiple views is learned. For each feature point, the image neighborhood is warped to represent different camera viewpoints and the signs of the differences are collected. The distribution of these vectors of signs (one component for each difference) is calculated and stored in an approximation called Ferns [Ozuysal2007]. This approximation only captures part of the full distribution but is enough to reliably identify the feature point again.

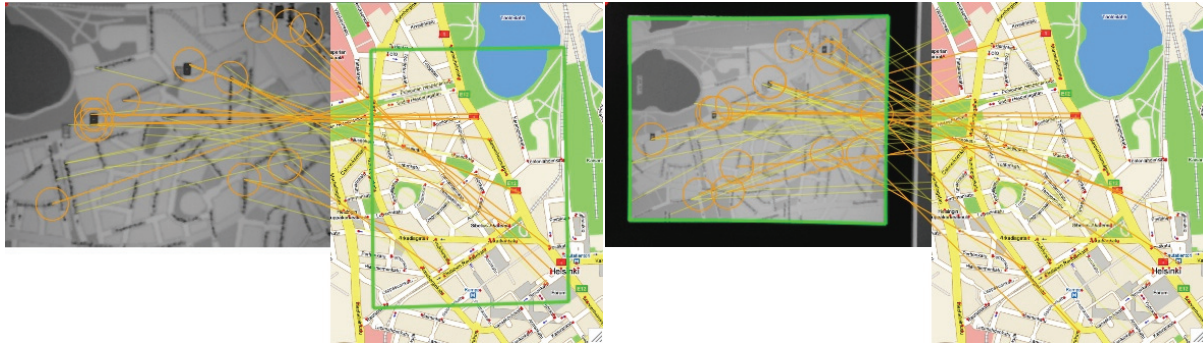


**Figure 11:** Arrows represent differences, the sign of the difference is mapped to 0 or 1 and the vector of signs computed for different views. The occurrence of different vectors yields a distribution of the possible outcome vectors.

At runtime, first interest points in the image are established using an interest point detector. The same differences as used in training are computed for detected points. Then the probability of observing this outcome is computed for each trained feature point. The most likely interest point for each feature point is then chosen as a correspondence. The camera pose can then be recovered from a set of correspondences using robust statistical methods (RANSAC) to reduce the effects of wrong correspondences (Figure 12).

This method was implemented and optimized to operate on the Nokia N95 platform used in the trials. A more detailed account including extensive testing of different parameters can be found in Wagner et. al [Wagner2008]. In general the system operates at between 5 and 12 frames per second, depending on the speed of motion of the camera allowing for interactive use.

For this application a template image was used that allowed operation from about 15cm to 40cm distance between the printed map and the lense of the camera. Note, that the closer border is at limit of the focal range and the camera will not focus properly anymore. The following figure shows the matching operating within these ranges. Tilt between the map and the camera is tolerated up to about 30 deg, while in plane rotation is handled over the full range of rotations. Due to the use of differences in the classification, the method is largely invariant to lighting conditions.



**Figure 12:** The left gray image shows the video input, while the colored maps show the template image. Orange lines connect corresponding feature points between the two images. The green outline show the view in the template image (left) or the outline of the map in the video (right).

#### 4.1.4.3 Related Work

Natural feature tracking approaches differ mostly by the image features that are associated between the video image and a model of the object or environment to be tracked. The dominant trade-off is between the reliability of relocating the features and the computational work required to do so.

Point-based approaches use interest point detectors and matching schemes to associate 2D locations in the video image with 3D locations. The location invariance afforded by interest point detectors is attractive for localization without prior knowledge and wide-base line matching. However, computation of descriptors that are invariant across large view changes is usually expensive. Skrypnik and Lowe [2] describe a classic system based on the SIFT descriptor [3] for object localization in the context of AR. Features can also be selected online from a model [4] or mapped from the environment at runtime [5][6]. Lepetit et. al [7] recast matching as a classification problem using a decision tree and trade increased memory usage with avoiding expensive computation of descriptors at runtime. A later improvement described by Ozuysal et. al [1] improves the classification rates while further reducing necessary computational work. Our work investigates the applicability of descriptor-based approaches like SIFT and classification like Ferns for use on mobile devices which are typically limited in both computation and memory. Other, potentially more efficient, descriptors such as SURF [8] have been evaluated in the context of mobile devices [4], but also have not attained real-time performance yet.

1. Ozuysal M, Fua P, Lepetit V (2007) Fast Keypoint Recognition in Ten Lines of Code, IEEE CVPR 2007
2. Skrypnik, I., Lowe, D., Scene modeling, recognition and tracking with invariant image features. In Proc. of ISMAR 2004, pp. 110-119, 2004
3. Lowe, D., Distinctive image features from scale-invariant keypoints. Int. Journal of Computer Vision, Volume 60, Issue 2, pp. 91-110, 2004
4. Bleser, G., Stricker, D., Advanced tracking through efficient image processing and visual-inertial sensor fusion. In Proc. of IEEE VR 2008, pp. 137-144 2008
5. Davison, A.J., Mayol, W.W., Murray, D.W., Real-time localisation and mapping with wearable active vision. In Proc. of ISMAR 2003, pp. 18-27, 2003
6. Klein, G., Murray, D., Parallel tracking and mapping for small ar workspaces. In Proc. of ISMAR 2007, pp. 225-234, 2007
7. Lepetit, V., Lagger, P., Fua, P., Randomized trees for real-time keypoint recognition. In Proc. CVPR 2005, pp. 775-781, 2005
8. Bay, H., Tuytelaars, T., Gool, L. V., Surf: Speeded up robust features, In Proc. ECCV 2006, 2006.

9. Chen, W.-C., Xiong, Y., Gao, J., Gelfand, N., Grzeszczuk, R., Efficient extraction of robust image features on mobile devices, In Proc. ISMAR 2007, 2007.

#### 4.1.4.4 Testing and Public Demonstration

The natural feature tracking method was used in the field trials of the MapLens system in August 2008 in Helsinki. See section 5.1.4.4 for more details on the results of the trials.

#### 4.1.4.5 Evaluation

Evaluation results of the MapLens system using the natural feature tracking method are described in more detail in section 5.1.4.5.

#### 4.1.4.6 Specification

Hardware and OS	Nokia N95 and Symbian OS 6.2
Software	Symbian OS and StbMath library
Core Features	Pose estimation from natural features
Status	Prototype
Intended users	Any number of users
Showcases	WP6 - MapLens
Relevance beyond project	Demonstrated the feasibility of natural feature tracking on mobile phones.

#### 4.1.4.7 Publications

1. Daniel Wagner, Gerhard Reitmayr, Alessandro Mulloni, Tom Drummond, Dieter Schmalstieg

Pose Tracking from Natural Features on Mobile Phones. Proc. 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, (ISMAR'08), pp. 125-134, Cambridge, UK, Sep. 2008.

2. Morrison, A., Jacucci, G, Peltonen, P., Juustila, A., & Reitmayr, G. Using locative games to evaluate hybrid technology. British HCI workshop: Using locative games to evaluate hybrid technology, Liverpool, September 1-5, 2008. Accepted as workshop paper and presented.

3. Morrison, A, Oulasvirta, A, Peltonen, P, Jacucci, G, Lemella, S, Juustila, A., & Reitmayr, G. "Like Bees Around the Hive": Interaction In the Wild with a Mobile AR Map. CHI 2009. Submitted.

### 4.1.5 Color Table Tracking

#### 4.1.5.1 Review and Re-design

Due to the integration process of the ColorTable and UrbanSketcher the tracking of the color tokens on the centralized round table in the MR Tent needs to be managed. The color tokens also referred to as ColorBlobs are identified by IDs which unexpectedly change due to the computer vision tracking. The tracker only distinguishes between colors but can not distinguish uniquely between different tokens of the same color that's why a ColorBlobManager is needed to assign the tokens to the appropriate virtual objects. Furthermore this Manager should organize the scene graph regarding all the tokens which are dynamically placed and arranged on the table. The major goal here is to implement the Manager for organizing the scene graph as the tracking components already exist and the further development of these localization capabilities are described below.

#### Tracking colored objects on Color Table (AAU):

AAU provides a tracking system for the Color Table application in WP6. In this system, colored objects are detected with color and shape information (see Figure 13). Color tracking technique has to be invariant under different imaging conditions (light source, shadow, and illumination) [1,2]. Four different shapes are detected namely round, square, rectangle and directional triangle. The current version of the system is stable and fast, for examples the processing time of a frame with around 30 objects is ~0.30 second.



**Figure 13:** An example of the ColorTable with different objects (colors and shapes).

#### 4.1.5.2 Development & Research

The demanded software component for managing the tracked color tokens was implemented and tested as required.

#### 4.1.5.3 Related Work

1. T. Gevers and A.W.M. Smeulders, "Color based object recognition", Pattern Recognition 32, pages 453-464, 1999.
2. J.M. Geusebroek, R. van den Boomgaard, A.W.M. Smeulders, and H. Geerts, "Color invariance". *IEEE Trans. Pattern Anal. Machine Intell.*, 23(12):1338-1350, 2001.

#### 4.1.5.4 Specification

Hardware and OS	PC hardware, Linux
Software	C++ and openCV library
Core Features	Tangible tracking
Status	Prototype
Intended users	Any number of users
Showcases	WP6
Relevance beyond project	available

#### 4.1.6 Augmented Map Table

The augmented map table implements a tabletop augmented reality environment for enhancing paper-based artifacts, such as maps, with dynamic information and interaction capabilities. This system is a result of prior work by the University of Cambridge and will be included into showcases WP6 and WP8 to provide tangible user interfaces to the authoring systems deployed by the showcases. For a description of it's general user interface see D4.2.

#### 4.1.6.1 Review and Re-design

Now also a physical map can be used in combination with the ColorTable. The tracking is done via the Map Tracker which broadcasts the position and orientation of the map placed on the table, two different ColorTable Components. Users can move and rotate the map, while the system calculates the positions of the color tokens relatively to the map and translates them into global coordinates.



**Figure 14:** ColorTable with tokens and physical map

The integration of the ColorTable with the Map Tracker provoked also the implementation of a couple of calibration matrices for the ColorTable, which resulted in a much higher consistence between the tangible and digital representation, then the one achieved before. The usage of maps in different magnification levels of one single location will compensate the loss of flexibility in comparison with a digital map while the usage of physical maps greatly increases the tangibility and intuitiveness of the system (Figure 14).

#### 4.1.6.2 Related Work

Implementations of tangible user interfaces use a variety of tracking methods. The metaDESK (Ullmer and Ishii 1997) project used both 6D positioning devices to track handheld tools as well as computer vision techniques to track tangible artefacts placed on a back projection surface. Here marker-based techniques provided robust localisation of the tangible artefacts. Similarly, the ARToolkit (Kato et al. 1999) marker has become a ubiquitous tangible artefact, due to the simple localization implementation. For example, (Regenbrecht et al. 2001) describe a collaborative, tangible desktop environment using solely ARToolkit for tracking. Contrary to these approaches the augmented map table (Reitmayer et al. 2005) uses natural feature-based approach similar to the SIFT (Lowe 2004) feature descriptor to localize unmodified paper artefacts on the table surface. Similarly, tangible tools are localized based on their shape using a Hough transform technique without any further modification. (Wilson 2005) describes a similar approach to interact with paper artefacts in a natural desktop setting.

1. Kato, H. and Billinghurst, M. (1999). Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In Proc. IWAR'99, pages 85–94, San Francisco, CA, USA. IEEE CS.
2. Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *Int. Journal of Computer Vision*, 60(2):91–110.



3. Regenbrecht, H., Baratoff, G., and Wagner, M. (2001). A tangible AR desktop environment. *Computer & Graphics*, 25(25):755–763.
4. Reitmayr, G., Eade, E., and Drummond, T. W. (2005). Localisation and interaction for augmented maps. In *Proc. ISMAR 2005*, pages 120–129, Vienna, Austria. IEEE and ACM, IEEE CS.
5. Ullmer, B. and Ishii, H. (1997). The metaDESK: Models and prototypes for tangible user interfaces. In *Proc. UIST'97*, pages 223–232, Banff, Alberta, Canada. ACM Press.
6. Wilson, A. D. (2005). Playanywhere: A compact interactive tabletop projection-vision system. In *Proc. UIST'05*, Seattle, Washington, USA.

#### 4.1.6.3 Specification

Hardware and OS	The augmented map table operates under Linux on a standard desktop PC with state-of-the-art CPU and graphics hardware. USB or Firewire cameras can be used as input.
Software	The system is written in C++ using the Toon numerical library, libcvd computer vision library and Coin scene graph library. Communication with other software through CORBA and OpenTracker
Core Features	Tracking of printed maps and rectangular interaction tools on a table surface, augmentation of maps and tools with projected graphics.
Status	Stable demonstrator to be integrated with other showcase technology.
Intended users	Any user
Showcases	Mainly WP6, potentially WP8 as user interface to authoring software.
Relevance beyond project	General user interface prototype for Command & Control type environments, such as first responder management.

#### 4.1.7 Bluetooth Media Dispatcher

Bluetooth media dispatcher is a service for receiving and dispatching media files and related metadata over a Bluetooth™ connection using OBEX (Object Exchange) protocol. The service works in Linux and MacOS X and uses a modified open source Avetana and BlueZ Bluetooth™ stack.

The service is used by the MMS Extractor interaction tool, which receives Multimedia messages (MMS) on a Symbian smartphone and extracts media and metadata from the message and sends the media files to the Bluetooth Media Dispatcher. These two tools can be used together in the IPCity showcases for receiving and handling media, e.g. in Large scale events (event participation, for example), Urban renewal (community feedback, for example) and City tales (users entering tales of the city using MMS (recorded sound, 3GPP videos). The media files with metadata are then stored to the HyperMedia DataBase (HMDB) to be accessed by other components and services within the IPCity.

#### 4.1.7.1 Review and Re-design

Implementation to allow MMS and SMS integration for WP7 show case CityWall was begun. The system was ready from Oulu to implement for the European City of Sciences event but TKK was still dealing with porting and new interaction and implementation issues so used a simple SMS and MMS system instead for this event.

#### 4.1.7.2 Development & Research

A new version of the tool will be implemented for the next set of field trials for CityWall in summer 2009. For this reason, the tool has already been ported to S60 3<sup>rd</sup> Edition to enable using it in newer phones. In addition to MMS messages, it will also be handling SMS messages for text only input.

#### 4.1.7.3 Related Work

MMS Entrance is using this component to pass the media ahead to the HyperMedia Database (HMDB). HMDB is used as a storage. Previously, tools and showcases related to this functionality have used email based sending and retrieving of media, requiring an internet connection. This work enables entering and retrieving media to HMDB without internet connectivity.

#### 4.1.7.4 Testing and Public Demonstration

Bluetooth Media Dispatcher will be included in field trial studies for CityWall planned for Summer 2009.

#### 4.1.7.5 Evaluation

Analyses of videos, field notes, interviews, questionnaires and user-created content will be part of the field trials system.

#### 4.1.7.6 Specification

Hardware and OS	PC hardware, Linux
Software	<ul style="list-style-type: none"> <li>Modified Avetana open source Bluetooth™ OBEX implementation</li> <li>BlueZ Bluetooth™ stack</li> <li>Component implemented in Java</li> </ul>
Core Features	<ul style="list-style-type: none"> <li>Imports media into any storage supporting the VFS interface using mobile devices, current implementation will be using HMDB as the storage</li> </ul>
Status	<ul style="list-style-type: none"> <li>Beta prototype</li> </ul>
Intended users	<ul style="list-style-type: none"> <li>Any number of users</li> </ul>
Showcases	<ul style="list-style-type: none"> <li>WP7, others</li> </ul>
Relevance beyond project	<ul style="list-style-type: none"> <li>Is reusable to any similar purpose (entering media using Bluetooth and OBEX)</li> </ul>

#### 4.1.8 Vision Based Localization

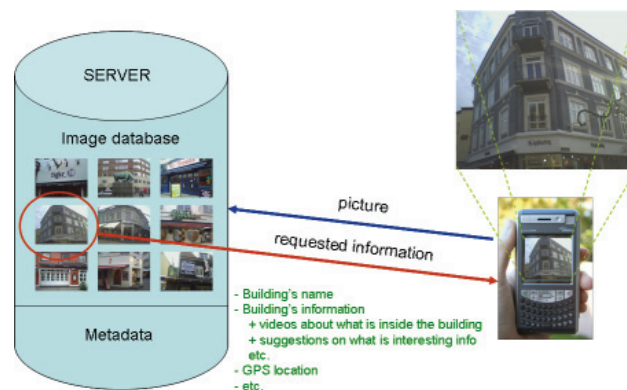
In urban environment, buildings are usually used as objects for detecting locations. We concentrate on developing system for urban building recognition. To that purpose,

techniques using local feature descriptors are commonly used [1]. We employ a recent technique proposed by Brown et al. [2], which is called multi-scale oriented patches (MOPS).

#### 4.1.8.1 Development & Research

##### System for building recognition and localization:

AAU has been working on the communication between a server and a user device (see Figure 15). The server contains of image data, videos and metadata. A user device is a PDA with camera built-in. The data transfer between the two has to be fast for interactive purpose. In the current system, processing time in recognition step is less than one second for each search query. We aim at having the processing time approximately one second in total including the data transfer step.



**Figure 15:** Scheme of data communication between server and user device.

#### 4.1.8.2 Testing and Public Demonstration

A complete application for building recognition has been created where a user can navigate in Aalborg center with a provided PDA by taking picture of an interested building, then send the image to the server (located at AAU), and get back related information of that building (text information, video information, etc.)

#### 4.1.8.3 Evaluation

The system is very stable, with accuracy of 95% with GPS filtering. The processing time is within a second.

#### 4.1.8.4 Related Work

1. K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors", IEEE Transactions on Pattern Analysis and Machine Intelligence, number 10, volume 27, pages 1615-1630, 2005.
2. M. Brown and R. Szeliski and S. Winder, "Multi-Image Matching using Multi-Scale Oriented Patches", In proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, volume 1, pages 510-517, 2005.

#### 4.1.8.5 Specification

Hardware and OS	Windows (possibly also Linux)
Software	C++ and OpenCV library
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	

#### 4.1.8.6 Publications

1. G.P. Nguyen, H.J. Andersen, and M.F.Christensen, “Urban building recognition during significant temporal variations”. In proceedings of the IEEE 2008 Winter Vision Meetings, Workshop on Application of Computer Vision, January 2008.
2. G.P. Nguyen and H.J. Andersen, *Uniqueness filtering for local feature descriptors in urban building recognition*. In proceeding of the International Conference on Image and Signal Processing, Cherbourg-Octeville, France, July 1-3, 2008 Proceedings. Berlin / Heidelberg : Springer Verlag, 2008. s. 85-93 (Lecture Notes in Computer Science).

#### 4.1.9 Model-based Outdoor Tracking

Traditionally, AR systems developed for outdoor use rely on GPS for position measurements and magnetic compasses and inertial sensors for orientation. Within the scope of the project the AR Scouting application (see section 5.4.2) in WP6 and the mobile AR System (see section 5.4.3) application in WP8 both use this solution. While GPS has satisfactory accuracy and performance in open spaces, its quality deteriorates significantly in urban environments. Both the accuracy and the availability of GPS position estimates are reduced by shadowing from buildings and signal reflections. Similarly, inertial sensors are prone to drift and magnetic sensors are disturbed by local magnetic fields encountered in urban environments.

This section describes the development of a computer vision-based localization component for outdoor mobile AR systems that can provide both accurate localization and robustness.

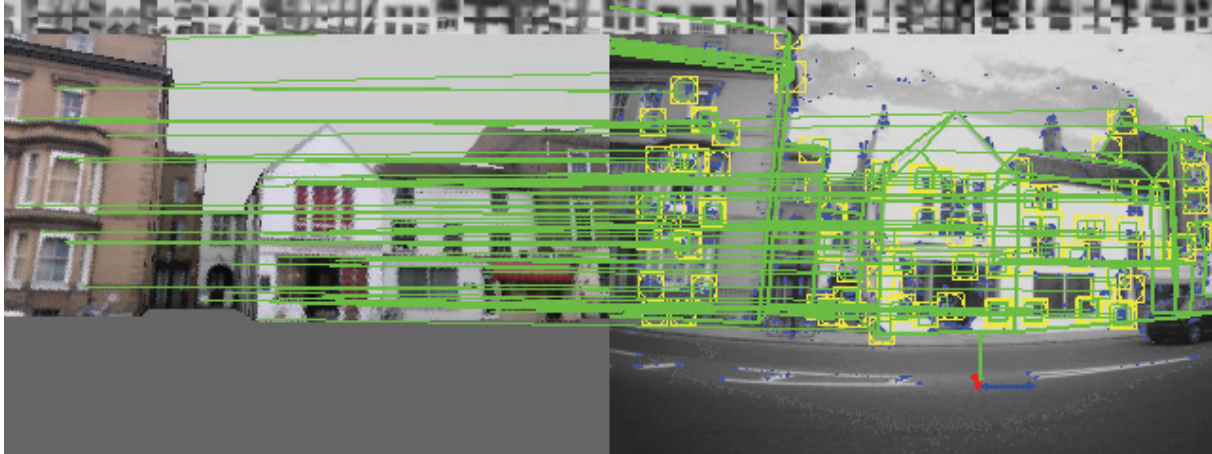
##### 4.1.9.1 Review and Re-design

Model-based tracking approaches appear to be the most promising among the standard vision techniques currently applied in AR applications. While marker-based approaches such as ARToolkit or commercial tracking systems such as ART provide a robust and stable solution for controlled environments, it is not feasible to equip a larger outdoor space with fiducial markers. Hence, any such system has to rely on models of natural features such as architectural lines or feature points extracted from reference images.

Polygonal, textured 3D models [me, Bleser] can serve as a good basis for feature-based tracking. The features to be tracked can be dynamically determined at runtime by performing feature extraction on a rendering of the 3D model from the current pose prediction. This provides two significant advantages: The system automatically performs detail culling and only searches for features that are likely to be visible at the current scale. The system is able to extract an appearance signature for each feature that it wishes to localize in the live feed, thus increasing the accuracy of matching.

#### 4.1.9.2 Development & Research

The system uses a low-polygon textured 3D model as the basis for tracking. From a given starting pose, a view of the model is rendered. Then interest points are detected and matches around them are extracted and normalized to account for lighting changes. In the live video image, corresponding locations for these patches are found and used as 2D-3D correspondences to update the camera pose (see Figure 16).



**Figure 16.** An view from the outdoor tracking system. On the left side, the rendered 3D model is shown while on the right side the current video frame is shown. Corresponding feature locations are connected by green lines. The grey image patches on the top show the corresponding patches in the two images.

Robustness is improved through two methods. Firstly, a tightly-coupled IMU provides updates on rotational velocity, gravity and magnetic field direction between frames and improves the system's ability to deal with fast motions. Secondly, a dedicated recovery system keeps track of video frames that were tracked reliably and matches new images against the old ones, if the normal visual tracking fails.

At startup the system needs to either be in a known location or have a database of known video frames to match against. To remove this limitation, we are in the process of integrating the tracking system with the Vision Based Localization system developed by AAU (see section 5.1.9). A network interface allows the tracker to send the current interface to the server at AAU and retrieve the 5 best matching frames. Then it uses these frames and their known location to initialize the visual tracking.

#### 4.1.9.3 Related Work

Vision based localisation in urban environments is an ongoing topic in both the Computer Vision and Augmented reality community. Johansson & Cipolla[5] investigated pose estimation from a single image in a city scene based on architectural patterns such as grids of windows and doors. This work was extended by Robertson & Cipolla[9] by matching a single image into a database of views with known positions. Such a system provides a very general approach to the problem of localisation without any prior knowledge, but does not solve accurate localisation in realtime.

To operate in realtime, computationally less expensive approaches are required. Imaged-based methods reduce the number of reference images by employing panoramic images. Stricker[12] describes a system that matches video images into a reference panoramic image using Fourier transform-based techniques.

Using line features has been proposed as well. Coors et al.[3] describe a localisation system based on a 3D city model where visible edges are globally matched to edges extracted from a video image. Behringer[1] uses an edge-based method to align the horizon silhouette extracted from elevation grid data with a video image for estimating orientation of a mobile AR system. Our system is an extension of another former system by Reitmayr & Drummond [7] extracted edges from a textured 3D model. Jiang et al.[4] propose a similar system for

outdoor tracking that integrates line-based tracking with gyroscope measurements to accommodate fast rotations. Lines are defined offline in a CAD model and matched using a heuristic line selection method to avoid wrong line correspondences.

The use of point features can be based on feature descriptors, such as Skrypnik & Lowe [11] work using a database of highly discriminate feature descriptors. Alternatively, features can be tracked frame to frame such as in Simon et al.[10] work on a point-based pose tracker that requires no further model and is demonstrated with outdoor scenes. Ribo et al.[8] introduce a hybrid tracking system for outdoor augmented reality using a database of feature points established offline from a 3D model and a set of reference images. Gyroscopes are used to increase the robustness of the system. Bleser et al.[1] describe a system extracting feature points from renderings of a 3D model which is closest to our system.

The recovery system is inspired by work from Klein & Murray [6] on SLAM for small AR workspaces that includes a similar cheap recovery mechanism.

1. Behringer, R. (1999). Registration for outdoor augmented reality applications using computer vision techniques and hybrid sensors. In Proc. IEEE VR'99, Houston, Texas, USA.
2. Bleser, G. and Stricker, D. (2008). Advanced tracking through efficient image processing and visual-inertial sensor fusion. In Proc. IEEE VR 2008, Reno, Nevada, USA.
3. Coors, V., Huch, T., and Kretschmer, U. (2000). Matching buildings: Pose estimation in an urban environment. In Proc. ISAR 2000, pages 89–92, Munich, Germany. IEEE and ACM.
4. Jiang, B., Neumann, U., and You, S. (2004). A robust hybrid tracking system for outdoor augmented reality. In Proc. VR 2004, pages 3–10, Chicago, USA. IEEE.
5. Johansson, B. and Cipolla, R. (2002). A system for automatic pose-estimation from a single image in a city scene. In IASTED Int. Conf. Signal Processing, Pattern Recognition and Applications, Crete, Greece.
6. Klein, G. and Murray, D. (2008). Improving the agility of keyframe-based slam. In Proc. ECCV 2008, volume 5303 of LNCS, pages 802–815. Springer.
7. Reitmayr, G. and Drummond, T. W. (2006). Going out: Robust tracking for outdoor augmented reality. In Proc. ISMAR 2006, pages 109–118, Santa Barbara, CA, USA. IEEE and ACM, IEEE CS.
8. Ribo, M., Lang, P., Ganster, H., Brandner, M., Stock, C., and Pinz, A. (2002). Hybrid tracking for outdoor augmented reality applications. IEEE Comp. Graph. Appl., 22(6):54–63.
9. Robertson, D. P. and Cipolla, R. (2004). An image-based system for urban navigation. In Proc. BMVC 2004, London, UK.
10. Simon, G., Fitzgibbon, A. W., and Zisserman, A. (2000). Markerless tracking using planar structures in the scene. In Proc. ISAR 2000, pages 120–128, Munich, Germany. IEEE and ACM.
11. Skrypnik, I. and Lowe, D. G. (2004). Scene modeling, recognition and tracking with invariant image features. In Proc. ISMAR 2004, pages 110–119, Arlington, VA, USA. IEEE.
12. Stricker, D. (2001). Tracking with reference images: A real-time and markerless tracking solution for out-door augmented reality applications. In Proc. VAST 2001, Glyfada, Athens, Greece. Eurographics.

#### 4.1.9.4 Testing and Public Demonstration

The system is currently in development and undergoes local testing. Future integration with the TimeWarp showcase will make it available to public demonstration and evaluation.

#### 4.1.9.5 Evaluation

The system has not yet been deployed in any showcase, therefore there is now evaluation available yet.

#### 4.1.9.6 Specification

Hardware and OS	UMPC, either Windows XP or Linux
Software	TooN, Libcud, Coin3D
Core Features	Accurate tracking of camera location with respect to a 3D model of a city scene.
Status	Prototype
Intended users	Showcase developers
Showcases	TimeWarp WP8 and AR Scout W6
Relevance beyond project	Tracking in urban outdoor environments is generally a hard topic and of relevance for any location-based mobile system. The developed library will be reused in related future projects.

#### 4.1.9.7 Publications

No publications submitted yet, as the system is still in development.





## 4.2 Storage/Content

### 4.2.1 Distributed Media Entrance and Management

Distributed Media Entrance and Management includes services for entering media (pictures, video, etc) into a central media storage (HyperMedia DataBase; HMDB). This media can then be used by several showcases from different interaction tools, which currently include MapLens, Bluetooth Media Dispatcher and Mobile Media Collector. Also the Atelier infrastructure contains services that use the HMDB (such as the eMailEntrance service).

HMDB is used in the implementation of MapLens (WP7) in caching the media related to an event and produced during the event. Atelier infrastructure is used as a platform for distributed communication and data sharing in the Illuminate showcase (WP7). The components for entering media (Bluetooth Media Dispatcher and eMailEntrance) are already finished and can be used in any trials and showcases for entering media into the HMDB from stationery and mobile devices using different protocols.

The Mobile Media Collector (MMC; See D4.3) uses HMDB as a data storage when uploading locative media to the database. It also fetches media from HMDB in the form of KML, containing placemarks of the media around the mobile device (place identified by GPS on the mobile device). KML exported from the HMDB contains, in addition to the location, author and date/time data, a hyperlink to the actual media, which can be downloaded separately.

#### 4.2.1.1 Review and Re-design

HMDB has been enhanced with thumbnail support. This has so far been tested with MMC, and further tests and trials are expected to be made with other HMDB clients. Especially with mobile clients it is important to be able to fetch a smaller version of a big image, due to network issues and device restrictions.

The Bluetooth Media Dispatcher has been integrated to work with HMDB so that media sent from a mobile phone as a MMS message goes through the MMS Entrance, then to the BT Media Dispatcher and to the HMDB. This has been used in test situations, but as the MMS Entrance and BT Media Dispatcher is trialled with CityWall, this setting needs further testing.

#### 4.2.1.2 Development & Research

Currently the features of the HMDB seems to be sufficient to the client applications. Some features need further testing with trials, but it is expected that no major development efforts in addition to bug fixing is not needed during 2009.

If there are resources, the Browser UI for HMDB may be further developed. Currently the Browser (only „generic“ UI for HMDB) is too slow and cumbersome to be used, so effort directed there can be done if there are needs to this, in addition to the other clients of the HMDB.

Research related to HMDB will be conducted in trials and fieldwork utilizing the HMDB via clients (MMC, MapLens, BT Media Dispatcher).

#### 4.2.1.3 Related Work

MARS, Multimedia Analysis and Revival System, realizes an integrated multimedia information retrieval and database management system, that supports multimedia information as first-class objects suited for storage and retrieval based on their semantic content. MARS includes the conception of multimedia data model, for content indexing and retrieval, and for database management. The MARS system uses only unstructured metainformation calculated from the images and does not support structured metainformation.

Multimedia Data Cartridge (MDC) is an extension to Oracle 9i DBMS providing a multimedia query language, access to media, processing and optimization of queries. It is designed to support MPEG-7 ISO standard called "Multimedia Content Description Interface". MDC have three main concepts: Multimedia Data Model, which is based on MPEG-7; Multimedia Indexing Framework, which is based on GiST framework; and last Libraries for media access and communication to the MDC. (Döller & Kosch 2003.)

Afghan Community Information System for Cultural Heritage Management (ACIS) combines the use of GIS systems with Hypermedia supporting multimedia and semantic web standard to construct a Web Community. In ACIS the metainformation of multimedia files is saved as XML data, the multimedia is saved in multimedia repository supporting different multimedia transport protocol for access and the map information in GIS system. These backend system is used via servlets to combine the information for the client to use. The implementation used Oracle database technologies that consist of technologies for spatial database, multimedia database and XML database (Klamma et.al 2006).

ParaGrab uses relational database to store metainformation about images, that it have been crawled from the Internet. In addition to the structured metainformation it calculates indexes from visual features. Although the main feature of ParaGrab is the crawl of images from the Internet it supports local file uploads. (Joshi et. al. 2006.)

1. Joshi, D., Datta, R., Zhuang, Z., Weiss, W. P., Friedenber, M., Li, J., and Wang, J. Z. 2006. PARAGrab: a comprehensive architecture for web image management and multimodal querying. In Proceedings of the 32nd international Conference on Very Large Data Bases (Seoul, Korea, September 12 - 15, 2006). U. Dayal, K. Whang, D. Lomet, G. Alonso, G. Lohman, M. Kersten, S. K. Cha, and Y. Kim, Eds. Very Large Data Bases. VLDB Endowment, 1163-1166.

2. Mario Döller & Harald Kosch, An MPEG-7 Multimedia Data Cartridge, Proc. of the SPIE conference on Multimedia Computing and Networking 2003 (MMCN 2003), Santa Clara, California, USA, 2003, 126-137

3. Ralf Klamma, Marc Spaniol, Matthias Jarke, Yiwei Cao, Michael Jansen and Georgios Toubekis. 2006. A Hypermedia Afghan Sites and Monuments Database in Geographic Hypermedia. Springer Berlin Heidelberg. ISBN 978-3-540-34237-3. pp 189-209.

#### 4.2.1.4 Testing and Public Demonstration

HMDB has been tested in connection with MapLens and MMC, as well as with WP6 work. These include public trials in Helsinki (MapLens) as well as in Sevettijärvi (MMC).

The Atelier Infrastructure as part of the Illuminate was present in Barcelona review but since the Illuminate consisted of only one set of leds showing the light control, and with two mobile devices working as Badges, this didn't quite well present the functionality nor the features of the system. Due to the changes in WP7, no further trials are yet planned.

#### 4.2.1.5 Evaluation

Analysis of data revealed how participants used MapLens in contrast to a digital system.

#### 4.2.1.6 Specification

Hardware and OS	PC hardware, Linux, MySQL Camera phones, Surveillance cameras (with smtp support)
Software	<ul style="list-style-type: none"> <li>• Atelier infrastructure</li> <li>• Atelier Hypermedia database for storing the media and related metadata</li> <li>• EMailEntrance Atelier External service</li> </ul> <p>All these are implemented with Java</p>
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)

#### 4.2.1.7 Publications

First paper discusses the Atelier infrastructure requirements and implementation. The second paper describes a trial where the HMDB was used to store images received from a remote surveillance camera for supporting architectural design.

1. Juustila, A., Räisänen, T. Atelier Infrastructure for Ubiquitous Computing. Tiainen, Tarja, Isomäki, Hannakaisa, Korpela, Mikko, Mursu, Anja, Paakki, Minna-Kristiina, and Pekkola, Samuli (Eds.) (2007), Proceedings of 30th Information Systems Research Seminar in Scandinavia, IRIS30 (11.-14.8.2007, Murikka, Tampere, Finland), Department of Computer Sciences, University of Tampere, Finland, Series of Publications: D - Net Publications D-2007-9, September 2007, <http://www.cs.uta.fi/reports/dsarja/> 1330 pages. ISBN 978-951-44-7048-6, ISSN 1795-4274.

2. Juustila, Antti; Kangas, Tanja; Räisänen, Toni; Kuutti, Kari; Soudunsaari, Leena (2007). "Bringing Urban Design Site to Studio by using a Remote Surveillance Camera". Workshop paper in: CHI 2007. Imaging the City, Exploring the Practices & Technologies of Representing the Urban Environment in Human-Computer Interaction. 25th International Conference on Computer-Human Interaction, San Jose, California, U.S.A., 28 April-3 May, 2007. See: <http://www.andrew.cmu.edu/user/cdisalvo/chi2007workshop/papers.html>

#### 4.2.2 HMDB Interfacing

Since the introduction of Muddleware to the project partners WP6 has thought about switching from using the Atelier Infrastructure to Muddleware, some integration already happened in year 1 influenced by the fact that Studierstube and Muddleware are tightly integrated.

Furthermore the Atelier framework is focused on message passing, while Muddleware provides more persistence and supports message passing. The number of cross component communication could drastically be reduced by making use of these persistence features.

A generic graphical user interface (MuddlewareGenericGUI) was developed that allows monitoring and manipulating the current state of the system. It hence improves the development and debugging of the whole system.

#### 4.2.2.1 Review and Re-design

The redesign of HMDB has been focused on performance improvements on the other hand, and adding new features needed in the showcases on the other hand. Performance improvements are needed in WP6 in order to be able to access many large multimedia files in certain scenarios. New features are related to the WP7, where we need importing of media from outside sources such as the Flickr image service. Also the KML export feature is enhanced to support different types of mobile devices (smaller KML files with less metadata) and other clients (richer KML and metadata in the KML file). Also, a php interface to HMDB is implemented to support web based applications to access and import media to the HMDB. Tutorials and examples have been provided to partners on using the HMDB from various applications and HMDB APIs.

#### HMDB Web Interface

The TUW is continuously improving its webinterface for the HMDB to allow comfortable administration of the contents inside the Database. Some of the searching and browsing functionalities have been now implemented by directly accessing the MySQL Database to achieve better performance, while many of the other functions still make use of the HMDB Java API. Furthermore a list of all existing keywords allows now fast access to the different types of stored content and gives a quick overview of used tags.

#### 4.2.2.2 Related Work

A number of articles have been published in the field of middleware development and integration of databases. The most related work can be found in the area of ubiquitous computing and event passing frameworks. CORBA and other communication frameworks are currently investigated and will probably be used in the system in the future.

#### 4.2.2.3 Specification

Hardware and OS	PC hardware, Windows, Linux, MacOSX
Software	Java, C++
Core Features	Cross programming language communication (Java <-> C++)
Status	stable prototype
Intended users	Any number of users
Showcases	WP6,7,9
Relevance beyond project	Any mixed reality or ubiquitous project based on distributed input/output processing

#### 4.2.3 ContentManager

ContentManager is software designed for downloading content from Internet's social media services such as Flickr and to insert this downloaded content into a database. Currently ContentManager is used by CityWall in WP7 for retrieving content to be shown on the display.

##### 4.2.3.1 Review and Re-design

ContentManager was implemented for WP7 to replace the Flickr download script used by CityWall earlier. There were three reasons for this: 1) Media from other services (YouTube) was also required that the current script couldn't download 2) CityWall's storage of meta data was changed to use database instead of simple XML files and the current script did not support this and 3) the current download script could download only a one file at a time, making the download times quite long.

ContentManager is a collection of Linux Bash shell and PHP scripts that use the social media service's public API's to download media files. A record of downloaded files is kept in a MySQL database. Downloading many files in parallel and checking for stalled downloads is supported. The downloaded files are stored on disk and the meta-information relating to them is inserted into the production database of the target application (at this point only CityWall).

#### 4.2.3.2 Development & Research

ContentManger was developed in 2008.

#### 4.2.3.3 Testing and Public Demonstration

ContentManger was tested in public with CityWall at European City of Sciences exhibition in Paris 2008.

#### 4.2.3.4 Evaluation

Data from field trials is yet to be analysed from European City of Sciences exhibition

#### 4.2.3.5 Specification

Hardware and OS	PC hardware, Linux
Software	<ul style="list-style-type: none"> <li>• MySQL (open source)</li> <li>• Linux bash shell (open source)</li> <li>• PHP (open source)</li> <li>• phpFlickr (open source)</li> <li>• phpYoutube (open source)</li> <li>• youtube-dl (open source)</li> <li>• ContentManager scripts written in bash and PHP</li> </ul>
Core Features	Imports media meta-data and files from social media services into database
Status	Stable prototype
Intended users	Any number of users
Showcases	WP7
Relevance beyond project	Is reusable to any similar purpose (retrieving media from social media services)

#### 4.2.4 Muddleware

The idea of Muddleware is to provide a general framework for distributed off-line communication of different participants (Wagner 2007). 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. Muddleware is mainly used in WP6 and the development work in this context is also described in the previous Section 4.2.2.

#### 4.2.4.1 Review and Re-design

Redesign issues here concerning Muddleware are addressing the need emerging out of showcase demands (see WP6). The use of Muddleware for interfacing the UrbanSketcher application allows the communication for a wide range of devices or other applications like the ColorTable. The clear demand here is a Muddleware based API which contributes to merging the currently two applications UrbanSketcher and ColorTable. The API should give remote access to functions like sketching tool selection, scene input and output functionality and scene reconfigurability. To be able to realize this functionality a Scene Graph node is needed to synchronize Muddleware node elements with fields of the Open Inventor.

#### 4.2.4.2 Development & Research

The SoSceneField implementation realizes the demand for persistent open inventor fields and serves as a basis for developing an API open for various devices and applications. The node takes care of updating the MWDB field value with the value of the corresponding open inventor field and as well as the other way around, keeping the two field types synchronized.

#### 4.2.4.3 Testing and Public Demonstration

See Section 5.4.6 on Urban Sketcher and Expression Integration

#### 4.2.4.4 Specification

Hardware and OS	PC hardware, Windows, Linux, MacOSX
Software	Java, C++
Core Features	Cross programming language communication (Java <-> C++)
Status	stable prototype
Intended users	Any number of users
Showcases	WP6, (7,9)
Relevance beyond project	Any mixed reality or ubiquitous project based on distributed input/output processing

#### 4.2.4.5 Publications

1. Wagner Daniel, Schmalstieg Dieter, „Muddleware for Prototyping Mixed Reality Multiuser Games“, Proceedings of IEEE Virtual Reality 2007 (VR2007), IEEE, IEEE, 2007-March

### 4.2.5 Audio/Video Streaming

Video clips and audio files can enhance the usability and operating range of VR/AR applications in many different ways. Therefore, a core component of the MORGAN VR/AR framework is the Media Library. The Library provides the entire functionality to augment the real environment with media objects like images, sound and video clips.

Now, the library has been extended supporting audio and video streaming as well as a combined file capturing. The developed capturer can be divided into three different classes:

- **Live-Streaming Capturer** enabling broadcasting, video conferences and out-of-core applications
- **Media-Streaming Capturer** enabling efficient distribution of locally stored media files in multi-user scenarios and client-server architectures
- **File Capturer** enabling documentation and evaluation of user attitudes

To be compatible to common media players and other streaming applications the standardized Real-Time Streaming Protocol RTP/RTSP is used. Nowadays, RTP/RTSP is

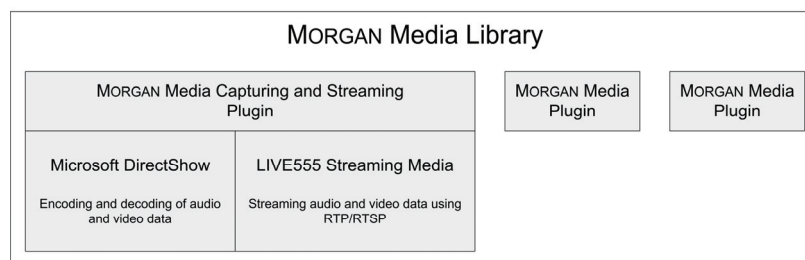
supported by several media players and applications like e.g. the VLC Media Player or the QuickTime Player.

#### 4.2.5.1 Review and Re-design

The MORGAN Media Library has been extended with new video and especially audio streaming features. In comparison to our former design the amount of supported video compression techniques has been extended. Additionally, fully support for audio streaming has been added. Now the streaming library supports most common audio and video compressor techniques e.g. H263, MPEG4, MJPEG, MP3 and G723. Therefore, on less powerful hardware devices less time consuming encoding methods can be selected and vice versa. Additionally, the CPU load has been reduced by taking advantage of already encoded video or audio data. A once encoded audio or video frame can be used for several streaming and file writing tasks concurrently. To support the new streaming features the device abstraction layer DEVAL is being extended and modified. Eventually, DEVAL will provide a simple interface which is independent of the streaming content.

#### 4.2.5.2 Development & Research

The streaming component is realized as a MORGAN plugin based on the MORGAN Media Library. The plugin is comprised of the Microsoft DirectShow framework and the LIVE555 Streaming Media library. While DirectShow is used to encode and decode the media data LIVE555 streams and receives the data over RTP/RTSP. Figure 17 shows the structure of the MORGAN Media Library and the realized streaming plugin.

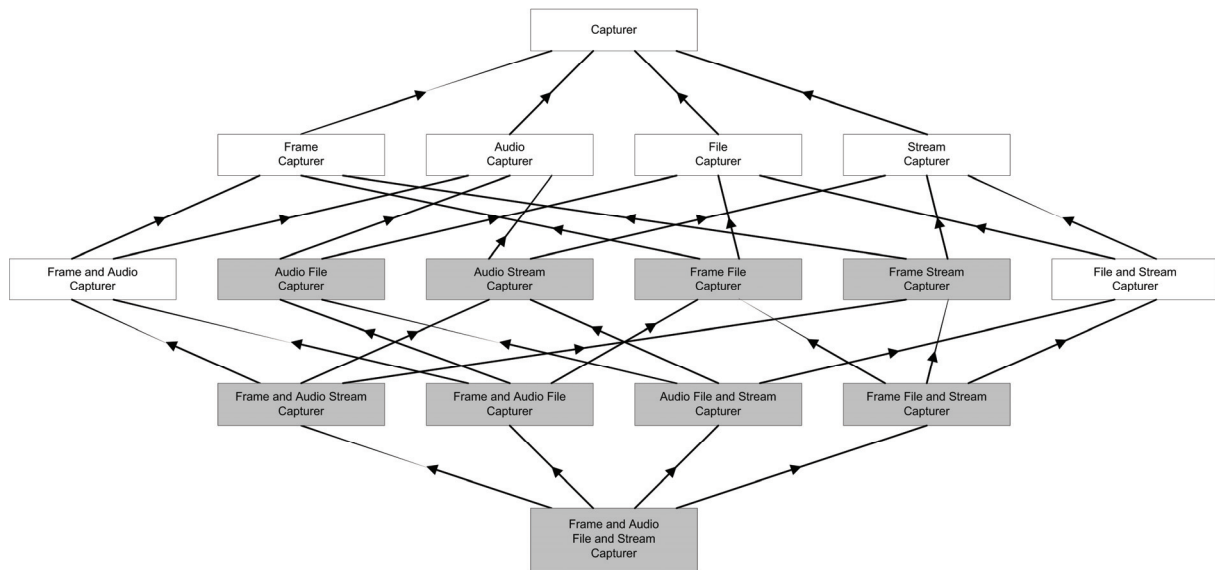


**Figure 17** Structure of Morgan Media Library and the streaming plugin

The library provides several types of capturer appropriate for different media types and tasks. All capturers are defined in a complex inheritance hierarchy sharing common objects and functionalities. The available MORGAN capturers are based on an arbitrary combination of the following abstract capturer:

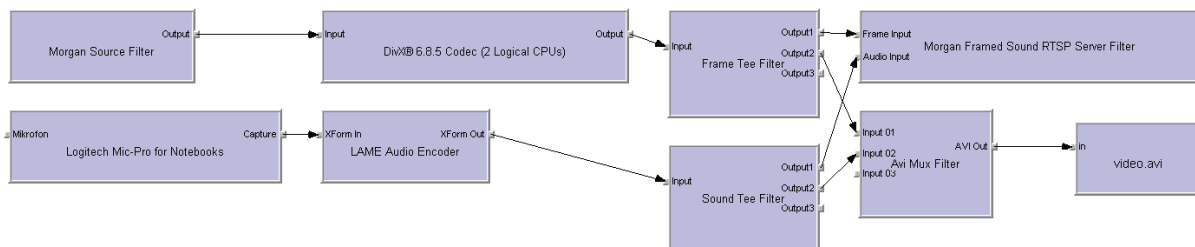
- **Frame Capturer** able to capture frame objects
- **Audio Capturer** able to capture sound objects
- **File Capturer** able to capture data into a media file
- **Stream Capturer** able to stream data to a remote client

Therefore, now MORGAN provides e.g. a simple audio stream capturer as well as a complex audio and video stream capturer which concurrently creates a media file. The structure of the capturer inheritance hierarchy is given in Figure 18.



**Figure 18:** Structure of the capturer inheritance hierarchy, white capturer are abstract

As already mentioned the streaming library uses DirectShow for data encoding. Each capturer holds an internal DirectShow filter graph handling the entire capturing, encoding and streaming tasks. Therefore, several new DirectShow streaming filters have been implemented. The streaming filters encapsulate a RTSP server or client using LIVE555. Depending on the selected DirectShow encoder the appropriate streaming RTP payload type is selected and registered for streaming. Figure 19 shows such a DirectShow filter graph combining audio and video streaming and file capturing.



**Figure 19:** DirectShow filter graph for audio and video streaming and file capturing

#### 4.2.5.3 Related Work

1. H. Schulzrinne, A. Rao and R. Lanphier, Real Time Streaming Protocol (RTSP), RFC2326, 1998



#### 4.2.5.4 Specification

Hardware and OS	Windows XP
Software	LIVE555 Streaming Media Microsoft DirectShow C++
Core Features	Video and audio streaming to local files and remote clients. The standardized Real-Time Streaming Protocol RTP/RTSP is used allowing common media player reception of audio and video streams
Status	stable prototype
Intended users	Any number of users
Showcases	WP8
Relevance beyond project	Any mixed reality or ubiquitous project based on distributed input/output processing

## 4.3 Computation

### 4.3.1 Spatial Sound

Support for spatial sound was a new request. We address this demand by integrating OpenAL into the AR/VR Framework Morgan allowing us to render sound depending on the user's position. In addition, we support audio effects to model a more convincing sound scape.

Spatial Sound support was also added to the the ColorTable Application with Max/MSP.

#### 4.3.1.1 Review and Re-design

In the first prototype of the soundapplication for the ColorTable the following basic functions were implemented.

- Receiving OSC data messages from the ColorTable.
- Connecting the tokens with the assigned soundfile.
- Spacial positioning in the 360° Soundfield with Ambisonics
- Playback of the soundfile.

The soundapplication was designed to change dynamically its audiochannels with adding and removing every token. Due to performance limits and simplifying the development workflow we changed to a fixed amount on audiochannels.

The top few video-tracking implies also certain interruption caused through shadows and moving the tokens to fast. This results in annoying soundfile interruptions. Smoothing functions had to be implemented to get a more homogeneous Soundscape. The behaviour of the tracking had to be better and also the soundapplication got its smoothing by implementing fadein and fadeout algorithms. We are still testing these two smoothing items to improve the homogeneous aspects of the created soundscape.

Another big item is the mixing of the individual volume levels of the different soundfiles of the library. Therefore we added a list of volume values witch relates to the data routing and reads its volume value for each individual soundfile. This was a big challenge for the balance of the soundfiles in the soundscape. As a next step, we did some test to get some kind of matrix for finding the right volume levels for each soundfile. The difficulty of this task is the "natural" volume curve of the Ambisonics witch we have to be aware.

#### 4.3.1.2 Development & Research

##### General

After a few tests with PureData which were not very successful we decided to use Max/MSP to develop the Audioapplication for the ColorTable. As a main item we use the Ambisonics Algorithm to generate the 360° surround soundfield related to the created scenarios of the users. The Ambisonic decoding and encoding Algorithms are implemented with some Max/MSP externals which were developed at the ICST-Zürich by Phillipe Kocher and Jan Schacher. (released under the terms of the GNU Lesser General Public License LGPL)

Additional to the Ambisonics externals we build a framework to get all the necessary data send from the ColorTable to Max/MSP routed and connected with the soundfiles inside our audio library. For the playback of the Soundfiles a special audiochannel had to be designed. The audiochannel includes fadein and fadeout algorithms and the automatisation of the individualized volume levels for each soundfile.

##### Routing

The Soundapplication is connected with the ColorTable via Ethernet and gets its data with the OSC protocol, receiving them at the [udpreceive 55555] object of the Soundapplication.

The data from the ColorTable is containing the space coordinates of the tokens on the Colortable, the number of the connected soundfile, as well as the play and stop commands. The space coordinates were send in the polar coordinate (AED) format.

Azimuth: in degrees, 0° due front, angle increases clockwise

Elevation: in degrees, 0° horizontal, angle increases upwards

Distance: in units (from 0. to 10.)

angles can take any positive or negative value

The AED coordinates are pre-prepared in the ColorTable Application. The relative AED positions of the tokens to the different hearing positions and the different scales of the maps on the table are calculated by the ColorTable Application. The flows are pre-prepared in the way of randomly choosing a soundfile out of a thematically assigned folder of the audio library. By changing the panorama view the data messages will contain the assigned 4 panoramasounds with their predefined positions.

#### 4.3.1.3 Related Work

The application of spatial audio is increasingly important in Augmented Reality (AR) applications (Yamazaki 2000). AR is a field of computer science which enables the user to perceive reality superimposed with computer generated data like graphics, audio, haptics, etc. Spatial audio is widely used in virtual environments and provides additional cues and enhancements like useful spatial information for human performance and an increase of involvement, realism, immersion and sense of presence (Lombard 1997, Lentz 2006). However, the possibility of noticing the natural acoustic environment differentiates the concept of AR audio from the traditional concept of a virtual audio environment, where a user typically dives into a completely synthetic acoustical environment. In contrast, AR audio combines real and virtual sound in a form that virtual sounds are noticed as an extension of natural sounds. Compared with real sound, virtual sound either is artificially produced or previously captured in real environment.

Current investigations like (Zhou 2004) show that three-dimensional sound with non-individualized HRTFs increases realism and immersion in AR environments. Zhou et al. examined how 3D sound improved the localization of visual objects, the immersion and the operating time of tasks in an AR application. They showed that 3D sound contributes significantly to the improvement of the number of correct localizations and depth perception of visual objects.

Though quite some AR audio systems exist (Härmä 2003, Lyons 200), in most cases they take place within a closed and/or firmly defined area. In contrast, our research focuses on spatial acoustics in mobile outdoor AR systems. Additionally to the visual representation, our AR system provides the user with binaural signals and thus creates a three-dimensional audio space around him.

Furthermore, recent advantages in digital signal processing (DSP) and audio hardware allows creating convincing interactive audio environments simulating realistic 3D audio effects. This is mainly driven by game companies, as sound is essential for a game experience. Several technologies have aroused that support software or hardware accelerated rendering of 3D audio output and room acoustics (Gardener 1999). A quasi-standard for environmental audio in games is the DSP library implementation Environmental Audio Extensions (EAX) (Creative Labs, Inc. EAX 4.0 Documentation, 2007) that adds room acoustic effects to Microsoft's DirectSound3D or alternatively OpenAL (OpenAL SDK for Windows, 2007). Augmented Reality and Virtual Reality (VR) applications could profit from these advantages, but still most applications disregard advanced audio effects due to the lack of proper modelling possibilities in X3D (Extensible 3D) (X3D and Related Specifications - X3D International Standards), which only includes spatial sound (i.e. sound localized in 3D). An X3D scene description may contain an *AudioClip* node to include an audio file and a

*Sound* node to position that sound source. This position is defined as 3D point and an ellipsoidal pattern to control distance dependent linear attenuation.

MPEG-4 BIFS (Binary Format for Scenes) which is based on VRML concepts, and its extension Advanced Audio BIFS (AABIFS) (Väänänen 2004) allows to describe virtual acoustics with room effects. Special audio nodes to create an audio scene graph were added, that overcomes the limitations of the X3D sound specification. However, this specification is optimized for streaming and doubles some of the grouping nodes. In (Topol 2001) a similar approach is given, integrating a perceptual acoustic model more closely to the X3D concept. A grouping node (called *RoomEffect*) for attaching a room effect to its children is added as well as an audio filter property (called *soundFiltering*) in the geometry node *Shape*. Therefore, the acoustic scene description is closely linked to the geometrical scene description and it is not possible to describe an audio-only scene.

Our approach to enhance the X3D sound description is based on the idea of giving a sound an appearance. We chose a perceptual approach, too, but benefit from EAX by exposing presets to the scene description. In contrast to Topol, our approach allows to describe audio-only scenes. Furthermore, our lean extension does not double grouping nodes as AABIFS does, but transfers the concept of X3D textures to specify the visual properties of a geometry to the concept of effects to specify the audible properties of a sound.

1. Yasuhiro Yamazaki and Jens Herder: Exploring spatial audio conferencing functionality in multiuser virtual environments, The Third Int. Conf. on Collaborative Virtual Environments (San Francisco, USA), ACM, September 2000, pp. 207–208.
2. Matthew Lombard and Theresa Ditton, At the heart of it all: The concept of presence, J. Computer-Mediated Communication 3 (1997), no. 2.
3. Tobias Lentz, Ingo Assenmacher, Michael Vorländer, and Torsten Kuhlen, Precise Near-to-Head Acoustics with Binaural Synthesis, Journal of Virtual Reality and Broadcasting 3 (2006), no. 2.
4. Z. Zhou, A. D. Cheok, X. Yang, and Y. Qiu, An experimental study on the role of 3d sound in augmented reality environment, Elsevier, Interacting with Computers 16 (2004), 989–1016.
5. A. Härmä, J. Jakka, M. Tikander, M. Karjalainen, T. Lokki, H. Nironen, and S. Vesa, Techniques and applications of wearable augmented reality audio, in AES 114th Convention Paper (2003).
6. K. Lyons, M. Gandy, and T. Starner, Guided by voices: An audio augmented reality system, 2000.
7. Creative Labs, Inc. EAX 4.0 Documentation
8. W. G. Gardner. 3D Audio and Acoustic Environment Modeling, Wave Arts Inc., 1999
9. OpenAL SDK for Windows, referenced Nov. 2007. Available at <http://developer.creative.com/>
10. A. Topol and F. Schaeffer. Enhancing sound description in VRML. In ICMA ICMC'2001, Havana, Cuba, September, 2001.
11. Riitta Väänänen and Jyri Huopaniemi. Advanced AudioBIFS: virtual acoustics modeling in MPEG-4 scene description. IEEE Transactions on Multimedia, no. 5, volume 6, 2004.
12. Web 3D Consortium, <http://www.web3d.org>, referenced Nov. 2007.
13. X3D and Related Specifications - X3D International Standards <http://www.web3d.org/x3d/specifications/>, referenced Nov. 2007.

#### 4.3.1.4 Specification

Hardware and OS	<ul style="list-style-type: none"> <li>• Creative Labs Sound Blaster AUDIGY 2 ZS notebook sound card</li> <li>• Koss Porta Pro headphones</li> <li>• Windows XP</li> </ul>
Software	OpenAL, Morgan
Core Features	<ul style="list-style-type: none"> <li>• Sound RSG <ul style="list-style-type: none"> <li>○ Rendering specialized sound</li> <li>○ Rendering sound effects</li> </ul> </li> <li>• X3D XSG <ul style="list-style-type: none"> <li>○ Loading of Sound and Audio Clip node</li> <li>○ Audio Effect Extensions</li> </ul> </li> </ul>
Status	beta prototype
Intended users	Showcase Developer
Showcases	Showcase 8, Showcase 9
Relevance beyond project	Spatial Audio improves user experience and provides additional cues like spatial information for human performance.

#### 4.3.1.5 Publications

Katharina Garbe, Iris Herbst, Jens Herder: Spatial Augmented Reality, Proc. of 10<sup>th</sup> Int.Conf. on Human and Computers 2007 (HC2007), Dec, 13<sup>th</sup>-15<sup>th</sup>, 2007, Duesseldorf, Germany

Katharina Garbe, Iris Herbst: Extending X3D with Perceptual Auditory Properties, Poster at VR 2008

## 4.4 Mobile AR/MR

### 4.4.1 MR-Tent

The design discussion of the MR tent enclosure was continued during the second year and the Tent different elements of the MR Tent were produced. After tests with the MR Tent and technical equipments on site at the general Meeting in Graz the first design studies according to the requirements of the WP6 partners were refined.

#### 4.4.1.1 Review and Re-design

The first testing of the MR-Tent in Graz during the general meeting led to some necessary adjustments (Figure 20). A few corrections concerning the tent construction are needed, such as allow lifting the canvas for better water drainage and a proper fixing of the entrance curtain to keep it in place in windy weathers.



**Figure 20:** MR Tent first set up

During design process, the possibility of partly lifting the canvas was discussed controversial and it now became clear that facilitating to opening the enclosure will have a positive effect on taking into account the actual environment of the site.

Another demand is to enhance the flexibility of setting up the technical equipment inside the tent. In order to enable the variability of adjustments of technical equipment for the color table application, sound system and urban sketcher, the bridge construction will be upgraded with additional beams and relocatable joints.

To ease handling, prevent messiness and help organizing people's movement during the workshop, the distribution of technical equipment must be arranged more properly: wires will be conducted beneath the platforms and two small stand-up tables that carry lap-tops and other equipment, which is needed for running the application will be mounted directly on the bridge on each of the tent's side walls in order to enlarge the actual space of interaction between participants and the color-table/urban sketcher (Figure 21).



**Figure 21:** Distribution of technical equipment.

Although the design of the tent already provides a sun-shield to control light condition for the see-through projection the first tests made clear that the possibility of additional dim-outs is necessary. A black box at the size of the window will be placed outside the tent and protect the see-through from flare. We will also further investigate in additional see-through materials (Figure 22).



**Figure 22:** See-through, Video Augmentation and Panorama Screen

#### 4.4.1.2 Development & Research

The needed corrections concerning the tent construction for water drainage or fixing of the entrance curtain was easily solved. The acoustic layer of the environment already strongly affects the atmosphere within the Tent and we now also enabled lifting the canvas on one of the sides.

The more challenging task concerned the organization of the interior space. Although the movement of people and the arrangement of technical equipment was carefully documented in earlier workshops (e.g. TGI-Workshop), the effect of messiness and people standing in their own way is strongly related to the possibility of storing the necessary equipment. We upgraded the bridge construction with additional beams, brackets and clamps and we also added 6 aluminium pipes and supports to fix up to 8 loudspeakers according to average ear-levels of the future participants (Figure 23).



**Figure 23:** Panorama of the MR Tent interior with Position of Loudspeakers

Further furniture like the stand-up tables with appropriate chairs, create two side areas where the laptops are stored and can be easily accessed by research staff. Also the redesign of the ColorTable that now houses the printer clears up the space in order to let participants freely manoeuvre around the table (Figure 24).



**Figure 24:** Stand up Tables and Color Table

In order to protect the see-through from flare an additional sunshield at the size of the window is placed outside the MR Tent. In order to enhance flexibility in using the different types of projection and viewpoints of the panoramas we decided to place a roller projection screen above the see-through that can be easily handled (Figure 25).



**Figure 25:** Sun-Shield for Light Control; Additional Roller Projection Screen

#### 4.4.1.3 Testing and Public Demonstration

Apart from the set up in Graz Graz, we tested the MR Tent during two workshops which provided different settings concerning environmental aspects and groups of participants. At the WP6 Workshop in Cergy-Pontoise the MR Tent was installed on site where the spatial set up created different interior and exterior zones for discussion, demonstration and breaks. The second set up took place in an interior space: the large hall of the Grand Palais in Paris at the European City of Science (Figure 26).



**Figure 26:** MR Tent in different Environments: Cergy-Pontoise and Grand Palais



#### 4.4.1.4 Evaluation

With the corrections of the structure and canvas the MR-Tent provides now useful setting as mobile mixed reality environment for weather conditions are to be considered very carefully. The different types of openings, such as the main entrance with adjustable curtain, side entrance and See-Through creates a useful configuration of exterior and interior spaces. Although additional furniture allow a better organizing people's movement as well as storage and handling of the technical equipments, the size of the Tent limits the number of participants, that have access to the ColorTable and urban Sketcher at the same time. Transportation and setting up the Tent is still changeling, especially the amount of weight that is needed in order to provide stability in heavy winds. Although setting up the Tent is not complicated at least four people are needed, and practical knowledge decreases the amount of time it takes.

Further testing of See-Through material created better results concerning the visibility of virtual objects onto the real scene and but led to some unexpected result. Depending on the standpoint of the participants, the loose layering of a dark foil and the perforated membrane causes areas of reflections (Figure 27).



**Figure 27:** See Through: Perforated Membrane with Dark Foil

#### 4.4.1.5 Specification

Hardware and OS	<p>Tent Hardware:</p> <ul style="list-style-type: none"> <li>• Sanyo PLC-XP57L projector, 5500 ANSI lumens</li> <li>• PTU (pan-tilt unit) by RoboSoft</li> <li>• Sony HDV 1080i camera</li> <li>• A.R.T. optical tracking system</li> <li>• Interaction table with a projector-camera setup</li> <li>• Different kinds of projection walls</li> </ul>
Software	Urban Sketcher and Color Table
Core Features	Hardware equipment for the MR tent
Status	beta/stable prototype
Intended users	Urban Presence Explorers
Showcases	WP6 and WP7
Relevance beyond project	yes

#### 4.4.2 AR Scouting

The capability of roaming the real environment is considered necessary, in order to obtain different and dynamic points of view over the augmented world. The idea of AR Scouting is to introduce a person in an MR application scenario who is able to interactively explore an environment from an egocentric viewpoint. The exploration is typically driven by instructions of the user of a remote application who has access to an exocentric view of the environment. By supplying the AR Scout with mobile sensors he becomes able to collect data from the environment which is selected online by the remote instructor and therefore very specific to its current needs. By using the idea of the AR Scout, a very task driven use of all available data is possible, allowing for even interactive refinements of the data which is going to be augmented in the remote application. The delivered data can be used either directly by simply presenting them registered to the remote application or the data can be used as input for further computations which afterwards leads to the augmented data. Such a computation could be a 3D reconstruction system, which outputs a textured 3d mesh initially created or refined by images delivered by the AR Scout.

We developed a system for streaming live video and positioning information from *scouts* that are sent to physically explore the environment with a handheld device. Furthermore, we integrated into our desktop systems the capability of receiving and augmenting such live video streams.

##### 4.4.2.1 Review and Re-design

We implemented a first system for streaming live videos in real time over UDP connections. This system works on any camera-equipped device running Windows platforms. We extended this preliminary system with the ability of streaming both video and information about the spatial position and orientation of the scouts. Our system interfaces to an electronic compass, a GPS sensor and an orientation sensor. For the design of the scouting system, we decided to target only Windows-based devices, e.g. UMPCs. This design choice was taken in order to have sufficient computing power for performing real-time compression

of the data to stream and therefore reducing the network bandwidth required to transmit such data (Figure 28).



**Figure 28:** UMPC scouting device

For network communication, we decided to rely on WLAN connections up to the coverage of our network infrastructures. In the case of loss of connectivity, we designed as a fallback the capability to save the whole stream of information into the local hard drives of the scouts. This adds value to the scouting system, allowing exploration of areas where no network coverage is possible, and permitting archiving, easy recovery and replay of previous scouting sessions.

#### 4.4.2.2 Development & Research

For solid network communication, we chose to rely on a standard protocol (RTP) and an already available and established library (Live555: <http://www.live555.com>). On the server side, we designed and implemented support for our streaming system through the creation of a custom RTP receiver, also based on Live555. This receiver interfaces to OpenVideo and OpenTracker, libraries that are already employed for UrbanSketcher and other WP6 applications. The implementation for the Video-Scout features automatic reconnection on WLAN signal loss, auto resize for the scout stream, synchronized transfer of tracking data and was tested with a standard range access point. The hardware was assembled with all sensors but not yet fully tested and calibrated.

#### 4.4.2.3 Related Work

Since the idea of AR Scouting is very new no directly related work is available.

#### 4.4.2.4 Testing and Public Demonstration

First test of the Video-Scout were performed during the workshops of WP6 but more testing is needed to integrate and calibrate the gps and orientation sensor data with the augmented scouting view on the base station.

#### 4.4.2.5 Evaluation

Specification of the Mobile AR System Vesp'R is detailed in the following table. The second column shows wishes for a future specification and its sensors which would optimally be integrated into the future state of the art device.

Mobile AR Device	<u>UMPC Scout</u>	<u>Imaginary Future Device</u>
Processor CPU	1GHz	4GHz (Multi Core e.g. Quad)
Graphics Processor GPU	none	GForce10 (4 Watt)
Display	1024x600	1280x720 / 1920x1080
Memory	500MB	8 /16GB
Camera	640x480p @ 30fps	1280x720p / 1920x1080p @ 60fps
GPS Accuracy	+/- 5 Meter	+/- 0,01 Meter
Orientation Sensor Accuracy	1 deg (Inertia Cube 3)	0.001 deg
WLAN/Bluetooth	Yes/1.0	High Range @ Low W / 2.0
Weight	1249 Grams	500 Grams
Average Battery Lifetime	1,5H/Battery	8H/Battery - unlimited

#### 4.4.2.6 Specification

Hardware and OS	<p><i>Prototype was tested on the following devices:</i></p> <ul style="list-style-type: none"> <li>• Sony Vaio U70</li> <li>• Sony UX90/280</li> <li>• Samsung Q1</li> </ul> <p><i>Sensors: uBlox ANTARIS GPS receiver, Socket BT GPS receiver, HSDPA/UMTS modem</i></p> <p>The scout application runs under WindowsXP on these devices</p>
Software	Studierstube, OpenTracker, OpenVideo
Core Features	<p>Mobile outdoor AR interface</p> <p>Outdoor data gathering by an expert</p> <p>Platform for outdoor tracking prototypes</p> <p>Client for online 3D reconstruction</p>
Status	Prototype which will be further developed and refined
Intended users	Expert user which explores the urban environment
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.

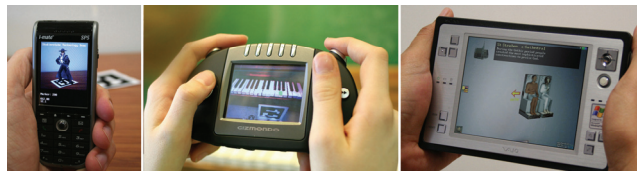
#### 4.4.2.7 Publications

1. Reitinger, Bernhard, Zach, Christopher, Schmalstieg, Dieter, „Augmented Reality Scouting for Interactive 3D Reconstruction“, IEEE Virtual Reality '07
2. Gerhard Schall, Erick Mendez, Ernst Kruijff, Eduardo Veas, Sebastian Junghanns, Bernhard Reitinger, Dieter Schmalstieg, Handheld Augmented Reality for Underground Infrastructure Visualization, ACM Personal and Ubiquitous Computing Journal, 2008
3. Schall, Gerhard, Mendez, Erick, Reitinger, Bernhard, Junghanns, Sebastian, Schmalstieg, Dieter, Mobile Geospatial Augmented Reality using Urban 3D Models, Workshop on Mobile Spatial Interaction (in conjunction with ACM CHI '07), 2007

#### 4.4.3 Studierstube ES

As a foundation for Handheld Augmented Reality, a software framework called Studierstube ES was developed. The framework is available for Windows CE and Windows XP, targeting small form factor devices ranging from smartphones to ultra-mobile PCs (Figure 29). Experimental versions also exist for Symbian and Linux.

All processing is done natively on the handheld device, so that applications can run independently of any infrastructure and scale to an arbitrary number of simultaneous users. Typical frame rates on smartphones are in the order of 5-20 fps, depending on the content and device.



**Figure 29:** Devices running Studierstube ES range from smartphones and game consoles to ultra-mobile PCs

The client software framework is based on a component design, and allows customizing the runtime environment to the needs of the application and the capability of the handheld device. In particular, memory footprint can be optimized to as little as 500K for a basic system. However, an extensive set of components is available. The core components essential for AR are Studierstube Tracker, a real-time fiducial tracking component, and Studierstube Scene Graph, a rendering engine running on top of OpenGL ES or Direct3D Mobile. Studierstube ES also offers scriptable components for networking, 2D user interfaces, Macromedia Flash playback, keyframe animation, audio, and video. Application code is managed through dynamically linked libraries, which simplifies memory management and downloading on demand.

#### Large Scale Localization/Detection

Work has recently started on localization/detection from a camera image on a large scale (building level or large logo database). Specifically this includes building a dense 3D reconstruction for model based tracking. This work package is currently in an early stage. In the future a mobile phone will be able to send an image to a server and get back an absolute position plus data for tracking from natural features on the client.

#### 4D Barcodes

4D barcodes are markers with animated and colored content. Due to their high bandwidth they are suitable for transmitting simple content (a few Kbytes) quickly over the air without using a network connection.

### **Skeletal Animation Support in Studierstube SG**

Studierstube SG now supports high-speed skeletal animation at CPU level. In the future shader-based skeletal animations will be added.

### **Video Texture Node for Studierstube SG**

A 3D quad capable of showing live video feed has been implemented for the Studierstube scene-graph. Currently the node shows only camera feeds. Playback of video files will be added in the future.

### **Universal Format Converter “Any2StbES” for Studierstube SG**

A converter framework supporting arbitrary formats has been created. It allows implementing importers for any kind of format, exporting into the StbSG format. Modules for supporting

- Milkshape3D (MS3D)
- Autodesk FBX (work in progress)
- Wavefront OBJ

have been implemented. VRML, as supported by the soon deprecated VRMLGui tool we be added soon.

### **Studierstube ES Support for Linux**

Studierstube ES has been ported to Linux. Currently only tests on a desktop Linux have been performed due to lack of a Linux-operated mobile phone. Desktop Linux support is a first, big step in supporting Google Android and other Linux-operated mobile phones.

### **Studierstube ES Support for MacOS**

Studierstube ES has been partially (~60%) ported to MacOS. The ported modules (StbCore, StbMath, StbTracker) build for MacOS on the desktop as well as the iPhone.

### **Natural Feature Tracking**

Partner U. Cambridge and TUG worked together on natural feature tracking for mobile phones, which was used in WP7 by partners HIIT and U. Oulu. A paper describing the natural feature tracking won the best paper award at IEEE International Symp. on Mixed and Augmented Reality 2008 [Wagner08].

#### **4.4.3.1 Related Work**

A few other projects dealing with Augmented Reality on mobile phones or PDAs have been reported in literature. Early work used these devices as thin clients, outsourcing most processing tasks to PC-based servers via wireless connections, which was necessary as the early PDAs did not have enough capability for stand-alone AR applications.

The Batportal3 used non-mixed 3D graphics streamed from a PC to the PDA via VNC, while the AR-PDA project (Gausemeier et al 2003) used digital image streaming from and to an application server. Shibata's work explored load balancing between client and server: the weaker the client, the more tasks are outsourced to a server.

Like the work reported here, later projects discarded the idea of outsourcing processing tasks in order to gain infrastructure independence. Möhring et al. (2004) were among the first to

target Symbian smartphones for mobile AR. The scarce processing power of mobile phones at that time allowed only a very coarse estimation of the object's pose on the screen and rendering of simple objects. Later Henrysson ported ARToolKit to the Symbian platform and created the first two-player AR game (Ingram et al 2001) for mobile phones. Rohs created the VisualCodes marker system for Symbian phones to explore novel interaction techniques. Similar to the work of Möhring, the tracking system provides only coarse pose estimation. VisualCodes was used for several simple AR games (Möhring et al 2004) augmented product packages, exploring AR as a tool for marketing. ULTRA (Marki et al 2005) uses PDAs for augmenting "snapshot" still images. Since this approach does not require real-time tracking, it allows using sophisticated natural feature tracking methods and therefore requires no fiducial markers.

1. Gausemeier, J., Freund, J., Matysczok, C., Bruederlin, B., Beier, D., Development of a real time image based object recognition method for mobile AR-devices, Proceedings of 2nd international conference on Computer graphics, Virtual Reality, Visualisation and Interaction in Africa, pp. 133-139, 2003
2. Henrysson, A., Billinghamurst, M., Ollila, M. Face to Face Collaborative AR on Mobile Phones. International Symposium on Augmented and Mixed Reality (ISMAR'05), pp. 80-89, 2005
3. 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), p. 77, 2001
4. Makri, A., Arsenijevic, D., Weidenhausen, J., Eschler, P., Stricker, D., Machui, O., Fernandes, C., Maria, S., Voss, G., Ioannidis N., ULTRA: An Augmented Reality System for Handheld Platforms, Targeting Industrial Maintenance Applications, Proceedings of 11 th International Conference on Virtual Systems and Multimedia (VSMM'05), 2005
5. Möhring, M., Lessig, C. and Bimber, O., Video See-Through AR on Consumer Cell Phones, International Symposium on Augmented and Mixed Reality (ISMAR'04), pp. 252-253, 2004
6. Michael Rohs Marker-Based Embodied Interaction for Handheld Augmented Reality Games Journal of Virtual Reality and Broadcasting (JVBR), Vol. 4, No. 5, March 2007
7. Shibata, F., Mobile Computing Laboratory, Department of Computer Science, Ritsumeikan University, Japan, <http://www.mclab.ics.ritsumeik.ac.jp/research.html>

#### 4.4.3.2 Publications

1. Daniel Wagner, Gerhard Reitmayr, Alessandro Mulloni, Tom Drummond, Dieter Schmalstieg, Pose Tracking from Natural Features on Mobile Phones, Proc. 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, (ISMAR'08), pp. 125-134, Cambridge, UK, Sep. 2008.

#### 4.4.4 Studierstube - Urban Sketcher and Expression

The integration process of the functionality for the infrastructure of ColorTable and UrbanSketcher results in the Urban Expression framework which is based on the MR-Framework Studierstube. The application of the framework is driven by the user centered design from the urban reconstruction scenarios in Showcase WP6 but is not limited to this particular field. Changes and demands emerging out of the integration process are reported here from an engineering point of view. The former ColorTable functionality goes into a component which can be loaded dynamically by applications in need of its functionality. The main application of the Expression framework is an extension of the former Urban Sketcher.

##### 4.4.4.1 Review and Re-design

The MR application needs new functionality for the objects in the scene which will be controlled by the ColorBlobManager (see section Tracking 5.1.5). This Manager needs the

ability to “clone” objects which means that a copy of an object is created which still shares certain parameters or abilities in the scene. This is for instance needed if two tokens of the same color are tracked on the table and represent a virtual object. If this object is modified e.g. painted with another texture its corresponding clone should also change in the same way. Clones will have some individual parameters like their position on the table.

Another demand is the easy change of operation modes so the scene can be changed fast from AR to VR e.g. for using Panoramas as background with varying tracking inputs for the virtual camera of the rendered MR scene.

The well known pipette functionality of Photoshop is now demanded for UrbanSketcher so colors can be sampled out of the scene for painting. Furthermore a texture copy stamp for applying parts of the scene as texture is on the wish list.

The continuous saving e.g. every 5 minutes of the scene for documentation purposes is demanded as well as functions to save the active texture or active object. Also saving of an image of the augmented scene to the hard disk drive is needed.

The integration process of UrbanSketcher and the ColorTable requires the creation of a new component for Studierstube which comprises all the functionality needed by the ColorTable. This new component will be loaded as a dynamic library by the new unified application.

#### 4.4.4.2 Development & Research

Integration of ColorTable and UrbanSketcher was initiated by a visit from TUW at TUG. First of all we ported ColorTable nodes into a Studierstube component available for all Studierstube based applications and therefore also in Urban Sketcher. The clone functionality was implemented as demanded for the so called “Raffaello Objects” creating the ability to instantiate clones by the tracking manager of the color tokens. Like this tokens of similar color are represented by objects of similar appearance in the scene graph. The user interface of the application Urban Sketcher was extended to allow interactive change of various view ports as demanded and an API based on Muddleware was created so all the needed major functionality of the application can be controlled by the barcode interface of the ColorTable component. The API was realized with a special scene graph node “SoSceneField” (section 5.2.3) in Studierstube which synchronizes field values of various types with Muddleware database altered commonly by the user via the Barcode Scanner component. Furthermore the demands for saving were implemented and tested in several iterations including debugging together with the remaining functionalities of the integration.

The Sketching functions were extended by a pipette tool for color sampling and stamping just like the well known functionality in Photoshop, but for the 3D scene. A reload function for the painting objects was implemented in Urban Sketcher to meet some interaction demands from the Color Table interface so a smooth collaboration is possible.

Another point is the wireless laser pointer; it was redesigned for a better reliability of the radio transmission from the button and for better visibility of the laser dot on the projection screen.

Also 3D content was generated suitable for the painting process in the WP6 Workshops.

#### 4.4.4.3 Related Work

Tables with architectural scale maps and models are established tools in architectural communication, enabling an observer to quickly grasp an overview of the planned design from an exocentric view. Interactive tabletop displays with MR capabilities (Aish et al. 2004, Kato et al. 2003, Pilgrim et al. 2001) and tangible user interface approaches have been developed to facilitate architectural education and also design negotiation (Underkoffler et. al. 1999, Ishii et. al. 2002). Neumann et al. 2004 describe Augmented Virtual Environments combining VR models with live video textures. Grasset et al. 2005 present an approach for video-see through MR painting. However none of these systems is suitable for collaborative outdoor use.



1. F. Aish, W. Broll, M. Stoerring, A. Fatah, C. Mottram, Arthur - an augmented reality collaborative design system, Visual Media Production, 2004. (CVMP). 1st European Conference on, Vol., Iss., 15-16 March 2004 Pages: 277- 281
2. R. Grasset, J.-D. Gascuel, D. Schmalstieg, Interactive mediated reality, AUIC 05: Proceedings of the 6th Australasian User Interface Conference, 2005
3. H. Ishii, J. Underkoffler, D. Chak, B. Piper, E. Ben-Joseph, L. Yeung, Z. Kanji, Augmented urban planning workbench: overlaying drawings, physical models and digital simulation, ISMAR 2002: Proceedings of the International Symposium on Mixed and Augmented Reality 2002, Pages: 203- 211, 2002
4. H. Kato, K. Tachibana, M. Tanabe, T. Nakajima, Y. Fukuda, A city-planning system based on augmented reality with a tangible interface, ISMAR '03: Proceedings of the The 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality, IEEE Computer Society, 2003
5. U. Neumann, S. You, J. Hu, B. Jiang, I. O. Sebe, Visualizing reality in an augmented virtual environment, Presence: Teleoper. Virtual Environ, 13, 2, 222—233, MIT Press, Cambridge, MA, USA, 2004
6. M. Pilgrim, D. Bouchlaghem, D. Loveday, M. Holmes, A mixed reality system for building form and data representation, In Proceedings of Fifth International Conference on Information Visualisation 25-27 July 2001 Page(s):369 – 375, London, 2001
7. J. Underkoffler, H. Ishii, Urp: A Luminous-Tangible Workbench for Urban Planning and Design, CHI, 386-393, 1999

#### 4.4.4.4 Testing and Public Demonstration

The locations where Cergy-Pointoise and the European City of Science event.

#### 4.4.4.5 Evaluation

Some experiments were performed so the time critical painting and sketching procedure can be done using the GPU of the computer system. This would allow a significant speedup which is demanded and results out of the testing and public demonstration.

#### 4.4.4.6 Specification

Hardware and OS	PC hardware, Windows
Software	UrbanSketcher Application, Studierstube Framework
Core Features	MR, AR Rendering, Painting and Sketching
Status	Prototype
Intended users	Any number of users
Showcases	WP6, available for others
Relevance beyond project	Available

#### 4.4.4.7 Publications

1. M. Sareika, D. Schmalstieg, Urban Sketcher: Mixed Reality on Site for Urban Planning and Architecture, In Proceedings of 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Pages: 27-30, 2007
2. M. Sareika, D. Schmalstieg, Urban Sketcher: Mixing Realities in the Urban Planning and Design Process. CHI 2008 - Workshop.

## 4.4.5 Color Compensation for See-Through Displays

Optical see-through displays are an established technology within augmented reality. Wearing such a display the users eyes automatically adapt to the luminance of the real world environment, while the virtual part is displayed using a steady brightness. This often results in clear differences between real and virtual elements. We developed a technique for colorimetric compensation which avoids this effect.

### 4.4.5.1 Development & Research

Our work is based on algorithms for photometric compensation. The appearance of background shapes and colours arise from the combination of the luminance of the background and the projected luminance of the object. These “ghosts” are photometrically compensated.

In order to capture the environmental luminance we attached a HDR camera to a head-mounted display. As the camera and head-mounted display are aligned it is easy to extract the region of the image which is visible for the user. As both differ in horizontal position and to avoid mapping errors the viewing distance is calculated.

Colorimetric compensation is used to simulate the visual adaptation of the virtual object. First the environmental luminance is extracted from the captured image. The fields of a GretagMacBeth colour checker are measured under various luminance levels in order to provide pixel-to-luminance mapping. Afterwards this information is used within a ring buffer slice which contains random pixel positions and is updated for each frame. From this the total luminance is computed.

As a reference we chose an image of 1000 cd/m<sup>2</sup> and images of lower luminance have been approximated using optimized box filters. The filters adjust visual acuity, color intensity and chrominance. When the filter parameters for a specific image luminance level are known, the see-through display image is adapted accordingly (using the environment luminance as input). Box filter parameters in-between the known samples are linearly interpolated.

Eye adaptation changes with time and this is also included in our model as well as the eye's responses to rapid luminance changes. After appliance of all effects the result is rewritten to the backbuffer.

Photometric compensation reduces the visibility of real background in relation to the virtual object. Such an algorithm must fulfill four quality criteria: preservation of luminance and contrast in the virtual image, background compensation and speed. For efficiency reasons the photometric compensation is implemented as per-pixel operators.

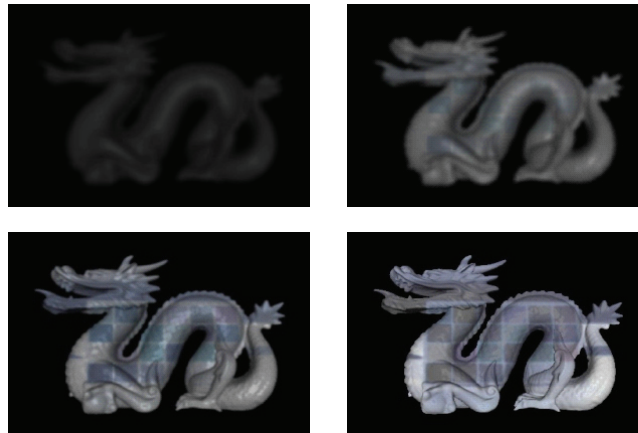
### 4.4.5.2 Related Work

1. Ferwerda, J., Pattanaik, S., Shirley, P., Greenberg, D. 1996. *A Model of Visual Adaptation for Realistic Image Synthesis*. ACM Transactions on Graphics, 1996, p. 249–258.
2. Irawan, P., Ferwerda, J., Marschner, S. 2005. *Perceptually Based Tone Mapping of High Dynamic Range Image Streams*. The Eurographics Association 2005: Eurographics Symposium on Rendering
3. Bimber, O., Emmerling, A., Klemmer, T. 2005. *Embedded entertainment with smart projectors*. Computer 38 (2005), Jan., Nr. 1, p. 48–55. DOI 10.1109/MC.2005.17

### 4.4.5.3 Testing and Public Demonstration

Both compensation systems improve the visual experience for users. Colorimetric compensation gives good results for the whole luminance range provided by the camera and includes time course as well as blinding effects (Figure 30). The mechanisms described in tone reproduction models of Ferwerda and Irawan [Irawan et al. 2005] are approximated closely. The subtraction operators give best results for photometric compensation, but have to be configured manually for each scene. Compensation by the smooth operator is less

intense but is robust to changes in environmental luminance. Framerate on a 3.6 GHz Dual Core system reaches 16.75 fps.



**Figure 30:** Colorimetric compensation applied on the Stanford Dragon. With increasing luminance (upper left to lower right) sharpness, brightness and chrominance are modulated accordingly. The negative checkerboard pattern arises from photometric compensation.

#### 4.4.5.4 Evaluation

Photometric compensation reduces the visibility of real background in relation to the virtual object (Figure 31). Such an algorithm must fulfill four quality criteria: preservation of luminance and contrast in the virtual image, background compensation and speed. For efficiency reasons the photometric compensation is implemented as per-pixel operators.

We evaluated several different versions. The simple ones subtract the background pixel  $b_i$  from the virtual pixel  $v_i$  but keep a fixed virtual part  $k < 1$  to avoid contrast loss:  $r_i = k * v_i + (1 - k) * (v_i - b_i)$ . An operator concentrating on luminance instead of chrominance will falsify the image instead. An advanced “smooth” operator scales the virtual pixel using an equation that changes with virtual and background pixel intensity to eliminate hard transitions.

The approximations made for Ferwerda’s model allow colorimetric compensation at an adequate speed. With further optimizations our system should be able to exceed the 30 fps necessary for real-time rendering.



**Figure 31:** Images captured by a webcam through the see-through display. Left is the uncompensated image, the right one is compensated with Subtraction Compensation.

The colorimetric compensation approximation mechanisms can be improved and new photometric operators can be developed. However, a focus must be kept on luminance preservation as both systems reduce luminance in the virtual image. We conclude that the environmental luminance range in which both system work can be extended by adding further measurements.

#### 4.4.5.5 Specification

Hardware and OS	Windows XP
Software	MORGAN framework
Core Features	Advanced color compensation for AR/MR applications using see-through displays
Status	Prototype
Intended users	Any number of users
Showcases	Available to all showcases
Relevance beyond project	Available

#### 4.4.6 Device Comparison Tests

During Year 3 the TimeWarp system used a handheld device rather than a mobile AR system as was the case during year 2. This approach was adopted to try and avoid problems with members of the public taking too much interest in the players, and consequently interrupting the game play and to remove certain other usability problems associated with the visors.

In the device comparison test, we evaluated different display techniques in order to explore presence, experience and usability. We compared the following four display devices (Figure 32).

- A backpacked laptop with a monocular head-mounted optical see-through display which was connected to mouse to improve the interaction. This was identical to the system used during the year 2 prototype.
- A backpacked laptop with a binocular head-mounted optical see-through display and a mouse as interaction device. Since there are no binocular see-through displays available, we developed our own.
- A tablet PC which will be used with a “Magic Lens” metaphor. Instead of an optical see-through display, a live video image will be displayed on the screen.
- An Ultra Mobile PC which is also used with a “Magic Lens” metaphor.



**Figure 32:** Compared devices: monocular and binocular see-through display (left) and Tablet PC and UMPC (right).

During the tests, a subset of the TimeWarp game was played. The player has to complete three tasks. First they have to solve the tutorial challenge. Afterwards they have to pass the time portal which appears after the first challenge is solved successfully. The third task is to solve the future challenge.

For evaluation we prepared two questionnaires. The first questionnaire focuses on the usability especially on the interface elements and the interaction with game elements. The second questionnaire covers presence related issues.

Currently, the testing is still in progress and we are not able to provide first results. The final results of the study will be presented on the IPCity review on March 2<sup>nd</sup> 2009 in Barcelona.

## **Acknowledgements and Further Information**

---

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

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

*[www.ipcity.eu](http://www.ipcity.eu)*