Integrated Project on Interaction and Presence in Urban Environments

FP6-2004-IST-4-27571

ipcity.eu

Final Report for MR Infrastructure

Deliverable D5.4

Doc-Id:	**D 5.4**
Version:	**0.8**
Author(s):	**Markus Sareika, Giang Phuong Nguyen, Gerhard Reitmayr, Michael Idziorek, Antti Juustila, Thorsten Fröhlich, Ann Morrison, Zsolt Szalavari, Peter Peltonen**
Date:	**2010-03-10**
Status:	**Final**
Availability:	**Restricted**
Distribution:	**Project Partners / EC**

Table of Content

Intended Audience

This document is intended to all partners of the project and to the reviews for the last 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 it’s 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 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 integration and collaboration supporting user interfaces in the tent. Furthermore, progress towards mobile AR software 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 the final phase 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 final results of the development that took place.

2 Mixed Reality Infrastructure 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. 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 user lives in her 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 components from various 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.

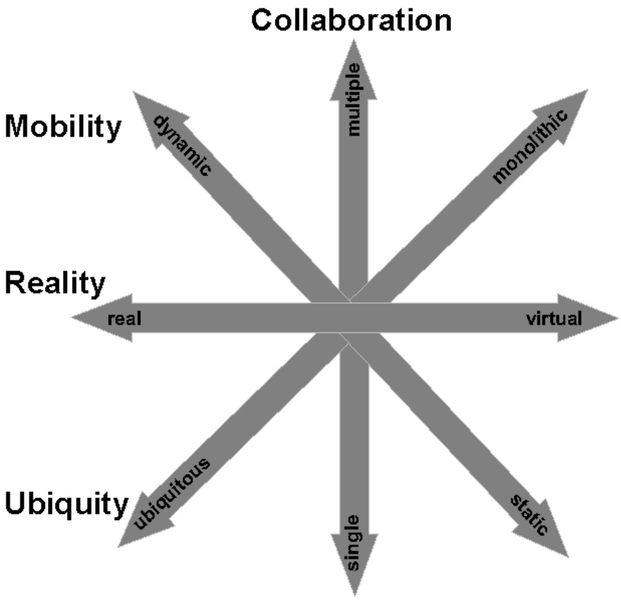


Figure 1: Multidimensional continuum spanning a design space

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. Therefore integration and the application of open interfaces are essential for large scale collaboration. The cross-reality interaction tools research work package 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.

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.[1] suggests to organize ubiquitous MR applications in a two-dimensions Milgram-Weiser continuum taking the quantity and density of spatial distribution in to 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.

We found 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 1 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 [2] (2002). The role of presence in mixed reality was outlined by Wagner et al. [3] (2009).

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.

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 components 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.

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.

---

[1] 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.

[2] 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.

[3] Ina Wagner, Wolfgang Broll, Giulio Jacucci, Kari Kuutii, Rod McCall, Ann Morrison, Dieter Schmalstieg, Jean-Jacques Terrin, On the Role of Presence in Mixed Reality. PRESENCE - Teleoperators and Virtual Environments, Vol. 18, No. 4, 249-276, MIT Press, 2009.

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.

The infrastructure is intrinsically tied close to technology and was divided into four major building blocks:

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

The developed components were formerly categorized by the building blocks. We decided to organize the final document structure by showcases to show where the developed infrastructure components are used. Nevertheless the conceptual building blocks were defined within an information space that lives around the users cross reality presence and are described in the next chapter.

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 six degrees of freedom (DOF) outdoor tracking in unconstrained environments. Even more importantly that new grounding breaking 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 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 (they are artificial landmarks added to a scene to facilitate locating point correspondences between images and a known model) 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 several 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 enhances user experience.

4 Workpackage Objectives

<table>
<tr>
<td>Objectives Final Phase</td>
<td>
The final prototypes give promising results concerning hardware and software developments. The software components for augmented and mixed reality applications on different mobile devices were continued according to the requirements of the various emerging showcase applications. In this context, sub-notebook but also UMPC-based as well as smart phone-based settings are useful for different showcase scenarios. Therefore, we continued the core development for these kinds of devices. In addition, a persistent collaborative database and message passing seems to be inevitable in order to exchange data between various devices. We further worked on the localization and tracking of outdoor users by fusing different types of tracking modalities such as GPS, inertia and vision-based systems. Specifically, we worked on software and hardware infrastructure for the following issues:
Urban Sketcher usability assessment
The Sketcher has been used in the MR Tent as part of multiple workshops with the main goal of collecting qualitative data on MR presence and related issues, relying mostly on ethnographic observation. However, the usability of the Sketcher interface has been neglected. We therefore performed assessment of the Sketcher’s usability through a quantitative evaluation. In particular, various assumptions about the 3D interactions using the Sketcher were scrutinized.
GPU Sketching
The Sketcher was rewritten to use advanced GPU shaders for painting, which speeds up this component significantly.
Performance optimization with Slow-Fast Rendering
The complex setup of the MR Tent featuring the main components ColorTable and Urban Sketcher has proven difficult, because of the variety of computer graphics tools used. In particular, these components can run at very different frame rates. We therefore refactored the MR Tent graphics subsystem into a mini-cluster composed of two PCs acting in parallel, using a slow-fast configuration with sort-last compositing.
Content Manager
The content managing was extended to support more services according to application needs and additional content moderation tools were added.
Scouting and multi-perspective MR
We extended the preliminary system with the ability of streaming both video and information about the spatial position and orientation of the scout. In order to account for extra performance demands by vision tracking, sensor fusion and communication support with the base station e.g. the MR-Tent, hardware re-considerations were inevitable and resulted in utilization of a dual core tablet PC. The integration with the Urban Sketcher interface allows collaborative experience of multi-perspective MR.
Outdoor tracking
Further development of the outdoor tracking with vision-based localization as well as various sensor combinations using vision tracking approaches were developed for outdoor tracking.
</td>
</tr>
</table>

Developed Components Final Phase	The focus of our work was on the following topics and tools realized in various components due to demands from the showcases: • **ContentManager** Extended support for creating the content "worlds" through ContentManager. Redesigned and implemented the YouTube support. Added web based content moderation tools. • **ColorTable:** Re-implemented the content assignment for using RFID, enhanced the camera sender component, sound improvements, integration of augmented maps for the table. • **Urban Sketcher usability assessment:** Performed user tests with Sketcher's interface through a quantitative evaluation. • **Urban Sketcher:** Introduced GPU supported sketching for a better performance and texture resolution independence. Added map tracking for interactive top view of the combined MR scene. Scouting scene integration for multi-perspective MR. • **Performance optimization with Slow-Fast Rendering:** Implemented a sort last compositing for multiple scenes connected by network rendered at independent framerates used for transferring the MR scene from ColorTable to integrate in Urban Sketcher. • **Scouting and multi-perspective MR:** Extend scouting capabilities by MR scene integration, optimized tracking and data transmission, introduced multiple perspective MR approaches. • **Outdoor tracking:** Further development of the outdoor tracking with Vision-based localization as well as various sensor combinations using vision tracking approaches were developed for outdoor tracking. • **HMDB:** Enhanced ATOM Publishing Protocol API, Session support was implemented as a basis for new features, voting and compass data structure elements were added. • **Mobile AR:** Development of navigation and panorama annotations, in-situ content creation, Morgan Light and Studierstube ES • **Physics Abstraction Layer:** A design and prototype implementation of a physics engine abstraction layer for the Morgan framework. • **Second City Database**: Development of geo-localized database for MR content management with flexible export interface and multi-user architecture

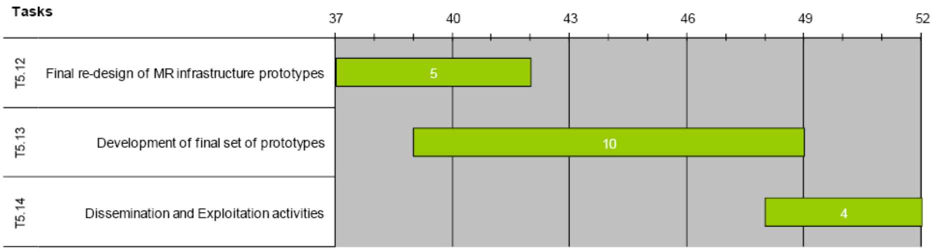
5 Results Concerning Research Questions

Feature/Addition	Research Question	Results
Integrating the outdoor tracking with Vision-based localization and further extend to use Geo-referenced images as model.	Can we create an outdoor tracking system that does not rely on expensive 3D models, but exhibits the same performance as model-based solutions?	A novel approach to use on-line available panorama images for live camera tracking was investigated. This allows initialization of the outdoor tracking.
Perform assessment of the Sketcher's usability through a quantitative evaluation.	Where can the user interface of the sketcher be more adaptive to the user?	A tracked architectural scale model was augmented with the Urban Sketcher interface and assumptions about the 3D interaction were studied in a user test, resulting in interface design suggestions.
Add additional content moderation tools to the Content Manager.	How can the Content Manager be Improved?	Web-based moderation tools were added to Content Manager, which allow browsing through and deleting inappropriate content from the system. Support for creating containers ("worlds") of content and their parameters through the manager was added: this is in use in the CityWall prototype.
Refactor the MR Tent graphics subsystem. Slow-fast rendering	How can all the MR Tent graphics systems be integrated efficiently?	Streamlining performance due to strong user demands was realized and tested with the slow-fast rendering component of Studierstube and the viewport synchronizations. Furthermore the painting performance was optimized by porting the painting implementation of Urban Sketcher to the GPU.
Scouting interface	How can multiple viewpoints in multi-user MR experiences be integrated?	Scouting is an important concept but up to now it was never fully realized due to other priorities. The scout is able to enhance the user experience by providing a dynamic viewpoint in addition to the otherwise available fixed views. A prototype system was created which is able to send a geo referenced video stream into the MR tent. The virtual data created by the users is overlaid using the streamed image and the available positioning data in Urban Sketcher. The current system extensions are improvement of its hardware capabilities and collaboration methods.

6 Infrastructure Workpackage

Mixed Reality (MR) infrastructure is focusing on basic research for mobile devices and their application for urban environments. Mobile settings in this context can vary in scale between light-weight systems such as smart phones or sub-notebooks, and semi-stationary devices such as high-end equipment in the MR tent.

The infrastructure year 3 prototypes have been re-designed, developed, tested and evaluated until Month 37 of the Project. They are the basis for the final tasks 5.12-5.14 the final redesign of the infrastructure. This final report states the final infrastructure developments on the prototypes for the showcases: Urban Renewal (WP6), Environmental Awareness (WP7), Time Warp (WP8), City Tales (WP9). Previously developed functionality was improved and new features which were demanded implemented.



We have organized the structure of the technical work packages WP4 and WP5 document to show how the individual developments relate to the showcases and how they are interlinked to one another. In the following a brief summary outlines the individual showcases.

Urban Renewal (WP6)

The Urban Renewal showcase is about communication, reconstruction and design in the context of urban issues. Stakeholders, citizens and professionals form the fields of urban planning, architecture, art, multidisciplinary design and mixed reality get together in the MR-Tent to take part in a process aimed at unifying their insights with the help of MR technology in development.

Environmental Awareness (WP7)

Environmental Awareness is about senstising people to their environment with the help of available cutting edge technology. We look to ensure people become more aware of their surrounds as well as more aware of environmental issues in their own environment (both are often overlooked). We enable local citizens, novices and expert users to work with new MR, AR and pervasive technologies, while exploring aspects of their own urban environments in playful ways. We aim to provide more sustained and immersed engagement for the participants and enable embodied participatory interactions.

Time Warp (WP8)

In Time Warp location based story telling game is developed while involving the user playfully. The natural border of time is overcome with the help of MR technology.

City Tales (WP9)

The City Tales work package is working towards establishing a digital layer over the urban tissue of a city that contains a story telling environment based on participant's input. The power of a community driven content creation approach has been shown in many Web 2.0 applications, such as encyclopedia, social networking portals and geo-information. Our aim is to focus this power to the creation of content into the digital layer of an urban environment, supported with easy to use yet powerful tools for mixed reality content communication.

7 Urban Renewal (WP6)

The Urban Renewal showcase is about communication, reconstruction and design in the context of urban issues. Stakeholders, citizens and professionals form the fields of urban planning, architecture, art, multidisciplinary design and mixed reality get together in the MR-Tent to take part in a process aimed at unifying their insights with the help of MR technology in development.

7.1 MR-Tent

With the corrections to the structure, canvas and interior design the MR-Tent provides now a useful setting as mobile mixed reality environment. Although the MR equipment is protected by the tent, weather conditions are to be considered carefully when planning workshops and events because the participants comfort is affected. 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 tangible table and the urban sketching interface. Transportation and setting up the Tent is still challenging due to the bulkiness and the weight as a whole, as well weights need to be added in order to provide stability for the structure 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.

Figure 2: 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 computers are stored and can be easily accessed by research staff. Also the redesign of the central tangible table clears up the space in order to let participants freely manoeuvre around the table and place content (Figure 3).

Figure 3: Stand up Tables and Color Table

7.1.1 Technology Components

The application developments are driven by the user centered design from the urban reconstruction scenarios in the showcase but is not limited to this particular field as parts of the technology has been evaluated in a user study with the aim to scrutinize interface characteristics in a quantitative and qualitative manner.

Tangible Table

See extra section 7.2 on the developments for the ColorTable.

Sketching Interface

See extra section 7.3 on Urban Sketcher and integrated components.

7.1.2 Testing and Public Demonstration

Apart from the initial set up in Graz, we tested the MR Tent during three workshops which provided different settings concerning environmental aspects and groups of participants. At the WP6 Workshops in Cergy-Pontoise the MR Tent was installed on two different sites were the spatial set up created different interior and exterior zones for discussion, demonstration and breaks. The other set up took place in an interior space: the large hall of the Grand Palais in Paris at the European City of Science (Figure 4).

Figure 4: MR Tent in different Environments: Cergy-Pontoise and Grand Palais

7.1.3 Specification

Hardware and OS	Tent Hardware: • Sanyo PLC-XP57L projector, 5500 ANSI lumens • PTU (pan-tilt unit) by RoboSoft • Sony HDV 1080i camera • Laser Pointer Tracking • Interaction table with a projector-camera setup • Different kinds of projection walls
Software	ColorTable, Urban Sketcher, Scouting, Muddleware, HMDB
Core Features	Hardware equipment for the MR tent
Status	prototype
Intended users	Urban Presence Explorers

Showcases	WP6 and WP7
Relevance beyond project	available

7.1.4 Publications

V. Maquil, M. Sareika, D. Schmalstieg, I. Wagner, MR Tent: A Place for Co-Constructing Mixed Realities in Urban Planning, In Proc. of Graphics Interface 2009, 211-214, 2009.

Wagner I., Basile M., Ehrenstrasser L., Maquil V., Terrin J., Wagner M. Supporting community engagement in the city: urban planning in the MR-tent Proceedings of Communities and Technologies, June 25 - 27, University Park, PA, USA, 2009.

7.2 ColorTable

The tangible table is mainly developed within the showcase but integrates and builds on several developments of the infrastructure work package. This includes the formerly developed tracking of color tokens, studierstube, opentracker, openvideo, muddleware and HMDB.

7.2.1 Components

Specifically developed components in the last period are listed below.

Slow-Fast Rendering

See Urban Sketcher section since the main implementation is integrated there. The developed component for sending the rendered scene over network is loaded and executed by the ColorTable renderer.

Camera Sender

Due to the integration process of the ColorTable and Urban Sketcher a unified camera model had to be adapted. Since we are now rendering two scenes separately on two different machines, to increase overall system performance, and combine them afterwards we have to be sure that all rendered frames match exactly. The Camera Sender Component connects to the Muddleware API of Urban Sketcher and sends/receives all required data for the Color Table (position, orientation, field of view) over the network so the rendering application gets the current virtual camera settings. The Camera Sender Component can receive, translate and forward camera orientation and position data from any of the following:

- Scout View
- Pan-Tilt-Unit View
- Bird Eye View
- Tangible Token View
- Rotatable Panorama View

RFID content assignment

Figure 5: Color content mapping

The RFID Component was developed to experiment with novel interaction techniques for the ColorTable and allows the connection of multiple RFID Readers and handle RFID Tags based on their id and the involved reader id.

Before usage, content of the ColorTable has to be first mapped to the colored tokens. To improve the intuitivite-ness of this process we decided to use RFID technology, instead of the previous combination of the Color Rotator and Barcodes. Every content card is equipped with a RFID-tag now, so the user has simply to place the desired content card on one of the colored areas in order to associate it with a token of this color. A small symbol on every content-card (triangle, rectangle or circle) acts as a hint for the users, the chosen shape is easily associated to the content. In general, sound, billboards and 3d objects are associated with triangles, while rectangles define the endpoints of paths, streets and other connections. Circular tokens are used to fill ground areas with different textures like grass or water.

Augmented Maps Integration

Figure 6: Augmented tracked map

During the workshops we work with physical maps of different scales and foci. To enable the user to rotate the physical maps instead of placing them only oriented, with north facing the projection, we implemented the possibility to use data from the Augmented Maps-Map Tracker (developed at University of Cambridge) to make the system aware of the maps orientation. However the demand and usage of this seemed low, so we did not use this option for later workshops in favor of system reliability and performance.

Link: http://mi.eng.cam.ac.uk/~gr281/augmentedmaps.html

SoundComponent

The SoundComponent is a standalone MAX/MSP application based mainly on Ambisonics externals and allows control via open sound protocol (osc) messages. It provides 32 stations, whose positions can be arbitrarily set in the virtual space and that are played back by multiple speakers to achieve spatial sounds. In parallel a sound library was developed and the existing sounds can be assigned freely to the stations. The application also allows the easy use of alternative sound libraries.

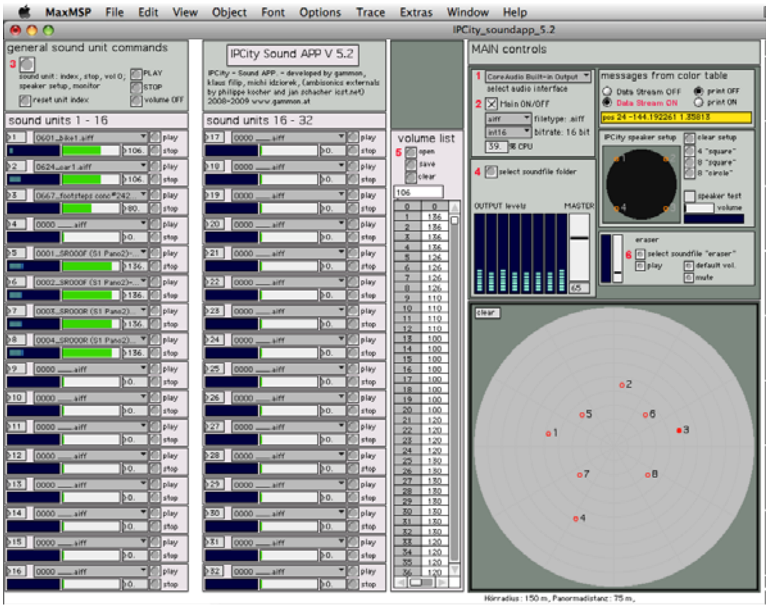


Figure 7: The SoundComponent GUI

- Summarized the supported features include:
- 32 fully configurable stations.
- Assigning sounds to stations via osc.
- Setting station positions via osc (x,y).
- Turning station on / off via osc.
- Customizable list of default volumes for every sound.
- Interpolation of the station positions between two messages.
- Graphical overview of all active sound stations and their positions in the available sound space.
- Alternative sound input from an external sound source (portable audio player device).

Currently the SoundComponent is used by the WP6 Showcase and is an integral part of the ColorTable. Most of the visual content is connected with default sounds while alternative sounds can be assigned with physical soundcards (by placing them in the assignment-area at the color of choice). The sound positions are calculated from the tokens positions relatively to the cameras orientation and position. Additionally the SoundComponent is also used for ambient sounds that can be defined for every panorama and for sounds generated by pedestrians, bicycles and vehicles that populate the streets and paths in the scene.

HMDB

The Hyper Media Database is used for media management by the ColorTable.

HMDB Interface

The HMDB Interface was developed at TUW to allow administration of the HMDB and can be accessed via a web browser. It provides a html based graphical user interface to administrate the contents inside the Hyper Media Database. Some of the supported functions are:

- Creating containers
- Uploading content to containers
- Moving content
- Adding and removing meta tags
- Browsing and searching the available content

7.2.2 Testing and Public Demonstration

In combination with the MR-Tent.

7.2.3 Specification

Hardware and OS	• Round table with top projection • RFID token assignment • Barcode interface
Software	ColorTable, Studierstube, OpenTracker, OpenVideo, Muddleware, HMDB
Core Features	Tangible Tokens, Content Assignment, Barcode Input
Status	prototype
Intended users	Urban Presence Explorers
Showcases	WP6
Relevance beyond project	available

7.2.4 Publications

Maquil V., Psik T., Wagner I., Wagner M., Expressive Interactions Supporting Collaboration in Urban Design, Proceedings of GROUP 2007, Nov 4-7, Sanibel Island, Florida, USA, 2007.

Maquil V., Psik T., Wagner I., The ColorTable - A Design Story Proceedings of TEI 2008, Feb 18-21, Bonn, Germany, *ACM, 2008.*

Boerner A., Maquil V. Enhancing synergies between computer science and urban disciplines: Semi-automated applications for tangible user interfaces, a case study Proceedings of CAAD Futures 2009, June 17 - 19, Montreal, Canada, 2009

7.3 Urban Sketcher

The MR application Urban Sketcher was further developed, with new components integrated. As well existing components were updated and improved. The slow-fast rendering concept was integrated to receive a real time stream from the colour table so the scenes of both applications can be combined in one scene graph without affecting one another's rendering performance. The showcase request also was to have multiple view port configurations consisting of panorama, top down (Bird Eye's View), Scout and augmented video see through view, which can be changed interactively or by the Muddleware API. These requirements lead to the implementation of the dynamic view port synchronization which made it possible to have a common view of both applications driven by the slow fast rendering component (see Components: Integration with Slow-Fast Rendering). In order to realize this combined functionality the camera parameters of both applications needed to be synchronized and matched to the currently active physical camera which provides the video background for the MR scene. The ColorTable was connected over the Muddleware API and the Scout was integrated via OpenTracker and OpenVideo components (see Components: Scouting) where as the Birds Eye View (see Components: Map Tracking – Bird View of MR Scene) was realised directly in Urban Sketcher.

The painting performance issue of Urban Sketcher was a long term demand which was finally met by the GPU painting implementation. The new concept for applying paint onto textures in the MR scene was integrated into the application. This required rewriting a large amount of the internal state management of the application also adapting the picking strategy for activating painting and non painting geometries in the mixed reality scene. See (Figure 8) for visual detail on the component GPU Painting in use.

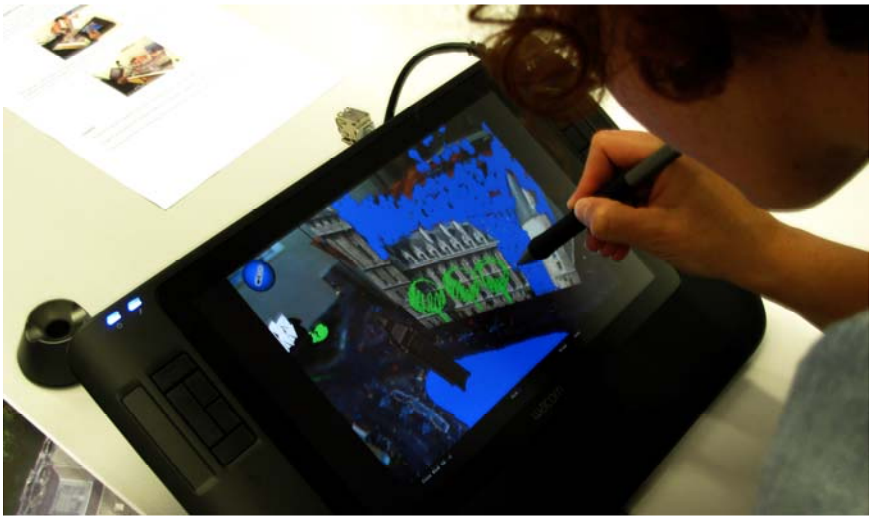


Figure 8: GPU based sketching

During the Cergy-Pontoise workshop it also became clear that the design and functionality of the laser pointer was still insufficient and the button needs improvement for sketching on the screen. The main issue was that the user had to concentrate on putting a lot of force on the right place in order to get the desired result of for instance applying paint on to an object. To this end, we improved the Laser Pointer pen design and functionality of the button. First, we reduced the weight by removing two batteries and the case of the transmitter of the button. The circuit board for the transmission of the button was taped to the outside of the laser pointer case and the power was connected directly to the supply of the laser pointer. The result did not look nice but is very ergonomic. Second, the old button was replaced by a new one reacting at high precision and consequently requiring a minimum of force. In summary the new Laser Pointer had an unconventional appearance but worked well, so the user could concentrate on her work and not be distracted by the pen interface as happened before these improvements.

7.3.1 Components

Integration with Slow-Fast Rendering

The integration process of Urban Sketcher and the ColorTable required the creation of a new component for studierstube which comprises all the functionality needed to send the rendered scene from one scene graph to another, uncoupling the render performance of the two applications. The demand for correct occlusions in the combined scene made it necessary to stream the depth information in addition to the rendered scene and alpha channel which separates the scene objects from the background. Several tests were performed in order to find an efficient way of encoding all the necessary data to be sent over a gigabit network connection while retaining near real time speed requirements. In addition to the sending scene node, also a receiving scene node was realized to sufficiently decode and integrate the incoming scene. The correct alignment and composition of the scenes is of course only possible when both rendering view ports are synchronized. This was realized by synchronizing the camera parameters of both scenes. See 7.3 Urban Sketcher and 7.2.1 Camera Sender component.

OpenTracker Zoom Node

The separate implementation of the Open Tracker zooming node was required because of timing issues and a clear separation of data channels which were formerly combined. The zoom node now acquires and sends data directly through the usb connected lanc module to the connected camera. The zoom can be set to a specific value now and does not rely on continued input from e.g. a joystick anymore.

Map Tracking – Bird View of MR Scene

The Bird View allows the user to adjust the mainly top down oriented viewing direction into the MR Scene by manually adjusting a web cam (Figure 9 left) to the desired area of the map. The captured image from the camera is augmented with the MR scene (Figure 9 right). Like this, an interactive view from a bird's-eye perspective is achieved. On the technical side we integrated an up-to-date version of a natural feature tracker into the Open Tracker module of Studierstube, so the map could be tracked in real-time without any view obstructing markers by the Urban Sketcher application in real time.

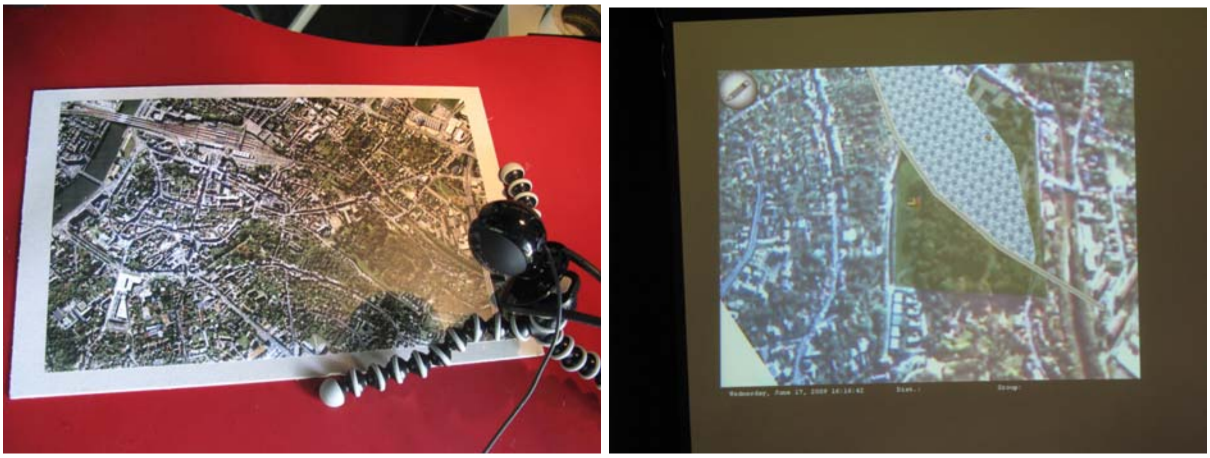


Figure 9: Webcam and overview map as well as wall projection of augmented map

GPU Painting

GPU based Texture painting was implemented to significantly improve the performance for applying paint onto a textured object in the MR scene. The ability to work fluently is determined by the speed and precision in which the visual feedback is given. In order to be able to paint and sketch at a reasonable precision even at moderate frame rates also with small pencil or brush sizes an algorithm is needed which accounts for intra frame texture paint updates. First of all we build a caching mechanism for mouse events so all available position information can be used to compute the stroke of the brush in between the rendering of two consecutive frames. After a series of tests we found that for very thin painting brushes near to the size of one texel the frequency of the available mouse events was not high enough to get a closed line when using rather high texture resolutions. This is why we implemented an interpolation algorithm to account for this issue. The implemented solution allows freely configuring the painting texture resolution which is only limited by the used graphics hardware responsible also for the final processing speed. Fluent work can be done at texture resolutions up to 1024 pixels at almost any brush size.

The implementation for determining the actual painting object and position in three dimensional space required us to place an ID into the rendered texture of the objects. This integrated a new GPU based picking method for activating painting objects and had to be put into action by the application using the GPU painting method. The Node for handling the models in the scene was also adjusted to work in the new configuration and needs further investigation for efficient future handling of various types of painting objects in the MR scene which is now possible due to the new painting method.

Scouting integration

The live video stream of the scout is received including all camera parameters. The integration of this video-stream and geometric data is used to render the MR scene in Urban Sketcher from the perspective of the scout dynamically moving in the environment. For more details see section 7.4 on scouting.

7.3.2 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.

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

R. Grasset, J.-D. Gascuel, D. Schmalstieg, Interactive mediated reality, AUIC 05: Proceedings of the 6th Australasian User Interface Conference, 2005

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

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

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

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

J. Underkoffler, H. Ishii, Urp: A Luminous-Tangible Workbench for Urban Planning and Design, CHI, 386-393, 1999

7.3.3 Testing and Public Demonstration

Exhibition in Paris: European City of Science Information, Demo in HIT Lab NZ Open House, Demo in Nara Japan ISMAR`07, Workshop in Sainte-Anne Paris, Exhibition in Vienna "Draußen in der Stadt", Demo in Greece at the 1st Peach Summer School, Demo in Graz at the Open Lab Night 2006, Two different workshops in Cergy-Pointoise (France), IPCity Summer School 2009, Open Lab Night at TUGraz 2009 and will finally be shown at IPCity's final event in Vienna.

7.3.4 Evaluation

The Sketcher has been used in the MR Tent as part of multiple workshops with the main goal of collecting qualitative data on MR presence and related issues, relying mostly on ethnographic observation. However, the usability of the Sketcher has been neglected. This is why we designed a user study in order to perform assessment of the Sketcher's usability through an evaluation which was partly done during the summer school. In particular, various assumptions about the 3D interaction in the Sketcher were scrutinized.

As a meaningful test setup, we use a table with a flat map and an architectural scale model consisting of block-shaped houses. The map is be tracked with a natural feature tracker, using a PC and handheld webcam. The house models partially occlude the map, but the tracking still works in most cases. The scene is presented in a video see-through MR mode on a tablet screen with pen input. View navigation is essential since viewport adjustments are elementary for interaction tasks in many use cases and direct the area of attention. In the study we concentrate on an imaginary urban planning scenario addressing a group of users with a wide range of previous knowledge on different fields and varying computer experience, which should all be equally well supported by the interface disregarding their experience and background. The goal of the evaluation is to clarify our research questions and hypotheses stated and provide data as well as insights concerning the proposed interface and device configurations. In summary the results should serve interface designers and assist them with design decisions ultimately guiding intuition on natural communication between human and machine. We evaluate in a quantitative and qualitative manner using measurements, questionnaires and video observation to find out which type of mixed reality view navigation is suitable for specific types of tasks when working with tracked table top models. The questionnaires used were NASA's TLX standard questionnaires and in addition to that we formulated 10 specific questions for the 31 participating users.

All the user feedback concerning the setup we received was very positive confirming that experiencing and expressing is done naturally, and was considered to be fun when using our bimanual MR interface. Independent of the users experience before the study all the tasks were solved after a brief introduction and no extra training. In conclusion the user interface supports a high bandwidth for information transfer between human and machine, which was necessary to complete the tasks in either of the two tested configurations. Input potential was integrated using bimanual interface operations with visual real time feedback giving the user the freedom to communicate efficiently supported by a natural and intuitive interface. In general the influencing factors are numerous and cannot all be quantified in a single statistical model. This is why we are in favor of the insights gained by triangulating methods and the qualitative user feedback containing rich information on the system in general. When working with users with varying professional backgrounds and skill levels giving options for individually optimizing the user interface configurations in order to address a wide range of individuals sounds tempting and can be cumbersome as the user is directly involved in designing the interface requiring her decision every time before the intended task can be accomplished. This blocks the natural workflow especially in collaborative situations around the table where users on the interface change frequently. Our findings indicate an advantage of the interface with the camera attached to the display in terms of task completion time and mental load. However, users did not express a clear preference for either interface.

7.3.5 Specification

Hardware and OS	PC hardware, Windows
Software	Urban Sketcher, Studierstube Framework, OpenTracker, OpenVideo, Muddleware.
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

7.3.6 Publications

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

M. Sareika, D. Schmalstieg, Urban Sketcher: Mixing Realities in the Urban Planning and Design Process. CHI 2008 – Workshop, 2008.

V. Maquil, M. Sareika, D. Schmalstieg, I. Wagner, MR Tent: A Place for Co-Constructing Mixed Realities in Urban Planning, In Proc. of Graphics Interface 2009, 211-214, 2009.

M. Sareika, D. Schmalstieg, Bimanual Handheld Mixed Reality Interfaces for Urban Planning, To appear in Proc. of AVI 2010, 2010

7.4 Scouting

The capability of roaming the real environment was included, in order to obtain different and dynamic points of view over the augmented world. We decided to develop 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 systems the capability of receiving and augmenting such live video streams. We implemented a 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 has an interface to an electronic compass, a GPS sensor and an orientation sensor. For the development 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 higher real-time compression of the data to stream and therefore reducing the network bandwidth required to transmit such data. For solid network communication, we rely on a standard protocol (RTP) and an already available and established library (Live555: http://www.live555.com).

On the server side, a custom RTP receiver based on Live555 was created. This receiver provides an interface to OpenVideo and OpenTracker, libraries that are also employed for Urban Sketcher. The network communications are established over WLAN using the latest standard (801.11n), which theoretically is able to transfer data for up to 250 meters in outdoor environments. Nevertheless, the performance degrades with the distance. Tests performed using standard WLAN hardware showed that a stable connection for transferring video data was possible for a range of around 30 to 40 meters.

The system was brought to practical use at a workshop in Cergy Pontoise (Figure 10). The extension of the previously available fixed viewpoints only allowing changes in orientation with one able to change its position was perceived as a valuable contribution by the workshop participants.

Figure 10: Scout in the surrounding of the tent

The current scout setup was improved in several aspects. The main limitation of the scout is its range. The scout user is not able to leave site of the tent. Therefore, the connection technology had to be upgraded. Previous tests of available mobile modem hardware have shown that the upload transfer rate was too low to send the video stream from scout to tent. Recently, new and faster mobile modem hardware became available. In addition to the extension of the mobile communications network to provide faster upload speeds, mobile internet technology became attractive again. Hence, the WLAN connection of the scout was exchanged for a mobile internet connection, removing the position constraints. The current system requires 120Kbyte/s upload bandwidth for a 160x120 image, thus requiring at least 1Mbit/s upload speed. Theoretically, the newest mobile technology supports upload speeds of up to 5.76Mbit/s. We will reevaluate currently available mobile modem technology, but practical experience gives reason for distrust. We trade the ability for changing the scout position freely for bandwidth.

Although the original design choice for the UMPC was the better processing performance, the practical results did not catch up to the expectations. Preparation and transfer of video data brings the CPU to its upper limits, even in low resolutions. To be able to enhance the scout functionality more system resources have to be available. Therefore, the UMPC was exchanged for a new portable tablet PC.

Using GPS and inertia sensor for position and orientation information leads to registration errors, because of sensor inaccuracies. This further leads to registration errors of the virtual content on the real world. To enhance the tracking, we are experimenting with various sensor combinations with vision tracking approaches developed for outdoor tracking.

Tent-scout communication consists of two parts. One part involves textual and voice communication in both directions using existing instant messaging solutions. This is easily possible using currently available mobile internet technology. However, voice communication will only be employed, if the bandwidth of the mobile connection does not suffer from the increased data traffic. The transmission of images has higher priority than voice communication.

Naturally, the communication of orientation changes is more efficient using visualization aids, which is the second part of communication. We implemented a simple graphical user interface where tent-users mark a certain spot on a 2d map, thereby requesting a position and/or orientation change from the scout. The selection can be refined in image space, where tent users are able to place a mark in the 2d plane. The scout user is presented with arrows indicating the shortest rotation direction, if the marked spot is outside of the current field of view. If the requested location is visible in the scout's view, a line is drawn from the center of the view to the respective location.

A position change involves finding a path from the current to the target location. Although the scout range is extended using mobile internet technology, we did not implement an automatic way-finding solution, because the deployment region of the scout is limited to the surroundings of the tent. Furthermore, scout users are able to refer to top-down maps of the surroundings to find appropriate paths by themselves.

7.4.1 Outdoor Tracking

See extra section 7.5 summarizing work on Outdoor Tracking essentially needed for the scout.

7.4.2 Related Work

Since the idea of AR Scouting is very new, no directly related work is available. Benford et al. 2006 performed a study on mixed reality games, which used a kind of scouting device. They developed a mixed reality game in which online players are chased through a virtual city by so called runners, who themselves are located in the real world. The runners navigate through the city using a handheld device, which receives its position from a GPS tracker. Data is transmitted using a Wifi connection.

S. Benford, A. Crabtree, M. Flintham, A. Drozd, R. Anastasi, M. Paxton, N. Tandavanitj, M. Adams, J. Row-Farr, Can you see me now?, ACM Transactions on Computer-Human Interaction, 2006, Vol. 13, Iss. 1, March 2006, Pages: 100-133

7.4.3 Testing and Public Demonstration

First test of the Video-Scout were performed during the workshops in Cergy-Pontoise and will be shown at IPCitys final event in Vienna.

7.4.4 Specification

Hardware and OS	*Prototype was tested on the following devices:* • UMPC • Tablet PC *Sensors*: uBlox ANTARIS GPS receiver, Socket BT GPS receiver, HSDPA/UMTS modem The scout application runs under WindowsXP
Software	Studierstube, OpenTracker, OpenVideo, Muddleware
Core Features	Mobile outdoor AR interface Outdoor data gathering by an expert Platform for outdoor tracking prototypes Multi-Perspective MR
Status	Prototype
Intended users	Expert user which explores the urban environment
Showcases	WP6, WP9 and also interesting for WP7 and WP8 and as a mobile outdoor client platform
Relevance beyond project	This prototype contributes to the mobile handheld and outdoor MR research community. It also contributes to related

	projects such as WikiVienna or Vidente, both Austrian funded projects.

7.4.5 Publications

Bernhard Reitinger, Christopher Zach, Dieter Schmalstieg, Augmented Reality Scouting for Interactive 3D Reconstruction, IEEE Virtual Reality ’07

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

Gerhard Schall, Erick Mendez, Bernhard Reitinger, Sebastian Junghanns, Dieter Schmalstieg, Mobile Geospatial Augmented Reality using Urban 3D Models, Workshop on Mobile Spatial Interaction (in conjunction with ACM CHI ’07), 2007

7.5 Outdoor Tracking

Outdoor Tracking from Panoramas

To overcome the problems of obtaining geo-referenced image databases for initialization of visual tracking systems, UCAM investigated the use of on-line accessible databases of image panoramas, providing 360 deg panoramas of many cities, sampled on roads at reasonable distances of between 10 – 50m. These panorama images can be queried at runtime or downloaded in advance and are annotated with their geographical location and orientation.

Figure 11 Match of a live image against a stored panorama image and annotations overlaid corresponding to the computed camera pose.

Our system implements a simple feature matching approach between a live input image and a stored panoramic image. First the GPS location of the client is used to download the nearest panorama image, typically yielding only one or two candidates, depending on the accuracy of the location and the density of panoramas. The panoramas are indexed multi-scale oriented patches (MOPS) as features stored in a K-D tree for sub linear indexing. From the live video frame we also extract features and match against the stored feature sets (see Figure 11).

The repeated structure of building facades with many similarly looking windows lead to many wrong correspondences in a feature matching-based approach. To overcome this problem we employ more geometric constraints in the later outlier removal step. First we accept more then a single correspondence between the input image and the stored panorama images. Then we build a hypothesis by first drawing a single correspondence and further considering only correspondences that fall into the viewing frustum defined by the first correspondences. From 4 such correspondences we then estimate a homography (a homography is an invertible transformation from the real projective plane to the projective plane) to allow for some additional warping due to the camera position not exactly aligning with the panorama's centre.

Improvement for building recognition and localization system (AAU)

AAU was working on improving the system for building recognition and localization. The main idea is to speed up the system to fit into any run-time application. In previous work, we have shown that using Multi-scale oriented patches (MOPS) gives the system a better performance over commonly used SIFT (Scale Invariant Feature Transform). We also pointed out that selecting a subset of features instead of using all extracted features could boost the system performance. The question is how the selection should work. We proposed two approaches in selecting informative features:

1. *Uniqueness filtering for local feature descriptors in urban building recognition*

   Existing local feature detectors such as Scale Invariant Feature Transform (SIFT) usually produce a large number of features per image. This is a major disadvantage in terms of the speed of search and recognition in a run-time application. Besides, not all detected features are equally important in the search. It is therefore essential to select informative descriptors. In this paper, we propose a new approach to selecting a subset of local feature descriptors. Uniqueness is used as a filtering criterion in selecting informative features. We formalize the notion of uniqueness and show how it can be used for selection purposes. To evaluate our approach, we carried out experiments in urban building recognition domains with different datasets. The results show a significant improvement not only in recognition speed, as a result of using fewer features, but also in the performance of the system with selected features.

   Definition of uniqueness: A unique feature has an identifiable property that distinguishes it from other features in the image.

   The uniqueness of a feature vector *Fi* in image *I* is formulated as:

   $$\mathcal{U}_{F_i} = \|\{S_{ij} < \epsilon\}\|_{j=1..k, j \neq i}$$

   where S_{ij} denotes the similarity between 2 feature vectors Fi and Fj.

   The Figure 12 shows results from different cases in which varying numbers of descriptors were used over two different test sets (AAU and Aalborg centrum set). k* represents our uniqueness filtering method, the last row is the default MOPS with all features taken into account. The results show that the number of features extracted in the default cases is much higher.

Method	Features/image	Total features
k^*	100	13500
k^*	200	27000
k^*	300	40500
k^*	500	67255
k^*	800	108000
k^*	1000	130831
default	1470	198479

Method	Features/image	Total features
k^*	100	44200
k^*	200	88400
k^*	300	132600
k^*	500	221000
k^*	800	353600
k^*	1000	442000
default	2160	954409

Figure 12: AAU and Aalborg datasets with different numbers of unique features per image vs. default MOPS detector

Figure 13 shows recognition results for the two test sets. System performances with different number of unique features are compared, and versus default performance with MOPS (black-dashed line). For example, in figure 1a, using 200 features can give better recognition rate over the default one. This means that instead of using all features, we can limit ourselves to only 1/7 of the total number.

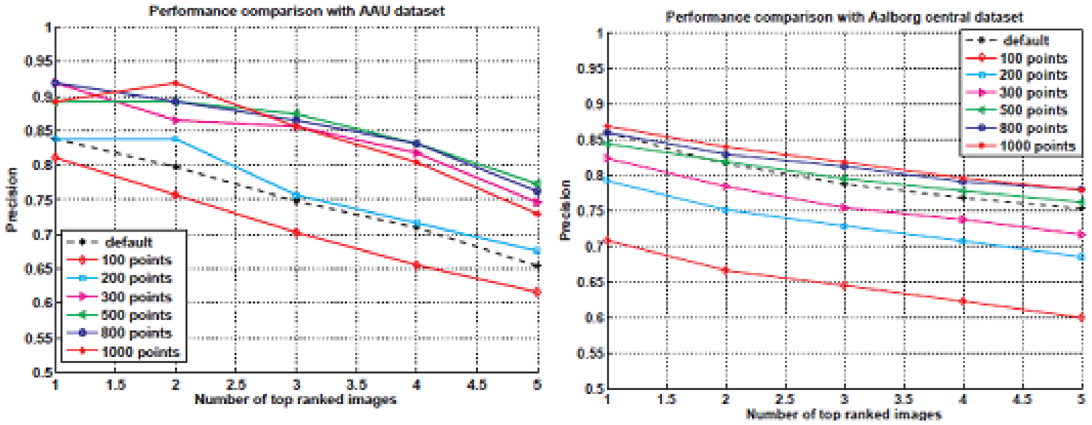


Figure 13: AAU and Aalborg dataset. Precision vs. the nummber of top ranked images. Results show performance when the default MOPS with all extracted features is used, versus our approach which focuses only on unique features.

Our experimental results with different datasets show that our approach improves recognition performance even where far fewer descriptors are used. However, the time needed for the complex process of extracting all the descriptors before the selection still remains. Finding informative features without extracting all descriptors should further speed up the system. In next paper, our investigation follows this direction.

2. *Context-based adaptive filtering of interest points in image retrieval*

As mentioned above, computing descriptors is the most expensive part in feature extraction. Therefore, our aim is find a solution to filter feature points before applying the descriptor extraction. We present a new technique to choose a subset of features. Our approach differs from others in a fact that selected feature is based on the context of the given image. The Btree triangular coding (BTTC) is employed to perform the filtering task. BTTC is a compressed representation of the image content, where an image is divided into a set of right angle triangles in which the content within each triangle can be interpolated from its three vertices.

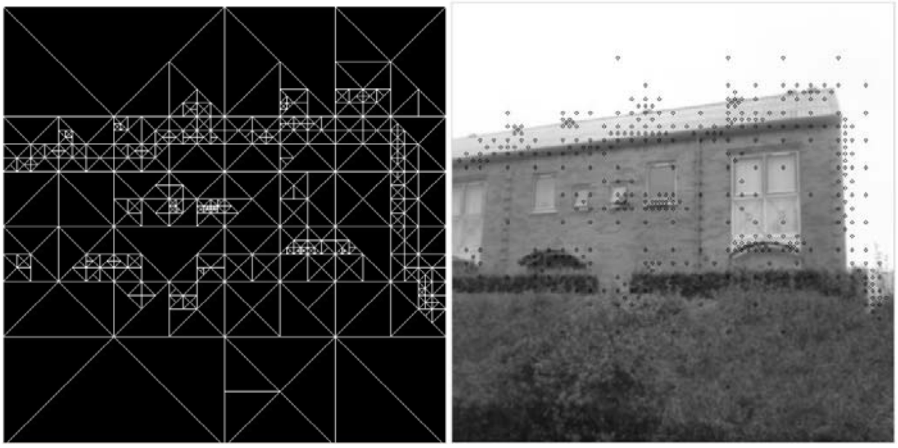
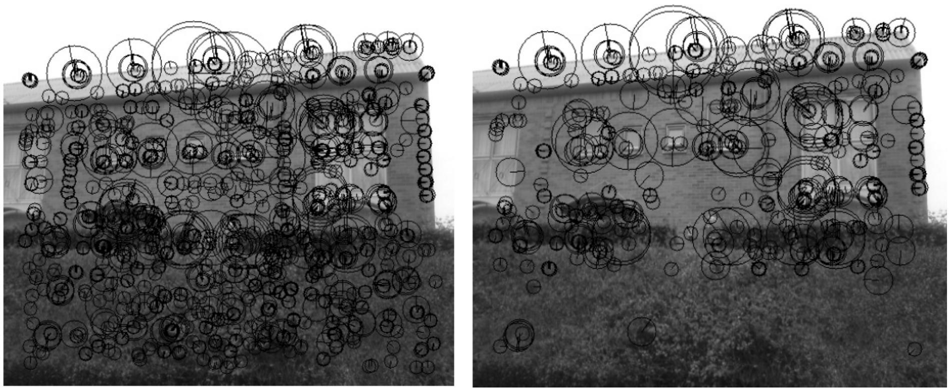


Figure 14: Examples using BTTC. Left: all triangles are drawn for observation. Right: vertices of all triangles are embedded to the input image

For the filtering, we first extract all interest points (for instance using Harris corner detector), then find the BTTC representation of the image. All the extracted points are assigned to their corresponding triangles, the one with highest corner strength within each triangle is selected as a representative point. The following figure shows an example of point filtering with BTTC:



Our first experiment is to show how our approach can reduce the number of features. For that purpose, we compute for each image the number of interest points with and without the BTTC filtering. In the Figure 15, the dotted lines represent the average reduction rate for the whole dataset, and the other lines show the rate at each image

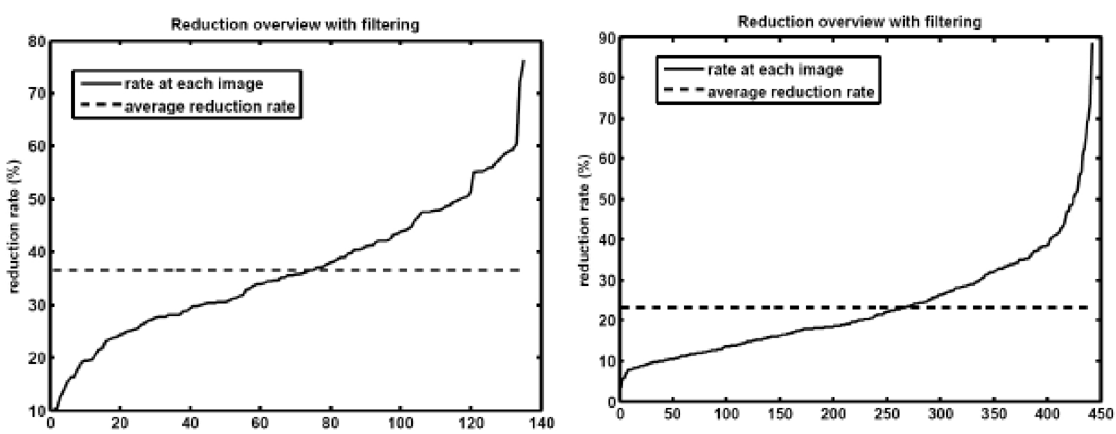


Figure 15: reduction rate for the two test sets

(in a sorted order). For example, in the first test set, we get on average approximate 50% of the images have more than 37% of reduction rate, and the highest reduction rate is 78%.

The next experiment is to see if the reduction does not come at the cost of lower recognition rate. We prove that reducing the feature points correctly will not reduce the system performance. Moreover, by reducing the size of feature, we can speed up the current system by 20 and 40%, for the AAU and the centrum dataset, respectively.

Panorama Tracking with Sensor Fusion

We proposed to improve the accuracy and robustness of the pose of a mobile device using a coarse positioning system with IMU measurements and the use of visual constraints provided by a camera used in a mobile Mixed Reality device.

A convincing tracking solution must overcome inherent limitations of individual tracking techniques by combining different complementary methods. Consequently, we have implemented a multi-sensor fusion approach. Next to the 3 DOF positioning system such as Ubisense UWB (or for example GPS in outdoor environments) and the video camera, we use an inertial measurement unit (IMU) as additional sensor. While the orientation information from the IMU and the position information from the 3dof positioning system only provide complementary measurements of the camera pose, the video stream encodes relative motion information about both translation and rotation. During operation, the proposed system records IMU measurements, Ubisense UWB measurements and video frames taken at the same time. Motion estimation and feature matching between pairs of video frames create epipolar constraints on the camera motion between the frames.

Inertial tracking has the advantages of range, and a system that is passive and self-contained. Its major disadvantage is its lack of accuracy and drift over time. The first effect of time-depended drift of the accelerometers angular rates we correct by an Attitude Kalman filter, which performs a sensor fusion of gyroscopes, accelerometers and magnetometer. Moreover, the gyroscopic angular rates have biases which are estimated in this filter as well to avoid a temporal drift of the attitude angles. This filter is also capable of eliminating the rather long transient oscillation behavior of the inertial sensor. The second effect of location-depended deviations of yaw can be detected and corrected by using a visual tracker, which does not require a model of the environment. By online mapping and learning of natural features of the unknown environment, this tracker allows for detecting and correcting the deviation of the 3-axis compass. The visual tracker works in two steps: First it updates the current rotation and then it maps new areas of environment. The rotation updating process uses these 2D-2D point correspondences between the environment map and the camera image. Second, after the rotation has been updated to the current camera image, we project the camera image into the environment map. This improves both the accuracy and the robustness of the orientation estimates. Thus, this makes the rotation much more stable with respect to the real world then the usual inertial tracking, which typically has some lag, drift or slight misalignment.

Experiments were conducted to assess the accuracies of the single sensors as well as the performance of the multi-sensor fusion tracking approach. While providing sub-meter accuracy position estimates using the coarse 3dof tracking system the accuracy and robustness of the orientation estimates of the mobile device could be increased significantly under real-world conditions.

7.5.1 Related Work

R. Azuma, J. W. Lee, B. Jiang, J. Park, S. You, and U. Neumann. Tracking in unprepared environments for augmented reality systems. Computer & Graphics, 23(6):787-793, 1999.

G. Bleser, Stricker, D., Advanced tracking through efficient image processing and visual-inertial sensor fusion. In Proc. of IEEE VR 2008, pp. 137-144 2008.

S. Feiner, B. MacIntyre, T. Höllerer, and A. Webster. A touring machine: Prototyping 3D mobile augmented reality systems for exploring the urban environment. In Proc. ISWC'97, pages 74, Cambridge, MA, USA, October 13, 1997.

E. Foxlin, Pedestrian Tracking with Shoe-Mounted Inertial Sensors, IEEE Computer Graphics and Applications, p.38-46, Nov 2005.

B. Hofmann-Wellenhof, K. Legat, and M. Wieser, Navigation - Principles of Positioning and Guidance. Springer, Wien/New York, 2003.

X. Hu, Y. Liu, Y. Wang, Y. Hu, and D. Yan. Autocalibration of an electronic compass for augmented reality. In Proc. ISMAR 2005, pages 182-183, Vienna, Austria, October 5-8 2005. IEEE and ACM.

W. Piekarski and B. H. Thomas. Tinmith-metro: New outdoor techniques for creating city models with an augmented reality wearable computer. In Proc. IEEE ISWC'01, pages 31- 38, Zurich, Switzerland, 8-9 October 2001.

G. Reitmayr and T. W. Drummond. Going out: Robust tracking for outdoor augmented reality. In Proc. ISMAR 2006, pages 109-118, Santa Barbara, CA, USA, October 22-25 2006. IEEE and ACM.

S. You, U. Neumann, and R. Azuma. Orientation tracking for outdoor augmented reality registration. IEEE Comp. Graph. Appl., 19(6), November 1999.

7.5.2 Publications

A book chapter 'Fast and efficient local features detection for image recognition' for book on: 'Innovations in Intelligent Image Analysis', Springer-Verlag. To be appeared.

Uniqueness filtering for local feature descriptors in urban building recognition, Giang P. Nguyen 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).

Context-based adaptive filtering of interest points in image retrieval, Giang P. Nguyen and Hans Jørgen Andersen. In IEEE proceeding of the International Conference on Intelligent Systems Design and Applications.

Global Pose Estimation using Multi-Sensor Fusion for Outdoor Augmented Reality, Schall Gerhard, Wagner Daniel, Reitmayr Gerhard, Manfred Wieser, Elise Taichmann, Schmalstieg Dieter, Bernhard Hofmann-Wellenhof, Proceedings of Int. Symposium on Mixed and Augmented Reality 2009 (ISMAR´09), IEEE, 2009-October.

7.6 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 interfaces HMDB, ColorTable and Urban Sketcher.

7.6.1 Testing and Public Demonstration

In combination with Urban Sketcher 7.3 and ColorTable 7.2.

7.6.2 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

7.6.3 Publications

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

7.7 Considerations for a Distributed Multi-Display Infrastructure

We investigated a system infrastructure for a collaborative multi-display environment combining displays of heterogeneous form factors (e.g. private monitors and large public projection surfaces on walls and tables) into a common interaction space (Figure 16). In particular, an API exposed by a multi-display framework was drafted. The conceptual API provides information about the physical display environment, the users interacting with the system, and provides functionality to render visual information on top of desktop content. With such a multi-display infrastructure, content of registered applications can be adjusted to spatial properties of the environment, as well as user preferences and backgrounds, thereby bridging the knowledge gap between experts of different domains.

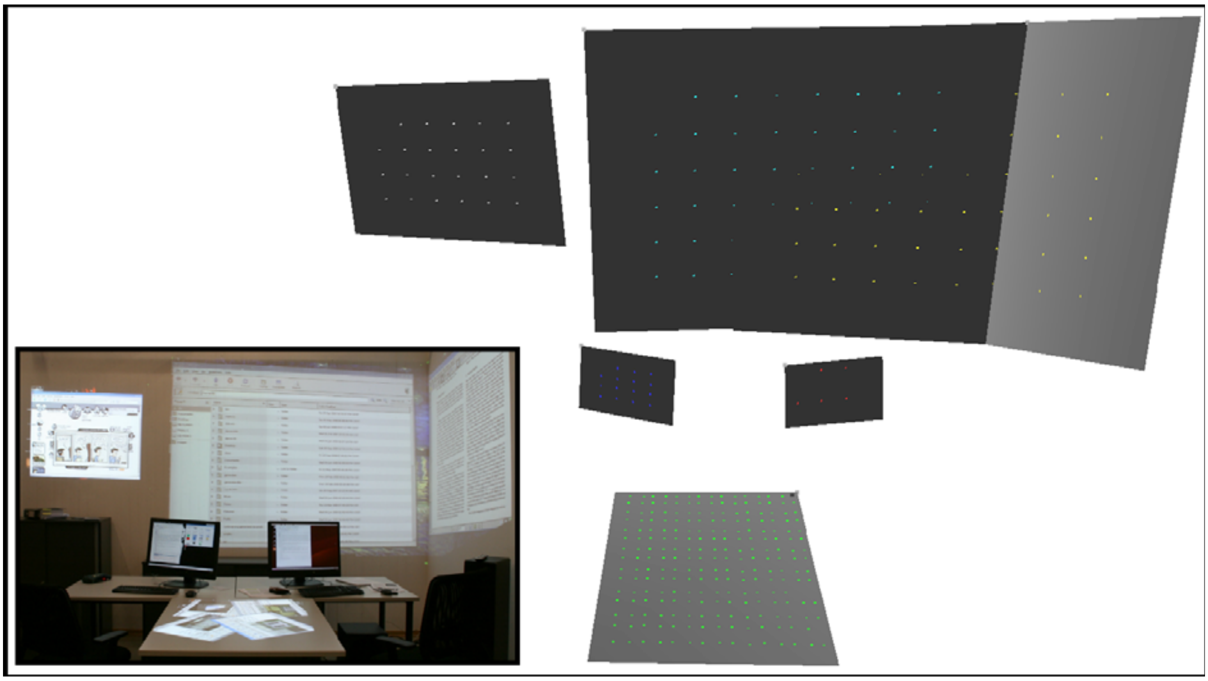


Figure 16: The spatial model of a multi-display environment.

7.7.1 Related Work

In their "rule of diversity", Baldonado et al. [2000] suggest that multiple views should be employed if users' preferences and knowledge backgrounds differ. Tang et al. [2006], as well as Forlines and Shen [2005] demonstrated systems providing each user with tools for filtering a single, shared view. Isenberg et al. [2007] created a collaborative visualization system for a multi-touch table. They also presented a set of design guidelines for collaborative visualization systems, which was extended by Heer et al. [2008]. Our design considerations differ, as we are more focused on MDEs with special emphasis on the influence of display geometries, display topologies, user locations and user preferences on the visualizations.

In an MDE, Forlines and Lilien [2008] distributed multiple coordinated 3D views of a protein to an interactive touch display, two wall displays, and a tablet PC for fine-grained interaction. Although their system supports multiple users by facilitating a multi-touch table, they do not provide special collaborative interaction features. Shen et al. [2006] developed a taxonomy of multiple-view visualization styles in multi-display environments. They proposed three visualization styles differing in their synchronization method. In contrast to their taxonomy, we propose the separation of the system into a multi-display framework -- for instance [Pirchheim et al., 2009] -- and an application layer to not limit the MDE's functionality to the employed applications and likewise, not to limit the applications' applicabilities to the specific MDE.

M. Q. W. Baldonado, A. Woodruff, and A. Kuchinsky. Guidelines for using multiple views in information visualization. pages 110–119, 2000.

C. Forlines and R. Lilien. Adapting a single-user, single-display molecular visualization application for use in a multi-user, multi-display environment. In AVI '08: Proceedings of the working conference on Advanced visual interfaces, pages 367–371, New York, NY, USA, 2008. ACM.

C. Forlines and C. Shen. Dtlens: multi-user tabletop spatial data exploration. In UIST '05: Proceedings of the 18th annual ACM symposium on User interface software and technology, pages 119–122, New York, NY, USA, 2005. ACM.

J. Heer, F. van Ham, S. Carpendale, C. Weaver, and P. Isenberg. Information Visualization, volume 4950/2008, chapter Creation and Collaboration: Engaging New Audiences for Information Visualization, pages 92–133. Springer Berlin / Heidelberg, 2008.

P. Isenberg and S. Carpendale. Interactive tree comparison for co-located collaborative information visualization. IEEE Transactions on Visualization and Computer Graphics, 13(6):1232–1239, 2007.

C. Pirchheim, M.Waldner, and D. Schmalstieg. Improving spatial awareness in multi-display environments. In Proceedings of IEEE Virtual Reality, pages 123–126, 2009.

C. Shen, A. Esenther, C. Forlines, and K. Ryall. Three modes of multisurface interaction and visualization. In Information Visualization and Interaction Techniques for Collaboration across Multiple Displays Workshop associated with CHI'06 International Conference, 2006.

A. Tang, M. Tory, B. Po, P. Neumann, and S. Carpendale. Collaborative coupling over tabletop displays. In CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems, pages 1181–1190, New York, NY, USA, 2006. ACM.

7.7.2 Publication

Manuela Waldner, Alexander Lex, Marc Streit, Dieter Schmalstieg, Design Considerations for Collaborative Information Workspaces in Multi-Display Environments, Proc. of Workshop on Collaborative Visualization on Interactive Surfaces (CoVIS'09), in conjunction with VisWeek, Atlantic City, USA, 2009.

8 Environmental Awareness (WP7)

Environmental Awareness is about sensitising people to their environment with the help of available cutting edge technology. We look to ensure people become more aware of their surrounds as well as more aware of environmental issues in their own environment (both are often overlooked). We enable local citizens, novices and expert users to work with new MR, AR and pervasive technologies, while exploring aspects of their own urban environments in playful ways. We aim to provide more sustained and immersed engagement for the participants and enable embodied participatory interactions.

8.1 Outdoor Tracking

See Urban Renewal showcase developments, section 7.5, Outdoor Tracking)

8.2 Tracking for Map Lens2

Within MapLens2 we implemented the first fully self-contained natural-feature tracking system capable of tracking full six degrees of freedom (6DOF) at real-time frame rates (20Hz) from natural features using solely the built-in camera of a mobile phone. Our tracking technique uses only textured planar targets, which are known beforehand and can be used to create a training data set. This fits particularly well with the planar map used for MapLens2.

We have achieved real-time performance by examining a leading approach in feature descriptors, namely SIFT. In its original published form, the SIFT approach is unsuitable for low-end embedded platforms such as phones, due to requirements in both memory size and computational performance. Our resulting tracker is 1-2 orders of magnitude faster than naïve approaches towards natural-feature tracking and therefore also very suitable for more capable computer platforms such as PCs. We present our changes to the original SIFT work, our optimizations and detailed evaluation results in (Wagner et al., 2010).

In (Wagner et al., 2010) we also present how we integrated a template-based tracker, which has orthogonal strengths and weaknesses compared to our SIFT-based approach. We combine the two approaches into a hybrid tracking system that is more robust and faster.

8.2.1 Testing and Public Demonstration

Tracking was tested in 2009 in the IPCity Summer School in Vienna, where students attempted to design a pervasive game to be used with the MapLens technology. Two MapLens user trials with participants (N=37) from the general public were organised in August before the summer school. In these trials the users used MapLens technology in various tasks both indoors and outdoors (see Figure 17).

Figure 17. Different ways to use MapLens: players either stand and hold a map or lay a map on a supporting surface like ground, table or a bench c) using while walking d) a team struggling with 3 maps at the beginning of the game.

8.2.2 Evaluation

To create comparable results for tracking quality as well as tracking speed over various datasets, tracking approaches and situations, we implemented a frame server that loads uncompressed raw images from the file system rather than from a live camera view. Our tracking approach was ported and evaluated on the mobile phone and on a PC (Microsoft Windows) platform. As typical in Computer Vision, we evaluated both the matching rate and the robustness of our tracker against several different sets of tracking targets. Our tracking approach resulted to have comparable and – in some cases – even better matching rate than the original SIFT implementation. Also, the hybrid approach proved to be robust also against most hard cases (strong tilts and lighting effects, high level of motion blur). A detailed analysis of the evaluation results can be found in (Wagner et al., 2009).

MapLens

Following previous studies on collaborative use in mobile Augmented Reality (see Morrison et al., 2009), we set up a field experiment to better understand differences in collaboration and tangible Mobile AR device use in urban environment in August 2009. In this field trial participants used MapLens (see Figure 18), an application on a mobile phone that works like a magic lens over a paper map, which provides an additional layer of digital information to the view seen through the mobile phone's camera. The application is based on the tracking technology developed in WP5/WP7.

Figure 18. The MapLens application showing a live video of the paper map underneath, augmented with icons and labels registered to map locations.

Our study was the first study of its kind to synchronously trial multiple, single and shared users and mobile devices in the field. The three configurations were: solo users with one device; a team of three sharing one device; a team of three with each one device. Each configuration completed the same game tasks in the same given time. We found that solo users could complete the game tasks in the given time therefore shared use as not required. However, in teams with more devices, the devices were used in a more expansive way. We observed divergent roles emerging and that the teams still decided to share only one paper map. We also noted that teams largely stayed together to complete tasks, despite it was not essential to complete the game. In teams sharing the device, looking at and pointing at each other's screens and the map beneath, occurred more than in the teams where everyone had their own device. The findings of this study have been reported in more depth in a paper submitted to CHI2010 (Morrison et al., 2009b).

For a detailed description of the evaluation process and results, please see D7.4.

8.2.3 Specification

Hardware and OS	Advanced Mobile Phones, Win Mobile
Software	Studierstube ES
Core Features	Natural Feature Map Tracking
Status	Prototype
Intended users	Mobile Navigators
Showcases	WP7, WP9

Relevance beyond project	yes

8.2.4 Publications

Morrison, A., Oulasvirta, A., Peltonen, P., Lemmela, S., Jacucci, G., Regenbrecht, H. and Juustila, A. (2009). *Like bees around the hive: a comparative study of a mobile augmented reality map*. In Proceedings of the 27th international Conference on Human Factors in Computing Systems (CHI '09) pp. 1889–1898.

Morrison, A., Lemmela, S., Oulasvirta, Schmalstieg, D., Peltonen, P., Mulloni, A., Regenbrecht, H., Jacucci, G. and Juustila, A. (2009b). *From Single to Multi-Lens Collaborative Augmented Reality on Mobile Phones*. Submitted to CHI2010.

Wagner D., Reitmayr G., Mulloni A., Drummond T., Schmalstieg D., "Real Time Detection and Tracking for Augmented Reality on Mobile Phones," IEEE Transactions on Visualization and Computer Graphics, 18 Aug. 2009. IEEE computer Society Digital Library.

8.3 Content Manager

8.3.1 For CityWall

Content Manager is software designed for downloading content from open web services and to insert this downloaded content into a content database. ContentManager is used by CityWall for retrieving content to be shown on the display and for content moderation (browsing through and deleting content from the database via a web interface). ContentManager has been used to retrieve content from Internet's social media sites such as Flickr and YouTube. In the European City of Sciences exhibition, CityWall was also used to show content created in the IPCity's MR Tent. As the content management happens through web interface, content moderation can be done remotely which can be useful for installations setup in the city or other locations visited more rarely.

8.3.2 Technology components

Content Manager consists of two parts:

1. Downloader component
2. Moderator component

Downloader component is a collection of PHP classes that are called by a bash script to do the media fetching from the different services in parallel and to insert the metadata of the downloaded media files into the databases.

The Moderator component is a web application designed for moderating the content fetched by the Downloader component and to group the content into categories ("worlds").

Downloader component improvements

YouTube video downloading was previously done using links in Google cache. This functionality was now replaced by using the open source youtube-dl program.

Also, multiple (parallel) file download reliability and time was improved. To retrieve masses of data using the public social media APIs can be difficult, as the services are sometimes under heavy load and stop functioning properly. Error checking was improved to overcome failures in the services.

Moderator component

Content Manager can now be also used for creating content containers that we call "worlds". These worlds are used to group downloaded social media into categories and themes that have certain attributes. For example we could create a world for "environmental" awareness and "urban issues", define tags (search criteria) for these worlds and then let the system collect social media to these worlds based on the tags we defined. This feature is used by City Wall to group different topics into “worlds of information” that can be represented through the 3D interface. Figure 19 shows the Content Manager main view in which four worlds are listed: HelsinkiBall, ParisBall, MultitouchBall and PinBall.

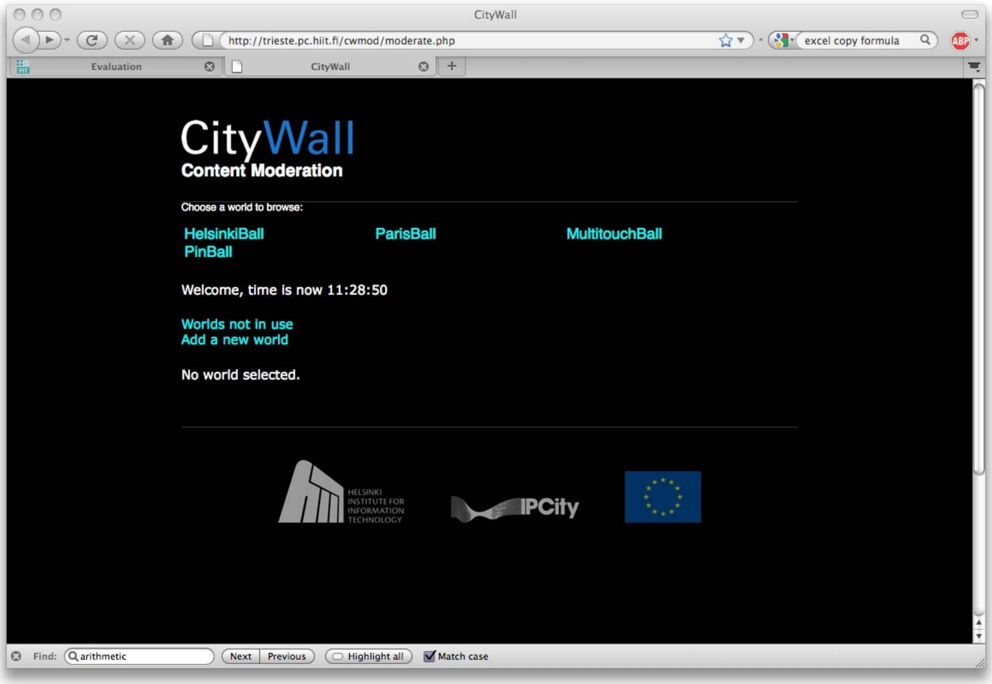


Figure 19. ContentManager Main view.

Each world name is a link, which can be clicked and that will take the moderator to browse the content of that world. In the content view (see Figure 19) all content to that world is presented in a list that shows the following attributes of a content item:

- Title: user created, this is shown below the thumbnail of the media item on CityWall
- Database Id: integer, for internal use only
- Type: image or video
- Published: date when the the item was created, shown on CityWall
- Tags: the metatags attached to this items, for example “Helsinki, samba, carnival”, these are used to organize the content to the different worlds
- Description: user created, at the moment this information is not used on CityWall, but could be in the future
- Picture: user created, after the textual information, if the item’s type is image, the image is shown

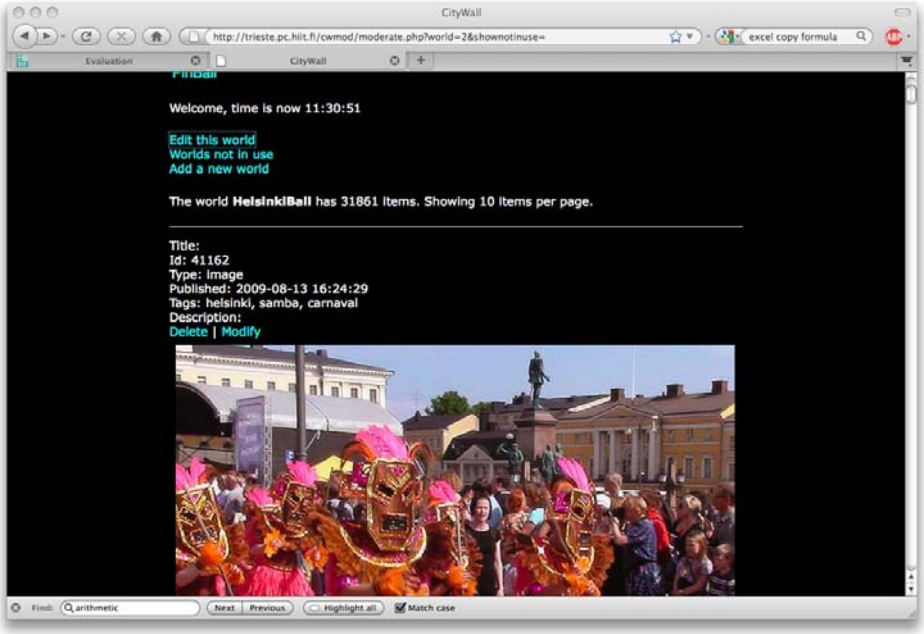


Figure 20. Showing the content for a world in ContentManager.

By clicking the link "Modify", which is presented after the information listed above, the moderator can change the attributes. The moderator has also the option to edit the selected world's parameters by clicking the "Edit this world" link, which will take the moderator to the Edit world view (see

Figure 20). In the Edit World view it possible to setup the different parameters for the different worlds.

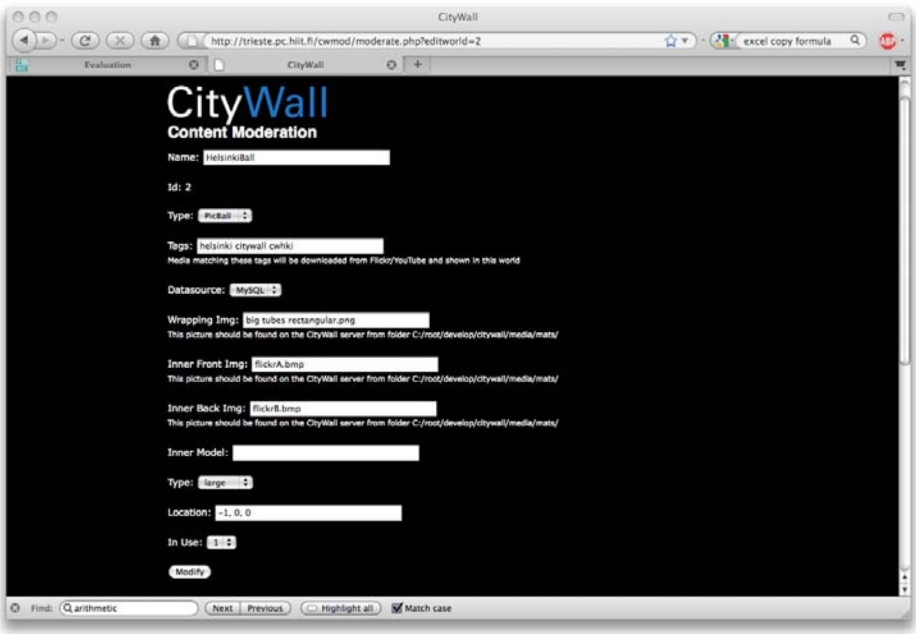


Figure 21. ContentManager Edit world view.

The Edit world view (see Figure 21) allows the moderator to control the following attributes of a world:

- Name: name of the world, for internal use only
- Id: database id of the world, for internal use only

- Type: type of the world that can be either picball (includes downloaded images or videos from Flickr/YouTube), pinball (includes the on-site user created content, such as SMS messages and drawings) or tentball (a special world that was created to show pictures created in the MR Tent)
- Tags: if the type is picball, media matching these tags will be downloaded from Flickr/YouTube and shown in this world
- Datasource: by which means this world tries retrieve its content from, can be MySQL/Flickr/SMS/MMS/RSS
- Wrapping Img: the filename of the image that is shown when the world is closed
- Inner Front Img: the filename of the image that is shown on the front the world is open
- Inner Back Img: the filename of the image that is shown on the back the world is open
- Inner Model: the model how the world should behave
- Type: how big the world should be (tiny/small/medium/large)
- Location: the coordinates where on screen the world should be located when CityWall is started
- In Use: 1/0, with this attribute one can temporarily disable the world so, that it not loaded and shown on CityWall

City Wall and database schema replication

In 2009 both Downloader and Moderator components were installed in the two CityWall installations we have at Lasipalatsi (Helsinki city centre) and at Spektri (for our internal use). Database schemas has been replicated between the two sites, so both databases can be transferred to another site if needed, which has made maintenance easier.

8.3.3 Testing and Public Demonstration

Content Manger was tested in public with CityWall at European City of Sciences exhibition in Paris 2008 and in the City Wall permanent installation at Lasipalatsi 2009. Specification

Hardware and OS	PC hardware, Linux
Software	• MySQL (open source) • Linux bash shell (open source) • PHP (open source) • phpFlickr (open source) • phpYoutube (open source) • youtube-dl (open source) • Content Manager scripts written in bash and PHP
Core Features	Imports media meta-data and files from social media services into database and offers a web UI for content moderation and creation of content collections (“worlds”)
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)

8.3.4 Conclusions and Future Work

ContentManager has proved to be a useful tool for managing the worlds displayed on CityWall and moderating the content shown on it. With the web based tool worlds can be easily created, edited and disabled if needed. If some content is not suitable to be shown in public, with ContentManager a non-technical person can remove these media objects with a few clicks.

ContentManager could be improved still in many areas:

1. *User management*. At the moment the system recognizes one superuser that can use all the features. If we would like to have separate moderators that should just watch for the content, a better user management system would need to be created that would allow having different levels of access for the users (a content manager that has no access to edit the worlds, for example).
2. *Searching*. Currently there is no possibility to search for a specific published content item. If there is a lot of content, it can be problematic to find a certain item.
3. *Downloading content by area*. Instead of just specifying certain textual tags like “Helsinki” that is used as the filtering criteria for downloading, it could be useful if one could use Google Maps like interface to specify an area from where the content should origin from.

8.4 HMDB

HMDB (HyperMediaDataBase) is a relational database specifically intended to support storing, searching and retrieval of location based media with metadata elements over the internet. It provides several standard based interfaces to add and retrieve location based media:

- ATOM Publishing Protocol API (APP) over HTTP enables any client application to enter and retrieve media and associated metadata from the HMDB.
- APP enables also retrieving of media and metadata, but more commonly the HTTP and KML are used to access the media, especially when working with map based applications.
- Java API enables Java clients to access the HMDB content. The Java API is implemented on top of the HTTP and APP APIs for reusability and flexibility.

Example below illustrates the APP API usage, where the metadata of the media is packaged into a XML document. The XML document is then uploaded to the HMDB using APP, followed by the media files, using HTTP multipart upload protocol.

```
<?xml version="1.0" encoding="utf-8"?>
<atom:entry xmlns:atom="http://www.w3.org/2005/Atom"
xmlns:hmdb="http://www.atelieros.org/2007/hmdb#">
  <atom:id>urn:uuid:001c9afa-0df5-7a8e-f494-ad36342e461e</atom:id>
  <atom:title>Story</atom:title>
  <atom:content type="text">Story</atom:content>
  <atom:author><atom:name>Vesa-Pekka</atom:name></atom:author>
  <atom:category term="/session/started">2009-09-
15T14:01:39+03:00</atom:category>
  <atom:category term="/session/ended">2009-09-
15T17:04:58+03:00</atom:category>
  <atom:category term="/session/tool">MobileMediaCollector</atom:category>
  <atom:category term="/session/id">001c9afa-0df5-5d1b-13da-
953949f3645d</atom:category>
```

```
  <atom:category
term="/location/combined">26.1634426,65.1880875,124.0000000</atom:category>
  <atom:category term="/dc/tags">key,words,here</atom:category>
  <hmdb:type code="O">HyperObject</hmdb:type>
  <atom:link rel="hmdb-media"
href="e:\StoryMediaFiles\StoryCollectorImage2009-09-15_142634.jpg"
type="image/jpeg" length="142858" title="8"></atom:link>
</atom:entry>
```

The metadata entries included in the APP XML metadata file above are extracted then in the HMDB and stored in the hierarchical keyword tables. Searches can be performed on these keywords, e.g. to find content created by a specific author (<atom:author>), or content in a specific area determined by the GPS coordinates (<atom:category> where term=/location/combined metadata is specified).

As the media is fetched from the HMDB, the content is provided either in a similar XML metadata file following the APP, or as a KML file. Example of a partial KML file is below:

```
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://www.opengis.net/kml/2.2"
xmlns:atom="http://www.w3.org/2005/Atom"
xmlns:gx="http://www.google.com/kml/ext/2.2"
xmlns:hmdb="http://www.atelieros.org/2007/hmdb#">
<Document>
  <Folder><name>Session: Jane</name>
    <atom:author><atom:name>Jane</atom:name></atom:author>
    <ExtendedData>
      <atom:entry><atom:id>14f8c63f-d8f5-332f-8480-96125c34cda9</atom:id>
      <atom:author><atom:name>Unknown</atom:name></atom:author>
      <hmdb:type code="C">HyperElementContainer</hmdb:type>
...
    <Placemark>
      <name>Story</name>
      <Point>
      <coordinates>26.1250362,65.2444000,131.0000000</coordinates>
      </Point>
      <ExtendedData>
         <atom:entry><atom:id>1cacd054-8da1-3bc5-b6ad-
ae2e702039dd</atom:id>
        <atom:author><atom:name>Jane</atom:name></atom:author>
        <atom:title>Story</atom:title>
        <hmdb:type code="O">HyperObject</hmdb:type>
        <atom:category hmdb:id="7"
term="/location/combined"><![CDATA[26.1250362,65.2444000,131.0000000]]></at
om:category>
        <atom:link hmdb:id="29" hmdb:owner="28" href="http://hmdb-
proxy.ipcity.eu:80/hmdb/files/2009/09/15/14/35ff1c5f-7a61-35c8-9cb4-
dd181e4c3f5e-StoryCollectorImage2009-09-15_144347.jpg" rel="hmdb-media"
type="image/jpeg"/>
       </atom:entry>
     </ExtendedData>
    </Placemark>...
```

The extract describes a session stored with the MobileMediaCollector (MMC) used by Jane, containing a story told in GPS coordinates 26.1250362,65.2444000,131.0000000, containing an image.

Development of the HMDB during the project year 2009

- HMDB has been enhanced with new features needed in the MobileMediaCollector (MMC) tool. Previously, a location based story with media could hold only one media item. A location based story then could have either an image or a voice recording. New feature supports multiple media files in a story. With this enhancement, users can now create voice annotated photographs or if they prefer, illustrate their stories with a photograph. Also, new metadata was added to the story, including:

  - Compass direction where the user is looking at when telling the story. With our current hardware with no electronic compass, the direction is manually entered by the user. Already now several new devices include an electronic compass. Viewing tools which assist users in viewing the content from the HMDB (usually on a map) can now use the compass direction in showing where to the user was facing.
  - Is the story positive, negative or neutral, relating to the place. This feature was requested from the users, since in a specific usage scenario it was necessary to evaluate places with these categories.
  - Is the story public or protected. Public stories can be viewed by all, as protected stories are viewable, for example, to architects using the MobileMediaCollector for gathering material more interesting to the planners, not to the general public.

- A session support was added both to HMDB and MMC. This is needed as ground work for the saving of the path of the user travelling as she moves around collecting location based stories with MobileMediaCollector.
  - In our user tests, we have observed that while users view the stories on a map, they often try to remember the path they took while walking around. The context of the path helps the users in understanding the context of the location based story and to knit the stories in different places together, when viewing them afterwards. Since the stories are separate from the path, a concept of a *session* was designed. Media (with metadata; the stories) are saved and associated with a session. The path is saved with the session.
  - This enables us to export the session along with the path in the KML export feature of the HMDB, thus enabling the users to view not only their stories on a map, but also the path they took in their field trip. Sessions are included as KML Folders in the exported KML file. This also assists the users in limiting the content they want to view according to the session hierarchy. When the HMDB contains hundreds of stories, it is easier to view the stories within sessions, which are constructed by user and by date.
  - Currently, the session feature has been implemented and the path is stored on the mobile device in GPX format, and the session data transmitted to the HMDB, but the path of the user it is not yet uploaded to the HMDB.
  - In the future, the path information along with other metadata, can be used to generate Google Earth tours (http://earth.google.com/tour.html#v=5)
- Supported integrating of the MapLens also to the Imagination content server. As the MapLens implementation uses open standards in communication with the HMDB (HTTP, ATOM Publishing Protocol and KML), it has been relatively straightforward to enable MapLens in switching between using either HMDB or Imagination Content Server. The main issues have been in the format and structure of the KML file and how it effects the download and parsing speeds in real settings.

8.4.1 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 meta-information 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.)

Joshi, D., Datta, R., Zhuang, Z., Weiss, W. P., Friedenberg, 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.

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

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.

Saushik Chakrabarti, Kriengkrai Porkaew, & Sharad Mehrotra, Efficient query refinement in multimedia databases. Proc. of the IEEE International Conference on Data Engineering (ICDE), San Diego, California, USA, 2000, 196.

8.4.2 Testing and Public Demonstration

HMDB has been used in WP6 trials and is part of the MR-Tent technology. HMDB was in used in Fall 2008 trials of MapLens in Helsinki WP7 trials. HMDB is used by the Mobile Media Collector (MMC) in user tests held in Oulu.

8.4.3 Evaluation

HMDB provides the basic features needed in the management of location based media with metadata. The new features implemented during the year 2009 further enable new application functionality. Due to the prototypical nature of the HDMB, some reliability problems persevere, but this is expected in research projects. On the other hand, as client applications are often dependent on the HMDB and fail if the HMDB fails, in field trials this can be a real problem. During the project therefore, a considerable amount of work has been put into making the code more reliable and efficient. Currently a group of students conducting unit and systems testing to HMDB.

8.4.4 Specification

Hardware and OS	Linux/Unix, Tomcat web server, MySQL server.
Software	Java based servlet, additional software packages.
Core Features	Location based media management with metadata handling. ATOM Publishing Protocol and Java API for uploading media and metadata. ATOM and KML interfaces for downloading media.
Status	Advanced prototype
Intended users	Background service
Showcases	WP6, 7, 9
Relevance beyond project	Available

8.4.5 Publications

Morrison, A., Oulasvirta, A., Peltonen, P., Lemmela, S., Jacucci, G., Regenbrecht, H. and Juustila, A. (2009). *Like bees around the hive: a comparative study of a mobile augmented reality map*. In Proceedings of the 27th international Conference on Human Factors in Computing Systems (CHI '09) pp. 1889–1898.

9 Time Warp (WP8)

In Time Warp location based story telling game is developed while involving the user playfully. The natural border of time is overcome with the help of MR technology.

9.1 Mobile AR with Morgan Light

Development on the Morgan Light framework continues based on experience gained from the first prototype. In particular, addressing the need for higher performance and continued support of Morgan on mobile platforms, it was necessary to develop an abstraction that can cope with the incompatible OpenGL ES 1.1 and 2.0 APIs. While older mobile platforms such as Windows Mobile based PDAs support only OpenGL ES 1.1, more modern platforms such as the Apple iPhone 3G, have better support for OpenGL ES 2.0. Further, OpenGL ES 2.0 is, due to its forward looking shader-based architecture much more suitable for long-term evolution of Morgan rendering as it also provides almost plug and play compatibility with desktop OpenGL 3.x.

Considering all options, in year 3 of IPCity the design of a new Rendering Abstraction Layer (RAL) for Morgan began, building on the experience of the previous two designs that evolved in the eight years of continued Morgan development. The main focus was to learn from mistakes made in the first and second implementation, in particular the feature overkill in the second and current rendering abstraction of Morgan, which is becoming a burden when moving to newer rendering APIs and more modern hardware for rendering, both on mobile and desktop systems.

The newly designed Morgan RAL provides a much higher level interface to the most common rendering APIs, in particular OpenGL 2.x and 3.x and OpenGL ES 1.1 and 2.0 and Direct3D 9 and 10 (support for Direct3D 11 is a possible future feature). This new, higher level interface provides direct, single call activation of advanced rendering effects, such as soft shadows in VR environments, phantom objects and advanced lighting in AR environments, reflections and refraction in both environments (with additional hardware support). While Morgan does currently support some of these features as well, they are difficult to use and control, making development slow and error-prone when trying to use these advanced features.

Another benefit of providing a higher level of abstraction is the significantly easier integration of newly published advanced effect algorithms into RAL. While previously, for example porting a soft shadow algorithm published as OpenGL or Direct3D short code required converting of the OpenGL calls into the Morgan framebuffer equivalents, the new RAL keeps all this away from the programmer at a slight cost of additional code needed in both the Direct3D and OpenGL implementations of RAL. However, the most time consuming and error prone task of hand-converting existing published solutions for new algorithms is almost completely removed.

As such, the development of RAL has contributed to the overall improvement of Morgan beyond the end of the project. By improving the implementation quality of the Morgan framework, exploitation activities can (for example) focus on improved gameplay and usability, better sense of presence and more realism where appropriate, rather than time-consuming debugging of low-level Morgan framework functionality to use the research AR/VR framework in commercial grade applications.

9.1.1 Testing and Public Demonstration

As the development of Morgan RAL is still ongoing, it was decided not to use it for the last TimeWarp game. Instead, TimeWarp will be used as part of the remaining quality assurance process of RAL before production use. At this time, RAL will fully support TimeWarp on new platforms and systems not available at the time TimeWarp was conceived.

9.1.2 Evaluation

An extended evaluation of Morgan RAI has not been conducted yet. The current development plan expects final development of RAL to be completed within late 2010, about half a year after the conclusion of IPCity. As such, RAI will be highly valuable for the long-term exploitation activities at FIT of technologies and games developed as part of IPCity.

9.1.3 Specification

Hardware and OS	Windows XP and 7, iPhone OS, Android, Mac OS X, Linux, Window Mobile
Software	Morgan Framework
Core Features	Higher level abstraction of rendering
Status	Design specification complete
Intended users	All users of Morgan
Showcases	WP 8
Relevance beyond project	Long term evolution on Morgan and IPCity games and technologies based on Morgan

9.2 Physics Abstraction Layer

In recent years, physics for augmented environments has received much attention because it can significantly improve realism of augmented objects in the environment and hence increase the sense of presence. One of the highly developed areas is rigid body dynamics because it is most computationally affordable compare to other physics simulations for example soft body dynamics (Ohlenburg 2004, 2005). Some fields such as water simulation or fire simulation still require substantial computational power for simulating them in real-time. Consequently, the focus of development for a real-time physics abstraction layer for Morgan was in the area of rigid body dynamics as a first step towards a future, full physics abstraction adding effects such as water.

For IPCity, physics can add to the realism in the augmented environments created for example in TimeWarp, as they provide means for users to implicitly interact with the world around them. Players of the game would no longer be able to walk into moving characters in the game, as those could avoid the players. However, the opposite, preventing players from walking into virtual objects would require for example smart textiles, which is out of the scope of IPCity and the currently planned features of the Morgan framework (Ohlenburg et al. 2004).

VR/AR Physics abstraction layer has features similar to PAL and OPAL (see below) such as abstract design, which allows to support more than one low-level physics engine, basic features such as Rigid Bodies and Joints. Apart from these basic features, the Morgan PAL offers tight integration into the other parts of the Morgan framework and the following new and distinguished features:

High-level functionality: VR/AR PAL provides more high-level functionality such as support for Force Fields

Extensibility: VR/AR PAL is designed in mind to be easy to extend therefore developers can create unlimited variety of different physics effects. For example Motors are classified as Body Motors (ones which act on rigid bodies) and Joint Motors (ones which act on joints). Most of the body motors can be easily derived from the Body Motor class and the same applies for joint motors. Organized and transparent hierarchy helps to simplify creating of new motors.

Usability: One of the main goals of VR/AR PAL is to provide a library which is very simple and easy to use even for people with minimum knowledge of the physics simulation nature.

Therefore the approach was to create a high-level functionality which is transparent and easy to understand. VR/AR PAL provides end-user a set of external functions (so-called standard interfaces) which is necessary to create physics simulation and hides all irrelevant internal functionality in order to avoid errors and complications. However if the end-user wants to extend and modify VR/AR PAL it might be necessary to work with the internal functionality.

VR/AR Support: As the name suggests physics abstraction layer was developed with an idea to be used with virtual and augmented environments. In order to achieve this goal VR/AR PAL was integrated with Morgan VR/AR framework which provides support for a large variety of VR/AR specific features such as different tracking components, VR/AR specific I/O devices, etc. Morgan VR/AR framework also provides support for handling I/O and scripting behaviours by means of Interaction Prototyping plugin.

9.2.1 Related Work

In recent years a number of physics abstraction layers were developed (Boeing et al. 2007). There are Open Physics Abstraction Layer (OPAL) and Physics Abstraction Layer (PAL). The goal of these projects is to provide an interface to multiple engines and user-friendly, comprehensible functionality. The Physics Abstraction Layer has many features similar to the Open Physics Abstraction Layer. However currently this project is being actively maintained and new features are added. One of the main difference between PAL and OPAL is the physics engines support. While OPAL supports only Open Dynamic Engine, PAL provides access to 11 physics engines. Authors of PAL emphasize that one of the purpose of this abstraction is to provide users opportunity to experiment with different engines to achieve best results. On top of this PAL provides most of the functionality similar to what OPAL has. The table below provides an overview of OPAL and PAL features:

Feature	OPAL	PAL
Supported Physics Engines	One: Open Dynamic Engine(ODE)	Eleven: Ageia PhysX, Box2D, Bullet, IBDS (experimental), JigLib, Newton, ODE, OpenTissue (experimental), Simple Physics Engine (experimental),Tokamak, TrueAxis
Rigid Bodies and Geometries	Bodies and geometries: Box, Sphere, Capsule, Planes, Triangular meshes	Bodies: Box, Capsule, Compound, Convex, Sphere; Geometries: Box, Capsule, Convex Mesh, Concave Mesh (terrain), Heightfield (terrain), Plane (terrain), Sphere;
Joints	Hinge, Universal, Ball, Wheel, Slider, Fixed	Spherical (Ball and Socket), Revolute (Hinge), Prismatic (Slider), Generic 6DOF
Motors or Force Actuators	Attraction motor, Geared motor, Servo motor, Spring motor, Thruster motor	Force actuator (Generic), DC Motor, Servo Motor, Hydrofoil Motor, Propeller, Spring
Sensors	Acceleration sensor (linear and angular acceleration), Incline sensor (the angle the solid has rotated relative to solid local rotational axis), Ray cast (closes point of intersection), Volume sensor (returns all the solids,	Contact, Compass (Angular position), GPS, Gyroscope (Angular velocity), Inclinometer (Angular position), PSD (Position Sensitive Device-Ray casting), Velocimeter (Linear velocity), Transponder (Distance

	which exists in the range of the volume solid i.e. inside of it	between objects);
Event Handlers	Collision Event Handler Joint Breaking	No
Vehicle Creation Support	No	Yes

OPAL helps to simplify integration and usage of physics engine, in particularly ODE. It contains components for creating rigid bodies and constraints between them, event handling, acquiring information from simulation and applying forces on bodies. For the latter there are several general-purpose and specific-purpose motors. General-purpose motors include attraction, spring and thruster. Thruster motor corresponds to the functionality for applying general force which each physics engine provides. Specific-purposed motors can not be used in a big range of simulations, but rather in the specific cases. Geared motor is good for simulation car engine and servo motor is good in robotic simulation or controller simulation when user wants always achieve some desired angle.

However, for the (augmented reality) game developers broad specter of forces simulation functionalities are needed. Due to the great variety of game situations, which involve physics it is more preferable to have a number of abstract functionalities which can be easily used to build a specific case. PAL and OPAL have many common features such as similar structure of components. However the main focus of PAL is providing an interface to a significant number of existing physics engines. It has only several high-level functionalities which significantly simplifies the process of implementing physics effects. The main disadvantages of currently available physics abstraction layers are usability and extensibility problems. It is also beneficial to have an abstraction layer with more high-level functionality, which can allow developers and designers to create bigger variety of physics effects. VR/AR PAL is an alternative to these abstraction layers. New and distinguish features, which are provided by VR/AR PAL are high-level functionality such as support for Force Fields, extensibility, usability and VR/AR support.

Ohlenburg, J. Improving Collision Detection in Distributed Virtual Environments by Adaptive Collision Prediction Tracking. Proceedings of VR 2004, pp. 83-90

Ohlenburg, J., Müller, J.: Interactive CSG Trees Inside Complex Scenes. IEEE Visualization 2005, p. 106

Boeing, A., Bräunl, T. Evaluation of real-time physics simulation systems, Computer graphics and interactive techniques in Australasia and South East Asia 2007, pp. 281 - 288

Ohlenburg, J., Herbst, I., Lindt, I., Fröhlich, T., Broll, W.: The MORGAN framework: enabling dynamic multi-user AR and VR projects. VRST 2004, pp. 166-169

9.2.2 Testing and Public Demonstration

Apart from user studies evaluating the basic usability and performance of PAL, no public demonstrations or tests have been conducted within IPCity.

9.2.3 Evaluation

User studies have been conducted subsequent to the implementation of the physics abstraction layer and integration it into the Morgan VR/AR framework by means of Morgan X3D-plugin extension. The user study consisted of two parts: user study for content creators and user study for software developers. The user test for content creators is aimed to evaluate X3D rigid body dynamics specification developed along with PAL. Evaluation was focused on the usability of the given X3D physics specification. The goal of the user study for software developers was to evaluate functionalities of VR/AR PAL middleware. The user test aimed to evaluate how easy it will be for software developers, with small or no experience, to develop a physics simulation. They have been tasked to work with VR/AR PAL existing functionalities in order to add new functionalities i.e. extend physics abstraction layer with new features.

Early evaluation results showed that integrating physics into the existing TimeWarp gameplay would be too much work for the time remaining within IPCity. As such, it is currently planned to integrate physics at a later stage, possibly as part of post-IPCity exploitation activities building on TimeWarp.

9.2.4 Specification

Hardware and OS	Windows
Software	Morgan
Core Features	Rigid body dynamics
Status	Advanced prototype
Intended users	Morgan developers, Morgan users
Showcases	WP8
Relevance beyond project	Exploitation activities related to WP8 game TimeWarp

10 City Tales (WP9)

The City Tales work package is working towards establishing a digital layer over the urban tissue of a city that contains a story telling environment based on participant's input. The power of a community driven content creation approach has been shown in many Web 2.0 applications, such as encyclopedia, social networking portals and geo-information. Our aim is to focus this power to the creation of content into the digital layer of an urban environment, supported with easy to use yet powerful tools for mixed reality content communication.

10.1 Second City Database

Initially WP9 team investigated the use of HMDB to store media elements for our City Tales approach. The hierarchical structure of the HMDB puts the hierarchy of the media elements into the foreground and by meta-information added we would have had the possibility to add certain functionality to it.

But on progress in the design we also noticed that we need user management, geo location and time based queries, etc. from the very first start. The largest step towards a new database was however our internal decision to use KML as the major communication protocol to both export and import data into the database. One simple reason for that was to be compatible with "clients" like Google Earth that allows both retrieving, browsing of all data now contained in the DB but also provides a (simple) content creation tool in the same moment.

The decision was to develop and setup a new database to be provided also as service to all IPCity partners. We installed a dedicated server that can hold all types of (mime-typed) data files that are stored in geocoded indices. The server does validation of uploaded (KML or any type of XML) files against freely definable schemas. Upload and download is running using standard HTTP protocol to ease integration and provide access with a wide number of clients.

Automatic features provide package upload/download mechanism (ZIP compacted archives, like KMZ) in order to keep track also of the media files referenced inside a KML document. The server is built in order to allow also extensions to KML (which we need in IPCity, such as sound). We have a user management in the system that allows multi-user content creation and manipulation.

The dedicated server will be put to a public reachable IP in the internet, so any IPCity partner is invited to use the service then. We see also a chance that content of other IPCity prototype applications and demonstrators is stored in our database compared to the current situation of storing them in several dislocated and separated databases. Our idea was to demonstrate the possibility of combing different work packages also on the content level in order to provide a common view on data.

10.1.1 Related Work

Figure 22: Related projects

Several projects (Figure 22) investigate the possibility contained in the possibility of nowadays emerging technology to interlink locations with information. Global Positioning

System (GPS) provides for most approaches the basis to locate the user's "playback device" on global scale and to provide information from a database related to this location. Games, experiences, narratives are provided and many of the existing systems also try to involve the community to introduce the content themselves. Lack of standards, diverse "playback technologies", technical pitfalls, non oriented user base however let's most of the existing systems stay in a beta-phase. Also none of the investigated systems are using the power of 3D display to create a compelling mixed reality experience.

Urban Tapestries[3] investigated the social and cultural uses and behaviors to annotate a city; Hewlett Packard runs the project mscape[4] as a continuation of Mobile Bristol dating back to 2002 to allow participants to create games, guides, stories triggered by GPS location – currently containing about 300 "mediascapes". MIT's M-Views[5] is a context sensitive mobile cinematic narrative system that enhances real world with video clips based on WiFi triggering. MIT's Museum Without Walls Project[6] puts history and science in the user's hand and turns the world into a museum by adding location based information and stories to historic relevant places. The project is currently on hold due to funding issues.

A relatively new system is bliin YourLIVE![7] which is dedicated to creating a fluid social network for sharing personal experiences - photos, videos, sound & text - in real-time, located on a world map and focuses since 2007 directly as a Web 2.0 application on a young community with currently about 23.000 "shares". GPS Mission[8] is a GPS game for mobile phones where the community can create missions and riddles to other members. The last two systems are purely commercial, however all services are free so the business model behind is not obviously visible currently.

A special class of more content focused applications can be defined, such as Sonic City[9] which generates a personal soundscape co-produced by physical movement, local activity, and urban ambiance. Sony's Street Beat followed a similar approach to enhance the audio environment in a city. Landvermesser.tv[10] – a new Berlin art project creates fiction literature stories on real places in the city, written by 10 participating professional writers. The stories can be perceived as a combination of audio stories and video clips played on city embedded displays or on-line.

Combining existing resources and location based information in a mixed reality environment is demoed by the WikiTude[11] project that allows users of the new Android G1 phone to overlay the camera image of the user's environment with WikiPedia entries related to the actual location of the user with a definable range. Available geo-located Wikipedia entries (around 350.000 worldwide) are represented with balloons of information over the real buildings or points of interest. Selection enables to read the attached Wikipedia entry on-line.

An interesting approach is to omit the necessity of a continuous tracking throughout the story telling domain by not relying in GPS or similar approaches, but by using discrete points of

---

[3] http://research.urbantapestries.net/ – accessed last 19.12.2008

[4] http://www.mscapers.com/ – accessed last 19.12.2008

[5] Crow, D., Pan, P., Kam, L., Davenport, G., M-views: A system for location based storytelling, Proceedings of Ubiquitous Computing (Ubicomp), Seattle, Washington, October 12-15, 2003, pp 31-34.; more information available at http://ic.media.mit.edu/projects/M-Views/

[6] http://museum.mit.edu/mwow – accessed last 19.12.2008

[7] http://www.bliin.com/ – accessed last 19.12.2008

[8] http://gpsmission.com/ – accessed last 19.12.2008

[9] http://www.tii.se/reform/projects/pps/soniccity/index.html – accessed last 19.12.2008

[10] http://landvermesser.tv/ – accessed last 19.12.2008

[11] http://www.wikitude.org/ and http://www.youtube.com/watch?v=8EA8xlicmT8 – accessed last 19.12.2008

identification or tracking, e.g. 2D-markers. The method of "mobile tagging" becomes widely popular as the link between the real world and the digital content is achieved with a simple to create 2D barcode marker that is placed at an exactly defined location. As one example of the many-many we point to the Active Print[12] project which is exploring how printed materials and digital displays can be linked to online content, services and applications in all kinds of urban/suburban/rural situations.

10.1.2 Testing and Public Demonstration

Multiple testing scenarios have been carried out during YR4 of the project to test and demonstrate the system. In workshops organized by the UniAK and IMAF the technologies have been tested at Naschmarkt in scenarios where participants were invited to run the complete process of authoring, injecting and retrieving content on the go.

In these workshops based on the student's earlier experiences and in conjunction with the course topic the basic idea of the workshop was to design collaborative games. Several gaming ideas were proposed and brainstormed, two selected for implementation. A "Virtual Development" Monopoly-style game and a "Sound Meal" named new gaming concept has been implemented on the basis of the City Tales architecture.

The technology has been used to implement the original idea of city tales, that were to be written by invited authors. These writers did create fictional urban stories related to the specific location of the Naschmarkt. The stories are interwoven so characters can appear in multiple contexts – a true non-linear story experience.

The implemented games have been played by students (N=14) the story experience was used by young people (N=26).

10.1.3 Dissemination

The Second City Database has been successfully proven to act as the data management backbone of large-scale, multi-user, geo-located, multi-media data databases. The flexibility of the database structure, the application of industrial standards, the use of widely available development tools and accepted content formats allows the easy integration of existing augmented reality clients, approaches, prototypes into the system. This integration capability creates a great opportunity to AR and MR experts worldwide – both commercial as well as non-commercial – to build large-scale AR application scenarios based on our technology.

Apart from the internal use and integration of the SecondCity Database in following work-packages:

- WP9 – City Tales system with multiple clients, such as MR-Player, Wall Blogging, Walking Explorer, Authoring User Client, DB Management Interface;
- WP7 – MapLens application for data storage and retrieval of location based imagery

External projects are already making use of the database, such as research projects:

- AndroidAR – a HitlabNZ internal project to create AR applications based on the Android platform
- WorldWideSignpost – a TUG internal project to combine panoramic tracking with global coverage of augmented overlay data, see 10.3. and **Fehler! Verweisquelle konnte nicht gefunden werden.**.
- several other non-public prototype projects based on the aspect of storing, retrieving data in a hybrid tracking environment on global scale.

[12] http://www.activeprint.org/ – accessed last 19.12.2008

10.1.4 Specification

Hardware and OS	▪ Intel Based PC Platform ▪ SUSE Linux 11
Software	▪ MySQL DataBase Server ▪ Apache TomCat 5.5
Core Features	▪ Multimedia content database ▪ Geolocated indexing of content ▪ Marker Management ▪ User Management ▪ KML import/export ▪ XML schema checking
Status	Stable prototype
Intended users	> 10, content providers, story authors
Test users	~ 45
Research Workpackages	WP7, WP9
Relevance Beyond Project	Stable community server for geo-located content. Flexible and extendible and support further functions and request types. High.

10.2 Scouting

See section 7.4 Scouting.

10.3 Mobile Navigation and Panorama Annotation

Real-time creation, annotation and tracking of panoramic maps on mobile phones was developed. The maps generated with this technique are visually appealing, very accurate and allow drift-free rotation tracking. The method runs on mobile phones at 30Hz and has applications in the creation of panoramic images for offline browsing, for visual enhancements through environment mapping and for outdoor Augmented Reality on mobile phones.

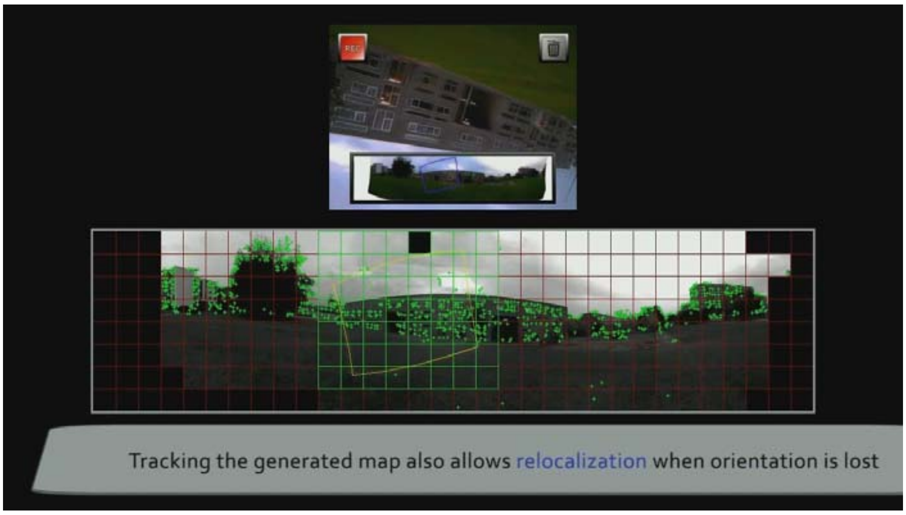


Figure 23: The panorama tracker creates a representation of the environment during the process of collecting images for the natural feature based tracking approach

10.3.1 Related Work

Panorama creation is a widely discussed topic in computer vision. Most of the existing approaches create panoramic images in an offline process [Szelisk 1997]. These methods typically use SIFT [Lowe 2004] or similar descriptors to match image features. Szeliski [Szelisk 2006] gives a good overview of the many existing techniques for image alignment and stitching. In contrast to these offline approaches, Adams et al. [Adams 2008] align camera images in real time on mobile phones in their View Finder Alignment work. Consecutive camera images are roughly aligned by calculating a histogram of gradients for four 2D directions. The alignment is then refined using feature points. Successively, tracking of the optical flow is used to create a panoramic image. While this approach works in real time on current mobile phones, it neither permits the creation of a closed 360° panorama, nor does it track the 3D motion of the phone. Baudisch et al. [Baudisch 2005] created a real-time preview for panoramic imaging based on the work presented in [Steedly 2005]. Their application stitches low quality panoramas on the fly and thereby provides an estimate of the maximum rectangular cropping area that can be created from the set of images taken so far. The visualization of the already captured panorama is similar to the visualization used by our system. Envisor [DiVerd 2008] tracks the orientation of a camera in real time and an environment map is created on the fly. This is achieved by calculating the optical flow between successive frames. Similar to [Adams 2008], the optical flow measurements are refined with computationally expensive landmark tracking to avoid the drift introduced by frame-to-frame feature matching. While the results of this approach are similar to ours, the system cannot run on phones due to the high computational cost of the method, which requires extensive GPU processing to run in real time. Montiel and Davison created a visual compass [Montiel 2006] based on single-camera SLAM [Davison 2003]. They used an extended-Kalman-filter formulation of the tracking problem to compute orientation from dynamically acquired landmark features. Since their approach creates a sparse 3D reconstruction of the environment, the system is not restricted to rotations only. Klein and Murray also introduced another successful approach of SLAM-based tracking for augmented reality running on a mobile phone [Klein 2009]. However, due to low processing power of mobile phones Klein's SLAM system is limited to a few hundred keypoints whereas our method can handle 1000s of keypoints and is several times faster on a similar device.

The related work discussed above either does not run in real time on current phones due to high computational costs, or it solves only one task between panorama creation and panorama tracking. Our approach combines panoramic mapping and orientation tracking, both working on the same data set. It can therefore be used for creating panoramic content as well as for browsing and augmenting previously created panoramic images.

A. Adams, N. Gelfand, and K. Pulli, "View Finder Alignment," Computer Graphics Forum (Proceedings of Eurographics) 27, 2008, pp. 597-606.

P. Baudisch, D. Tan, D. Steedly, E. Rudolph, M. Uyttendaele, C. Pal, and R. Szeliski, "Panoramic viewfinder: providing a real-time preview to help users avoid flaws in panoramic pictures," In Proceedings of the 17th Australia conference on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future, 2005, pp. 1-10.

A. Davison, "Real-time simultaneous localisation and mapping with a single camera," Proceedings of the Ninth IEEE International Conference on Computer Vision, IEEE, 2003, pp. 1403-1410.

S. DiVerdi, J. Wither, T. Höllerer, Envisor: Online Environment Map Construction for Mixed Reality. In Proc. of IEEE VR 2008, pp. 19-26.

G. Klein, D. Murray, Parallel Tracking and Mapping on a Camera Phone, In Proceedings of ISMAR'09, 2009, pp. 83-86.

D.G. Lowe, Distinctive image features from scale-invariant keypoints, In International Journal of Computer Vision, Vol.60, Nr. 2, pp. 91-110, 2004.

J. Montiel and A. Davison, "A visual compass based on SLAM," Proceedings of the 2006 IEEE International Conference on Robotics and Automation, ICRA 2006., IEEE, 2006, pp. 1917-1922

D. Steedly, C. Pal, and R. Szeliski, "Efficiently Registering Video into Panoramic Mosaics," Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, IEEE, 2005, pp. 1300-1307.

R. Szeliski and H. Shum, "Creating full view panoramic image mosaics and environment maps," Proc. of the 24th annual conference on Computer graphics and interactive techniques (Siggraph 1997), 1997, pp. 251 - 258.

R. Szeliski, "Image alignment and stitching: a tutorial," Foundations and Trends in Computer Graphics and Vision, vol. 2, 2006, pp. 1 - 104

10.3.2 Evaluation

The visual tracker works at a camera resolution of 320x240 pixels. The map is created at a resolution of 2048x512 pixels. For a typical field of view of 60°, the camera resolution is therefore close to the map's resolution: 320 pixels / 60° · 360° = 1920 pixels. The theoretical angular resolution of the map is therefore 360° / 2048 pixels = 0.176 degrees per pixel. In practice, mapping errors accumulate, especially along the rotation around the vertical axis (yaw). The mapping error was measured by creating several full horizontal loops, which resulted in offsets of 27-40 pixels horizontally and 3-7 pixels vertically. These offsets represent errors of 4.74°-7.03° horizontally and 0.44°-1.04° vertically. Executing the loop closing mechanism described in the paper removes this error by applying a linear correction term along the whole map.

10.3.3 Specification

Hardware and OS	Advanced Mobile Phones
Software	Studierstube ES
Core Features	Panorama Tracking and Annotation
Status	Stable Prototype
Intended users	Environmental taggers
Showcases	WP9
Relevance beyond project	Yes

10.3.4 Publications

Wagner, D., Mulloni, A., Langlotz, A., Schmalstieg, D. (2010). *Real-time Panoramic Mapping and Tracking on Mobile Phones*. To appear in Proceedings of IEEE Virtual Reality Conference 2010 (VR´09). Scouting

10.4 In Situ Content Creation on the Mobile Phone

The system tracks the mobile device in a large environment based on natural feature tracking whereas the tracking data is provided by a database on a remote server. Our system offers a set of simple, yet powerful modeling functions for in-place content creation. The resulting AR scene can be shared with other users using a content server or kept in a private inventory for later use.

Existing authoring solutions address a specific user group. Most of them target professional content creators and media artists. But the target audiences for the in-situ authoring are people who have never used modeling programs, but can handle their phones (taking pictures, sending short messages).

Figure 24: In-situ created objects: Left: Virtual objects textured with predefined textures. Right: Virtual objects textured with textures out of the real world

Therefore we limited the functionality to basic, yet powerful functions to create 3D content. One of these functions is extrusion. The user can create 2D-Objects (polygons, circles, freehand drawings) and later convert them into 3D objects by extruding the ground plane. Hence the supported 3d models range from cubes, tubes and spheres to objects with polygonal ground plane. After defining the geometry the system allows to assign different colors or textures. The user can select from a set of predefined textures, which can be mapped to the object. Furthermore it is possible to create own textures by using textures extracted from the camera image, which can later be assigned to the objects (see Figure 24). This is especially useful if the goal is to create virtual duplicates of real objects. In addition to the creation of 3D objects the system also supports a variety of 2D operations to draw or annotate the environment. Similar to basic paint programs the user can choose between pencils, brushes or a Graffiti spray tool. The creation and manipulation of the content is done directly on the mobile device. Since more and more mobile phones ship with no physical keyboard, all manipulations and authoring functions are controlled using a touch screen.

Furthermore we create an XML-based markup file to describe the content and add Meta information. We developed our own XML language schema to express all relevant information and call *it Augmented Reality Markup Language* (ARML). The resulting markup file is also placed in the container and later is used for indexing the content and expressing the relations. Beside the possibility of sharing the generated content with other users over a remote content server, each user has a private inventory. This allows placing created objects into the local inventory on the device for later use. Hence the user is able to create an object (e.g. annotated with a texture of the current environment), pick it up and insert it into a scene at a different location.

10.4.1 Related Work

Although research of content authoring has a long tradition in augmented reality, only little investigations have been done in the field of content authoring using mobile devices. In general, we can distinguish cross-platform authoring (at the desktop) and in-situ authoring. Only few of the latter exist. One example approach that allows the user to create AR applications in place was presented in [Lee 2004]. The designer can thereby interact with the virtual world by using a marker-based tangible interface.

Piekarski and Thomas described an architecture for supporting mobile augmented reality environments [Piekarski 2001]. This architecture enables the user to develop object-oriented applications that perform a variety of complex tasks, such as user interaction techniques and in-situ 3D modeling. OutlinAR **Fehler! Verweisquelle konnte nicht gefunden werden.** presents an approach for in-situ modeling. With OutlinAR the user is able to build wireframes models by using a handheld camera mouse, which is plugged into a standard computer.

There are a dozen of desktop-based authoring solutions, which extend existing 3D modelers like Maya and Blender. DART **Fehler! Verweisquelle konnte nicht gefunden werden.** for example, is one of the best-known authoring toolkits for AR. It comes as a plugin for Macromedia Director targeting professional content producers. However, none of the above solutions has been designed with a truly large user base of thousands or ten thousands users in mind. This is particularly important, as in the spirit of Web 2.0 every consumer of AR content is potentially also an author. Thus they are not suitable tools for user-centered content creation.

P. Bunnun and W.W. Mayol-Cuevas, "OutlinAR: an assisted interactive model building system with reduced computational effort," 2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, Ieee, 2008, pp. 61-64.

G.A. Lee, C. Nelles, M. Billinghurst, and G.J. Kim, "Immersive Authoring of Tangible Augmented Reality Applications," Symposium on Mixed and Augmented Reality, 2004, pp. 172 - 181.

B. MacIntyre, M. Gandy, S. Dow, and J.D. Bolter, "DART: a toolkit for rapid design exploration of augmented reality experiences," Symposium on User Interface Software and Technology, 2004.

W. Piekarski and B.H. Thomas, "Tinmith-evo5 -- An Architecture for Supporting Mobile Augmented Reality Environments," IEEE and ACM International Symposium on Augmented Reality (ISAR'01), 2001, p. 177.

10.4.2 Specification

Hardware and OS	Mobile Phones
Software	Studierstube ES
Core Features	In-situ content creation, authoring
Status	Stable prototype
Intended users	any
Showcases	WP9
Relevance beyond project	yes

10.4.3 Publications

Tobias Langlotz, Stefan Mooslechner, Dieter Schmalstieg, “In-Situ Content Creation for Mobile Augmented Reality”, In Proc. ISMAR09, ACM, 2009.

10.5 Studierstube ES

Studierstube ES is based on over 10 years of AR software development outside of IPCity and offers the richest feature set and highest performance for developing commercial and research applications exploiting mobile AR techniques and is distributed by IMAG. It addresses graphics, video, tracking, multimedia, persistent storage, multi-user synchronization and application authoring tools. A PC-based emulator eases application development. Studierstube ES (Figure 26) can be used to develop standalone as well as multi-user applications requiring client-server communication.

The framework is existing for Symbian 3rd Edition, Windows CE, Windows Mobile 6, iPhone OS 3.0 and Windows XP/Vista/7, targeting small form factor devices ranging from smartphones to ultra-mobile PCs (Figure 25). Experimental versions also exist for Android.

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 15-60 fps, depending on the content and device.

Figure 25: 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, Studierstube NFT Tracker, a natural feature 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.

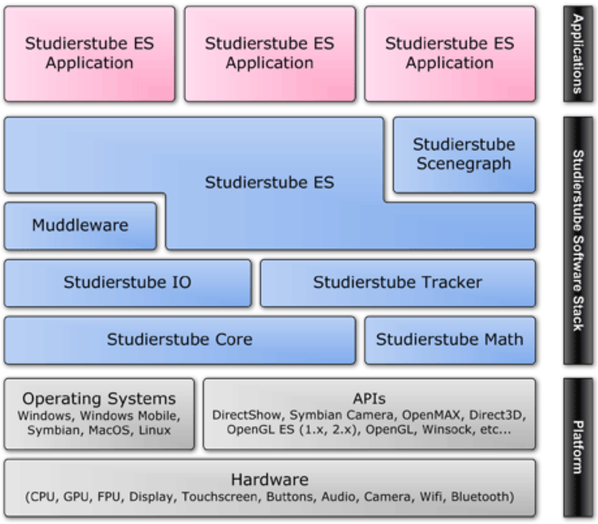


Figure 26: Studierstube ES Software architecture

Large Scale Localization/Detection

Large scale localization/detection from a camera image on a large scale is an approach to include building a dense 3D reconstruction for model based tracking. Panorama tracking with the extraction of natural features on the mobile phone overcomes the long roundtrip time approach to send an image to a server and get back an absolute position plus data for tracking from natural features on the client (Arth09).

It provides rather an in-situ creation of AR annotation of the environment directly on the mobile phone. Previous authoring or annotation tools were mostly bound to desktop computers, or could operate only at the accuracy of the employed mobile sensors. The new tracking approach going beyond the scope of IPCity developments allows creating annotations in place and storing them in a self-descriptive way on a server based on the City Tales system, in order to allow a later re-identification. GPS information is used for efficient indexing, but the label positions are identified using template matching against a panoramic map – which is created on-the-fly, compare chapter 10.3. This approach yields accurate and robust registration of annotations with the environment, even if seen from a slightly different position compared to where the annotation was created! The system can be used in large-scale indoor and outdoor scenarios and offers an accurate mapping of the annotations to physical objects.

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 Symposium on Mixed and Augmented Reality 2008 (Wagner08).

Further outcome of TUG research on this subject is the panorama tracking approach described in chapter 10.3. and the integration of natural feature.

10.5.1 Related Work

A few other projects dealing with Augmented Reality on mobile phones or PDAs have been reported in the 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.

Gausemeier, J., Fruend, 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

Henrysson, A., Billinghurst, M., Ollila, M. Face to Face Collaborative AR on Mobile Phones. International Symposium on Augmented and Mixed Reality (ISMAR'05), pp. 80-89, 2005

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

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

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

Michael Rohs Marker-Based Embodied Interaction for Handheld Augmented Reality Games Journal of Virtual Reality and Broadcasting (JVRB), Vol. 4, No. 5, March 2007

Shibata, F., Mobile Computing Laboratory, Department of Computer Science, Ritsumeikan University, Japan, http://www.mclab.ics.ritsumei.ac.jp/research.html

10.5.2 Testing and Public Demonstration

Based on the framework of Studierstube ES several mobile client applications have been developed within IPCity in work package WP9. The MR-Player described in detail in the Deliverable D9.3 and D9.4 and the Wall Blogging client described in Deliverable D9.3.

City Tales gaming and story-telling scenarios make extensive use of the Studierstube ES framework by basing the client application development onto this platform. The scenario of being connected to a distributed MR system calls for highly transportable applications as participants are possibly using different flavors of OS on their mobile devices.

MR-Player as one of the applications based on the Studierstube ES platform was presented at several testing and dissemination activities of IPCity, like the FET'09 Conference 'Science beyond Fiction' in Prague, the IPCity Summer School 2009 in Vienna, MIRACLE Workshop 2009 in St. Augustin and a number of other presentations to academic and commercial world.

10.5.3 Specification

Hardware and OS	Advanced Mobile devices and PC based Symbian 3rd Edition, Windows CE, Windows Mobile 6, iPhone OS 3.0 Microsoft Windows XP/Vista/7
Software	C++
Core Features	Mobile MR Framework: • Highly configurable to optimally use available hardware • Highly optimized fixed-point math classes • Small memory footprint (~500KB minimum for a useable system) • Support for OpenGL ES 1.x, OpenGL ES 2.x, OpenGL, and Direct3D Mobile • Dedicated render paths for software and hardware 3D accelerated devices • Lightweight XML-based scene graph for rapid application development • Integration of Studierstube Tracker • Content protection, binds Studierstube ES to a specific device and protects against reading or modifying the content outside the application
Status	Commercial product
Intended users	Academic and commercial application developers
Showcases	WP7, WP9
Relevance beyond project	yes

10.5.4 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. Arth Clemens, Wagner Daniel, Klopschitz Manfred, Irschara Arnold , Schmalstieg Dieter, *Towards Wide Area Localization on Mobile Phones*, Proceedings of Int. Symposium on Mixed and Augmented Reality 2009 (ISMAR´09), IEEE, October 2009

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:

ipcity.eu