MASTER-/DIPLOMARBEIT

Augmented Reality on Mobile Devices for Architectural Visualisation

ausgeführt zum Zwecke der Erlangung des akademischen Grades

eines Diplom-Ingenieurs / Diplom-Ingenieurin

unter der Leitung von

Ass.Prof. Dipl.-Ing. Dr.techn. Peter Ferschin

E259.1

Institut für Architekturwissenschaften

Abt. Für Digitale Architektur und Raumplanung

eingereicht an der Technischen Universität Wien

Fakultät für Architektur und Raumplanung

von

Stefan Niedermair, BSc

0403011

Am Steinbruch 15,

A-4048 Puchenau

Wien, am 04.10.2012

Abstract

During the last few years Augmented Reality (AR) applications have seen an enormous growth in different fields. These include advertising, project planning as well as complex information providing platforms. One area however that could profit greatly from using Augmented Reality is architectural visualisation.

The aim of this work is to show how and in what areas Augmented Reality can improve the presentation of information about an architectural project or content. First an introduction on the historical development and currently available AR applications is given. Following that the technical background and the relevant theoretical knowledge necessary to understand and build an AR system on mobile devices are presented. Special attention is thereby given to the differentiation between sensor based and computer vision-based AR, the two generally applicable methods for handheld AR. After a short introduction to the three use cases the development framework used to build the test applications for tablet computers is explained in further detail.

The evaluation of feasibility is conducted based on three real-world usage scenarios in the architectural domain. Following a demonstration of each example, challenges encountered during the implementation process are discussed. The chosen scenarios are:

- Table-top project presentation: Computer vision-based AR is used to replace a physical model with a digital, interactive pendant. An overview plan or another visual representation of the site can be used as a reference for positioning the digital model.
- On-site visualisation in context: The virtual model of a new or unrealized project is shown at the proposed site. While sensor based AR is used to guide the user to the site, computer vision-based AR is used to place the model precisely between the two existing buildings or to show modifications to an existing one. The facades of these buildings deliver the spatial reference.
- Urban cultural heritage information system: Content like pictures, videos and 3d models of destroyed buildings and other findings from an otherwise not easily accessible urban archaeological site is virtually positioned in its place using AR. Computer Vision and sensor based AR approaches are thereby used complementary to substitute each other's shortcomings in an inner urban environment.

While AR is currently not part of a typical architectural workflow these examples show to what extent it could be already integrated and how the design team as well as clients could profit from using this technology in the future.

Content

1. Introduction

The clear communication of an idea is indispensable in a multidisciplinary project, such as an architectural project. Over the time, visualisation has proven to be the most accurate way to accomplish that hence the importance of visualisation tools in the architectural design process. Traditionally these visualisations included three different forms of representations - physical scale models, orthographic plans - floor plan, section or frontal view - and perspective drawings. Physical models can thereby be traced back approximately 2000 BC in Egypt, and reached their peak in the 15th century where they became an essential part of the architectural design process. (Stalla 2008) The modern repertoire of visualisation methods was completed with Brunelleschi's development of a method to create perspective drawings in the early 15th century. This finally allowed more realistic representations of future buildings and brought the possibility to visualise them in their already built context.

The development of the computer and its introduction to the architectural workflow lead to a number of new possibilities and improvements for architects. The previously hand drawn orthographic plans were gradually replaced by CAD software which not only made modifications but also the reproduction of plans faster and easier.

Another important step was taken with the development of 3D modelling software. The use of such a tool offered a way to build a virtual three dimensional model of a design. This brought advantages in different areas of the design process. On the one hand visualisation was no longer limited to one chosen perspective that had to be drawn by hand. Based on a 3D model any desired number of image compositions could easily be generated. On the other hand virtual models became an essential part in the actual design process. Unlike a physical model, changes to such a virtual representation were easily reversible and intermediate steps and design alternatives could be saved at any time. Due to these advantages virtual models widely replaced their physical counterparts.

The missing connection between 2D CAD and 3D modelling programs was filled by so called building information modelling (BIM) software. These programs unified the production of 2D and 3D content in one process and offer the basis for most of nowadays architectural projects.

With all the advantages that the introduction of virtual tools brought, it also led to a decrease of connection to the real world on different levels. The first level is usability. Moving around a physical model to change the current view is easy and does not require any additional knowledge. A virtual model is undeniably a very useful tool for designers as it is easy to modify, adjust, save and recall but at the same time navigation in such a 3D environment is not trivial and a simple motion might be a difficult navigation task requiring rotations around multiple axis and several position changes. As there is no commonly suitable user interface (UI) for complex 3D navigation, such tasks cannot be expected from an inexperienced user. While machine-made physical representations of a virtual model can be produced in a process called rapid prototyping, the result is just a snapshot of one state of the model and does not keep any of the advantages of its virtual pendant. In short terms the possibilities that a virtual model offers seem to come hand in hand with the loss of the simplicity physical interaction offers and vice versa.
The second level of this decrease of connection relates to the context the model is shown in. 3D software makes it possible to generate almost photorealistic images of the future building from all desired perspectives. This might be sufficient to visualise a building by itself but in most cases it is going to be added to an assembly of existing buildings. To judge on its quality it has to be visualised in its future environment. Nowadays this is done in two ways. Either an abstracted version of the real environment is recreated in the 3D software or a rendered image of the virtual content is overlaid over a photo of the real scene. While in the first case the advantages of the virtual environment remain, the quality of the represented real content is very limited. In the second case a proper composition of virtual and real content can be made but the virtual aspects are lost again. Therefore an ideal platform for architectural visualisation has to establish a connection between virtual and real content on those two levels and unite the advantages of both worlds.

A technology with the potential to achieve that is Augmented Reality (AR). Augmented Reality in simple terms is the computer supported dynamic composition of virtual content and reality. [...], AR supplements reality, rather than completely replacing it. Ideally, it would appear to the user that the virtual and real objects coexisted in the same space, [...]. (Azuma 1997) The idea behind goes back to the late 1960's (Sutherland 1968) when a high performance computer and technical devices were necessary for such systems. The great development that smart phones are currently seeing in computational as well as in graphical performance combined with a wide range of built in sensors let them become an ideal platform for Augmented Reality.

The goal of this thesis is to prove that Augmented Reality on such mobile devices can indeed build this bridge between the virtual 3D model and reality by demonstrating its usability in different architecture related scenarios.
After a general introduction to the field of AR, the historical background relevant for this work is presented. Following that, existing AR applications are shown and explained. In chapter 2 the technological background, necessary to build an AR application on mobile devices, is established. In the third chapter the scenarios that are created to demonstrate the capabilities of such a system are introduced. In the first scenario AR is used as part of an advanced user interface for an interactive visualisation. The second and third scenarios demonstrate the possibilities for visualisation in context under different circumstances. Chapter four is dedicated to the implementation process. After the elements used to build the platform are described, the single applications are demonstrated and their problems discussed.

1.1. Overview on the development of Augmented Reality

As mentioned before Augmented Reality is the combination of virtual and real content and is defined as a sub category of Mixed Reality (MR). MR itself is also a sub category in the so called virtuality continuum that goes from the real environment without any virtual elements to the complete virtual environment and involves the merging of real and virtual worlds. Augmented Reality therefore is part of this and refers to all cases in which the display of an otherwise real environment is augmented by means of virtual (computer graphic) objects. (Milgram & Kishino 1994)

Fig. 1.1 Simplified representation of a "virtuality continuum" (based on Milgram & Kishino 1994)

This means that an Augmented Reality view consists of two basic parts:

- View of the real world
  This could be done either directly through a partially transmissive combiner display or through a live camera image on a monitor. These two techniques are also referred to as optical and video approach. (Azuma 1997)
- Computer generated image
  The virtual part of the augmented view is an overlay to the real world image. Its position in the view has to be calculated in the background according to the current position of the viewer.

To go one step further Azuma defines AR system by having the following three characteristics (Azuma 1997):

1) Combines real and virtual
2) Interactive in real time
3) Registered in 3-D

Until now this seems to be the most complete and universal definition of Augmented Reality. For that reason this is the definition that will be referred to in this thesis. The technical necessities to actually build an AR system will be described in chapter 2 and the above mentioned characteristics will be explained further.

1.1.1. Historical Development

The development of Augmented Reality systems has made considerable progress since its beginnings. Therefore the following chapter will not show the complete historical development of AR but focus on mobile AR and try to give an overview by selecting projects relevant for this thesis. A more complete list of developments can be found at (@ICG TU Graz) which was prepared by the members of the Christian Doppler Laboratory for Handheld Augmented Reality for the ISMAR society.

First approaches to generate AR go back to Ivan Sutherland when he in 1968 presented his head-mounted three-dimensional display which was also referred to as “The Sword of Damocles” as a reference to the Greek legend of Damocles. The display and the connected system to measure the head position with six degrees of freedom (DOF) were too heavy and bulky to be simply mounted to the users head as nowadays systems are. Therefore it had to be mounted to the ceiling of a room in a height adjustable way with the users head tied to it as shown in Fig. 1.2. The user was then able to move his head within limitations and the position in 3d space was measured by the sensors. A computer then used this coordinates to calculate the correct view of a simple wireframe model according to the users head position in real time. This wireframe view was then shown to the user via a binocular head-mounted display (HMD). (Sutherland 1968)

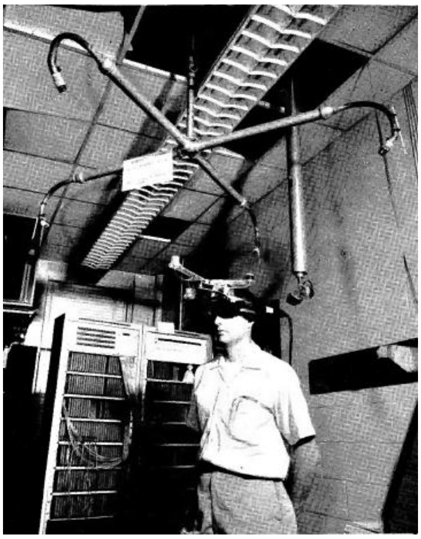

Fig. 1.2 Head-mounted display with mechanical head position sensor (Sutherland 1968, p.760)

During the late 1970s and 1980s the biggest area of development for AR related systems were head-mounted or helmet mounted displays for combat aircraft. The main tasks were on the one hand to make information about the current aircraft performance and targeting status visible to the pilot and on the other hand to use the pilots head angle as pointing device to direct the onboard weapons or target seeking systems. (@Wiki_HMD)

Fig. 1.3 Integrated Helmet and Display Sighting System (US Army 1982)

The first time the term Augmented Reality became really popular was in 1992 when Thomas Caudell and David Mizell introduced it for a project they were working on at Boeing. They created a prototype of a heads-up, see-through, head-mounted display with head position sensing and a real world registration system. This system should enable workers that are assembling an airplane to see diagrams and assembly instructions necessary for their current task superimposed and stabilized on a specific position on a real-world object. The main advantages of using such a system were thereby seen in the elimination of sources of error and additional expenses for templates, formboards or other masking devices. (Caudell & Mizell 1992)

Fig. 1.4 Boeing's prototype wire bundle assembly application. (Azuma 1997, p.360)

Following that, the most recognisable developments in AR in the mid and late 1990's have been made using so called wearable computers. The Touring Machine (Feiner et al. 1997) is considered to be the first mobile augmented reality system (MARS). After that a project called Tinmith was developed at the Wearable Computer Lab at the School of Computer and Information Science, University of South Australia and lead to a range of research projects and applications. Originally it started in 1998 as a project called "map-in-the-hat" and used a see-through head-mounted display, a differential GPS (dGPS) and a digital compass to help a user fulfil outdoor orienteering tasks in unknown terrain based on predefined waypoints.

Fig. 1.5 Map-in-the-hat; Tinmith-I prototype (1998)

Later on it was further developed under the name Tinmith which stands for “This Is Not Map In The Hat”. Tinmith was continued until 2006 and become one of the most powerful MARS at that time. One project based on Tinmith that got a lot of attention also outside the area of AR related researchers was the implementation of the ego shooter Quake into ARQuake in 2000 which combined tracking based on GPS, digital compass, and fiducial vision-based methods for an outdoor/indoor AR game (B. Thomas et al. 2000) (@ARQuake).

Fig. 1.6 ARQuake implementation on Tinmith-Endeavour (2002)

The final configuration of Tinmith in 2006 included a differential GPS with 50cm accuracy, wireless networking, Bluetooth for wireless peripherals, wireless video output as well as batteries for 3 hours of operation, wireless gloves and input devices and a specially designed helmet (@Tinmith).

Fig. 1.7 Final configuration of the Tinmith backpack (2006)

In 1999 H. Kato and M. Billinghurst presented their method for marker tracking for a video-based AR conferencing system. Their initial approach was to create a virtual whiteboard where users can collaboratively view and interact with virtual objects (Kato & Billinghurst 1999). One year later they expanded their idea by trying to create a table-top system that supports face-to-face collaboration to examine and work on an architectural design project and allows users to manipulate virtual objects in a natural and intuitive manner (Kato et al. 2000). Beside their research the contribution that had most impact on the general recognition and acceptance of AR was the development of ARToolKit. ARToolKit a software library for building real time AR applications using squared markers and a template-based approach for recognition and was made available freely for non-commercial use. This made it possible for people interested in the manner to experiment with AR without the necessity to know all the in-depth background and mathematics involved. Until today it is still available under the under the GNU General Public License and is very popular in the AR community. (@ARToolKit)

In 2003 Wagner and Schmalstieg created the first stand-alone AR system with self tracking running on an unmodified personal digital assistant (PDA). This has been done by implementing ARToolKit as most setups that included a HMD have been designed as a proof-of-concept and do not provide a useable form factor as a PDA would. (Wagner & Schmalstieg 2003)

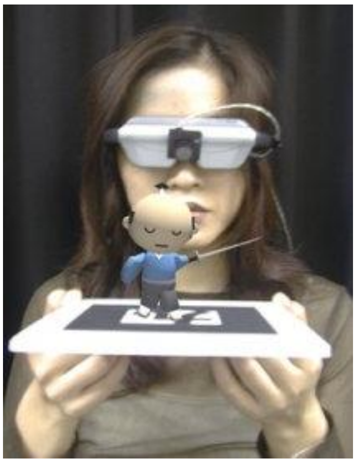

Fig. 1.8   Virtual object on a squared marker using ARToolKit

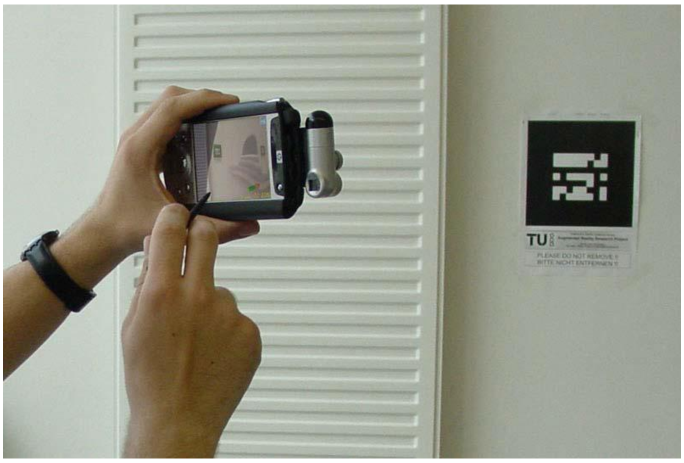

Fig. 1.9 ARToolKit Implementation running on PDA (Wagner & Schmalstieg 2003)

An interesting AR system for cultural heritage scenes was introduced under the name Archeoguide and is built around the historical site of Olympia, Greece. Using a combination of position detection based on differential GPS / compass and image recognition methods the main task is to provide location specific information about the site to the viewer on different devices like wearable computer, a tablet PC and a palmtop and provide different levels of detail on them. While the position calculation is done on the devices - dGPS / compass + image recognition on the wearable computer or dGPS /compass on the tablet and GPS on the palmtop – the information is provided by a site information server administering a multimedia object database storing 2D images, 3D models, audio and video clips and text objects on the archaeological site. (Vlahakis et al. 2002)

Fig. 1.10 Archeoguide: Video of ruins (left) and augmented temple rendered on top of live video (right) (Vlahakis et al. 2002, p.57)

The first serious considerations of implementing Virtual Reality and Augmented Reality into the architectural planning and decision making process have been taken by Drosdol et al. in 2003. During the design phase of the brand architecture centre for DaimlerChrysler in Stuttgart, Germany Virtual Reality (VR) and AR were used to support the design of car selling points in large European cities. While VR was used since the beginning of the project to work as a platform for communication for all professionals involved in order to gain some security in the decision making process, AR markers were used to arrange virtual objects in a CAVE for e.g. exhibition compositing in real scale. (Drosdol et al. 2003) While their experiences with VR / AR in the planning process have been very interesting, the integration of these technologies on such a level is very unlikely to become a standard in architecture as the costs and technical necessities are still very high.

A very interesting project dealing with the issues of inner urban AR was conducted by G. Reitmayr and T.W. Drummond in 2006. In their work titled "Going out: Robust Model-based Tracking for Outdoor Augmented Reality" they demonstrate how a textured low poly model of the facades in the surrounding of the user could be used as a reference for a vision-based AR system inside a city. Furthermore they showed how computer vision-based AR in such a system can be supported by sensors like compass and gyroscope in order to overcome some of the problems typically showing up when working in an such an environment. The sensors were thereby used to increase the performance of the system during fast motion and to avoid drifting. This allowed them to achieve very appealing results at sufficient frame rates (about 15 FPS) on a mobile device. (Reitmayr & Drummond 2006)

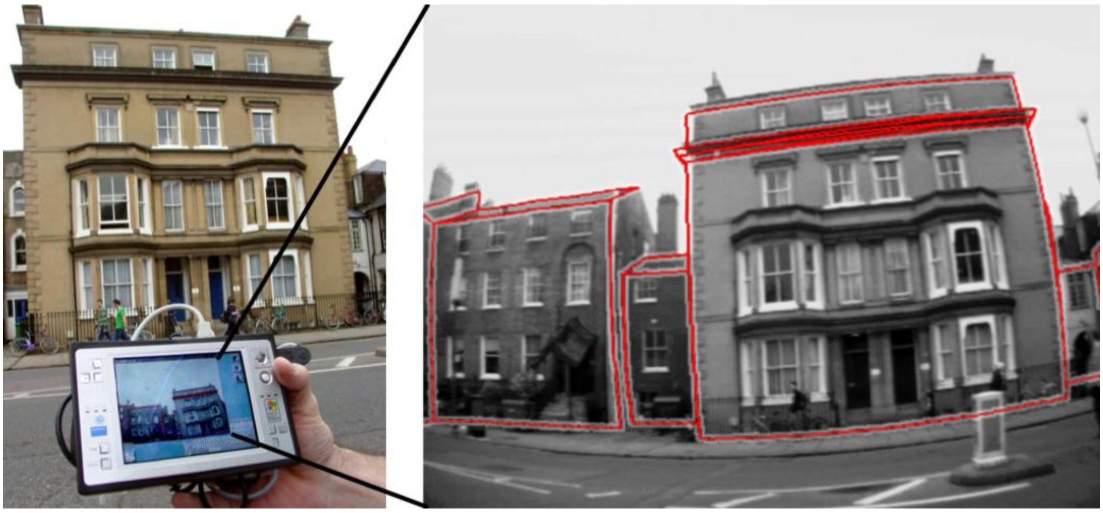

Fig. 1.11 Going Out: Buildings used as reference for AR (Reitmayr & Drummond 2006, p.109)

Alongside the above mentioned projects contributions and connections between AR and the architectural planning process in general have been made by (Belcher et al. 1997), (Bruce Thomas et al. 2000) and (Phan & Choo 2010).

1.1.2. Applications

Lead by the extensive research done in that area and the wide availability of smart phones that fulfil all the technical necessities for AR, since about 2008 the amount of Augmented Reality applications (apps) running on those devices and the interest in them has been growing constantly. The variety of functionalities goes from systems showing different kinds of location based information over apps providing marketing material to AR based games. This chapter will give an overview about the current state of available apps and the different ideas behind them.

Wikitude

One of the first AR apps that found wide acceptance is Wikitude. Wikitude is a so called AR browser. Similar to a web browser it is not creating content itself but rather using content provided by others or providing others the possibility to create content for it. The content thereby still belongs to the provider – in Wikitude called "Worlds" – and is just displayed in the app, not part of it. In 2008 the initial idea was to combine GPS and compass data with information found in Wikipedia entries. This made Wikipedia the first "world" in Wikitude and quite likely also influenced its name. The information found was then overlaid on top of the live camera view and represented by an icon (tag). Fig. 1.12 shows an example of that and Fig. 1.13 is a screenshot of the related Wikipedia article containing the necessary geographical information in form of GPS coordinates.

Fig. 1.12 Wikitude showing info tag on the Statue of Liberty

Right now Wikitude is available on Android, Blackberry, iOS, Symbian and Windows Phone operating systems and offers location based information for millions of places in different categories like events, tweets, Wikipedia articles, ATMs, restaurants, user reviews and much more. The worlds or sources of information can be divided in mainly two categories. On the one hand platforms like Wikipedia, Foursquare, Qype, Panoramio, YouTube, Flickr and so on, that already contain user created content, are searched for location specific data. On the other hand developers can create their own worlds filled with information about a certain topic. That way worlds containing locations of police stations, McDonalds restaurants or tourist guides for different cities have been added to the platform. While initially the user was able to access information from all worlds for free in 2011 Wikitude also added the possibility for developers to offer paid layers with premium content to the users. (@Wikitude)

Fig. 1.13 Wikipedia article about the Statue of Liberty – Geographical location highlighted

Layar

Layar is another Augmented Reality browser similar to Wikitude. Even though the main ideas of displaying different kind of information, separated in so called layers (equivalent to Wikitude's worlds), on top of the camera view is the same, there are numerous differences between the two platforms. Layar also offers location based information gathered from the community based, freely available platforms like Wikipedia, Flickr, Foursquare and many more. In addition to those and other custom created layers that show information tags like Wikitude, Layar also allows other kinds of layers with more advanced content. Enabling the developers to implement much richer content like 3D objects and animations in the augmentations these layers for example let users play games within their environment, browse for clothes in a 360-degree virtual shop, or even view artwork placed digitally into the real world. Even though Layar supports much more kinds of content to be displayed, the technology used for positioning the user is still based on the same approach used by Wikitude, a combination of GPS, compass and accelerometer. Additionally a newly introduced feature of Layar – Layar Vision – enables the creation of layers and applications that recognize real world objects based on computer vision algorithms and display digital experiences on top of them. This means that information is not necessarily shown in a certain geographical context but can also be positioned according to any previously known object like a poster or something similar. (@Layar)

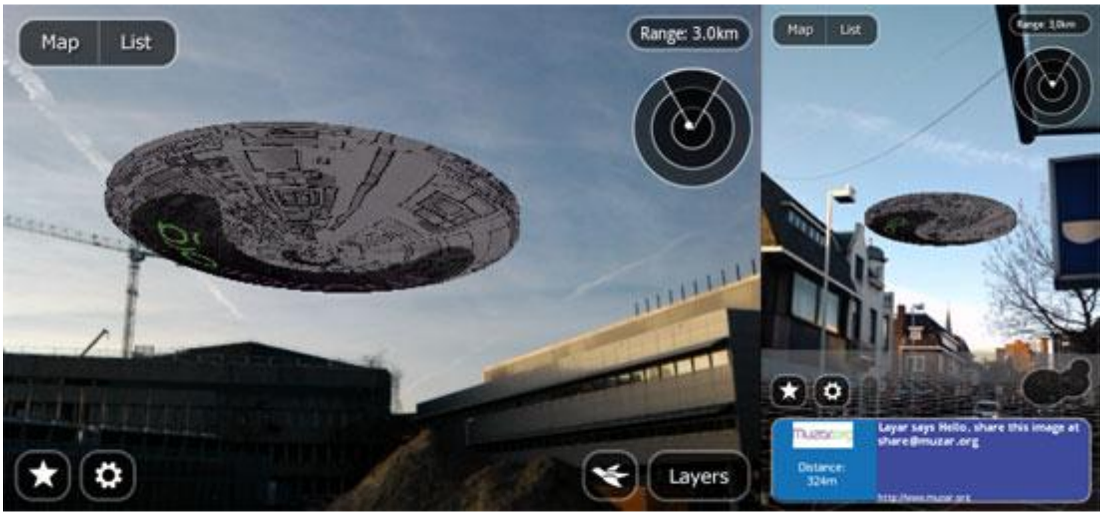

Fig. 1.14 **Layar displaying a GPS-referenced 3D model**

Fig. 1.15 **Layar vision demo**

Junaio

Junaio is a third AR browser released in 2010 that has to be mentioned. Even though it offers features similar to Wikitude and Layar, the company behind it, Metaio, is also researching in the field of AR based on image recognition. Therefore Junaio has some additional features like Barcode and QR code detection as well as recognition of 2D images, which allows developers to implement further functionality. This makes it possible to use Junaio also for completely different tasks like product visualisation. One demonstration for that was done by displaying a 3D model of a toy on top of its box as soon as the box was recognized in the camera view. Another area where Junaio has been used is exhibition guidance. There its capabilities made it possible to detect pictures or other 2D artwork for example in a museum in order to present enriched content or further information about it to the visitor using a smart phone. One more feature that differentiates Junaio from its competitors is the implementation of some indoor navigation features based on LLA-marker technology (latitude, longitude, altitude marker), compass and accelerometer information which was first demonstrated at the Kiosk Europe Expo 2010 and allows navigation in environments without GPS signal availability. (@Junaio)

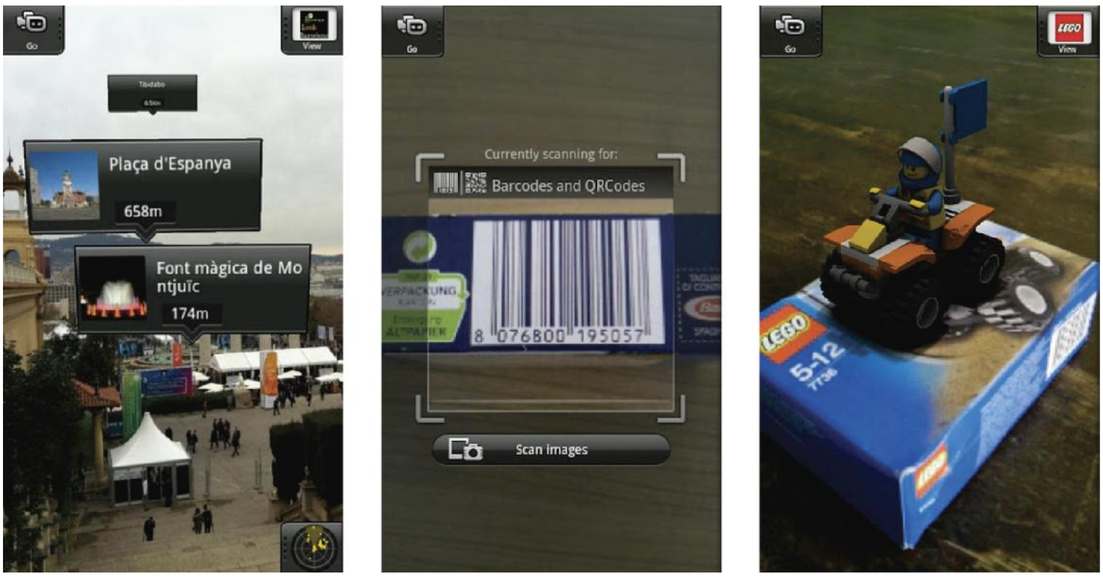

Fig. 1.16 Junaio functionality: GPS-referenced information (left), Barcode reader (middle) and image recognition (right)

3DOn ARchitecture

An application specialized in a bit different field is *3DOn ARchitecture*. While the main idea of this app is to work as a 3D viewer for architectural models built in Google SketchUp, it also offers AR functionality. By using the integrated Geo-tagging abilities of SketchUp the model can be placed at the correct geographic coordinates of the site and can then be shown on-site using AR on a supported iOS device. The current position of the viewer in that case is again calculated based on the devices GPS position, the compass and the accelerometer. Due to the impreciseness of GPS this application also offers a way to override the GPS position information by manually setting it on a map. (@3DOn)

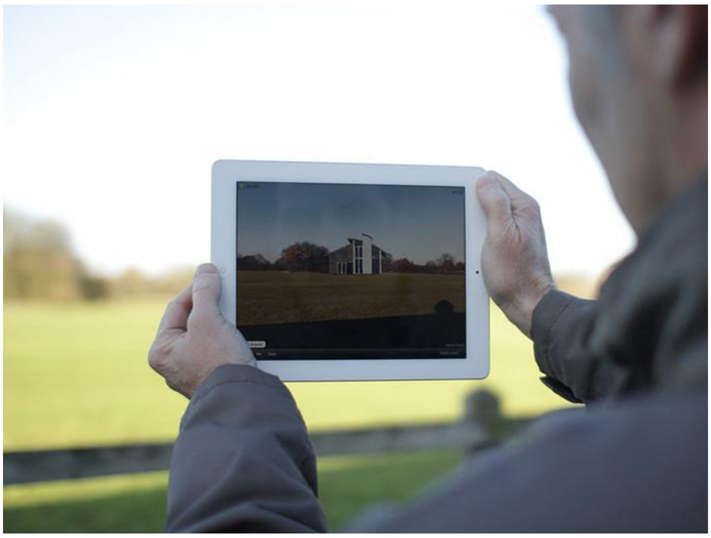

Fig. 1.17 3DOn's AR feature used to show a building on-site

Others

A further applications in the architectural field that demonstrates the possible use of AR to some extent is Magic Plan, an app that helps its user to generate floor plans (@MagicPlan). Besides these applications Augmented Reality on mobile devices is becoming more and more prominent in different areas reaching from marketing material over AR enabled magazines to games. Recent examples are the Juiced Up campaign for the Volkswagen Beatle where posters outdoor where used to trigger 3D animations and an app accompanying the 2013 IKEA catalogue where the user can see additional content when he is viewing it through the device (@IKEA).

2. Technological Background

While the previous chapter gave an overview on the development and the current state of AR, in order to build an actual AR application further technological background has to be established. Every Augmented Reality system consists of three steps that are necessary to achieve an augmentation. Those three steps in order of their appearance in such a system are 1. tracking of the viewer's current position in space, 2. computation of the correct view of the geometry that should be augmented and 3. displaying of the combination of real and virtual content based on the current position of the viewer. In order to achieve a convincing real-time AR experience, a frame rate of about 20 frames per second (FPS) is necessary, meaning that the whole process has to be completed at least 20 times per second. This chapter is going to explain these steps in more detail with respect to the capabilities and constraints of modern handheld devices.

For easier understanding parallels to the process necessary to create a still image composition will be shown. The process to create such a composition where virtual content is added to a picture of a real scene can also be split into three steps that correspond to the steps necessary for an AR system.

1) Camera pose matching

   After a picture of the scene has been taken the first necessary step is to find the matching position of the real camera in the 3D software. Thereby the position, rotation and field of view of the camera have to be adjusted accordingly. Based on this information the virtual content is displayed from the same point of view as the original camera.

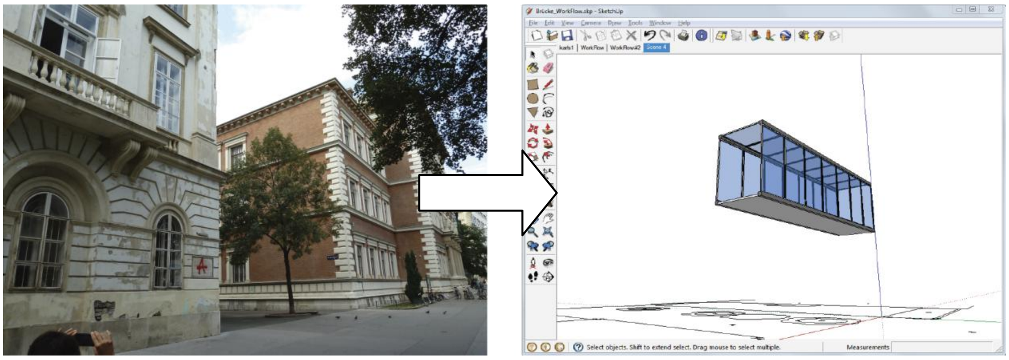

Fig. 2.1 Camera matching in 3D software

2) Rendering of the virtual content

   After the virtual camera has been set up correctly, the light setup can be modified in order to match the requirements in the image. After that the camera view can be rendered and is ready to be used further.

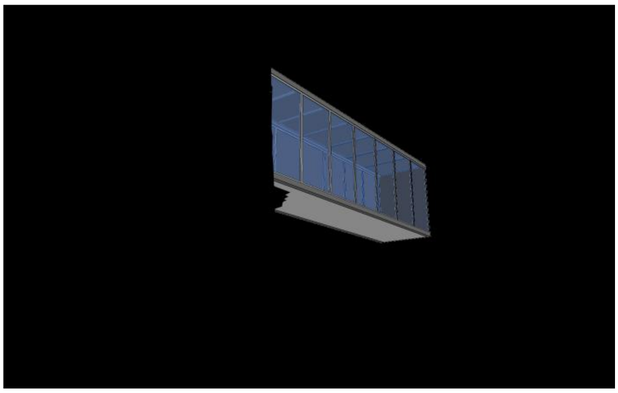

Fig. 2.2 Rendered content for overlay (black area will be masked out)

3) Merging real and virtual content

   The last step is to actually merge the virtual content that was rendered in the 3d software and to merge or overlay it to the camera image. The result is the final composition.

Fig. 2.3 Image combined from real and virtual view

2.1. Tracking of the Position

One of the basic necessities to generate AR is to find a way to track the user's current position in space. Depending on the situation there have been numerous different approaches to accomplish that. Looking back at Ivan Sutherland's head-mounted three dimensional display or the Touring Machine AR system, very bulky and expensive hardware was to be used to determine the user's position. Nowadays smart phones and tablets offer a variety of built-in sensors that can be used for position calculation and have all the necessary capabilities for AR. These available sensors are:

- Assisted Global Positioning System (aGPS-) Sensor
  Assisted GPS is an improved version of the commonly known GPS satellite-based positioning system, that additionally uses network resources to locate and utilize the satellites in poor signal conditions. In comparison to regular "standalone" GPS operation the usage of additional data provided by an aGPS server allows much faster position calculation in problematic situations like in cities. aGPS is used extensively with GPS-capable cellular phones and is a better way to determine a devices position globally as long as satellites are available. (@Wiki_aGPS) However the main problem about GPS on contemporary smart phones is the accuracy which is typically not better than 10 meters.

- Magnetometer (Compass)

  A magnetometer is a measuring device used to measure the strength or direction of a magnetic field. (@Wiki_Magnet) Like a compass it can be used to find the earth's magnetic poles. Due to the fact that the magnetic and geographic north does not match the magnetic declination, which is the angle between those two, has to be considered additionally. The magnetic declination varies from place to place and in most continental areas it has a value between 35°E and 35°W. (@Wiki_MagDecl)

- Accelerometer

  An accelerometer is an electromechanical device that can measure acceleration forces. These forces may be static, like the constant force of gravitation or they could be dynamic, caused by movement or vibration. By measuring the amount of static acceleration it is possible to find out the orientation or the angle the device is tilted at with respect to the earth. By sensing the amount of dynamic acceleration, the way the device is moving can be evaluated. (@DimensionEngineering) Most smart phones have a built-in 3-axis accelerometer in order to measure acceleration in all three axes simultaneously.

- Gyroscope

  A gyroscope is a device for measuring or maintaining orientation, based on the principles of angular momentum. It is not that widely spread among smart phones yet, but it can deliver more precise orientation information than an accelerometer if available. (@Wolfram) The main disadvantage compared to an accelerometer is, that it can only measure relative orientation changes and no absolute orientation of the device due to its lack of gravitation awareness.

- Camera

  The camera of the smart phone is essential as it delivers the background image for the augmentation and optical information about the surrounding.

While most of these sensors are available in almost all current handheld devices they can be used in two different ways to create an AR system. These two methods only differ in the way the current position of the user is evaluated and they will be explained further in the next chapters.

2.1.1. Sensor Based AR

Sensor based AR is a method where measurements from different sensors are taken to determine the current position of the user. These sensors could be of different kind. While Sutherland used a mechanical and an ultrasonic sensor to measure the current head position (Sutherland 1968), newer systems often use infrared cameras and reflective markers. While these systems offer very good results indoor, they are often not an option outside as sunlight is avoiding infrared systems from working properly. For outdoor navigation the Touring Machine already used a system based on differential GPS for position tracking and magnetometer/inclinometer for orientation tracking (Feiner et al. 1997).

No matter what kind of sensor is used, the system has to be able to measure the position and orientation of the camera. The transformation matrix consists of the positions along x-, y-, and z- axis as well as rotations around those three axes and has therefore 6 degrees of freedom (DOF) that need to be evaluated. A visual representation of this can be found in Fig. 2.4.

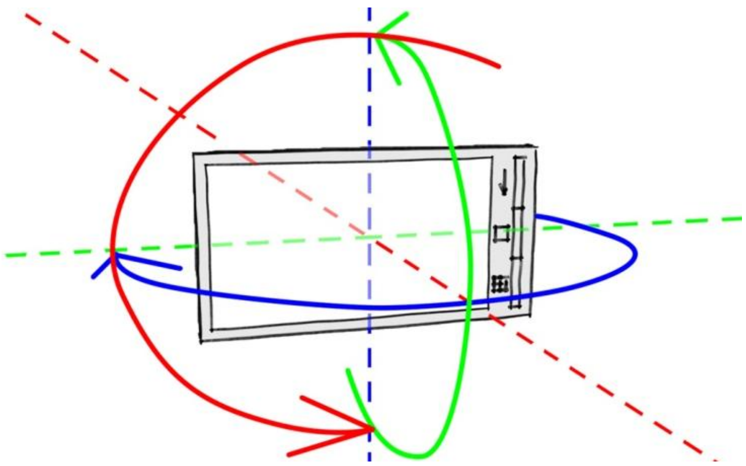

Fig. 2.4 6 DOF of the phone; x- (green), y- (blue) and z- (red) axis

On smart phones the most widely spread way to acquire the current position and orientation is by using the sensor values from GPS, compass and accelerometer.

GPS hereby is able to provide values for geographical width and length in decimal degrees as well as a height/altitude value. While width and length can be used to represent the x- and z- coordinates of the device in space, the height value can be used to represent the y- coordinate of the device. To reduce already one source of error it is possible to not use the altitude value of the GPS but to set the y-coordinate to a fix offset above the ground that approximates the height a user would typically hold such a device at and thereby to eliminate one DOF.

The second sensor used is the compass. It's capabilities to sense the earth's magnetic field are used to get horizontal orientation of the phone which is represented as the rotation around the y- axis. As already previously mentioned it is necessary to also consider the magnetic declination to get the real orientation.

The remaining two rotation values around x- and z- axis are retrieved from the accelerometer or the gyroscope if available. In order to eliminate another degree of freedom it is possible to set the rotation around the z-axis to zero thus requiring the user to always hold the device in the so called "landscape left" orientation and making it a system with just 4 DOF as it can be seen in Fig. 2.5.

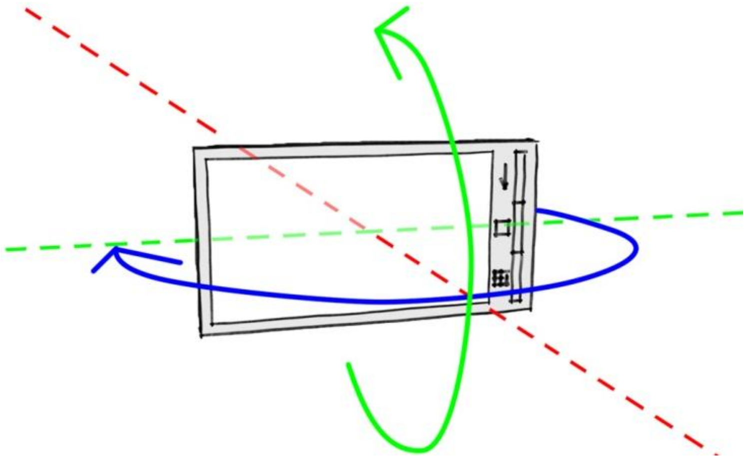

Fig. 2.5 4 DOF of the simplified system; x- (green), y- (blue) and z- (red) axis

2.1.2. Computer Vision-Based AR

Vision-based AR is a much newer approach and refers to a solution where computer vision (CV) algorithms are used to recognize known patterns in the current camera image in order to calculate the spatial relation between these markers and the camera. Currently there are mainly two ways this is used to track the actual position of the camera in a smart phone.

The first way uses special markers that are usually black and white to have as high contrast as possible. These markers are generate using predefined rules and are often built up from rectangular shapes. In addition to being a reference for pose estimation by the CV algorithms the markers can have information like a unique ID encoded to be clearly identifiable by the system. One of the first markers of that kind used for AR was a 2D matrix marker, a square shaped barcode that allowed identifying a large number (216) of objects (Rekimoto 1998). Nowadays QR-codes along with customized solutions are also used as markers for AR systems. A selection of different markers can be found in Fig. 2.6.

(a) (b) (c)

Fig. 2.6 Markers used for AR
(a) 2D matrix marker (Rekimoto 1998)
(b) QR-code
(c) Frame Marker from vuforia AR SDK

A more advanced method also relying on CV algorithms is called natural feature tracking and does not depend on artificial markers. Natural features are areas on the surface of an object that have a high level of local contrast. The idea is that a previously known object can be identified based on the pattern of these features visible in a camera image. In theory this makes it possible to use almost any planar or 3-dimensional object as reference as long as it contains a certain amount of visual information and contrast.

Fig. 2.7 Image with natural features detected (right) (© 2010 QUALCOMM Incorporated)

No matter if the system is marker based or natural feature based, once a known pattern is detected in the current camera view further algorithms are used to calculate the current camera position in relation to this pattern. By doing this all 6 DOF can be calculated directly and are then available for further steps.

2.2. Computation

Once the current position of the viewer is known by the system, the actual content that will be added to the real world view has to be calculated. To do this the acquired pose information of the camera is used in a 3D software that contains all the necessary virtual objects that should be visible in the augmentation and has a predefined spatial correlation to the real scene. At this point it does not make a difference if the pose information of the camera was recorded using a sensor based or a computer vision-based method.

It is again necessary to distinguish between two scenarios that could occur. While in some cases a simple overlay of the virtual elements in place might be sufficient for the augmentation, more complex scenes can require some objects of the real scene that partly or fully occlude any of the virtual object to be considered in the virtual environment as well.

2.2.1. Simple AR Overlay

Based on the transformation matrix that contains the pose information of the camera a virtual camera is positioned with the corresponding location and orientation in the 3d scene. In addition, light values that might have been gathered before such as intensity, direction or colour, can be used to improve the appearance of the virtual objects and to generate a more convincing augmentation. From this position the virtual content is rendered and the resulting image can be used further.

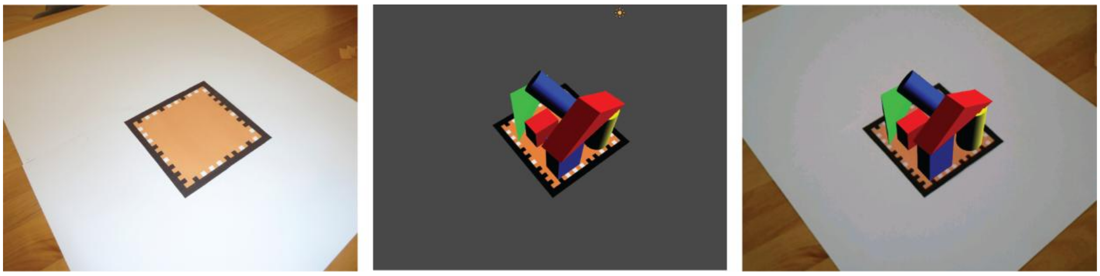

Fig. 2.8 Simple AR overlay; real scene (left), 3D setup (middle), final augmentation (right)

2.2.2. Occlusion by Real Objects

In more complex scenes it might not be possible to see the all parts of the virtual objects in the augmentation because parts of them are hidden behind some real world objects. In such a case also real objects that could at any time occlude virtual objects have to be modelled in the 3D software and need to be available for occlusion calculation. In such a case the correct positioning is just the first step to generate a good result and it needs precise calculation of what parts of the virtual model are actually visible and should thus be displayed. This is typically done by applying a special shader to the occlusion objects that make them not appear in the final render of the scene; it has a similar effect as an opacity mask in image processing software. Those parts of the virtual objects, that are hidden behind one of the occlusion objects are therefore clipped away and will not be visible in the rendered view. An example for such a scenario can be seen in Fig. 2.9.

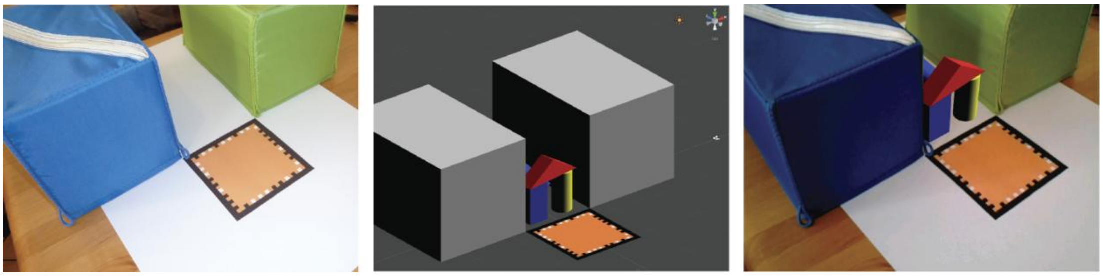

Fig. 2.9 Occlusion by real objects; real scene (left), 3D setup with occlusion objects (grey) (middle), final augmentation (right)

2.3. Display

In this third step the display defines the way the combination of virtual and real content will be presented to the user and thus plays an important role in creating a convincing AR experience. In “A Survey of Augmented Reality” Azuma describes two different methods that are used to combine virtual objects and real world view and calls it “*optical vs. video*”. Even though technology has made progress since then, this comparison still comes true.

2.3.1. Optical See-through Display

The first method, *optical*, refers to systems where the image that was calculated before is directly overlaid to the real view of the user by utilizing special displays. These displays are semi transparent, so that the user can see the real world through them. They are also semi reflective allowing the image with the virtual content, that is displayed by a monitor or beamer, to appear superimposed to the real world view. (Azuma 1997)

A popular member of this category is the Project Glass by Google which is an attempt to introduce HMD technology to a consumer eyewear (@ProjectGlass). A similar technology is also used in Head-up displays (HUDs) which are recently becoming popularity in consumer cars (@Wiki_HeadUp) but not as part of an Augmented Reality system as these HUDs show information to the driver that is not registered with the real world view.

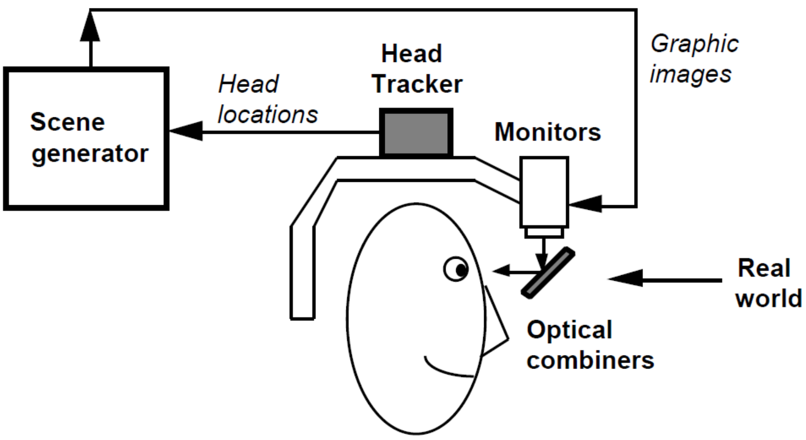

Fig. 2.10 **Optical see-through HMD conceptual diagram (Azuma 1997)**

2.3.1. Video Display

This second method uses another approach and is more commonly used today. In these systems the user does not have an actual view to the real world but he can see the video image recorded by a camera in front of him. The video is processed by the system in real time and the accordingly generated view of the virtual objects is overlaid. The composed image that consists of the real and the virtual objects correctly aligned can then be displayed via a HMD or through a monitor. As smart phones already have a camera as well as a monitor build in, this is the method corresponding to an AR system based on such a device.

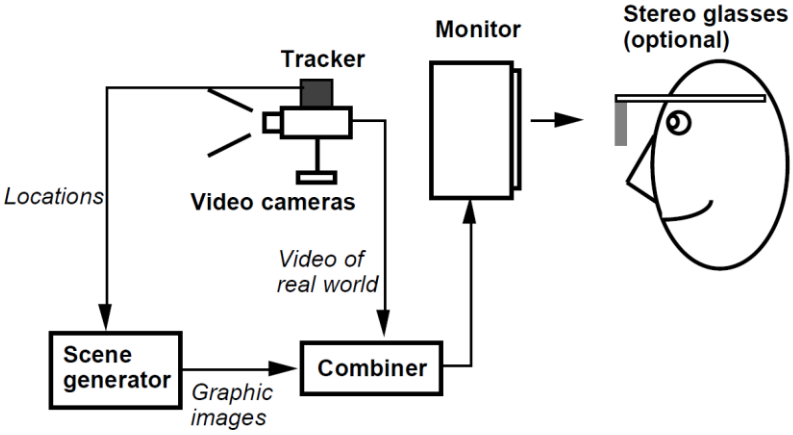

Fig. 2.11 **Monitor-based AR conceptual diagram (Azuma 1997)**

2.4. Workflow Overview

To summarise Chapter 2, Fig. 2.12 gives an overview of the steps required through every frame in an AR application.

I) A picture is taken from the devices camera. This picture will be the reference and background for the image displayed at the end of the frame.

Tracking of the position

CV based AR only:

II) The image taken in step I is analyzed using computer vision algorithms in order to detect available feature points.

III) The found feature points (yellow) are cross-checked with the set of trackables available to the application. A subset of these points is identified as belonging to trackable 4 (red).

IVa) Special algorithms are used to calculate the current position of the camera relative to the recognized trackable again using the feature points. All 6 degrees of freedom can be solved.

Sensor based AR only:

IVb) Based on the GPS, compass and accelerometer/gyroscope the position and rotation information of the device is calculated. Either all 6 DOF are calculated or some can be set to a static value if preferred.

Computation

V) The camera in the virtual environment is positioned according to the values calculated in IVa and/or IVb.

VI) The visible 3d content is rendered from that point of view. If necessary, occlusion elements are considered in this step.

Display

VII) In this final step the previously rendered virtual content is merged with the picture taken in step I. The resulting composition is shown to the user on the display of the device.

As mentioned at the beginning of the chapter, to achieve a convincing AR result this whole process has to be repeated at least 20 times per second.

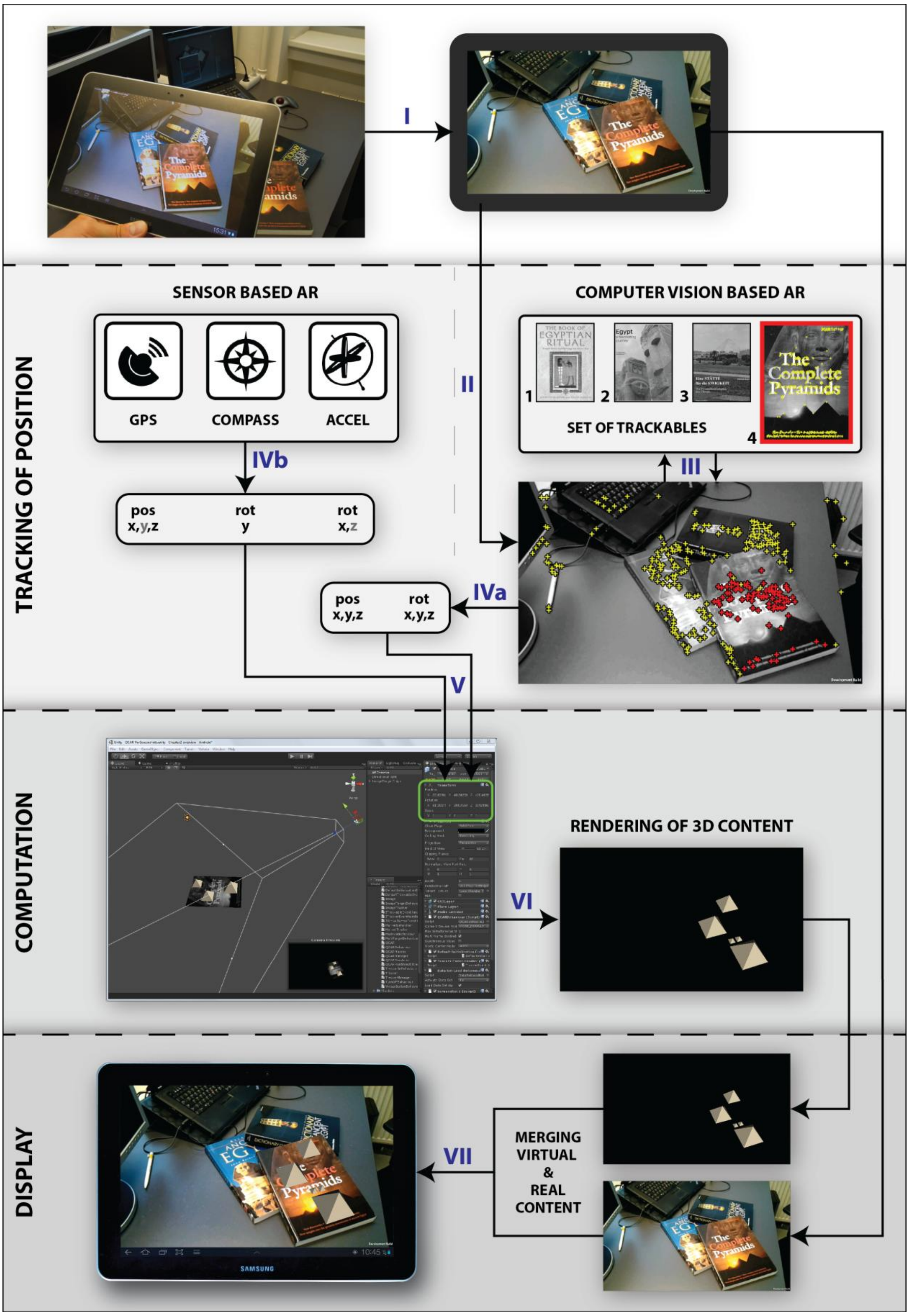

Fig. 2.12 Necessary steps of an AR application during one frame

3. Application Examples in the Architectural Domain

To prove the usability and show the capabilities of Augmented Reality for varying types of architectural visualisation, three scenarios have been chosen. These examples will be referred to as table-top project visualisation, on-site visualisation in context and cultural heritage information system. They differ not only in their content but also in their requirements to demonstrate the wide range of usability. This chapter will give an overview of these examples and their requirements. The actual implementation of these examples is explained in chapter 4.

3.1. Table-Top Project Visualisation

In many architectural offices the design process is supported by the generation of a 3d model. Even more when working with modern building information modelling (BIM) software, a 3D model is generated as a by-product during the planning process. On one side these models are then considered for the evaluation of preliminary design choices or to make further decisions. On the other side renderings, animations or even interactive walkthrough applications are created to present a project to the client or to the public. Even though these digital tools offer a broad palette of possibilities, physical models are still important parts of the design process and often obligatory for bigger project presentations. Some of the main problems of physical models however are the high price of a large scale presentation model, the time it takes to produce it as well as the fact that once it is built it can’t easily be modified.

Still there seem to be two reasons for the importance of a physical model. The first one is usability. With all the possibilities of implementing interactivity and different types of information in a 3D walk-/fly-through application they have one big problem as there is no common user interface for 3d navigation. While some applications use very simplified input methods, they very often limit the freedom of movement severely by for example restricting the user to operations like “move forward”, “move backward”, “turn left”, “turn right”. Other, more complex systems give way more flexibility but require simultaneous usage of mouse and keyboard which is not convenient for inexperienced users. The second reason for favouring a physical model is communication. While typical interactive 3D applications are used by one person at a time or by more people independently, a physical model is a place to gather around and discuss a project with all involved people. Everyone can directly refer to it during a conversation while no one is bound to a specific view.

To summarise there seems to be a gap between the possibilities digital presentation methods offer and the usability in different situations. This is exactly where the use of Augmented Reality can bring a big improvement as it allows the user to experience a combination of the advantages of both sides by showing him virtual content while maintaining his ability to move physically around it.

To build such an AR presentation platform a spatial relation between the virtual and the real world has to be established. As such presentations will typically take place indoors, a sensor based AR approach will not be the first choice. CV based AR seems to be perfectly suited for such a scenario. Thereby a well chosen marker or a suitable plan representing the project can be used as a trackable. Each participant of the presentation or discussion can then move around this trackable and use a mobile device like a tablet computer as his personal window to the virtual world showing his view of the 3D model. Any interactive content that is implemented to further explain the project can be made available to the user via a graphical user interface (GUI) and other touch inputs on the device or by a presenter who solely has control over the content and can decide what should be seen on all devices. In either case the user would still have control over his viewing position. While the first mode is preferable for conversations about the project or to give people the chance to experience it independently, the second mode is intended for presentations, where the presenter wants to make sure that everyone is following along.

3.1.1. Manhattan Skyscraper Design

The project chosen for implementation is a design for a highly energy efficient skyscraper in downtown Manhattan, New York (Fig. 3.1, left). As a main criteria for the design was to keep the ecological footprint of the building as small as possible, a number of environmental factors like solar gain, shadowing of surrounding buildings, wind movement, etc. played an important role (Fig. 3.1, right). It was clear, that it would not be possible to visualise these aspects of the design process in a physical model. Therefore in addition to posters, an AR based application showing the digital model seemed to be an ideal medium for the jury presentation and was considered from the beginning of the design process.

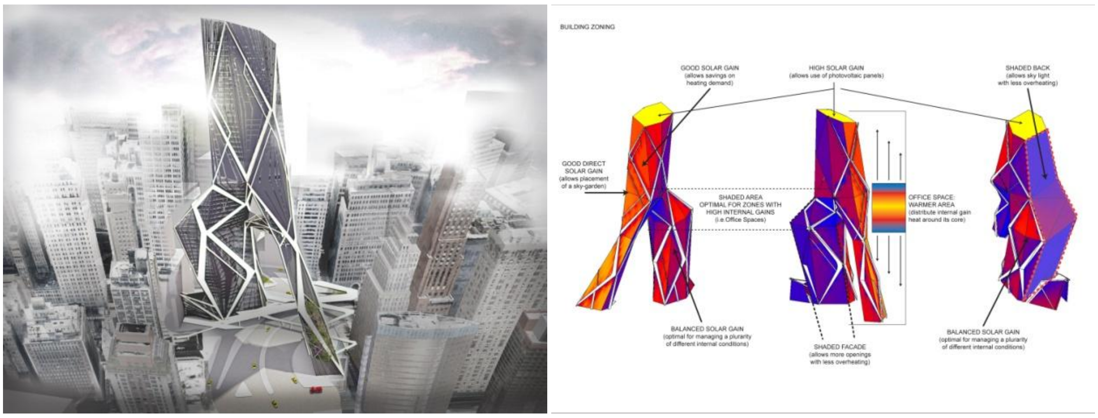

Fig. 3.1 Skyscraper design project; presentation image (left), zoning based on solar gains (right)

3.1.2. Application Requirements

For the purpose of presenting this project the application is required to offer certain functionalities. The main aspects are:

- An overview plan of the area should be used to visualise the proposed building and the nearby surrounding.
- It should thereby be possible to see the project in context of the neighbouring building as well as just by itself.
- Main considerations that influenced the design process should be visualised. Among them solar gain on the facade of the building and shadows in and from the surrounding.
- The application should be able to work in a "guided" mode during the actual presentation and in an "individual" mode afterwards for further, personal investigation.

3.2. On-Site Visualisation in Context

While a physical model gives a very good basis to judge a project on the whole, to really evaluate an architectural design it also has to be perceived in its actual context. The current approach to do so is by using image compositions. These are either fully rendered images showing the project and an abstracted version of the surrounding or they are created by inserting a rendered view of the project into a real picture of the site. In most cases the first method is used because it makes it easier to give the image a specific style without being bound to restrictions set by the picture. This however is very often used by visualisation artists in order to show the well designed parts of a project but avoid any problematic views. Furthermore, depending on the goal of the image, being realistic versus dragging attention, completely irrational situations might be created. Among them sunlight coming from the north side of a project, just because this is the side visible in the rendering or people being out of scale to fake the desired impression of proportions. While this might deliver some iconic images, they do not necessarily offer a reasonable representation of the project.

Augmented Reality could offer a completely new way to supply the demand for this kind of visualisations by bringing the virtual model to the actual building site. As AR applications for this purpose would in general be used outside both, sensor based and CV based approaches seem to be reasonable to calculate the devices position. To choose one method the situation in which the application should be used has to be evaluated in more detail. For buildings "in the green", meaning that they are going to be erected in the country side as solitaires, sensor based AR can be and sometimes has to be the first choice (see 1.1.2, 3D on). In fact sensor based AR is the only choice at locations where no neighbouring buildings or other appropriate geometry that could be used as reference for CV based tracking are available. The downside of such a system, the inaccurate positioning of the virtual content due to measurement errors, can in these cases still be coped with if the occurring misplacement does not influence the visual perception of the user too much. In an inner urban environment the situation is completely different. While at a free standing building an offset of the virtual geometry of about two meters might not even be recognized, a tolerance of a few centimetres is the maximum to believably position a 3D model in a built scenery. Also considering the fact, that the accuracy of the compass as well as the GPS precision typically decrease inside a dense city, sensor based AR is not suited for this kind of requirements but can be used as a navigation aid to direct the user to the site. For the actual visualisation a CV based solution is the better option. To make that possible a spatial relation has to be established again. To achieve that, facades of existing buildings at the site could be used as a reference. Once they are recognised in the camera image the 3d model could be placed accordingly.

This would allow the user to walk around the site while seeing how the planned project would look right in place. Not only would this give the user a more natural idea of how well the building integrates into its neighbourhood, but it would also allow him to use his natural sense for scale to understand the proportions, a fact that is always problematic in still image compositions. At the same time it would again be possible to implement interactive content or the possibility to switch between different versions of a design.

3.2.1. Pedestrian Bridge

To evaluate the feasibility of such a visualisation, two fictive modifications to the main building of the Vienna University of Technology at Karlsplatz have been created. The first example is the restoration and redesign of a part of the facade. It is supposed to demonstrate the capabilities of using a single facade of an existing building to partially or completely replace it with a virtual model. The second scenario is a bridge connecting two separate buildings. A less complicated task would be to completely fill the gap with a new building but a bridge with glass elements covers a wider range of possible necessities for such a visualisation like transparency in the virtual model or partial occlusion by a real world object.

Fig. 3.2 Test sites at Karlsplatz, Vienna; Facade refurbishment (left) and bridge project site (right)

3.2.2. Application Requirements

The final application is supposed to work as a visualisation tool for people interested in this project and as an design evaluation help for decision makers. As such a number of requirements have to be considered.

- GPS based guidance should help users not familiar with the project site to find it.
- The actual project visualisation should rely on CV based AR to deliver high enough precision to credibly position the 3D content. The facades available in the direct surrounding of the site should be used as trackables.
- The user should have the possibility to switch between different design alternatives for evaluation.
- It should be possible to adjust the appearance of the model to the light conditions as they can change drastically in outdoor scenes. This is an important factor for credibility.

3.3. Urban Cultural Heritage Information System

While the first two example applications deal with the visualisation of future or unrealized buildings, this third one is targeted at the area of urban cultural heritage. The analysis and interpretation of an archaeological site is a complex process and many layers of data and interpretations of it are generated while performing an excavation and studying the findings afterwards. Very often at urban sites, the excavation is closed again and most of the data is represented in publications, the findings themselves may be exhibited in different locations or are stored in archives. It is a challenging task to visualize the complexity of all available information at the context of the site. AR can thereby be used to provide information about destroyed, damaged or modified buildings and findings that have been removed from the site or are otherwise not accessible or visible to visitors.

While from a contextual point of view this example differs clearly from the previous examples, the considerations for the AR tracking system are similar to those made in 3.2. As a purely sensor based approach would not deliver precise enough positioning in such an environment, an alternative using facades in the surrounding for CV based tracking could be used. The dimensions of this system would however clearly exceed those of the previous example as it would be necessary to cover not only one specific project site but a whole area in which a user might move around freely and look into any possible direction. Based on the current camera view he could then see correctly positioned tags representing different types of available content. This might sound similar to how information is provided in Layar or Wikitude as mentioned in 1.1.2 but there are two clear differences. The first one is that the proposed system would be able to position the tag at an actual location instead of just giving an idea about the direction. The second one is that unlike in these applications, the consideration of buildings and their facades makes it possible to just show those tags that correspond to locations that are actually in the current field of view and not situated behind a building. In typical AR browsers the only possibility to simulate this effect is to choose a narrower radius for the display which is not an ideal solution.

3.3.1. Historical Site Michaelerplatz

The demo application is an information system for the historical site of Michaelerplatz. Situated at the intersection of Schauflergasse, Herrengasse, Kohlmarkt and Reitschulgasse Michaelerplatz is a calm, generous and beautiful square in the first district of Vienna surrounded by a group of interesting buildings. Because of its location and historic background it is a frequently visited hotspot for tourists. The history of the area however goes back as far as to Roman ages where in the 1st Century it was part of the so called canabae legionis which was part of the Roman camp Vindobona and was located at the intersection between two important streets at that time. After the Romans left the area in the 5th century it seemed to be abandoned until it was included in the medieval city in the 13th century. The St. Michael church was built before 1525 as a second church besides St. Stephan. After that the slow development to an actual square was characterized by imperial-private and public use until the Loos-House concluded the development into a square in the early 20th century. In 1990 and 1991 the Vienna Urban Archaeology made two excavations to investigate the remains under the square. During these excavations parts of buildings and basements have been documented. These helped to understand the history and development of the square from Roman and medieval times until the modern era. Besides that numerous other items of daily use have been found. Among these findings are pottery, coins, glass, jewellery and different items made of bone from different epochs. (Ranseder et al. 2011)

Presently the majority of the excavation is covered by the pavement of the square and only a small part of it remained visible for the public. The documented findings have been relocated and are in their majority inaccessible. Therefore an AR information system about the site could be a good way to make some of the hidden treasures available to everyone and increase the awareness for the importance of this square.

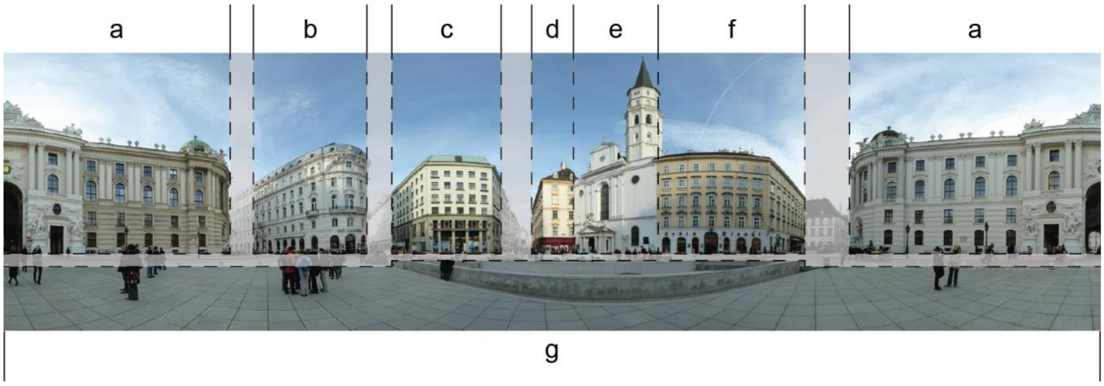

Fig. 3.3 Panoramic image with zoning of Michaelerplatz, Vienna

Fig. 3.3 gives an overview of nowadays Michaelerplatz and its surrounding buildings:

a. Hofburg – Michaelertrakt (Hofburg Palace – St. Michael's Wing)

   As a part of the Hofburg the wing oriented towards Michaelerplatz was intended to be a representative entrance form the city centre.

b. Palais Herberstein

   Palais Herberstein was built in 1879 and commissioned by the counts of Herberstein and was criticized for stylistically referring to the palace rather than the church and the other buildings around the square.

c. Loos-Haus (Loos house)

   Built between 1909 and 1911 by Architect Adolf Loos the so called Loos-House contained business and living areas for the tailor company Goldman & Salatsch. While there was criticism about the facade at first, today it is recognized as one of the most important buildings of early Modernism and was declared cultural heritage in 1947.

d. Großes Michaelerhaus (Big Michaelerhouse)

   This is a late baroque apartment building built by the Barnabites as a way to earn some money for their Order. Its ground plan is already visible in a plan from 1710.

e. Michaelerkirche (Church St. Michael)

   The Church itself goes back to the 13th century and got its name from Saint Michael. After parts of the church got destroyed several times its current appearance is mainly based on the state of the 16th century. Since 1626 the church belongs to the Order of Barnabites which was an important impulse for the development and design of the complex.

f. Kleines Michaelerhaus (Small Michaelerhouse)

   This is a late baroque apartment building built in 1732 which just saw minor modifications. The Small Michaelerhouse was commissioned by the Barnabites as part of the building complex including the Church St. Michael and the Big Michaelerhouse.

g. Excavation Site Michaelerplatz

(Krause 2007; Ranseder et al. 2011)

Fig. 3.4 Excavation at Michaelerplatz in 1990/91 (Museen der Stadt Wien – Stadtarchäologie)

3.3.2. Application Requirements

As a guide for visitors interested in the history and the findings at the archaeological site the application has to fulfil a number of requirements as follows:

- The user should be able to move freely around the square. The tracking, based on sensors and/or CV should deliver sufficient results at any position.
- Tags representing different kinds of information (text, images, videos, web links, 3d models) should be positioned on the square.
- If interested the user should be able to call further information about an item by clicking on the according tag.
- Only those tags actually visible from the current position should be displayed.

4. Implementation

Building up a complete mobile Augmented Reality system for the previously explained scenarios is an enormous task that involves a wide range of specialists and financial resources. Therefore this chapter will give an overview on how existing software and hardware can be used to actually implement an AR system suitable for architectural visualisation.

While the first part will give an introduction to different platforms and software development kits (SDK) available and suitable for that purpose, the second part will go through the necessary steps to provide required information to the system. Following this, the implementation process, challenges and issues of the three previously mentioned scenarios - table-top visualisation, on-site visualisation and urban cultural heritage - will be discussed.

4.1. Development Environment

4.1.1. Mobile Devices and Operating System (Android/iOS)

Even though modern smart phones fulfil the technical requirements for AR, the more suitable devices are the tablet computers. Unifying the advantages of smart phones, like the variety of sensors and the high degree of mobility with a much bigger screen makes them an ideal tool for AR visualisation. While there are numerous hardware manufacturers offering tablets in different price segments, most of them are suitable for our purpose. The main difference between the devices is the operating system (OS) they are running. When talking about tablets, there are two operating systems that need to be considered, Apple iOS and Google Android. With market shares of 54,7% (iOS) and 44,6% (Android) they shipped more than 99% of these devices worldwide in the 4th quarter of 2011 (@IDC). As both OS have their advantages and disadvantages, Android was picked as the main platform due to the better available development tools. The actual device picked for testing is the Samsung Galaxy Tab 10.1 (GT-P7500) offering a 1GHz dual-core processor, 1GB RAM and a 3 Megapixel camera running Android 3.2. It can be seen as a mid-range device in the tablet category. At a later point some applications were also ported to a 3rd generation iPad (A1416) running iOS 5.1.1 as proof of concept.

4.1.2. 3d Game Engine (Unity)

The implementation of an interactive 3d visualisation, no matter if it is running on a mobile device or on a desktop computer, requires a wide range of functionality. Among others it is essential to have a 3d rendering engine, physics simulation and collision, a scripting API, a GUI system, networking capabilities among other qualities. While it would be possible program the whole application in native code using libraries that offer these functionalities, it is more reasonable to take an already existing solution for this. Game engines, systems designed for the creation and development of video games (@Wiki_Game), typically offer a lot of the features needed. They are commonly used to create virtual reality environments taking the user into a completely artificial world. Besides being used for games, a strong tendency goes towards the development of so-called Serious Games, which is generally understood as the application of game concepts, technologies, and ideas to non-entertainment applications (Shiratudd et al. 2008). Among them a growing number of interactive architectural visualisations can be found. A large number of game engines are available at the moment but the majority is not suited for mobile development. Tab. 4.1 shows a selection of game engines that were considered as main development framework. A more complete list can be found at (@Wiki_GameList). While the availability and quality of "Getting Started" resources, the general documentation and the OS compatibility were considered in the first place, some other important factors made Unity the first choice. These were the ability to customize and extend the functionality of Unity with custom scripts, a huge amount of third-party plug-ins that expend the functionality of the engine even further and the fact that applications can be published to both desired platforms, Android and iOS, without any major modifications. Besides that the very clean and organized user interface and workflow as well as the support for a wide range of file formats for assets like 3d models or textures make it an ideal tool for non-programmers and creatives.

	Unreal Engine/UDK	SIO2	Unity
“Getting Started” Resources/Workflow	-	~	+
Documentation/ Support	~	~	+
Available Platforms (Android/iOS)	~	+	+
Pricing (test/production)	+/~	+/~	~/~

Tab. 4.1 Comparison of different Game Engines in relevant categories

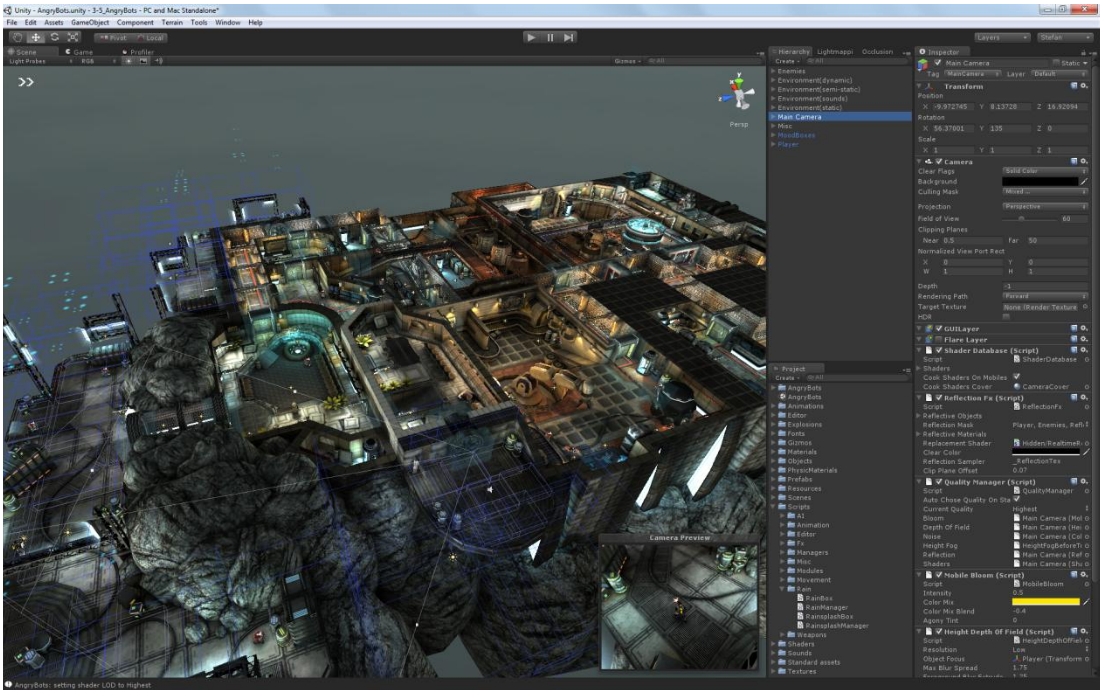

Fig. 4.1 Screenshot of the Unity user interface

4.1.3. AR Package (Vuforia SDK)

As game engines usually do not have a fully functional solution for Augmented Reality built in, it can be added via third party plug-ins. There are currently three computer vision-based Augmented Reality SDKs available that are capable of natural feature tracking and offer a plug-in for Unity. These are the commercial solutions String SDK (@String AR SDK) and metaio mobile SDK (@metaio mobile SDK) as well as the free vuforia SDK from Qualcomm (@vuforia AR SDK). Furthermore a Unity plug-in for Total Immersion´s D´Fusion (@Total Immersion) got available at a later point but was not considered.

All three packages are capable of natural feature tracking, but the way it is implemented and the resulting consequences make some of them better suited for the desired scenarios than others.

String's SDK is performance optimised and uses only very little memory while delivering robust tracking. This however comes at the price of one unacceptable necessity. In order to use an image for tracking, a high contrast outline (near black or near white) is required. While on architectural plans this is not common and very unlikely going to be accepted by possible clients, it is simply impossible to have this kind of outline in an outdoor scene. This makes the String SDK practically unusable for the considered purpose.

The metaio mobile SDK is probably the most complete solution when it comes to features. It offers tracking for different kinds of markers as well as tracking of 2D and 3D geometry based on natural features, making it suitable for indoor and theoretically ideal for outdoor scenes. The disadvantage of this SDK is a rather user-unfriendly documentation and it's very high price when it comes to production.

The third plug-in considered is the vuforia SDK from Qualcomm (originally called Qualcomm Augmented Reality or QCAR SDK). Like the metaio mobile SDK it can also use special markers for tracking, but the main functionality is natural feature based tracking of planar images. While real 3D tracking is currently not supported by the vuforia SDK, more complex geometry can be used for tracking by merging more of these planar trackables to a non-planar geometry. This should make indoor as well as outdoor scenarios possible. It's probably biggest advantage over the other two SDKs is the fact, that it can be used free of charge even for commercial projects. Taking all these factors into account, the most feasible solution for the test implementation appeared to be the vuforia SDK.

	String®	metaio mobile	vuforia
Platform (mobile OS) compatibility	~	+	+
Documentation/ Support	~	~	+
Indoor usability (2D tracking)	~	+	+
Outdoor usability (2D/3D tracking)	-	+	~
Pricing (test/production)	~/-	+/-	+/+

Tab. 4.2 Comparison of AR SDKs based on offered functionality, not actual tests

The workflow of setting up an Augmented Reality scene using the vuforia SDK is straight forward. After a planar reference image has been selected, it can be uploaded to the Qualcomm Target Management System (TMS) seen in Fig. 4.2. To do this an image in JPG or PNG file format with a file size of no more than 2MB is required. The system then automatically evaluates and processes the images uploaded and gives feedback on the quality of the trackables (Fig. 4.3). Contrast enhancement of the pictures can be very helpful to increase the available feature points for tracking where necessary. Once the Qualcomm web service reports sufficient tracking quality, the processed files can be downloaded as a Unitypackage. After loading the vuforia plug-in, the recently created trackable information from the TMS, can be imported into Unity making the vision-based AR system functional.

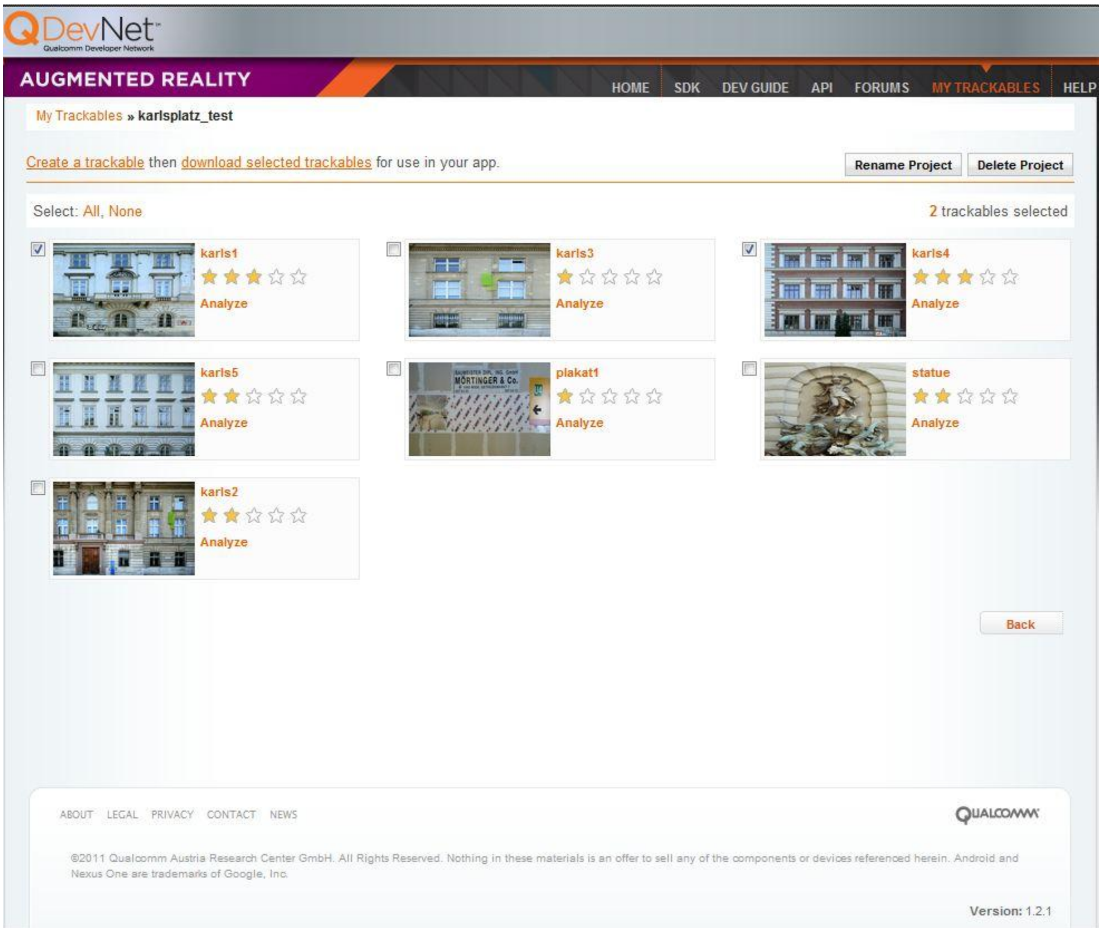

Fig. 4.2 Qualcomm AR target management system

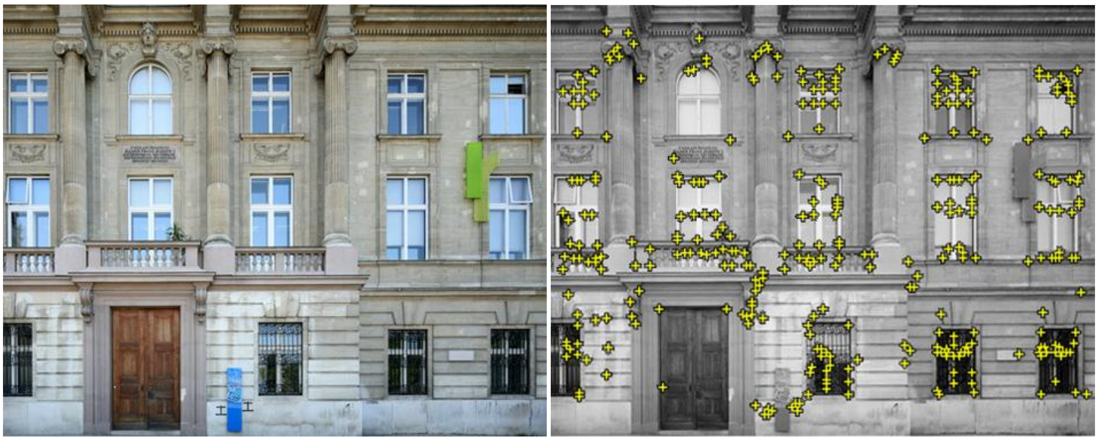

Fig. 4.3 Facade at Karlsplatz – Original image (left), Greyscale with detected Feature Points (right)

4.1.4. Additional Plug-ins

In order to also test sensor based AR, it is necessary to get access to the phones GPS sensor, compass, gyroscope and accelerometer. At the time of implementation the

GPS coordinates could be acquired directly inside Unity but other necessary sensors were not available. Therefore a plug-in called Gyrodroid (@Gyrodroid) is used to get the missing values and complete the system.

4.1.5. Platform Overview

To conclude chapter 4.1, the test implementation is supposed to run on tablet computers running the Android OS, iOS might be considered as an alternative. The main development platform is the game engine Unity which already offers a lot of the necessary features for this purpose. Furthermore a quick overview summarising chapter 4.1 and complementing Fig. 2.12 will be given and thereby follow the same three steps an AR system needs to fulfil in every frame.

Tracking of the position:

> Two methods, sensor based and computer vision-based AR are going to be used. In sensor based AR the GPS values are available in Unity by default. The compass, accelerometer and gyroscope are made available through an additional plug in called Gyrodroid. The obtained sensor values are then processed in unity and mapped to the 6 DOF in the Unity workspace (see chapters 2.1.1 and 4.2.1). Note: some of the sensors have been made available in Unity in the meanwhile making parts of this step obsolete. Vision-based tracking is handled by the vuforia plug-in from Qualcomm and the 6 DOF of the camera are assigned automatically.

Computation:

> Computation of the actual scene is based on the previously acquired camera position information and is entirely done in Unity. Custom interactions like user interface, physics simulation, networking features, etc. that can be added optionally are also handled in Unity and are processed before the scene is rendered.

Display:

The rendered content is overlaid on top of the camera image, which is always provided by the vuforia plug-in. This merged image is then shown on the display of the tablet.

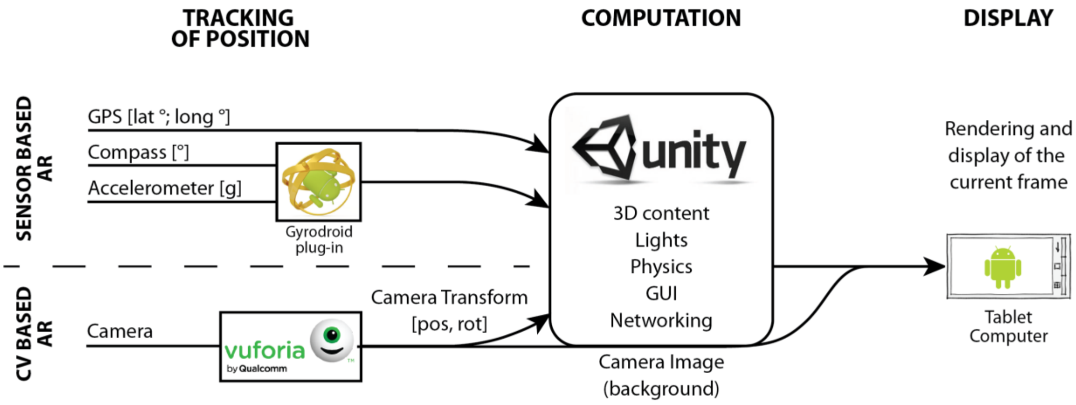

Fig. 4.4 Dataflow of the test platform during one frame

4.2. Data Acquisition and Preparation

In order to use this platform outdoors, data about the environment needs to be collected and prepared for the application. For the sensor based tracking system plans containing GPS reference points are necessary to position the virtual content accordingly. The CV based tracking system requires frontal facade images as planar trackables. In this chapter the necessary steps to acquire them are described.

Fig. 4.5 Measurement of GPS reference points (left) and photography with Tilt-Shift lens (right)

4.2.1. GPS Referenced Plans

One essential point when building a sensor based Augmented Reality application in a game engine like Unity is the mapping of the GPS polar coordinate system to the game engine's Cartesian coordinate system. To do this it is very helpful to have a plan of the used area that has some fix points with known GPS coordinates. As such plans were not available for the specific areas, they had to be created. The "Mehrzweck Karte" (MZK) (@Stadt Wien), a highly detailed map of the city of Vienna, was used as a basis. A Leica GPS1200+ Surveying System, a highly accurate GPS system usually used for geodesy and allowing measurements with precision of about 2 cm made it possible to pick some fix points like corners of stairs or manhole covers in the landscape and get the precise GPS coordinates. For the whole Karlsplatz and Michaelerplatz area 17 fix points were measured to be able to choose the best fitting ones during the actual process of setting up the application.

The mapping of the polar coordinates to the Cartesian coordinates does not consider the curvature of the earth and therefore gives a small error. This error can be ignored as it just accounts for an inaccuracy of about 30cm over a distance of about 1 km (0,03%) if the measurement in the plan is compared to a calculated distance using the Harvesine Formula (@Harvesine).

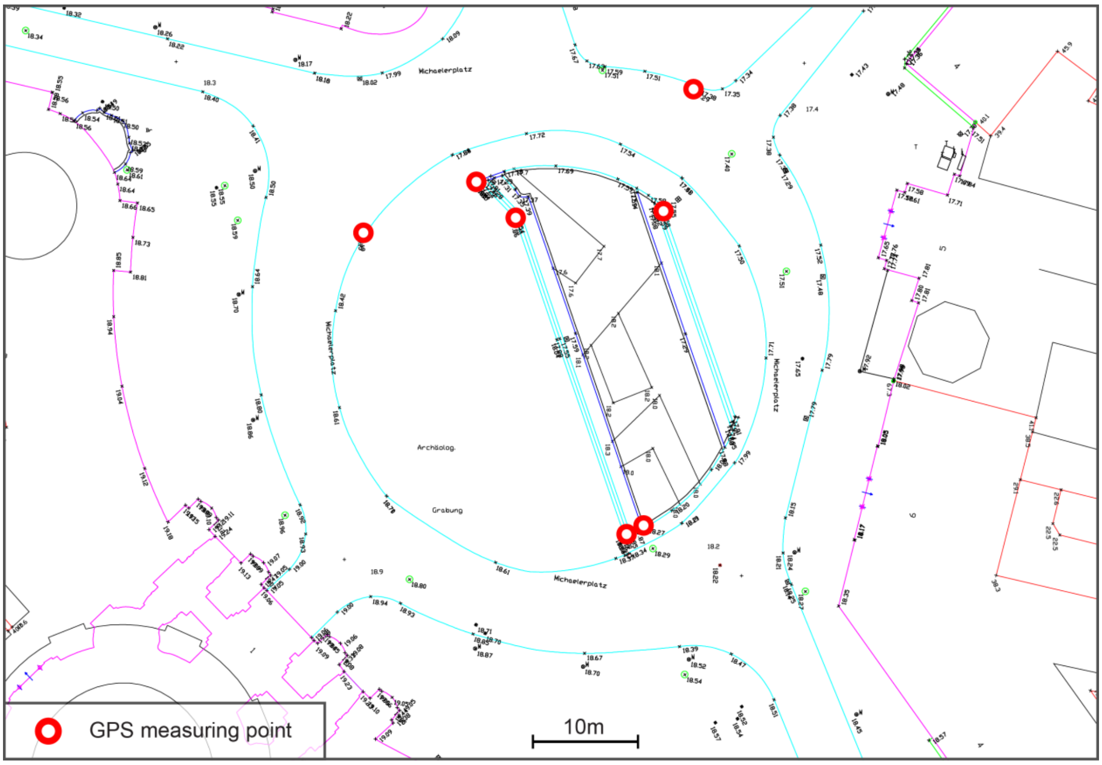

Fig. 4.6 MZK showing Michaelerplatz and the taken GPS measuring points (Plan Source: ViennaGIS, www.wien.gv.at/viennagis/)

Fig. 4.7 Leica GPS1200+ Surveying System

4.2.2. Distortion Reduced Facade Images

In order to use facades of buildings as trackables, the information necessary to recognise them later on has to be “taught” to the chosen vision-based AR package. As the vuforia AR SDK, described further in 4.1.3, is not capable of recognizing 3D geometry but only 2D image and assemblies of them, a step of abstraction has to be taken. This is done by considering facades of buildings as flat elements. Even though facades are typically not perfectly planar, in most cases the majority of their elements are more or less part of a single plane, making orthographic pictures of them reasonable trackables, that can be uploaded to the vuforia TMS.

The common approach to get these images would be to take a picture of the facade from a preferably high distance using a lens with rather high focal length resulting in an almost orthographic image. However in many cases, especially in architectural photography, the space necessary to shoot such images is not available. This might be due to narrow streets that make it impossible to reach the necessary distance or due to elements in the surrounding like trees or bushes blocking the direct view to the desired object. In such a case one could take pictures from a far less ideal position and apply distortion correction techniques in post-production. This however results in a quality loss due to unequal pixel distribution over the image. In this project orthographic photos of the facades have therefore been taken directly by using a digital camera together with a tilt-shift lens (Fig. 4.8), providing a high quality and almost distortion free image.

Fig. 4.8 Canon TS-E 17mm f/4 L Tilt-Shift Lens

While the Tilt function was not used, the Shift function was of great help. Shift hereby refers to a displacement of the lens parallel to the image plane that allows adjusting the position of the subject in the image area without moving the camera.

In effect the camera can be aimed by using shift movement. Shift can be used to keep the image plane parallel to the subject and can therefore be used to photograph a tall building while keeping the sides of the building parallel. (@Wiki_TiltShift) Fig. 4.9 shows a comparison between the same pictures taken without and with the Shift functionality of the Canon TS-E 17mm lens that was used. The originals are just cropped and some minor colour corrections have been applied in order to equalize the overall brightness of the images. Neither image was modified in terms of distortion correction.

Fig. 4.9 Facade image from 1,5m height shot with Canon TS-E 17mm (left); facade image from same position shot with Canon TS-E 17mm using the shift functionality (right)

4.3. Table-Top Project Visualisation

4.3.1. Technical Implementation

The implementation of the AR features in this case only required CV based tracking which was accomplished by the vuforia plug-in without any further modifications. After a suitable plan was chosen as a reference the necessary steps described in 4.1.3 were accomplished and the 3D model could be positioned in Unity according to it. The only necessary change that had to be made to the plan used as trackable was an adjustment to the line thickness in order to deliver enough contrast.

The most challenging task at this project was probably to find an appropriate workflow for the guided mode that allows one presenter to switch the content on the devices of all participating viewers. Fortunately Unity offers very powerful networking features which make it possible to send commands to other tablets within the same WLAN network. As this functionality is not OS specific but handled by Unity, it even allows devices running different systems, in our tests Android and iOS, to communicate flawlessly. Thereby one device, typically the one operated by the presenter, is defined as "server" and will manage the transfer of commands between all participants. The other "clients" do not have influence on the content and depend on the server to change it. The setup used in this server-client scenario is visualised in Fig. 4.10 and can be explained in four steps:

1) Before starting the application all devices have to connect to the same WLAN
2) Once running, one device is defined as server at runtime
3) The other devices connect to the server using its IP address
4) Upon request the server sends commands to the clients. The necessary feedback to make sure that the message was received by all clients is handled by Unity.

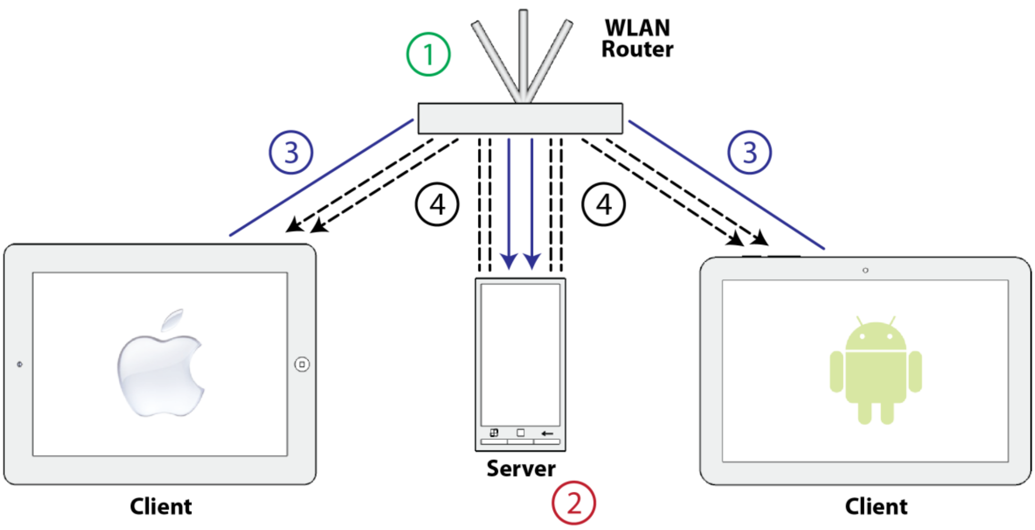

Fig. 4.10 Server-Client setup used for the guided mode

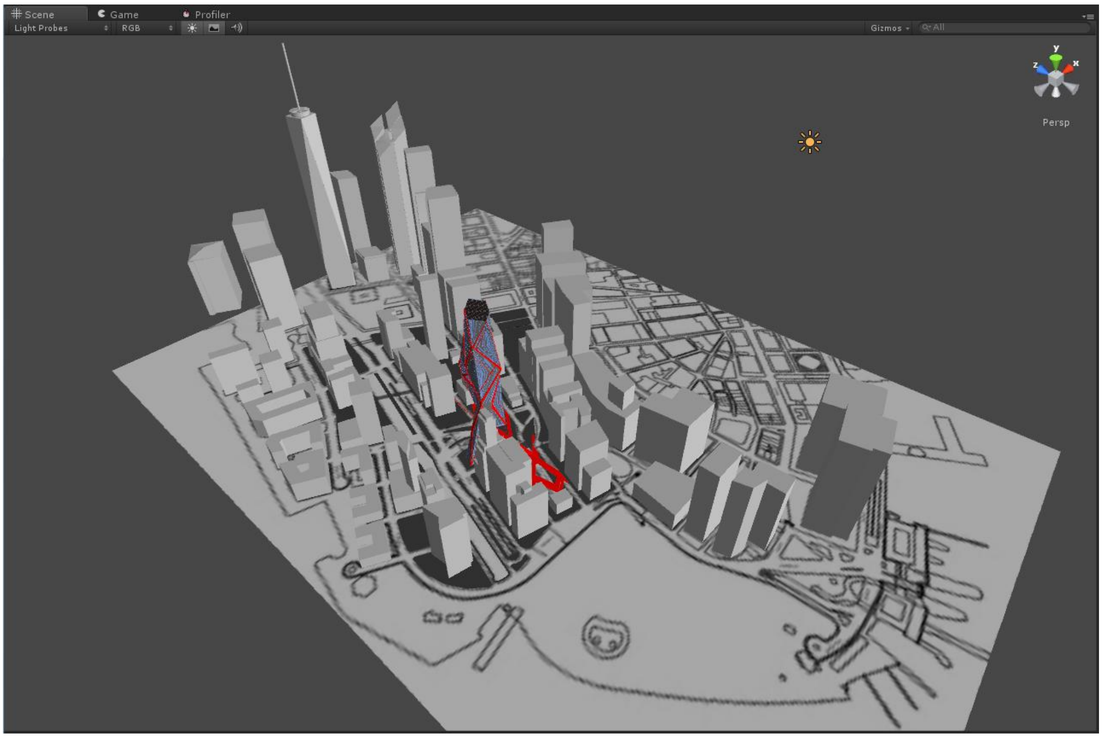

Fig. 4.11 Scene setup in Unity with all the necessary geometry

4.3.2. The Application in Use

As previously mentioned the application can be operated in two different modes, a guided mode, where a server-client hierarchy is used to centrally control the content and an individual mode giving every user complete control over what is shown. As they are just two different operational modes, not different applications, the available options on the user interface as well as the actual content shown are exactly the same and just the degree of control is different.

In a typical presentation scenario the jury would be given tablets in client mode and the presenter would be in server mode. Thereby the server does not necessarily have to be a fully AR enabled version of the application. In fact it might often be more useful and convenient for the presenter to use a torn down version running on a smart phone and just exposing the GUI. This way it works like a remote control for the application. Having the referenced plan in front of them the jury members can look at it through the tablets. As soon as the plan is recognised by the application the model can be viewed from every position. The UI visible on the client devices is in this stage limited to a minimum.

Fig. 4.12 Table-top application in guided mode

Fig. 4.13 Table-top application in individual mode; solar gains visualised (right)

Once the presentation is finished the presenter can make the full GUI accessible to the audience and give them a chance to investigate the project on their own. The user interface can be seen in Fig. 4.14 and offers the following functionality to the user:

1- Skin on/off: Allows the user to switch the skin or facade of the building on and off making the inner structure and connections visible.
2- City on/off: Can be used to turn the model of the surrounding off. By doing this the building can be seen without other buildings blocking the view.
3- Solar gain: Changes the skin of the building to a graphical representation of the thermal gains across the facade. This has been a key feature in the design process.
4- Shadows: Switches the application to shadow mode and brings up the shadow UI (5) on the left side.
5- Shadow UI: This exposes the control over the shadow settings in the application and consists of three buttons. “animate” circles through one day in two-hour steps and shows the influence of the buildings shadow on the surrounding. “+” and “-“ allow to manually adjust the time and the appropriate shadows.
6- Network UI: This is not a necessary part of the GUI but was implemented in the test application to handle the network connection.

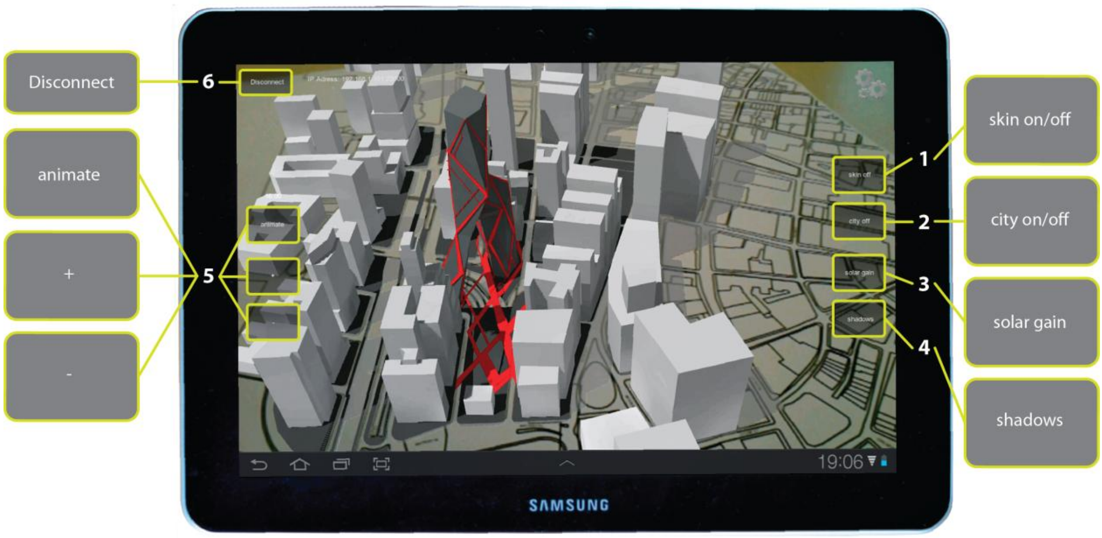

Fig. 4.14 **User Interface and functionality of the table-top application**

4.3.3. Discussion

It was possible to demonstrate how a traditional physical model could be replaced by a virtual 3D model by using Augmented Reality on mobile devices. Different aspects of the physical model have thereby been considered and integrated. The application uses computer vision-based tracking with a site plan as a trackable. By offering a guided and an individual operation mode it proved to be well suited for presentations and combines the advantages of digital and virtual presentation techniques.

While the presented application delivered very appealing results in terms of user experience as well as in terms of AR tracking quality different issues were observed in other tests. A first issue is based on a necessity of this kind of CV based AR, the fact that at least some part of the trackable has to be in the cameras field of view. This can cause problems especially with virtual models that reach very high above the trackable. In such a case the user might not be able to inspect the top area of the model from all angles. The only possibility to see this area and still have the trackable in view would be to stand almost straight above the model which is most likely not be the desired view. The only way to overcome this issue with the current configuration is to choose the proportion between the size of the reference image and the height of the 3D model properly in order to minimize such situations. A sensor based fallback system that takes over when the trackable is lost could be a solution in some cases, but due to the fact that only the rotation, not the position of the phone can be reliably tracked it was not considered any further.

Fig. 4.15 Collaboration while using the table-top AR application

A second issue observed is related to the design of the plans used. While from a CV point of view high contrast lines and non-repeating patterns are clear advantages for detection and tracking, architects often tend to chose light colours with relatively few contrast when it comes to plans for design presentations like competitions. In addition plans in early planning stages often show a fair amount of either unpopulated area (meaning that no furniture or other elements have been added to the plan) or highly repetitive elements like the same tree in different scales, chairs and tables. Tests with actual design stage plans from different project were thereby in many cases completely unusable due to the mentioned problems factors. Therefore the use of an AR application for presentation purposes either has to be considered in the design and layout of the plans (NOT the design of the building) or better suited version of it has to be created for the trackable. In later phases of the design the effect seems to be less problematic as the natural addition of measurements and other elements automatically leads to an improved tracking performance. Furthermore the tendency to used light colours in the plans decreases and leads to higher overall contrast.

4.4. On-Site Visualisation in Context

4.4.1. Technical Implementation

The demands for this application required the implementation of both, sensor and computer vision-based tracking of the position. The whole setup of the scene was done using information from the MZK. For the sensor based system one of the previously measured fix points on the map was used as reference to calculate the current position of the user based on the GPS coordinates. Furthermore it is limited to the 4 DOF explained in chapter 2.1.1 based on the assumptions made there. For the CV based mode a total of 5 trackables have been generated. One of them is used alone for the facade refurbishment example. The other four are joined to one “multitarget” and cover the two corners that enclose the bridge. Furthermore volumetric models of the two existing buildings the bridge is supposed to connect are added to the scene for occlusion calculation.

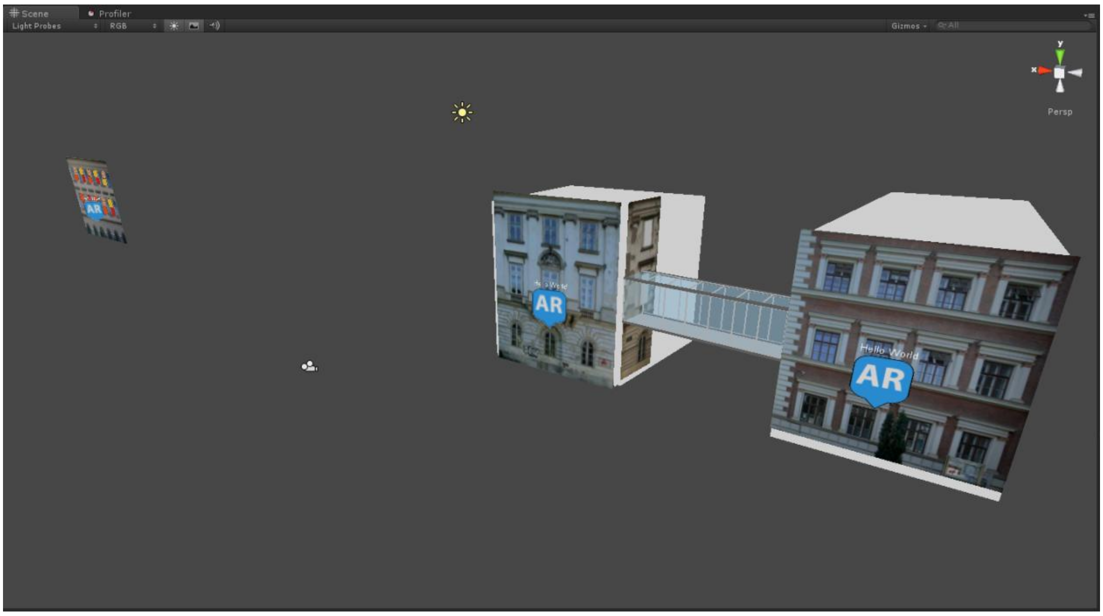

Fig. 4.16 Scene setup in Unity; trackables, occlusion objects (grey), 2D-tags and 3D models

4.4.2. The Application in Use

Once the application is started a simple geo referenced 2D icon (tag) shown on the screen indicates the availability of a project as well as the direction and the distance to it. In this stage, sensor based AR is used to help a user that is not familiar with the site to find the exact location of the project and the trackable environment (see Fig. 4.17). A click on one of the icons can then bring up further information about the project like a description, contact information or the name of the architect.

Fig. 4.17 Sensor based AR showing 2D tags to indicate a project

After approaching a registered facade and it's detection by the application the 2D tags disappear and the 3D geometry is shown in the correct position (see Fig. 4.18). In the facade refurbishment example it was not necessary to consider any of the surrounding geometry for occlusion. The augmentation is just a correctly positioned overlay of the virtual geometry above the camera image. While the solid parts of the facade are thereby virtually replaced, the windows are kept from the background image making the augmentation more realistic as the real reflections in the glass are visible.

The second example is shown in Fig. 4.19. The bridge is again correctly positioned and based on the tracking from the neighbouring facades. As the adjacent buildings are modelled as occlusion objects, the parts of the bridge that are hidden behind the corner are correctly clipped off. As long as the user keeps some parts of the trackable facades in the camera view he can move around freely and investigate the project.

Fig. 4.18 **Computer vision-based AR used to show a facade refurbishment and redesign**

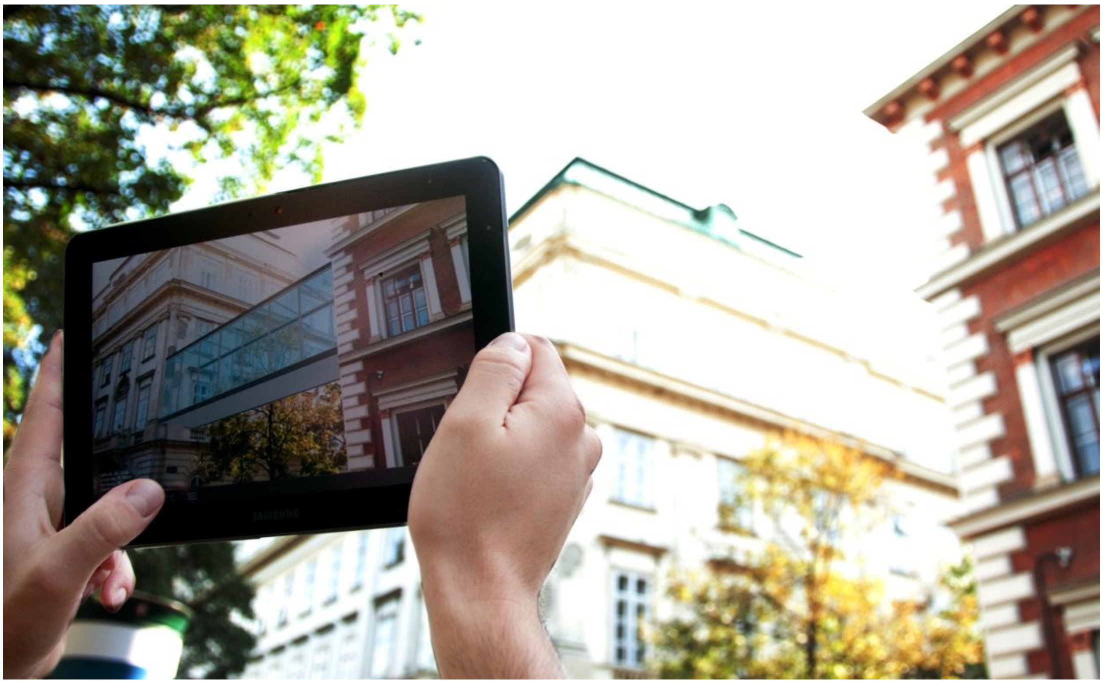

Fig. 4.19 **Computer vision-based AR used to show a bridge between two existing buildings**

While running the application further interaction is possible through the user interface as can be seen in Fig. 4.20. The left side is mainly intended to give control over the sensor based AR functionality and to give information about the scene while the right side offers control elements for the actual 3d scene.

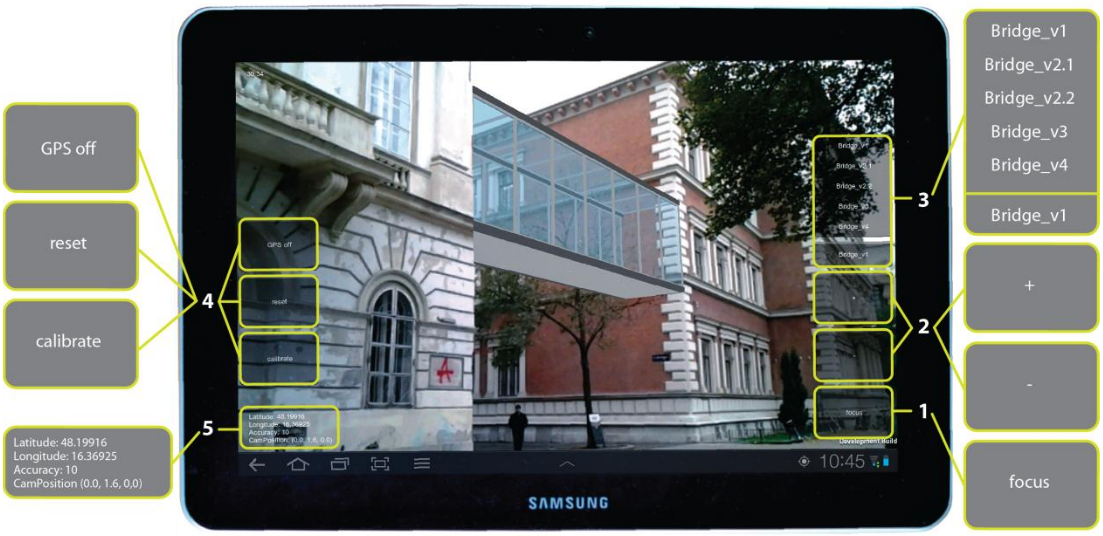

Fig. 4.20 User Interface and functionality of the on-site application

The elements of the user interface are:

1- Focus: This button is used to start the autofocus process of the camera in order to reduce tracking problems due to a blurry camera image and to increase the quality of the background image.

2- Light +/-: Makes it possible to change the brightness of the virtual elements. This allows fitting the environmental conditions and giving a more realistic appearance.

3- Load: Brings up a list of all available models for a certain scene. This can be useful to switch between different design alternatives.

4- GPS-Control: The GPS functionality can be switched on and off as needed. The "calibrate" button allows setting the origin of the current scene to the current position. This can be useful for situations with inaccurate GPS positioning.

5- Info: In this area information about the current position of the device is shown. GPS latitude, longitude and horizontal accuracy as well as the current position of the camera in the scene are updated frequently to give additional feedback.

4.4.3. Discussion

It was possible to build a prototype application suitable for architectural on-site visualisation using Augmented Reality. The system used sensor based AR as a guidance system and natural feature tracking with the available facades in the environment for the actual visualisation. This allowed positioning of 3d content on top of a facade or between two existing buildings while also considering occlusion by those buildings. During the implementation and evaluation process a number of observations were made.

A very important factor for the quality of a facade for tracking seems to be the amount of glazing. While traditional facades seem to work quite well as long as they show some level of non repetitive patterns it is basically impossible to use glass facades for that purpose as the reflections change constantly according to the current position. In general, tracking results were quite good with diffuse, sufficient lighting but problems came up when too many shadows were visible on the facade. This could probably be avoided by generating multiple trackables for different lighting situations for each facade. A long term evaluation showed that trees and bushes might cause tracking problems if they partly conceal the facades and are not cut back properly. Another issue is the use at night which is mainly based on hardware limitations of the currently used cameras and cannot be overcome at the moment.

4.5. Urban Cultural Heritage Information System

4.5.1. Technical Implementation

As Michaelerplatz is entirely surrounded by buildings the initial idea was to use all of the facades to achieve CV based tracking in every direction. The first tests with this implementation however showed that even though many of the facades around the square delivered sufficient tracking capabilities they were by far not perfect and interruptions occurred frequently. In order to reduce the impact on the user experience sensor based tracking was implemented as a backup in case CV based tracking is not available.

Considering that a typical user would be in a stand up position holding the device about 1.5m above the ground, it proved to give a more pleasant experience to set the y-value (height) of the camera to that value resulting in a system with just 5 DOF for the CV based AR. As long as facades in the surrounding are recognized and CV based tracking is available the user can move freely on the square. As soon as the tracking is lost the position of the camera is considered as static and the accelerometer or if available the gyroscope is used in a sensor based approach in order to keep track of the user's rotation in a system with just 3 DOF. Convincing AR at that point is still possible as long as the user does not change his position. Once the user changes his position without being in vision-based AR mode the application has to recognize one of the facades again in order to work as expected.

The spatial layout of the 16 single trackables, which were joined to one multitarget enclosing the whole Michaelerplatz, was again based on the MZK. While all referenced facades contribute to the positioning, the final application tests showed that only some of them were actually suited for initial detection and therefore to start the vision-based AR. Additionally an almost frontal position (+/- 20°) and a distance of at least 15m to these facades were usually necessary in order to recognize them. Considering those restrictions Fig. 4.21 shows that computer vision-based AR could still be started from almost everywhere on the square.

Different types of content were referenced to this surrounding model and placed either in direct spatial relation to one of the facades or in their position on the square (Fig. 4.22).

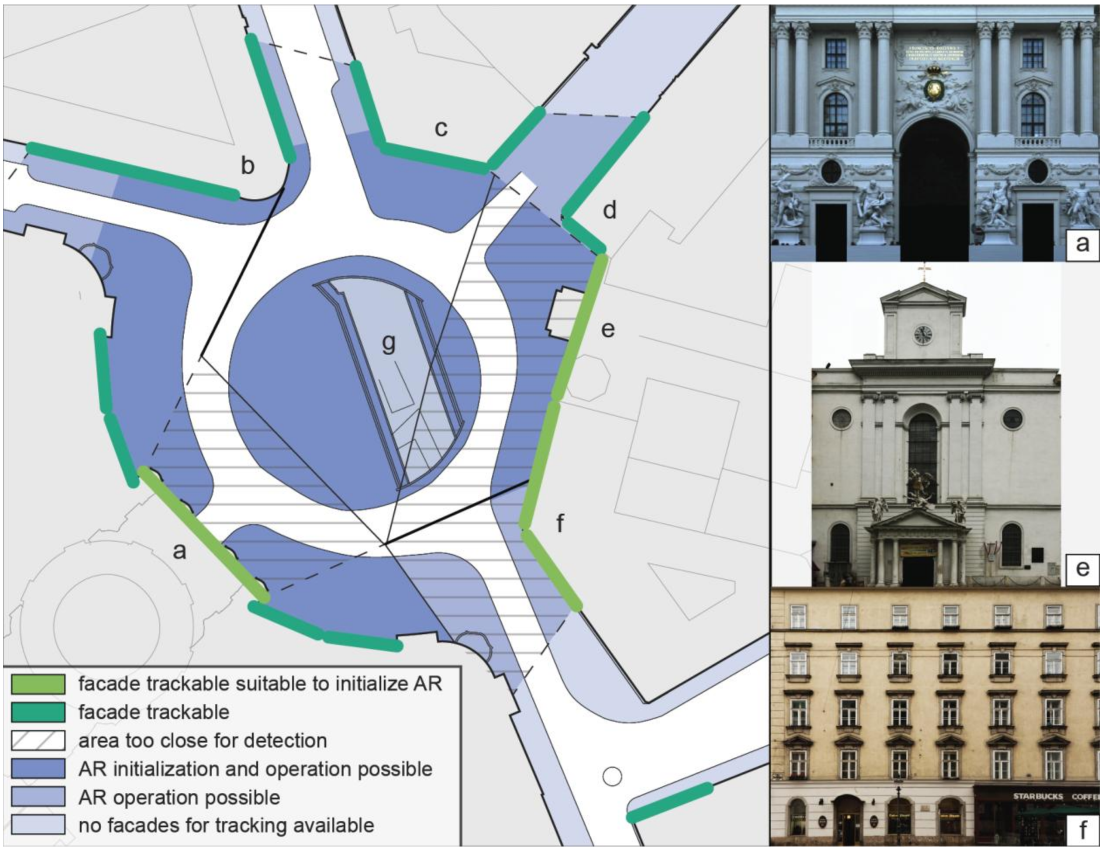

Fig. 4.21 Trackable layout and AR capabilities at the site (based on data from: Stadt Wien - ViennaGIS, www.wien.gv.at/viennagis/)

Fig. 4.22 Layout of referenced facades, 2D-tags and 3D models

4.5.2. The Application in Use

At the beginning an introduction screen is shown to the user asking him to initialize the AR functionality by pointing the devices camera at one of the previously described facades. As soon as one of these facades is recognized by the application, which in most tests took between 2 and 10 seconds, the system is running and the user can start to explore his surroundings. The user can see different types of information, texts, images, videos, web pages and 3d models that are related to the historical development of the square, indicated by info tags representing each category (Fig. 4.23). While the user moves and typically rotates around to see all the available information the tags are shown at their correct position inside the square (Fig. 4.24).

Fig. 4.23 Info tags for different content – from left to right: text, image, video, website, 3d model

Fig. 4.24 Running application showing 2d info tags

To find out more about one of the tags the user can point the device towards a particular marker. By bringing it into the centre of the monitor the appearance of the marker will change to a preview of the available content if applicable. In addition to that further information about the item is displayed in an info-box at the lower end of the screen (Fig. 4.25, left). The user can then either click on that preview or directly on a tag to get access to the whole information. Depending on the tag the content is displayed in different ways. While texts and images are displayed in pop-up windows (Fig. 4.25, right) videos can either be played by the devices integrated media player or they can be streamed from the internet using a platform like YouTube or Vimeo. Web pages typically linked to Wikipedia or other articles are being opened in the standard web browser. For 3D models there has to be differentiated between two different types. Findings from the site are opened in a new window allowing the user to further investigate it using standard gesture controls to rotate it as well as to zoom in and out. Models such as building reconstructions on the site will be shown as part of the augmentation over the camera image. This is the most critical case and requires precise results from the tracking system to place the model at the right spot (Fig. 4.26).

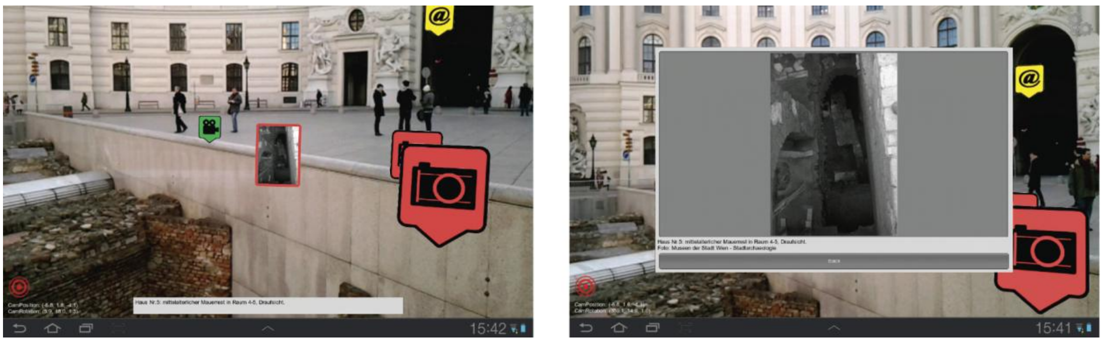

Fig. 4.25 Screenshots of the application showing image preview (left) and detailed view (right) (Image Source: Museen der Stadt Wien – Stadtarchäologie)

Fig. 4.26 **Application showing 2d info tags and 3d content**

4.5.3. Discussion

It was possible to implement a solution for an archaeological on-site visualization in an urban environment using CV based tracking and a sensor based backup system. As a reference for the CV based system, 16 images of the facades surrounding the site of Michaelerplatz in Vienna were used and allowed to accurately position 2D and 3D content on the square.

While the tracking system in most cases worked quite well we encountered a number of issues that need further development. Alongside the general technical issues of outdoor AR already mentioned in chapter 4.4.3, temporary architecture and facade decoration changing their appearance in wintertime came up as unexpected problems. Furthermore, as mentioned above, the usability of facades as trackables greatly varies on their design and arrangement of elements not allowing the same quality and stability of CV based tracking all over the place. Nevertheless the results were promising and even though it is not yet a perfect system, using AR with facades as a reference for inner urban visualization of archaeological data seems to be a very promising approach and opens a wide range of new possibilities.

5. Conclusion

Visual communication plays an essential role in all stages of the architectural planning process and is nowadays supported by different media like physical scale models, computer generated perspective images or interactive applications. While they already cover a range of possibilities it is demonstrated how the combination and integration of Augmented Reality can improve their capabilities as well as the experience for the user severely. Even though the ideas for AR can be traced back to the late 1960s, the developments of the last 20 years and the recently increasing availability of smart phones and tablet computers make the technical devices necessary for such applications available to a wide user base. There is no ready-made software solution so far but a platform suitable for different kinds of AR visualisation can be built based on a combination of available tools including the game engine Unity and the vuforia AR SDK.

The three example projects in which this platform is used are designed to show how Augmented Reality can change and improve the ways architecture is visualised. The visualisation of a sky rise building in Manhattan demonstrates how AR can combine the advantages of a physical model with the possibilities digital visualisations offers by using CV based tracking referenced to a plan of the site. The application enables the user to comprehend the future project based on a 3D model and different visualisations of other aspects that were important during the design phase. The additional implementation of a guided mode makes it a well suited tool for client or jury presentations. While the possible aesthetics of a physical model are not questioned, an AR application of this kind can be a valid and in many aspects superior replacement making the time consuming and expensive process of producing a real model unnecessary.

A bridge connecting two existing buildings at Karlsplatz is used to introduce AR to the important area of contextual visualisation. AR makes it possible to challenge some of the problems of the currently used computer generated still image compositions. By taking the facades of existing buildings as a reference for a CV based tracking system a 1:1 scale visualisation right at the actual future project site

can be realised. Thereby the user is no longer limited to the restrictions set in an artificial still image, but he can perceive the site with the future project in a natural way. Due to the obvious limitations of such a system, like the necessity to be at the site to use it, it is not a suitable replacement for existing contextual visualisation methods but rather an extension to them.
While the first two examples deal with future projects the third one is dedicated to the field of urban cultural heritage where an AR based application is used to keep archaeological findings virtually on the site. Facades around the historic location of Michaelerplatz in combination with a sensor based backup system are used to precisely place tags indicating available content. This allows the user to find out more about the items found in previous excavations and the numerous structures hidden under the pavement.

Even though the current framework already offers a well suited basis for AR visualisations it is just a first approach to that matter and a number of improvements and future developments are likely to be necessary to make it an indispensable part of the architectural planning and visualisation process.

In table-top visualisation one of the most beneficial improvements could be an extension to the current network based guided mode. At the moment it is implemented as a hierarchical structure granting one person all rights whilst the others are passive observers. The next steps however could go towards a multiuser collaborative peer oriented platform to discuss and develop a project together based on AR and a virtual model. Such a platform would give every user the ability to communicate with his peers on different levels. Besides the current possibilities of sending commands from one device to trigger predefined actions on the others it should for example offer the possibility to share the screen content among all devices. Live drawing tools for annotations and remarks on necessary changes could be implemented and should also be visible to others when the screen is shared. Screenshots of such annotations could be directly saved to the tablet or a remote server for further consideration during the design process.
A second area of development is workflow related. While at the moment it is a

rather simple process to set up a basic AR application, the steps necessary to extract the needed data from the CAD software are an unnecessary burden. However by integrating the functionality directly into the CAD software it might be minimized to a one-click task as these programs already have all the necessary data available. For example if used correctly a BIM program generates a 3D model of the project while the plans are drawn. A plug-in could be used to automatically generate the necessary trackables and export the according 3D models. If desired even the layer structure of the CAD program can be exported with the 3D model allowing the user to change the visibility of particular layers at runtime. While such an automated system would just offer a standard set of interactions like switching layers on and off, the direct integration into the already used tools could increase the usability enormously and make it useable from early stages of the design process on.

Future systems for on-site visualisation could see improvements and additional functionalities. One of the most crucial steps for the quality of the final CV based position tracking is the trackable generation and layout. Every little mistake during the process of taking the pictures, post processing until the final layout, decreases the reliability of the system severely. This whole procedure could however be replaced by using programmes for photogrammetric reconstruction. A freely available example for this is 123D Catch by Autodesk (@123D catch). These programmes have the ability to generate a fully textured 3D model of a certain object or building based on a series of images showing it from different viewing angles. Using such a model as reference in combination with a CV based AR system that is capable of real 3D tracking might be a far superior system in terms of tracking quality and in terms of usability.

Another way to improve on-site applications is by implementing networking functionality in combination with a content server. Among other features this would allow maintaining the application remotely over the internet. Once started it could connect to a remote server to check for updated tracking information or for additional content to be displayed. A more ambitious use of internet connection is crowd based decision-making. Thereby the application could give citizens the possibility to see different project proposals and to vote for a favourite. Especially

for municipal projects this could be a great way to increase public involvement and awareness.
On the side of urban cultural heritage a network connection could be used to build an urban archaeology AR browser instead of an application for a specific site. Based on the coarse GPS location of the device the trackable information for the surrounding could be downloaded at runtime. This would have two big advantages. First it would allow the application to work in a larger environment like whole cities, if trackables are made available on the server. Second loading the trackables from a server at runtime makes it possible to easily extend or update them without installing an update to the application. At the same time the server could also provide the necessary information that should be displayed based on the location. Furthermore by storing the available information in a database with different categories of classification, a GUI could allow the user to filter and display only the kind of content he is interested in.

Finally the visual appearance of the final composition could greatly be improved by considering the real world light situation when rendering the virtual content. A basic approach to do so would be to use a virtual light source representing the sun in the 3D scene. This light could be positioned and oriented correctly based on the devices location, date and time. Furthermore real time weather information could be acquired from a web service in order to adjust light intensity as well as the appearance of shadows.

Augmented Reality is a field of research that offers very promising opportunities and a number of possibilities for further applications in the architectural domain. While architectural presentations are currently still dominated by analogue media, AR might be the right technology to make the step to a digital era.

Acknowledgements

First of all I want to thank my parents and my grandmother for their great support before and during this thesis. Special thanks go to Peter Ferschin, the main supervisor of this thesis for his valuable input, support and feedback throughout the thesis. Furthermore I thank Paola Otero for her help with the photos and illustrations as well as for proofreading the work.

In addition a number of organisations and university institutes contributed to this project and were important for its success. Financial support was granted by the Vienna University of Technology through a scholarship. The Institute of History of Art, Building Archaeology and Restoration made their Tilt-Shift lens available and the Institute of Geodesy and Geophysics provided the GPS measuring system. Furthermore the “Stadtarchäologie Wien” provided the archaeological information about the historical site of Michaelerplatz and the team at Qualcomm Research Austria proved to be a valuable partner for discussions and feedback on the visual tracking system.

References

Azuma, R.T., 1997. A survey of augmented reality. *Presence-Teleoperators and Virtual Environments*, 6(4), pp.355–385. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.35.5387&rep=rep1&type=pdf [Accessed August 8, 2011].

Belcher, D. et al., 1997. ARchitectureView An Augmented Reality Interface for Viewing 3D Building Information Models. *Human-Computer Interaction*, pp.561–568.

Caudell, T.P. & Mizell, D.W., 1992. Augmented reality: an application of heads-up display technology to manual manufacturing processes. *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences*, pp.659–669 vol.2. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=183317.

Drosdol, J., Kieferle, J. & Wössner, U., 2003. The Integration of Virtual Reality (VR) into the Architectural Workflow. In *21th eCAADe Conference Proceedings*. Graz (Austria), pp. 25–28.

Feiner, S. et al., 1997. A touring machine: prototyping 3D mobile augmented reality systems for exploring the urban environment. In *Digest of Papers. First International Symposium on Wearable Computers*. Cambridge, MA , USA: IEEE Comput. Soc, pp. 74–81. Available at: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=629922 [Accessed November 25, 2011].

Kato, H. et al., 2000. Virtual object manipulation on a table-top AR environment. In *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*. IEEE, pp. 111–119. Available at: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=880934 [Accessed October 18, 2011].

Kato, H. & Billinghurst, M., 1999. Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In *Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*. IEEE Comput. Soc, pp. 85–94. Available at: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=803809 [Accessed October 18, 2011].

Krause, H., 2007. Von der Straßenkreuzung zum Platz - Die Geschichte des Michaelerplatzes vom Mittelalter bis ins 20. Jahrhundert. In Magistrat der Stadt Wien - Stadtarchäologie, ed. *Fundort Wien 10/2007. Berichte zur Archäologie*. Vienna: Phoibos Verlag, Wien, pp. 4–42.

Milgram, P. & Kishino, F., 1994. A taxonomy of mixed reality visual displays. *IEICE Transactions on Information and Systems E series D*, 77(12), pp.1321–1321. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.102.4646&rep=rep1&type=pdf [Accessed August 8, 2011].

Phan, V.T. & Choo, S.Y., 2010. Using AR for real-time crosscheck of ventilator ducts at worksite. In *Proceedings of the 9th ACM SIGGRAPH Conference on Virtual-Reality Continuum and its*

Applications in Industry - VRCAI '10. New York, New York, USA: ACM Press, p. 293. Available at: http://dl.acm.org/citation.cfm?id=1900179.1900241 [Accessed March 7, 2012].

Ranseder, C. et al., 2011. *Wien Archäologisch, Band 1, Michaelerplatz - Die archäologischen Ausgrabungen* 2nd ed., Vienna: Stadtarchäologie Wien.

Reitmayr, G. & Drummond, T., 2006. Going out: robust model-based tracking for outdoor augmented reality. In *2006 IEEE/ACM International Symposium on Mixed and Augmented Reality*. IEEE, pp. 109–118. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4079263 [Accessed January 15, 2012].

Rekimoto, J., 1998. Matrix: A Realtime Object Identification and Registration Method for Augmented Reality. In *Proceedings. 3rd Asia Pacific Computer Human Interaction (Cat. No.98EX110)*. IEEE Comput. Soc, pp. 63–68. Available at: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=704151 [Accessed October 29, 2011].

Shiratudd, M.F., Kitchens, K. & Fletcher, D., 2008. *Virtual Architecture - Modeling and Creation of Real-Time 3D Interactive Worlds*, Lulu.com. Available at: http://itunes.apple.com/us/book/virtual-architecture/id478871292?mt=11.

Stalla, R., 2008. "VO Architektur- und Kunstgeschichte / 257.072", Vienna Univrsity of Technology.

Sutherland, I.E., 1968. A head-mounted three dimensional display. *Proceedings of the December 9-11, 1968, fall joint computer conference, part I on - AFIPS '68 (Fall, part I)*, p.757. Available at: http://portal.acm.org/citation.cfm?doid=1476589.1476686.

Thomas, B. et al., 2000. ARQuake: an outdoor/indoor augmented reality first person application. *Digest of Papers. Fourth International Symposium on Wearable Computers*, pp.139–146. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=888480.

Thomas, Bruce, Piekarski, Wayne & Gunther, B., 2000. Using Augmented Reality to Visualise Architecture Designs in an Outdoor Environment. *International Journal of Design Computing*, 2. Available at: http://www.tinmith.net/papers/thomas-dcnet-1999.pdf.

Vlahakis, V. et al., 2002. Archeoguide: An Augmented Reality Guide for Archaeological Sites. *IEEE Computer Graphics and Applications*, 22(5), pp.52–60. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1028726.

Wagner, D. & Schmalstieg, D., 2003. First steps towards handheld augmented reality. In *Seventh IEEE International Symposium on Wearable Computers, 2003. Proceedings.* IEEE, pp. 127–135. Available at: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1241402 [Accessed January 15, 2012].

Web Sources

123D catch	http://www.123dapp.com/catch
3D on	http://www.3don.co.uk/
Android	http://source.android.com/index.html
Archicad	http://www.graphisoft.com/products/archicad/
ARQuake	http://wearables.unisa.edu.au/projects/arquake/; visited 13.01.2012
ARToolKit	http://www.hitl.washington.edu/artoolkit/; visited 15.01.2012
DimensionEngineering	http://www.dimensionengineering.com/accelerometers.htm; visited 05.01.2012
Gyrodroid	http://u3d.as/content/prefrontal-cortex/gyro-droid/2aR; visited 17.08.2012
Harvesine	http://www.movable-type.co.uk/scripts/latlong.html; visited 05.01.2012
HiddenCreative	http://hiddenltd.com/blog/augmented-reality-floor-plan-visualiser; visited 25.09.2012
ICG TU Graz	https://www.icg.tugraz.at/~daniel/HistoryOfMobileAR/; visited 13.01.2012
IDC	http://www.idc.com/getdoc.jsp?containerId=prUS23371312; visited 31.07.2012
IKEA	http://augmentedblog.wordpress.com/2012/07/24/ikea-2013-catalog-has-augmented-reality/; visited 25.09.2012
iOS	http://www.apple.com/ios/
Junaio	http://www.junaio.com/; visited 07.03.2012
Layar	http://www.layar.com/
MagicPlan	http://www.sensopia.com/english/index.html; visited 25.09.2012
metaio Mobile SDK	http://www.metaio.com/software/mobile-sdk/; visited 01.08.2012
MySQL	http://www.mysql.com/
ProjectGlass	https://plus.google.com/+projectglass/posts and
	http://en.wikipedia.org/wiki/Project_Glass; visited 20.09.2012
Stadt Wien	http://www.wien.gv.at/stadtentwicklung/stadtvermessung/geodaten/mzk/; visited 04.10.2011
String AR SDK	http://www.poweredbystring.com; visited 01.08.2012

Tinmith	http://www.tinmith.net/backpack.htm; visited 13.01.2012
Total Immersion	http://www.t-immersion.com/products/dfusion-unity/dfusion-unity; visited 03.08.2012
Unity3D	http://unity3d.com/
vuforia AR SDK	http://developer.qualcomm.com/dev/augmented-reality; visited 01.08.2012
Wiki_aGPS	http://en.wikipedia.org/wiki/Assisted_GPS; visited 05.01.2012
Wiki_Game	http://en.wikipedia.org/wiki/Game_engine; visited 01.08.2012
Wiki_GameList	http://en.wikipedia.org/wiki/List_of_game_engines; visited 02.08.2012
Wiki_HeadUp	http://en.wikipedia.org/wiki/Head-up_display; visited 05.01.2012
Wiki_HMD	http://en.wikipedia.org/wiki/Helmet_mounted_display; visited 27.07.2011
Wiki_MagDecl	http://en.wikipedia.org/wiki/Magnetic_declination; visited 05.01.2012
Wiki_Magnet	http://en.wikipedia.org/wiki/Magnetometer; visited 05.01.2012
Wiki_TiltShift	http://en.wikipedia.org/wiki/Tilt-shift_photography#Shift; visited 30.09.2011
Wikitude	http://www.wikitude.com/en/
Wolfram	http://demonstrations.wolfram.com/Gyroscope/; visited 05.01.2012

List of Figures

Abbrevations

aGPS	Assisted Global Positioning System
AR	Augmented Reality
BIM	Building Information Modelling
CV	Computer Vision
dGPS	Differential Global Positioning System
DOF	Degrees Of Freedom
FPS	Frames Per Second
GPS	Global Positioning System
GUI	Graphical User Interface
HMD	Head-Mounted Display / Helmet-Mounted Display
HUD	Head Up Display
MARS	Mobile Augmented Reality System
MR	Mixed Reality
QCAR	Qualcomm Augmented Reality
SDK	Software Development Kit
TMS	Qualcomm Target Management System
VR	Virtual Reality