The 11th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST (2010)
A. Artusi, M. Joly-Parvex, G. Lucet, A. Ribes, and D. Pitzalis (Editors)

Snapshot Augmented Reality - Augmented Photography

M. Zöllner, M. Becker, J. Keil

Fraunhofer IGD, Germany

Abstract

With the current generation of smartphones augmented reality (AR) finally gets in the hands of end users. This is a giant leap for cultural heritage presentation. But due to software and hardware limitations of consumer devices the AR experience is still lacking the quality we have seen in research projects over the last years.
In this paper we are proposing a scalable method for high quality AR presentations for cultural heritage on a wide range of consumer devices: Snapshot Augmented Reality. Instead of a live video stream superimposed with jittering annotations we are "freezing" the scene and enabling Augmented Reality Photography. The result is an interactive scene superimposed on a still image taken by a visitor.
In order to outsource processing power and deliver content for a wide range of smartphones most of the sophisticated software works in the cloud. We are describing a reliable and scalable server infrastructure for tracking objects and environments and delivering context aware content to the visitors' devices.

Categories and Subject Descriptors (according to ACM CCS): Methodology and Techniques [I.3.6]: Interaction techniques—; Installation [I.3.m]: —;

1. Introduction

Current generations of mobile phones like the iPhone 3GS and Google's Android phones are taking mobile information systems for cultural heritage to the next level. With decent processors, 3D rendering capabilities and localization technologies like GPS and electronic compasses, these devices are becoming the target platform for location-based information and will displace common audio guides soon. Compared to mobile platforms like Tablet PCs and Ultra Mobile PCs (UMPC) in the past, current Smartphones are finally devices common people own and are using everyday.

After years of research and development AR finally becomes a mainstream medium. A real augmented reality wave started in summer 2009 after the premiere of the iPhone 3GS with its new digital compass, improved camera and faster processor. Broad media coverage about mobile augmented reality in mainstream media is still ongoing. The wave started with simple applications like Wikitude and Accrossair's NearestTube, which showed the nearest metro stations around and was followed by TwittARound, an augmented reality Twitter viewer. Meanwhile there are more than 100 AR applications available in the App Store. Due to the limitations of processing power, battery life, sensor accuracy (GPS, compass) and software development kits of current smartphone platforms there is a lack of visualization quality and tracking accuracy of commercial AR applications.

In our paper we are proposing the concept of Snapshot Augmented Reality. It enables high quality augmented reality location-based services on a broad range of today's smartphones. This Augmented Photography approach is a very familiar interaction for tourists. In addition to standard location-based services, like lists and maps of points of interest (POI) around, the system overlays the POI on the user's view. Therefore the image of the Smartphone's camera is displayed on the screen to create a see-through effect. Additional annotations suggesting POI are visible on the horizon. After the user takes a picture of the point of interest (i.e. a historic monument), the image will be augmented with the contextual information and 3D models and displayed on the screen. Finally the user sees a snapshot of the real environment enriched with contextual information, i.e. a 3D Model of the Roman Coliseum in its original state on top of today's ruins.

While computational capabilities and battery life are limited on today's smartphones, mobile broadband coverage is guaranteed in most areas of the world. Although data plans

DOI: 10.2312/PE/VAST/VAST10S/053-056

are affordable today even while on holiday, we are trying to limit the transfered data in our system to a minimum.

Our paper is structured as follows: We are starting an overview of related work in section 2. Section 3 is a global overview of our system's components. In the final section 4 we are presenting initial cultural heritage projects based on our system.

2. Related Work

Augmented Reality in cultural heritage has a long history and promises new possibilities for the presentation of cultural content to visitors. An early and very important mobile augmented reality project was ARCHEOGUIDE [VKT*01] where ancient Greek architecture and ancient Olympic sports events were displayed as visual overlays through a head mounted display direct at the site. Large and heavy bag backs with high technology gear, large GPS antennas and heavy head mounted displays were needed in order to reach the goal of positioning the user at the site and displaying context sensitive information on the spot. It featured one of the first markerless tracking approaches for outdoor environments.

CIMAD (Context Influenced Mobile Acquisition and Delivery of CH data) [Rya05] was an EPOCH (FP6) NEWTON project with the aim at the implementation of a framework for smart cultural heritage environments. These environments supported distributed and mobile on-site applications, from data capture to public dissemination.

During the locally funded MUSE [CMR*01] project by University of Bologna a proprietary, context-aware wearable terminal called WHYRE was developed for the domain of cultural tourism. By using wireless, GPS, gyroscope and compass sensors, WHYRE delivered contextual information without requiring input from the user. WHYRE MUSE technology was applied to the Museo e Certosa di San Martino in Naples, the Institute and Museum of the History of Science in Florence, and the archeological site of Pompeii. Later the technology was commercialized and is still used at the Roman Forum and the Colosseum.

3. Description of the system

Since a large percentage of today's Smartphones are still not capable for high quality real time computer vision we are proposing a cloud computing solution for tracking and visualization in mobile AR. A picture taken by a tourist is sent to a server where the location and the exact orientation relative to a point of interest is calculated. The resulting tracking information together with available content is sent back to the device for visualization. In the following sections we are describing the different modules of our system 1.

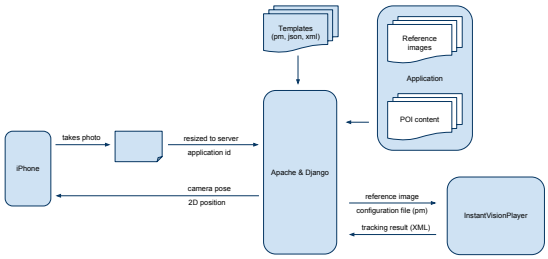

Figure 1: *System overview*

3.1. Smartphone Client

3.1.1. Overview

Until today most AR applications are developed as native standalone applications using programming languages like C++ and graphics libraries like OpenGL. Especially when developing for different phones with different software development kits (SDK) this means a lot of work and requires knowledge and experience. Thus the development is restricted to computer scientists.

Our solution consists of a slim client application based on open standards. It's main tasks are:

1. Showing abstract contextual POI in the area (annotations)
2. Taking a photo of a possible POI
3. Sending the photo with the geographical coordinates to the tracking server
4. Receiving detailed tracking information and contextual content
5. Visualizing the interactive content on the still image

The mobile client sends a picture together with an application ID and meta information like GPS coordinates, compass angle and return data format (JSON, XML, CSV) to the tracking server via HTTP. After the processing it receives the results from the server within seconds. The visualization is completely open to the developer and the device's capabilities. On iPhone and Android our visualization takes place in an embedded web browser view called Webview. Our applications are built on open standards like HTML5, Javascript and CSS. Thus available content on the web can be integrated easily and the content can be maintained by persons without programming skills.

3.1.2. Interaction

While using a photo instead of a live video AR scene looks like a limitation at first, it has several advantages regarding interaction and usability. After the photo was taken and is processed by the server, the user takes the smartphone in the same comfortable position he is used to while reviewing his common photos. The locus of attention now lies on the

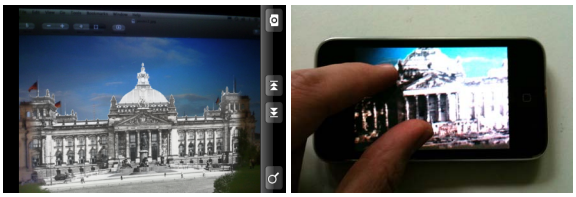

Figure 2: *Pinch and zoom gesture to zoom the AR still*

RT preprocessing	RT registration	RT storage
300s	0.008	25Mb
SIFT preprocessing	SIFT registration	SIFT storage
0.3s	0.04s	0.1Mb

Table 1: *CV methods evaluation. values are taken for 10 and 100 reference images, averaged and shown as per reference*

resulting superimposed photos instead of trying to point the phone at the POI and to interact simultaneous.

In this situation the user is able to interact with the still scene with well known multi-touch gestures. The pinch and zoom gesture zooms into the scene in order to see more details, which would not be possible on a live video stream. Pointing with the finger on the screen and moving it pushes the focus area of the enlarged scene. Additional annotations with contextual information like text, images, videos and 3D scenes are selected with a touch.

3.2. Tracking Server

The tracking server is managing the connection between the mobile client on the smartphone, the database and the computer vision. We are using the open source Apache [Apa10] web server, the Django [Dja10] web framework based on Python and a MySQL database. The computer vision is done by our software platform InstantVision. Thus all software components are platform independent and the main components except InstantVision are available for free. This is a main argument in favor of sustainability of cultural heritage platforms.

When a new image is sent from a mobile client, the server checks for the application ID and the meta information (i. e. GPS). Therewith it cuts down the amount of relevant tracking references from the database. Thus it only selects POI around the user's current location.

The server fills an XML template with the tracking references via Django's built-in template engine. The resulting XML file is handed over to InstantVision, which is running as a sub process via Python. After detection the tracking results are replied as an XML file. The server extracts the tracking information like the detected reference, the camera pose and its 2D position on the photo. This information is sent back to the client via JSON, XML or CSV.

3.2.1. Computer Vision Methods

For the positioning of the overlays on the screen we are using markerless poster tracking based on randomized trees [LLF05] like described in our previous work [ZPP*08]. In addition we also evaluated the SIFT [Low03] for object detection and registration.

We use these methods to identify the objects or locations in the image and also try to determine the position and orientation of the camera in relation to the cameraview used in the reference images. Thus the problem we need to solve is not just "object retrieval" as described in the work of Philbin et al. [PISZ07] but also registration between object and input image. In fact our CV methods are derived from previous work in Augmented Reality and are registration and tracking algorithms and not so much retrival algorithms. For retrival we just register the input image with all the references in our database and select the one which gives a valid registration result.

We also use KLT [TK91] for frame to frame tracking on the mobile device once the registration result was retrived from the server.

3.2.2. Randomized Tree Method

The randomized tree we are using here is mainly used for the location calculation from a set of simulated object views. We do not yet use these trees for recognition of different object as it was proposed by others. Our randomized tree method needs a long preparation phase, a set of random camera poses needs to be learned for each reference image, which takes a lot of time. The disk space needed for the trees is also quite large and makes the method impractical for our server architecture.

3.2.3. SIFT Method

The SIFT method can be used with just the images as reference data. As such it needs relatively small amounts of diskspace but the processing of all the reference images takes a while. Thus we implemented a preprocessing step where we extract the SIFT features from the images and we keep only the feature descriptors as reference data. In this way we need to run only one SIFT feature extraction per query. We then match the features from the query image with all the features in our database to find the best match.

3.2.4. Computer Vision Methods - Conclusion

Our performance measurements with the above methods are shown in table 1. We measured preparation time and registration time as well as storage requirements for the same set of images. The storage needed is actually influencing the

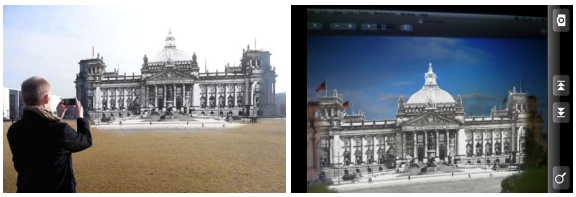

Figure 3: *SnapshotAR at German Reichstag in Berlin*

Figure 4: *Tourists in front of Palazzo di Diana*

registration performance quite a bit since we need to load the reference data at some point, thus large amounts of data slow down the process. For our purpose the SIFT method gives good results with acceptable cost.

4. Results

4.1. 20 Years Fall of the Berlin Wall

In 2009 Berlin celebrated "20 Years of the Fall of the Berlin Wall". Additionally to a real time AR application on UMPCs showing urban development and historic overlays [ZKWP09], we brought the content to the iPhone later. This Snapshot AR variant superimposes historic photos of the German Reichstag and the Brandenburg Gate seamless on a photo taken by a tourist. He can slide through different views in time from 1901 until today and zoom into them via common touch gestures.

4.2. iTACITUS - Reggia Venaria Reale

We started the the EU funded project iTACITUS [iTA08] in 2006 when we only could imagine the mobile revolution which would take place later. At the end of the project we ported the UMPC-based AR applications to the iPhone via Snapshot AR. We superimposed the buildings of one of the field test areas of the Augmented Reality applications at Reggia Venaria Reale, an UNESCO World Heritage site in Italy close to Turin.

In this early prototype we used randomized trees tracking 3.2.2. Thus we were able to use the tracking reference data from the live UMPC version [ZKWP09] on our server. For rapid development and testing the test server ran on a notebook on the site and the clients connected via local Wifi.

5. Conclusions

In this paper, we have presented a distributed cloud system for mobile augmented reality presentations of cultural heritage sites on a wide range of smartphones. The system is an interim solution for high quality AR visualizations until the technical capabilities and battery life of smartphones are sufficient for computer vision and visualization tasks. Meanwhile our system outsources sophisticated computing operations into the cloud and visualizes the results with a slim and efficient client on the smartphone.

6. Acknowledgments

Parts of the system described in this paper were developed in the iTACITUS project funded under the 7th Framework Programme of the European Union.

References

[Apa10] APACHEHTTPDSERVERPROJECT: The number one http server on the internet, 2010. http://httpd.apache.org.

[CMR*01] CINOTTI T. S., MALAVASI M., ROMAGNOLI E., SFORZA F., SUMMA S.: Muse: An integrated system for mobile fruition and site management. In *In Proc. ICHIM* (2001), pp. 609–621.

[Dja10] DJANGOWEBFRAMEWORK: High-level python web framework that encourages rapid development, 2010. http://www.djangoproject.com.

[iTA08] ITACITUS: Intelligent tourism and cultural information through ubiquitous services, 2008. http://www.itacitus.org.

[LLF05] LEPETIT V., LAGGER P., FUA P.: Randomized trees for real-time keypoint recognition. In *In CVPR* (2005), pp. 775–781.

[Low03] LOWE D. G.: Distinctive image features from scale-invariant keypoints, 2003.

[PISZ07] PHILBIN J., ISARD M., SIVIC J., ZISSERMAN A.: Object retrieval with large vocabularies and fast spatial matching. In *In Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (2007).

[Rya05] RYAN N.: Smart environments for cultural heritage. In *In Reading the Historical Spatial Information in the World* (February 2005), pp. 609–621.

[TK91] TOMASI C., KANADE T.: Detection and tracking of point features. *Carnegie Mellon University Technical Report CMU-CS-91-132* (Apr. 1991).

[VKT*01] VLAHAKIS V., KARIGIANNIS J., TSOTROS M., GOUNARIS M., ALMEIDA L., STRICKER D., GLEUE T., CHRISTOU I. T., CARLUCCI R., IOANNIDIS N.: Archeoguide: first results of an augmented reality, mobile computing system in cultural heritage sites. In *Virtual Reality, Archeology, and Cultural Heritage* (2001), pp. 131–140.

[ZKWP09] ZOELLNER M., KEIL J., WUEST H., PLETINCKX D.: An augmented reality presentation system for remote cultural heritage sites. *VAST 2009. Proceedings* (2009).

[ZPP*08] ZOELLNER M., PAGANI A., PASTARMOV Y., WUEST H., STRICKER D.: Reality filtering: A visual time machine in augmented reality. *VAST 2008. Proceedings* (December 2008).