Pers Ubiquit Comput (2012) 16:623–630
DOI 10.1007/s00779-011-0430-0

ORIGINAL ARTICLE

Sketching up the world: in situ authoring for mobile Augmented Reality

Tobias Langlotz · Stefan Mooslechner · Stefanie Zollmann · Claus Degendorfer · Gerhard Reitmayr · Dieter Schmalstieg

Received: 30 January 2011 / Accepted: 30 May 2011 / Published online: 27 July 2011

Abstract We present a novel system allowing in situ content creation for mobile Augmented Reality in unprepared environments. This system targets smartphones and therefore allows a spontaneous authoring while in place. We describe two different scenarios, which are depending on the size of the working environment and consequently use different tracking techniques. A natural feature-based approach for planar targets is used for small working spaces whereas for larger working environments, such as in outdoor scenarios, a panoramic-based orientation tracking is deployed. Both are integrated into one system allowing the user to use the same interaction for creating the content applying a set of simple, yet powerful modeling functions for content creation. The resulting content for Augmented Reality can be shared with other users using a dedicated content server or kept in a private inventory for later use.

Keywords Mixed and augmented reality · Content authoring

T. Langlotz (✉) · S. Mooslechner · S. Zollmann · C. Degendorfer · G. Reitmayr · D. Schmalstieg
Institute for Computer Graphics and Vision, Graz University of Technology, Graz, Austria
e-mail: langlotz@icg.tugraz.at

S. Mooslechner
e-mail: mooslechner@icg.tugraz.at

S. Zollmann
e-mail: zollmann@icg.tugraz.at

C. Degendorfer
e-mail: degendorfer@icg.tugraz.at

G. Reitmayr
e-mail: reitmayr@icg.tugraz.at

D. Schmalstieg
e-mail: schmalstieg@icg.tugraz.at

1 Introduction

Augmented Reality (AR) is a term describing a view of a physical real-world environment whose elements are augmented by virtual computer-generated imagery [1]. In the late 1990s, first experiments were conducted on presenting geo-referenced content using Augmented Reality as interface. The Touring Machine [4] introduced mobile outdoor AR and showed a campus tour guide.

Even with the limited possibilities of this first prototype, the power of in situ presentation of geo-referenced information became apparent. The emergence of the World Wide Web as a mass phenomenon around the same time prompted Spohrer in his seminal paper to suggest the *WorldBoard* [16], a combination of distributed online information systems and geo-referenced indexing. Information could be published like in the Web, but indexed by a geographic position rather than by a symbolic URL. He explicitly highlights to use Augmented Reality as interface to present the published information to the user. Unfortunately, his vision has not been fully realized yet, likely because key technologies were not mature enough.

Nevertheless the first commercial AR applications like Wikitude[1] were released to the market, which shows the potential of mobile AR. Wikitude, Layar[2] and other AR platforms form the genre of so-called AR-browser applications, as their main goal is to browse information, which are visually overlaid over the real world. These AR applications were made with the mass market in mind and target millions of users. Thereby solving technical problems like tracking and visualization are only some important aspects for mobile AR applications. Especially for the end user, the availability

[1] http://www.wikitude.org.

[2] http://www.layar.com.

of content has a high significance. This fact is widely known as the content problem and addressed in the research for AR authoring toolkits, allowing the users to create content for AR. But so far most applications are desktop based and were designed for professional users.

In the following, we want to present our work on in situ AR authoring and content distribution, which targets an inexperienced audience by allowing them to create content for Augmented Reality. This will allow the users to actively participate in the content creation process by creating content in an AR environment, which can be useful among others for:

- Sketching of annotations and 2D drawings, which can be used to place or highlight information, which are displayed using Augmented Reality.
- Creation of geo-referenced virtual duplicates of real objects, e.g., buildings, which can later be used in different visualization systems, such as 3D data viewer. Furthermore, virtual duplicates can be placed in different environments at new locations or can be stored as 3D model for inventory.
- Creation of 3D models that can be used in AR-based games or for phantom rendering for occlusion handling.

Requirements for our system are that it works in situ that the system does not need a prepared environment, e.g., printed marker or existing features databases and allows spontaneous content creation while in place.

In this paper, we will start by introducing existing work focusing on authoring tools for AR as well as content creation for AR. Later, we describe the concept of *Augmented Reality 2.0* (AR 2.0) as the next evolution of Augmented Reality, which is based on an active participation of users in the content creation process. We present why new approaches for in situ authoring in AR are important in an AR 2.0 system before we start to present our system, which is based on three parts: tracking the camera in the scene, modeling the content and content providing.

2 Related work

Research of content authoring has a long tradition in augmented reality. There are a dozen of desktop-based authoring solutions, which extend existing 3D modelers like Maya[3] and Blender.[4] DART [11], for example, is one of the best-known authoring toolkits for AR. It comes as a plug-in for Macromedia Director targeting professional content producers.

As DART does not allow in situ authoring further research was conducted for systems allowing authoring of content while in place. OutlinAR presents an approach for in situ modeling [3]. With OutlinAR, the user is able to build wireframe models by using a tracked handheld camera mouse, which is plugged into a standard computer.

Hengel et al. [5] presented an approach, which is based on a SLAM (Simultaneous Localization And Mapping) tracking system presented by Klein and Murray [8]. They use real-time camera tracking, and high-level automated image analysis, for an immersive modeling process, which generates 3D models of real objects. While this system allows spontaneous authoring of AR content on desktop computers, it is computationally too expensive to run on a phone. Although Klein and Murray have ported their SLAM approach to the iPhone, they state that it has some practical limitation compared to the PC-based version [9]. Especially, they stated that it works only for small working spaces and is reduced in robustness. That leads to the assumption that as of now it is not possible to extend the system of Hengel et al. to an in situ authoring tool capable for indoor and outdoor use on current generation smartphones. Furthermore, their system supports the user by reconstructing existing 3D shapes but it does not improve the modeling process in the case of non-existing content.

Simon presented an approach for modeling in an outdoor environment [15]. He identifies parallel lines and other information in the image to establish a coordinate system. Based on that, he can create geometry, which can also be used for tracking.

Pikarski et al. [12] proposed a 3D constructive solid geometry approach for constructing models within the Tinmith AR system, while Baillot et al. [2] introduced a more CAD-like interface for creating content in AR.

Despite these existing solutions for creating AR content within desktop/notebook applications only little investigations have been done in the field of content authoring for mobile devices such as smartphones as it is needed for an AR2.0 strategy. In general, we can distinguish cross-platform authoring (at the desktop) and in situ authoring using smartphones. Only few of the latter exist. Wang et al. [20] presented cross-platform content authoring conducted on desktop PCs and uses a PC simulating the behavior and the interface of a smartphone.

One of the few examples of in situ authoring conducted by using a smartphone is presented by Henrysson et al. They enable the user to build a simple scene by using a smartphone as interaction device to interact with augmented Lego blocks [6]. The main drawback for this authoring approach is that only pre-built models, in this case Lego Blocks, can be placed in a real-world scene. The authoring options are limited to translating and rotating the pre-built models. Thus, Henrysson's [7] approach allows

[3] http://autodesk.com.

[4] http://www.blender.org/.

not for authoring complex indoor and outdoor scenes. Later work showed how to manipulate an augmented mesh using a smartphone.

However, none of the above solutions has been designed with a truly large user base of thousands or ten thousands users in mind using mobile phones as their primary device for AR content and content creation. This is particularly important, as in the spirit of AR 2.0 every consumer of AR content is potentially also an author. Furthermore, the existing solutions for smartphones are only designed for small working environments such as desktop surfaces and require a prepared environment. Thus, they are not suitable tools for user-centered content creation. To overcome these limitations, the idea of our prototype is to implement a full pipeline including a backend for providing datasets used for tracking and the authored content that can be also used for further research on content providing for mobile AR. In the following, we want to present our approach for creating content for AR using smartphones, which tries to tackle some of the presented drawbacks of existing solutions.

3 Introducing AR 2.0

Assuming millions of people using mobile AR applications, the amount of content should reflect this huge user base and cannot only be created by a few professional content creators.

Fortunately, the recent development of the World Wide Web showed how to create content for millions of users: User participation. We think this is transferable to the described content problem in AR, which implies that the only solution is to allow the user to participate in such a way that they can create their own content rather than only using content created by a small number of professional modelers. The trend of making users to content creators and not only consumers is one of the main characteristics of the Web 2.0. Following this, we defined AR 2.0 as the next step of Augmented Reality [13] (sometimes also referred to as *Social AR*) where users can create and share content (see Table 1).

If AR applications are going to be deployed on a massive scale in an AR 2.0 approach, there are several key areas of technology that are needed:

1. A low-cost platform that combines AR display, tracking and processing.
2. Mobility to realize AR in a global space.
3. Large-scale AR tracking solutions, which work in real time.
4. Easy to use authoring tools for creating AR content.
5. Backend infrastructure for the distribution of AR content and applications.

1 and 2 are solved by using smartphones as platform. In the following, we want to present our current work, which focuses on a prototype of a mobile authoring solution for non-professionals. Furthermore, we will present the used tracking solution and the backend infrastructure to distribute the content. All this together allows inexperienced users the authoring of AR content while in place.

4 Tracking for mobile interaction

There are different scales in which in situ authoring solutions for smartphones could work. In this work, we differentiate between small working environments like desktop environments and larger environments like outdoor scenes. Small workspaces describe scenarios where the working volume is mostly bound to one specific object, which can nearly always be seen in the phones camera view.

Larger working environments describe scenarios where the environment around the user contains many objects, which cannot always be seen. Both scenarios require different tracking techniques; therefore, we treat them separately in this paper.

Table 1 Comparison of Web 2.0 and AR 2.0 characteristics

Web 2.0 characteristics	AR 2.0 characteristics
Large number of users and web sites (already true for Web 1.0)	Large-scale in number of users as well as working environment
No clearly visible separation between accessing local data and remote data	No clearly visible separation between visualizing local data and remote data
Applications running in a browser behave like local applications, encouraging the user to interact with them	Applications locally running on the device can transparently download modules or new features from remote servers
A huge amount of non-technical people retrieve data and contribute or modify it as well	Non-expert users can create or update the AR content at specific locations
Information from different sources can be combined and create a new value-added application, in so-called *Mashups*	*Mashups* which access data from sources like traditional web services and combine them with AR content to display them in 3D space

4.1 Tracking infrastructure for small working environments

We decided to use a natural feature-based approach for tracking the smartphone in case of a small working environment. Therefore, we employed the natural feature-based tracking system presented in [19] as this system works in real-time on smartphones while being robust against partial occlusions. This tracking approach tracks the position of the phone using a feature database containing features of an object in the current camera view. Thus, it requires a feature database of the currently tracked target to be available on the phone.

As we assume an unprepared environment, we created a tool, which allows the user to create this feature database using her smartphone. Therefore, the user has to point her phone toward the object she wants to use as tracking target, which should have a planar surface. After sketching the outline of the planar surface, the system starts to analyze the selected region concerning the amount of features and their distribution. Furthermore, the system analyzes if the target was captured frontal by checking if the target has a rectangular shape (see Fig. 1). We allow small errors as they can be corrected by reprojecting the area into a rectangle, but larger errors produce problems creating the feature description. A last check calculates the size of the object in the camera frame alarming the user if the object is too small to create a feasible tracking database from it.

Once all requirements toward the target are fulfilled, the system sends the picture of the target to a server and the server creates the natural feature database. Optionally the client running on the smartphone can send the GPS address of the current position (if available) to geo-reference the newly created feature database.

Once created, the natural feature database is then transmitted back to the client. Furthermore, it is stored on the server optionally using the transmitted GPS position as a filter in case another client is later asking for a tracking target on that position.

This means that the user running her AR application on the phone can create a new natural feature database for tracking using her smartphone or query the server for already existing ones. Both assures that the phone is tracked in 6DoF in a previously unprepared environment with minimal effort and minimal delay as the creation of the natural feature database takes 2 s in average and the transmission takes between 5 and 10 s assuming a typical database size of less than 200 k and a 3G internet connection.

4.2 Tracking infrastructure for large working environments

As mentioned previously, we define a large working environment by having an environment with many objects, which are not always in the current camera view. The typical scenario for using such a large working environment is authoring in outdoor environments. We therefore replaced the tracking for small working environments requiring a static tracking object with an approach assuming a static position and allowing only rotational movements. In outdoor environments, these assumptions are appropriate, since typically the user will stay at one position and explore the environment by rotating the phone.

Tracking in outdoor environments has very challenging requirements, such as an accurate registration to a given coordinate system, robustness and real-time processing. To satisfy these requirements, we used a sensor-fusion-based approach [18] that combines panorama-based vision tracking, accelerometer and compass data. Thereby, the magnetic compass and the accelerometers provide absolute orientation measurements in the earth coordinate system, while the panorama-based vision tracking provides relative orientation tracking.

Since the output of the accelerometers and the compass may suffer from jitter and external influences, visual tracking provides stable but relative orientation measurements. Therefore, the panorama-based tracking creates

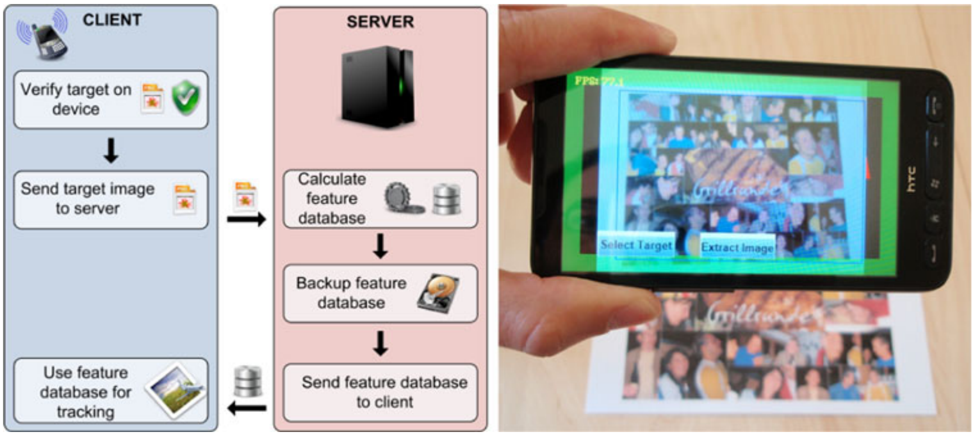
CLIENT
Verify target on device
Send target image to server
Use feature database for tracking
SERVER
Calculate feature database
Backup feature database
Send feature database to client
Select Target
Extract Image

Fig. 1 Showing the online creation of a feature database, which can be used for natural feature-based tracking. *Left* Illustration of the needed steps to create a feature database using a remote server. *Right* Showing the client for creating the feature database. The *green* display border indicates that the outlined object in the current camera fulfills all criteria for a good target for natural feature-based tracking

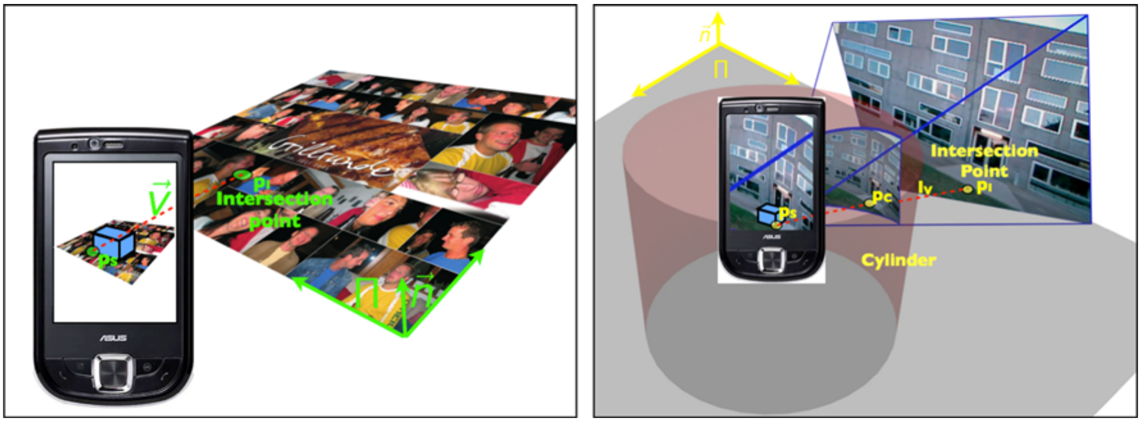

Fig. 2 Illustrating the tracking and estimation of the 3D coordinates of the current modeled object. *Left* Tracking the phone position in a small workspaces using natural feature-based tracking of a known planar surface. *Right* Tracking the phone position in a large environment using GPS and panorama-based vision tracking combined with sensors for tracking the orientation

panoramic maps online and tracks simultaneously from the map [17]. Natural feature mapping and tracking is used to generate the panoramic map of the environment from the live camera stream while using it for tracking at the same time.

To combine the accuracy and robustness of vision tracking with the absolute orientation of the inertial and magnetic sensors, the results of the panorama-based tracking and the sensors are fused in a Kalman filter-based framework.

As this approach only tracks rotational movements, we use the GPS-sensor of current generation smartphones to estimate the current user position.

5 In situ authoring

Existing authoring solutions address a specific user group. Most of them target professional content creators and media artists. The few targeting an inexperienced audience are desktop-only applications like BuildAR.[5] In this system, we target an inexperienced audience to create content for mobile Augmented Reality. The minimal requirement is that they can handle their phones (taking pictures, sending short messages). Therefore, we limited the functionality to basic, yet powerful functions to create 3D content. The functions comprise the generation of 3D primitives, such as cubes and cylinders, but also polygonal geometry, and the geometrical modification of the generated primitives, such as scaling and moving and changing the texture of these objects. Additionally, 2D annotations registered to the ground plane can be created.

5.1 Generating registered 3D primitives

To generate 3D primitives, the user starts to draw the 2D shape according to the footprint of the primitive by touching the screen. The 2D footprint of the object is generated on the ground plane.

In the case of a small tracked environment, the ground plane is the marker plane formed by the planar target used for natural feature tracking. To determine the 3D coordinates of the footprint, the intersection point p_I between the line given by the touched spot on the screen p_S and the viewing direction $\vec{v}$, and the marker plane Π is calculated (see Fig. 2 left).

In case of the large working environment, the ground plane is the physical ground. In this case, the intersection between a line given by the touched spot on the screen p_S and the viewing direction and the physical ground is determined by using the cylindrical representation of the panoramic map used for tracking. Therefore, the 2D point p_S is mapped from the screen space to a 3D point on the cylinder p_C (see Fig. 2 right). The intersection line l_V is then given by p_C and the coordinate of the origin of the camera. Intersecting l_V and the ground plane Π determines the 3D intersection point $\mathrm{p_I}$.

The calculated intersection points are then used as input to generate the footprint of the object in the 3D world space. After defining the 2D footprint on the ground plane, the user can extrude this shape to form a 3D object. Hence, the supported 3D models range from cubes, tubes and spheres to objects with arbitrary polygonal ground plane.

5.2 Modifying registered 3D primitives

After creating different 3D objects, the user may want to manipulate the object. Therefore, we integrate different

[5] http://www.buildar.co.nz.

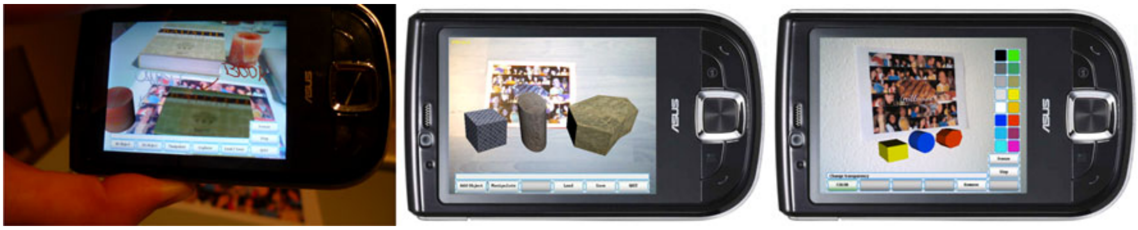

Fig. 3 Example showing created augmented 3D content and interface using the system configured for small working environments. *Left* Augmented scene with augmented duplicates of real objects (book and candle). *Middle* Augmented scene showing different object shapes with different applied textures. *Right* Augmented scene showing different objetcs with different colors and interface for selecting colors

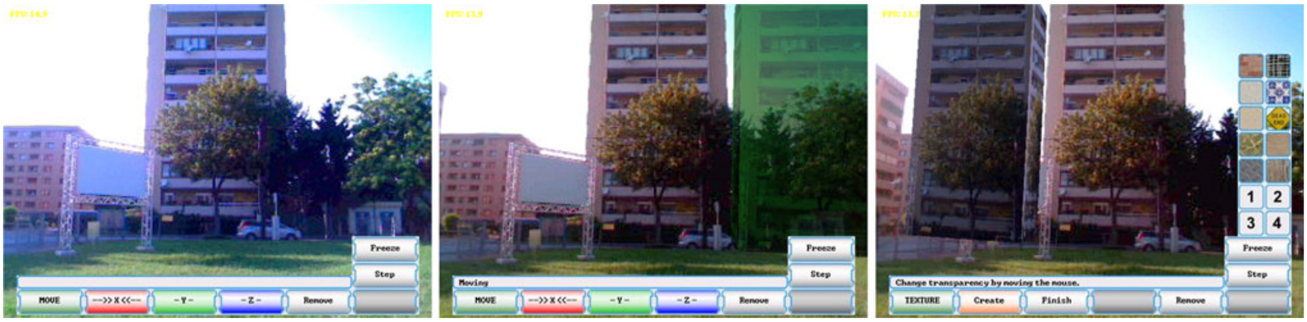

Fig. 4 Example showing the created augmented 3D content using the system configured for large working environments. *Left* Real scene as displayed in the camera view of the smartphone. *Middle* Augmented scene showing the created virtual object (highlighted in *green*) during placement operation. *Right* Final scene showing a created duplicate of an existing building augmented left to the real building

manipulation tools. The object can be translated, scaled and rotated by selecting the axis of transformation in the GUI and by sliding in the corresponding direction over the touchscreen.

5.3 Applying textures to the registered 3D objects

To enable the creation of more realistic models, we provide the functionality to assign different colors to the objects as well as to texture the object (see Fig. 3). To assign a new color to the object, the user can select from a color palette as shown in Fig. 3 (right). To assign a new texture, the user can select from a set of predefined textures, which can be mapped to the object (Fig. 3 middle). Furthermore, it is possible to create new textures by selecting a region of the current camera image, which can later be used as texture and assigned to the objects (Fig. 3 left). This is especially useful if the goal is to create virtual duplicates of real objects (see Fig. 4). In this case, it can be also convenient to immediately use the corresponding physical texture of the object. This can be achieved by back projecting the created 3D object into the camera image and using the projected area of the camera image as texture. This can be comfortable if objects have to be created fast, because no texture cutout is required.

5.4 Generating registered 2D objects

In addition to the creation of 3D objects, the system also supports a variety of 2D tools and objects to draw or annotate the environment. They comprise:

- Freehand
- Line
- Box
- Circle

Similar to basic paint programs, the user can choose between pencils, brushes or a graffiti spray tool. In order to create registered 2D objects, the objects are not created in the camera space, but in the world space. In small working environments where objects are registered in relation to a natural feature tracking target, the created 2D objects are inserted in the plane of the tracking target. Therefore, the intersection point between the selected point in the camera image and the ground plane is determined analog to the creation of 3D objects and used as coordinate of the newly created 2D object.

In large working environments, the usage of the intersection point between the ray of the selected point on the camera view and the ground plane as input for 2D objects only allow the creation of objects that are attached to the earth surface. But typically, users want to attach 2D annotations in relation to the objects in the scene. Therefore, we create the 2D objects in relation to the panoramic map, similar to the approach of Langlotz et al. [10]. With this approach, the user is able to create 2D objects attached to each visible object in the scene. The disadvantage of this approach is that the 2D objects can only be viewed from positions not too far from the position where the objects where created, since they are registered to the panoramic map of this location.

5.5 Interacting with the mobile device

The creation and manipulation of the content is done directly on the mobile device. Since more and more mobile phones have no physical keyboard, all manipulations and authoring functions are controlled using a touch screen. This raises the problem how to accurately operate the device while pointing the camera to the designated spot.

We solved this problem by implementing a freeze mode. Once the user is in the position where she wants to start the authoring task, she can freeze the view by pressing a button on the touch screen. This simulates a fixed position by freezing the current camera frame and allows moving the device with no implication regarding the tracking. After a successful completion of the selected task, the user can unfreeze the view.

The whole system running on the smartphone is implemented using the C/C++ based Studierstube ES [14] as underlying framework. Studierstube ES is an AR framework specifically designed and optimized for mobile phones as target devices. This efficient implementation allows real-time performance for the tracking as well as rendering during the content creation process. While Studierstube ES is a cross-platform framework, this work was mainly tested on an Asus P565 (600 MHz) and a HTC HD2 (1 GHz) both running Windows Mobile.

6 Content providing

The idea of AR 2.0 raises the question of how to publish the created content once created to all target devices. Assuming millions of users with devices and massive amount of content, we have to select the displayed items wisely to minimize visual clutter (e.g., showing too much content on a small screen) and to preserve bandwidth (e.g., transmitting irrelevant content to the client).

To address these goals, we use a server infrastructure as backbone, which shares many techniques and features with a server hosting a Web 2.0 application. This server for hosting content for AR supports among others user accounts, tagging with meta information, geo-referencing of content with GPS coordinates, referencing content to a ground truth in 6 *Degrees of Freedom* (DOF) (marker, natural feature tracking database) and allows to query that stored content using this information. The server application is implemented using standard web frameworks such as Apache Tomcat together with a MySQL database. The content is made available via a web services as well as via a programming interface, which can be accessed using C/C++.

This system can easily be adapted to only download content of known friends (e.g., by using Facebook's friends list) or updating friends (e.g., via Twitter or Facebook) if new content is published.

We used this server to host more complex content types, which was created with our prototype for 3D in situ authoring described in Sect. 5. In this system, the generated AR content and all related files are submitted to the server and stored in a zip container as this content is more complex and can consist of many files like 3D models, textures, text information and a XML markup file to describe the content and add meta information. We developed our own XML markup language schema to express all relevant information and call it *Augmented Reality Markup Language* (ARML). The resulting XML markup file is also placed in the container and later used for indexing the content and inferring the relations concerning place, time, user etc. (see Fig. 5).

Beside the possibility of sharing the generated content with other users over a remote content server, each user has a private inventory. This allows placing created objects into the local inventory on the device for later use. Hence, the user is able to create an object (e.g., annotated with a texture of the current environment), pick it up and insert it into a scene at a different location.

7 Conclusion and future work

We presented a system for mobile AR authoring, which can be used for user content creation in an AR 2.0 manner. We

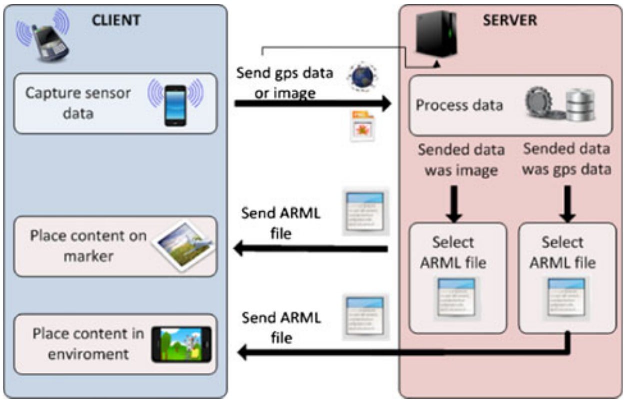

Fig. 5 Illustrating the dataflow for content providing using a remote server and a client with AR capabilities

showed how users could create content in place by utilizing two different approaches for tracking the device, which depend on the size of the working environment while not requiring any preparations. We combined this tracking with simple, but powerful authoring functions.

This allows the user to create new 2D as well as 3D content as well as to replicate existing 3D objects, which can be used for various AR applications like phantom rendering, games, art installations and social AR applications like AR browsers. We allow the user to sketch in AR using augmented graffiti allowing to color the environment or place annotations. Embedding a client–server approach to support content sharing between multiple users completes the system, which fulfills all aspect of an AR 2.0 environment.

During preliminary user tests, we identified the need for an improved interface, which we try to realize in a future version. While most of the users were able to handle the content creation process, they stated that they have to make too many actions in terms of the amount of finger moves, button clicks and clicks to create the content. Hence, we will revise the interface to decrease the number of buttons and user actions. For instance, we plan to support the user by using information from the current camera image such as edges and texture information as additional input for the content creation. Another possibility is to allow the picking of the object instead of switching through the objects to select the current active one.

Other future works include a user study to evaluate this system in term of usability as well as to integrate approaches for an image-based automatic reconstruction in case of existing geometry. We are also working on a desktop-based interface for accessing the content server to manipulate and improve content, which was created while in place. Furthermore, we are interested in cross-linking with other conventional online resources such as Wikipedia to create mash-ups of content, which is displayed using Augmented Reality as interface.

Acknowledgments This work was sponsored by the Christian Doppler Laboratory for Handheld Augmented Reality and by the Austrian Research Promotion Agency (FFG) under contract no. FIT-IT 820922 (SMART Vidente).

References

1. Azuma R (1997) A survey of augmented reality. Presence: Teleoper Virtual Environ 6(4):355–385
2. Baillot Y, Brown D, Julier S (2001) Authoring of physical models using mobile computers. In: Proceedings of the 5th IEEE international symposium on wearable computers, p 39
3. Bunnun P, Mayol-Cuevas WW (2008) OutlinAR: an assisted interactive model building system with reduced computational effort. In: Proceedings of the international symposium on mixed and augmented reality, pp 61–64
4. Feiner S, MacIntyre B, Höllerer T, Webster A (1997) A touring machine: prototyping 3D mobile augmented reality systems for exploring the urban environment. Pers Ubiquit Comput 1(4): 208–217
5. Hengel Av, Hill R, Ward B, Dick A (2009) In situ image-based modeling. In: Proceedings of the international symposium on mixed and augmented reality, pp 107–110
6. Henrysson A, Ollila M, Billinghurst M (2005) Mobile phone based AR scene assembly. In: Proceedings of the 4th international conference on mobile and ubiquitous multimedia, pp 95–102
7. Henrysson A, Billinghurst M (2007) Using a mobile phone for 6 DOF mesh editing. In: Proceedings of the 7th ACM SIGCHI New Zealand chapter's international conference on Computer-human interaction: design centered HCI, pp 9–16
8. Klein G, Murray D (2007) Parallel tracking and mapping for small AR workspaces. In: Proceedings of the international symposium on mixed and augmented reality, pp 1–10
9. Klein G, Murray D (2009) Parallel tracking and mapping on a camera phone. In: Proceedings of the international symposium on mixed and augmented reality, pp 1–10
10. Langlotz T, Wagner D, Mulloni A, Schmalstieg D (2010) Online creation of panoramic augmented reality annotations on mobile phones. IEEE Pervasive Computing 99 (preprint)
11. MacIntyre B, Gandy M, Dow S, Bolter JD (2004) DART: a toolkit for rapid design exploration of augmented reality experiences. In: Proceedings of the symposium on user interface software and technology
12. Piekarski W, Thomas B (2001) Tinmith-metro: new outdoor techniques for creating city models with an augmented reality wearable computer. In: Proceedings of the fifth international symposium on wearable computers, IEEE Comput Soc, pp 31–38
13. Schmalstieg D, Langlotz T, Billinghurst M (2010) Augmented reality 2.0. In: Coquillart S, Brunnett G, Welch G (eds) Virtual realities. Springer, Vienna, pp 13–37
14. Schmalstieg D, Wagner D (2008) Mobile phones as a platform for augmented reality. In: Proceedings of the IEEE VR 2008 workshop on software engineering and architectures for realtime interactive systems, pp 43–44
15. Simon G, Henri U, Inria P (2010) In situ 3D sketching using a video camera as an interaction and tracking device. In: Proceeding of the Eurographics 2010
16. Spohrer JC (1999) Information in places. IBM Syst J 38:602–628
17. Wagner D, Mulloni A, Langlotz T, Schmalstieg D (2010) Real-time panoramic mapping and tracking on mobile phones. In: Proceedings of the 2010 IEEE virtual reality conference (VR), pp 211–218
18. Schall G, Wagner D, Reitmayr G, Taichmann E, Wieser M, Schmalstieg D, Hofmann-Wellenhof B (2009) Global pose estimation using multi-sensor fusion for outdoor augmented reality. In: Proceedings of the international symposium on mixed and augmented reality, pp 153–162
19. Wagner D, Reitmayr G, Mulloni A, Drummond T, Schmalstieg D (2008) Pose tracking from natural features on mobile phones. In: Proceedings of the international symposium on mixed and augmented reality, pp 125–134
20. Wang Y, Langlotz T, Billinghurst M, Bell T (2009) An authoring tool for mobile phone AR environments. In: Proceedings of the New Zealand computer science research student conference 09 (NZCSRSC '09), pp 1–4