Mobile Augmented Reality System for Construction Site Visualization

Charles Woodward, Mika Hakkarainen

VTT Technical Research Centre of Finland

ABSTRACT

This paper describes our mobile Augmented Reality (AR) system for construction site visualization and interaction. The mobile AR system can be operated either stand-alone, or as a client-server solution scaling down to mobile phones and tablets based on our lightweight tracking solution. The system is validated in field tests, covering architectural AR visualization with photorealistic rendering effects, up to construction time applications at a real building site.

KEYWORDS: BIM, 4D, augmented reality, mobile devices.

1 INTRODUCTION AND RELATED WORK

The Architecture, Engineering and Construction (AEC) sector is widely recognized as one of the most promising application fields for Augmented Reality (AR). Building Information Models (BIM) and in particular the Industry Foundation Classes (IFC) data format are another main technology driver increasingly used for data sharing and communication purposes in the AEC sector. Combined with mobile Augmented Reality and time schedules, 4D BIMs could facilitate on-the-spot comparisons of the actual situation at the construction site with the building's planned appearance and other properties at the given moment. Shin and Dunston [1] evaluated 17 classified work tasks in the AEC industry. They concluded that eight of them (layout, excavation, positioning, inspection, coordination, supervision, commenting and strategizing) could potentially benefit from the use of AR. Additionally, related application areas would be communication and marketing prior to construction work, as well as building life cycle applications after the building is constructed.

A wealth of earlier work has been done around mobile AR and architectural building models by researchers, starting from pioneering work by Feiner et al. [2]. Among the first to address practical AEC applications, Schall et al. [3] presented a mobile handheld AR system for visualizing underground infrastructure. For further references see the thesis by Bezhadan [4]. However, not much work has been done to integrate mobile AR with real world building models, often containing millions of triangles and being hundreds of megabytes in size. Another topic seldom covered in previous literature is integrating the time component to mobile AR solutions. Goldparvar-Fard et al. [5] provide 3D reconstruction tools to compare the situation at construction site against 4D plans, but these are computed off-line (not mobile) from photographs taken of the site.

Our long term research goal has been to prove the technical validity of bringing real world BIM models to the construction site, for augmenting with lightweight mobile devices. Our work on mobile AR dates back to 2003 with the client-server implementation on a PDA device [6]. The next generation implementation [7] produced a marker-free UMPC solution by combining the building's location in Google Earth, the user's GPS position, optical flow tracking and user interaction for tracking initialization. This work lead to the first version of the current system architecture [8] to handle arbitrary OSG formats and IFC (instead of just Google Earth's Collada), 4D models for construction time visualization (instead of just 3D), and mobile feedback from the construction site to the design system ("augmented virtuality"). The system was further extended in [9] to cover more accurate map representations, mobile interaction, operation with data glasses, efficient client-server architecture, tracking methods, as well as discussion on photorealistic visualization for mobile AR.

This article gives an overall presentation of our software system, its background, current state and future plans. Among the most recent developments, we present the client implementation on mobile phones, based on a lightweight optical tracking solution, as well as the results of our field trials in different pilot cases, including application during the construction work, and comparing the visualization results with the appearance of a partially ready building.

2 SYSTEM OVERVIEW

This Section presents the general implementation of the system. The discussion is given mainly from functional point of view, while a more detailed discussion is provided in [9].

2.1 Software modules

Our system is divided into three parts; 4DStudio, MapStudio and OnSitePlayer. The Studio applications fulfill the authoring role of the system and are typically used at the office, while OnSitePlayer provides the augmented reality view and mobile feedback interface at the construction site. OnSitePlayer can be operated either as a stand-alone, or as a client-server solution, distributing heavy 3D computation to the OnSiteServer extension, and tracking and rendering to the OnSiteClient extension.

The tracking algorithms are based on our software library ALVAR – A Library for Virtual and Augmented Reality [10], and the OpenCV computer vision library. The GUI is built using the wxWidgets 2.8.10 framework. For rendering, the open-source 3D graphics library OpenSceneGraph (OSG) version 2.8.2, is used. The applications can handle all OSG supported file formats via OSG's plug-in interface (e.g. OSG's internal format, 3DS, VRML). The TNO IFC Engine3 [11] is used as a basic platform to process IFC building model files.

2.2 4DStudio

The 4DStudio application takes the building model (in IFC or some other format) and the construction project schedule (in MS Project XML format) as input. 4DStudio can then be used to link these into a 4D BIM. 4D IFC models defined with Tekla Structures can also be read directly by 4DStudio. Once the model has been defined, 4DStudio outputs the project description as an XML file.

Vuorimiehentie 3, Espoo, Finland
email: firstname.lastname@vtt.fi

For interaction, 4D Studio provides various tools to select elements for visualization, user definable color coding, clip planes, and viewing the model along the time line. Feedback report items generated with the mobile AR system can also be viewed with 4DStudio. Each item contains a title, a task description, a time and location of the task, and optionally one or several digital photos.

2.3 MapStudio

The MapStudio application is used to position the models into a geo coordinate system, using an imported map image of the construction site. The geo map can be imported from Google Earth, or for more accurate representations geospatial data formats like GeoTiff. The image import is done using the open source Geospatial Data Abstraction Library (GDAL).

The models are imported from 4DStudio, and can be any OSG compatible format or IFC format. The model can either be a main model or a so-called block model, which is used to enrich the AR view, or to mask the main model with existing buildings. The system can also add clipping information to the models, for example the basement can be hidden in the on-site visualization.

The user can position the models on the map either by entering numerical parameters or by interactively positioning the model with the mouse (see Figure 1). Finally, the AR scene information is stored as an XML based scene description, ready to be taken out for mobile visualization on site.
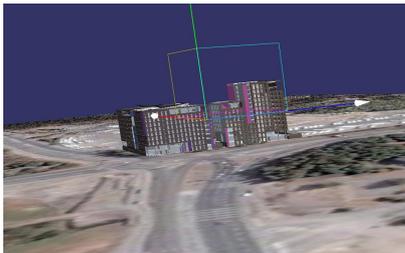

Figure 1. Building placed in geo coordinates with MapStudio.

2.4 OnSitePlayer

OnSitePlayer is launched at the remote location by opening a MapStudio scene description, or by importing a project file containing additional information. The application then provides two separate views in tabs: a map layout of the site with the models including the user location and viewing direction, and an augmented view with the models displayed over the real-time video feed.

The user is able to request different types of augmented visualizations of the model based on time, for example defining the visualization start-time and end-time freely, using clipping planes, and/or showing the model partially transparent to see the real and existing structures behind the virtual ones. OnSitePlayer is also able to store augmented still images and video of the visualization, to be later reviewed at the office.

As GPS positioning does not always work reliably or accurately enough (e.g. when indoors), we provide the user with the option to indicate his/her location interactively. The system presents the user the same map layout as used in the MapStudio application. The user is then able to zoom into the map and place the camera icon to the his/her currently know location. Additionally, the user's elevation from ground level can be adjusted with a slider.

Interactive means are also required as backup for the compass. In our implementation, known elements of the real world are marked in MapStudio as "placemarks". The mobile user then selects any of the defined placemarks with the "viewfinder" to initialize real time tracking (see Figure 2).

Figure 2. Mobile AR view showing viewfinder for the placemark, and building model augmented with OnSitePlayer.

With OnSitePlayer, the user can also create mobile feedback reports consisting of still images annotated with text comments. Each report is registered in the 3D environment at the user's location, camera direction, and moment in time. The reports are attached to the BIM via XML files and are available for browsing with 4DStudio, as explained above.

3 CLIENT-SERVER IMPLEMENTATION

Virtual building models are often too complex and large to be rendered with mobile devices at a reasonable frame rate. This problem is overcome with the client-server extension for the OnSitePlayer application. The client extension, OnSiteClient, is used at the construction site while the server extension, OnSiteServer, is running at the site office or at some other remote location. Data communication between the client and server can be done using either WLAN or 3G.

The client and server share the same scene description as well as the same construction site geospatial information. The client is responsible for gathering position and orientation information, but instead of rendering the full 3D model, the client just passes the user location and viewing direction to the server. The server uses this information to calculate the correct model view, which is then sent to the client for augmenting on the mobile device.

In our implementation, the view is represented as a textured spherical view of the virtual scene surrounding the user [9]. The sphere is approximated by triangles (icosahedron). Only those triangles that contain model information need to be transmitted to the client. The same sphere projection can be used as long as the user remains at the same location.

Our solution generally assumes that the user does not move about while viewing. This is quite a natural assumption, as viewing and interacting with a mobile device while walking would be quite awkward or even dangerous, especially on a construction site. The user is still free to rotate around 360/360° and view the entire sphere projection.

4 MOBILE PHONE IMPLEMENTATION

In the PC based client-server implementation, the client and server extensions were obtained by direct modifications to the OnSitePlayer application. With the mobile phone implementation this was not feasible due to the difference of platforms. Also, to create as lightweight solution as possible, we implemented a whole new client application for the Nokia N900 smart phone.

The mobile phone client still supports the network connection and data stream provided by the original server on the PC. The application framework is built using Qt SDK 1.0 and Qt Mobility. The rendering is done with OpenGL ES 2.0. The network connection is ad-hoc WLAN.

The functionality of our first mobile phone version is restricted to visualizing the architect's visualization models, without time component or other advanced features. User positioning is done using the integrated GPS module, without any user interaction. On

the other hand, the N900 does not have a compass so the user is responsible for defining the initial viewing direction.

All the user interactions are done via the touch screen. On the mobile phone we show all of the pre-defined viewfinder positions first in arbitrary direction. The user is then able to swipe the screen and choose the valid viewfinder(s). After locking the model in the correct position, the viewfinder images are removed from the view and feature based tracking is started (see Figure 3).

Figure 3. OnSiteClient running on N900

Model rendering is based on the sphere projection method, as described above. Downloading the sphere images from the server depends on the number of images required. New sphere initialization typically takes some 5 seconds, though in the worst case scenario (20 images, model all around the user) it takes up to 30 seconds. Alternatively, "hot spot" viewing positions can be defined at office using MapStudio. In this case the sphere images are stored beforehand in the OnSiteClient's scene description and no downloads or even connection to the server are required.

5 TRACKING

We have developed altogether three vision based tracking methods for different use cases. Two lightweight solutions were developed for the OnSiteClient application, one for PC and one for mobile phone. These solutions assume the user stands at one position, at least a few meters away from the target object, and explores the world by panning with the mobile device (camera). A separate solution was developed for the stand-alone OnSitePlayer on PC, allowing the user also to move freely while viewing. The PC based tracking solutions have been described in our previous article [9]. The light-weight solution for the mobile phone client is more recent, details to be published in future articles.

From performance point of view, the use of RIFF descriptors and FAST corners in the mobile phone implementation gives two clear benefits over the previous PC based solution. Firstly, detecting FAST corners is much faster than the previously used interest point detector of Shi and Tomasi. With a carefully optimized implementation we are able to reach a real-time performance of 30 FPS on the N900 mobile phone. Secondly, by tracking features using descriptor matching instead of the optical flow method of Lucas and Kanade, the system gains ability for local recovery. If the tracker fails to match enough feature descriptors, the user can rotate the camera to bring more inlier features back into the camera view, thus restoring the previously found orientation.

6 RENDERING

On-site visualization of architectural models differs somewhat from general purpose rendering [12] and the methods should be adapted to the particular characteristics of the application for optimal results. The following special characteristics typical for mobile architectural visualization were identified in [9]:

- Uneven tesselation of 3D CAD building models
- Shadow mapping methods, related to the previous
- Complex and constantly changing lighting conditions

- Aliasing problems with highly detailed building models
- Sharp computer graphics vs. web camera image quality

We have experimented with some rendering and light source discovery methods described in [12] and integrated them into the OnSitePlayer application. The present implementation of the rendering methods covers: determining of sun light direction based on GPS, date and time of day; interaction with sliders to adjust day light intensities; screen-space ambient occlusion; soft shadows based on shadow maps; and adjusting the rendered image quality to web camera aberrations. Figure 4 shows an example of applying our rendering methods with a pilot project.

Figure 4. Photorealistic rendering with OnSitePlayer.

Automatic lighting acquisition from the real scene [12] has not been integrated into our system yet, and the current implementation has been done for the stand-alone OnSitePlayer system only. We plan to implement more advanced features also with the client-server solution, using separate feedback mechanisms for interaction and passing of lighting conditions of the real world scene to the server.

7 FIELD TRIALS

We have performed several iterations of field trials with three pilot cases. The first mobile use experiments were done in summer 2009. We used an 4D IFC model of the Koutalaki hotel in Lapland as an artificial example and augmented it behind our Digitalo offices in Espoo. The experiment enabled us to verify that most of the functionality was already operational, including e.g. visualizing the building in various modes and along the timeline, masking the virtual model with the real one, creating and viewing of mobile feedback reports, etc. However, the poor accuracy of the compass as well as GPS was noticed to be a major problem in practice. This stimulated our decision to develop interactive positioning methods as backup for the sensors.

A second round of experiments was carried out in fall 2009 in a case of the Forchem oil refinery in Sweden, with the purpose of augmenting new equipment to be installed. A Sony Vaio UX was used as the mobile device. Video of these experiments is available in [13]. In this case we relied completely on our 3D feature based tracking solution without sensors. Tracking was initialized manually by having the user indicate point correspondences between the video image and the 3D model of the factory. As hypothesis for future work, we believe this initialization step could be avoided by first roughly aligning the video and the model using compass information, and based on that, finding the actual point correspondeces automatically.

Our most comprehensive field tests were conducted in a series of experiments with the new Skanska offices in Helsinki 2010-2011. In summer 2010 before the building work started, we compared AR visualization of the planned building with different display devices: laptop PC on a podium, attached data glasses, and UMPC client. The two first devices were used in stand-alone mode while the UMPC was used in client-server mode. For rendering, we compared standard computer graphics without adjustments against our photorealistic rendering methods to

account for light direction, intensity and other visual properties. See Figures 2-4 and video [14].

In October 2010 when the construction work had already started, we finally received the complete 4D model of the Skanska building (IFC model size 60 MB) and went out to try it at the actual construction site. We could then verify that our solution also worked in practice with this rather demanding experiment. With some user interaction, we were able to augment the complex model on site, and display the construction elements to be installed at different time frames and from various view points. With respect to tracking initialization, managing altitude information interactively was considered to be the biggest problem. Stand-alone laptop PC version was used in these experiments. See Figure 5 and video [14].
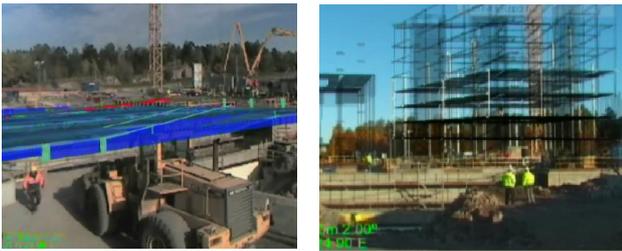

Figure 5. Mobile AR during construction work.

Harsh winter interrupted our field tests for almost half a year. The most recent experiments with the Skanska pilot were done in May 2011 when the back part of the building was already completed and also the first version of our mobile phone implementation was ready. In these experiments we were able to verify that our mobile phone solution using the new tracking method and pre-defined placemarks on the scene provided a stable augmented view of the building (see Figure 3). Comparison of the OnSitePlayer view which we had computed nine months earlier (Figure 4) against the real situation at the site (Figure 6) also validated the quality of our photorealistic rendering methods.

Figure 6. Photo of the Skanska building partly ready.

8 FUTURE WORK

For practical reasons, we have a number of stand-alone OnSitePlayer features yet to be integrated in the client-server solution. Also, improved tracking initialization, integration of feature based tracking methods with sensor data as well as photorealistic rendering technology into the AR system is still under way. Positioning accuracy could be improved by applying more accurate methods, e.g. differential GPS, Real Time Kinematics (RTK) and other measurement tools that are routinely employed at construction sites.

The first user studies with the system were conducted in late September 2011 in a real use case, involving augmented on-site presentation of a new hotel complex to a group of twenty municipal delegates and civil cervants. User studies will be reported in a separate publication. As one clear conclusion, laptop PC was clearly preferred over mobile phones for outdoors AR visualization. In the future, tablet devices seem to provide the most viable solution as an AR platform.

9 CONCLUSIONS

In this article, we have described a software system for mobile augmented reality interaction with complex 4D Building Information Models. Our system supports various native and standard CAD/BIM formats, combining them with time schedule information, fixing them to accurate geographic representations, using augmented reality with feature based tracking to visualize them on site, applying photorealistic renderinge, with various tools for mobile user interaction and feedback. The client-server solution is able to handle complex models on mobile devices, and an efficient tracking solution enables implementation also on mobile phones.

Future work and user studies are required before the system is ready for daily use at construction sites, and there are also some general concerns for applicability such as weather conditions. However we hope this article serves to prove the technical validity of applying mobile AR with real world 4D building models, and to stimulate further research in this important application domain.

REFERENCES

[1] D.H. Shin. and P.S. Dunston. Identification of application areas for augmented reality in industrial construction based on technological suitability. *Automation in Construction* 17, 2008, pp. 882-894.

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

[3] G. Schall, E. Mendez and D. Schmalstieg. Virtual redlining for civil engineering in real environments. *Proc. The 7th IEEE International Symposium on Mixed and Augmented Reality (ISMAR 2008)*, Cambridge, UK, September 2008, pp. 95-98.

[4] A.H. Behzadan. *ARVISCOPE: Georeferenced Visualisation of Dynamic Construction Processes in Three-Dimensional Outdoor Augmented Reality*. PhD Thesis, The University of Michigan, 2008.

[5] M. Goldparvar-Fard, F. Pena-Mora and S. Savarese. D4AR – 4 dimensional augmented reality – tools for automatred remote progress tracking and support of decision-enabling tasks in the AEC/FM industry. In *Proc. of The 6th Int. Conf. on innovations in AEC*, State College, PA, Jun 9-11, 2010.

[6] W. Pasman and C. Woodward. Implementation of an augmented reality system on a PDA. *Proc. The Second IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2003)*, Tokyo, Japan, October 2003, pp. 276-277.

[7] P. Honkamaa, S. Siltanen, J. Jäppinen, C. Woodward and O. Korkalo. Interactive outdoor mobile augmentation using markerless tracking and GPS. *Proc. Virtual Reality International Conference (VRIC)*, Laval, France, April 2007, pp. 285-288.

[8] M. Hakkarainen, C. Woodward and K. Rainio. Software Architecture for Mobile Mixed Reality and 4D BIM Interaction. In *Proc. 25th CIB W78 Conference*, Istanbul, Turkey, Oct 2009.

[9] C. Woodward, M. Hakkarainen, O. Korkalo, T. Kantonen. M. Aittala, K. Rainio and K. Kähkönen. Mixed reality for mobile construction site visualization and communication. *Proc. 10th International Conference on Construction Applications of Virtual Reality (CONVR2010)*, Sendai, Japan, Nov 4-5, 2010, pp. 35-44.

[10] ALVAR. http://www.vtt.fi/multimedia/alvar.html.

[11] TNO IFC Engine. http://www.ifcbrowser.com/ifcenginedll.html.

[12] M. Aittala. Inverse lighting and photorealistic rendering for augmented reality. The Visual Computer. Vol 26 2010, No. 6-8, pp. 669-678.

[13] Video on the Forchem pilot. http://youtu.be/xpZDXih9zLg.

[14] Video on the Skanska pilot. http://youtu.be/rVt86NGXQv4.