

UMAR - Ubiquitous Mobile Augmented Reality

Anders Henrysson
Norrköping Visualization and Interaction Studio
Linköping University, Norrköping, Sweden
andhe@itn.liu.se

Mark Ollila
Norrköping Visualization and Interaction Studio
Linköping University, Norrköping, Sweden
marol@itn.liu.se

ABSTRACT

In this paper we discuss the prospects of using marker based Augmented Reality for context aware applications on mobile phones. We also present the UMAR, a conceptual framework for developing Ubiquitous Mobile Augmented Reality applications which consists of research areas identified as relevant for successfully bridging the physical world and the digital domain using Mobile Augmented Reality. A step towards this we have successfully ported the ARToolkit to consumer mobile phones running on the Symbian platform and present results around this. We also present three sample applications based on UMAR and future case study work planned.

Categories and Subject Descriptors

H.5 [Information Interfaces and Presentation]: Multimedia Information Systems

General Terms

Human Factors, Experimentation, Design

Keywords

Augmented Reality, Pervasive Computing

1. INTRODUCTION AND BACKGROUND

In this paper we present exploratory research in the area of mobile computer graphics and interaction using augmented reality and context aware environments. Currently, mobile devices have limited processing power and screen size due to their small size and dependency on batteries with limited life times. However, the mobile device is the most ubiquitous device and a part of most peoples everyday life. Hence, as mobile devices become more advanced, the development of 3D information systems for mobile users is a growing research area [22]. Being mobile means that the context changes and that information, and computation can follow

the person. We would like to exploit the features of a mobile device while trying to work around its limitations in order to study its potential to enhance the experience of the users. We have chosen to work with smartphones since they have the capability of rendering and displaying graphics such as video and 3D animations. They are connected to data enabled networks such as GPRS and UMTS and many of them now feature built in cameras (with some models having GPS built in).

The idea behind augmented reality (AR) is to track the position and orientation of the users head in order to enhance his or her perception of the world by mixing a view of it with computer generated visual information relevant to the current context. AR is useful when we have to solve a real world problem using information from another domain such as a printed manual or a computer screen. By projecting the information onto a view of the real world there is no need for distracting domain switching. The information is fetched from a virtual representation of the real world environment to be augmented, using tracking, or it could also be derived from direct object recognition. We, however, will only discuss the case where tracking is involved. Tracking can be performed with a wide variety of sensors such as GPS, optical, inertia trackers, ultrasonic trackers etc. Since the smartphones we are working with has built in cameras we have chosen to work with optical tracking where we track the orientation and translation of the camera.

AR has traditionally been reserved for high-end computers while mobile augmented reality (MAR) has used custom-built hardware setups. Many of these consist of a laptop mounted on a frame carried as a backpack [20]. They often feature head mounted displays (HMDs) and as such, the majority are research platforms inaccessible to average consumers. Instead, in this paper, we have focussed on consumer level mobile phones where devices such as PDAs and smartphones have developed rapidly and have enough rendering power to do 3D graphics. As concluded in [13], the smartphone meets all of the requirements posed by AR to some extent.

We now present a background into relevant research regarding AR, mobile devices and tracking. The AR-PDA [16] setup consists of a PDA with a camera. The PDA is connected to an AR-server using WLAN and sends a video stream to the server for marker-less tracking and rendering. The augmented video stream is returned to the PDA for display after processing on the server. Similarly, AR-Phone [14] and PopRi [7] have still images sent to a server running ARToolkit [1] for augmentation. PopRi also allows contin-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004 October 27-29 2004 College Park, Maryland, USA
Copyright 2004 ACM 1-58113-981-0/04/10 ...\$5.00.

uous augmentation when run from a video enabled phone. The drawback of these client-server setups is that the user is dependent on a fast connection to a server. In a wide area application where no WLAN is accessible the images must be sent over a network such as GSM/UMTS that usually comes with a cost per byte or minute for data traffic and, obviously, latency issues. An implementation of the AR-ToolKit on the PocketPC platform was performed by [15]. They analyzed the performance of different functions in the ARToolkit and identified the most computationally heavy. These have been implemented using fixed-point arithmetic for significant speed up. However the PDA still requires WLAN or Bluetooth for communication, which limits its use in dynamic wide area applications where these networks are not available. The Real in Real project from Japanese operator NTT [9] uses a Tablet PC as the viewing device. It tracks a sensor cube with sides that consist of ARToolkit markers. The sensor cube is equipped with sensors that measure the incoming light. This information is then used to produce realistic renderings. Video See-Through AR and Optical Tracking with Consumer Cell Phones project [18] does 6DOF tracking on a smartphone at interactive frame rates. It tracks a 3D marker onto which the three coordinate axes are printed. By detecting these lines in the image the camera transformation can be estimated and used for rendering. The disadvantage is that it requires a marker with more complex topology than the simple 2D markers of for example, the ARToolkit. The SpotCode platform [4] uses markers to turn a camera-enabled phone into a virtual mouse pointer for interaction with digital content displayed on a screen. The camera phone identifies the marker and sends its id and relative position and orientation to a nearby computer using Bluetooth. The markers consist of data rings with sectors each of which encodes a single bit of information. Thus there is no need for the system to know the particular marker in advance and the markers are of known complexity. They can be used for real world hyperlinks where a marker id is mapped to an URL for instance.

IDEIXIS [21] uses images taken with a camera phone to recognize the location and provide context related information using a hybrid image-and-keyword searching technique. It works by using an image captured by a camera phone for content-based image retrieval (CBIR) in a limited image database. If there is a hit it continues by extracting a keyword that is used to search Google to find related images. These are matched to the captured image to determine relevance. Siemens has developed an AR game called Virtual Mosquito Hunt [12] where the real time video stream from the onboard camera is augmented by virtual mosquitoes for the player to kill.

An environment for context aware mobile services is SmartRoutaari [19]. It is based around PDAs and uses WLAN for communication and positioning. The context information also includes weather data. It provides services such as map-based guidance, mobile ads and interactive 3D visualization of historical data. Currently, there is no augmented reality in use.

Optical tracking can be marker-based or marker-less. In the case where we have markers the system calculates the orientation and translation of the camera in a coordinate system where the origin is in the center of a marker. In marker-less tracking we have to rely on features in the environment such as edges to track the camera. At a first

glance marker-less tracking seems more beautiful but having visible markers has the advantage of telling the user where environment has been augmented. The problem is analogous to that of hyperlinks in MPEG-4 video.

Marker-based optical tracking gives accurate results but is limited by the visibility of the marker. The markers have to be scaled by distance so that they are identifiable by the system. Here the complexity of the marker plays an important role. A low frequent pattern (large white and black regions) is easier to recognize but simple patterns limit us to a small set of distinct markers. Some marker-based systems need to learn a pattern before it can be used [1] while others contain information that can be translated into an identification number [4].

However, the benefits of camera tracking include, but are not limited to: 1) Virtual screen: By tracking a single marker it is possible to pan over a virtual screen up to 4 times bigger than the physical screen; 2) Positioning: Information tied to location. Tracking the camera in the environment yields accurate position information that can be used to adapt and configure services; 3) Camera phone as optical mouse pointer: Interact by moving the phone relative to marker.

The research we are conducting is to investigate how AR can be used to enhance the limited user interface of current smartphones and transform them into tools for interacting with the real world (through mobility). Mobility means that the context switches according to e.g. position, climate etc. By estimating the context a device or a service can be configured to better suit its user. The current context can be estimated by a wide variety of sensors e.g. GPS for positioning. In order to configure services as such, the system needs to know the user's personal preferences, which might also depend on context. The paper is organized as follows: Section 2 we present our framework for ubiquitous mobile AR. Section 3 describes the implementation of ARToolkit on a smartphone and also our implementation of context aware video as an example of a non-augmented context aware mobile application. Section 4 presents the three application and results. Section 5 examines future work planned.

2. UMAR FRAMEWORK

UMAR is a conceptual framework that consists of research areas required to perform Ubiquitous Mobile Augmented Reality where we bridge the digital and real domains with context aware information. For an arbitrary context we want to fetch the relevant information and display it using appropriate techniques depending on the spatial relationship between context and retrieved information. If there is a close spatial relationship we would prefer to use AR. If the spatial relationship is weaker we could use a 2D map similar to [19]. If there is no spatial relationship the information could be presented e.g. as a web page or as audio using text-to-speech depending on the user preferences. In contrast to specialized HMD configurations, the smartphone can easily switch between modalities.

Information retrieval based on context awareness and personalization is an important part of the overall framework but not a primary focus. Work in this area has been done as part of the Context Aware Pervasive Networking program [2] and also in the area of MPEG-7 when it comes to indexing media [5]. Future semantic web technologies [8] will be needed as we ultimately will need to search the entire web

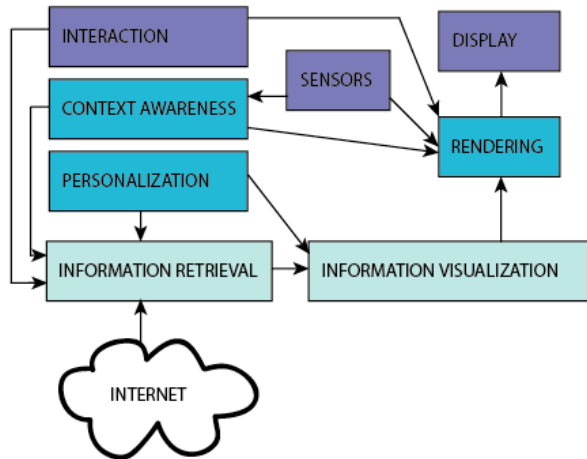


Figure 1: Overview of the UMAR Framework.

for relevant information and not be limited to custom-built databases. The hybrid search of [21] was able to search among 425 million indexed images and related web pages using a single image as input.

The retrieved information would need to be classified according to the mentioned spatial relationship to the real world scene surrounding the user and converted into graphical representations. This later is a problem of information visualization. Based on the user preferences, other media types e.g. web pages and audio could be synthesized. In the AR case we render a virtual object using the estimated camera before compositing it into the frame where the marker was detected. Ideally we would like to use context information for the rendering similar to [9] in order to produce photorealistic images. The rendering technique will depend on the object representation e.g. polygonal and must be adapted to the device in order to meet QoS demands such as interactive frame rate. The overview of the UMAR framework is shown in Figure 1. The arrows can be seen as data flow between possible outcomes from different areas but they can also represent research questions. Darker color means that the issues are more likely to be studied on the client side.

The goal for UMAR is to perform as much as possible on the client to reduce the data traffic and avoid being dependent on fast network access. While the sensors, display and interaction UI is tied to the device itself the remaining issues can be server-based if necessary. The information retrieval and visualization issues where information is retrieved and converted to graphical representation will most likely be studied on the server side for non-trivial scenarios.

3. IMPLEMENTATION

We have chosen to work with the ARToolkit [1] which is an open source toolkit for optical tracking. Besides a main library for tracking and marker identification, it also contains camera calibration software. It works by identifying markers in a video stream and calculate the orientation and translation of the camera in a reference coordinate system centered at a marker image. It performs the following steps: 1) Turn captured images into binary images; 2) Search the

binary image for black square regions; 3) For each detected square the pattern is captured and matched against templates; 4) Use the known square size and pattern orientation to calculate the orientation and translation of the camera; 5) Draw the virtual objects.

We have implemented two applications based on this conceptual framework. First we implemented a simple context-aware video service based on a subset of the UMAR framework in order to have a simple case study (see Figure 2). Secondly, we ported the ARToolkit 2.65 on to a smartphone running Symbian [11] and Series60 [10] for the purpose of evaluating marker-based optical tracked AR on a consumer device. ARToolkit consists of an AR library foremost responsible for marker identification and tracking, calibration software and an application for learning new markers. We treated ARToolkit as a black box that takes camera parameters and an image as input and returns a camera transformation matrix. The camera matrix consists of intrinsic and extrinsic parameters. The intrinsic parameters relates to the properties of the camera such as focal length. Together with distortion parameters they are used to link a pixel coordinates with the corresponding coordinates in the camera reference frame. These parameters are calculated once in a calibration process. To do this we modified the calibration application to take still images captured by the camera phone. Five images containing 81 points each were used. The extrinsic parameters are estimated each frame in an iterative process and consist of the orientation and translation of the camera in the world coordinate frame centered at the marker. When rendering, the extrinsic parameters correspond to the view transformation in the rendering pipeline and the intrinsic parameters correspond to the perspective transformation.

4. EXPERIMENTS AND RESULTS

To test the performance we built a test program for the mobile device (a Nokia 6600) with a 104 MHz ARM9 CPU, 6 MB of volatile memory and a screen size of 176*208 pixels with 16 bits per pixel. It also has a 0.3 megapixel camera. We used images at a resolution of 160*120 pixels with 12 bits per pixel color depth. For our tests we used a single marker with low complexity (size was 8 by 8cm). As an UMAR application example we placed a marker on top of a tram route map similar to those found at tram stops. The idea was to augment the map with an animation showing the current position of the trams trafficking the line. The stations were identified on the map and by using the arrival/departure time for the end stations and the current system time, the relative position of tram could be calculated. The absolute positions in mm relative thealso marker center could then be calculated. The tram route was drawn by a polyline and the trams represented with sprites. Since each station can have a unique marker, the users position can be estimate accurately and location-based information could easily be added. In a more sophisticated case all the above steps, now done manually, would likely be made automatically by using appropriate web semantics to tag the timetable data (see Figure 3).

The second UMAR application involved context-aware mobile video. We developed a minimalistic setup consisting of a client and a combined context and video server. The setup targets parts of the UMAR framework e.g. sensors, context awareness and information visualization. The client

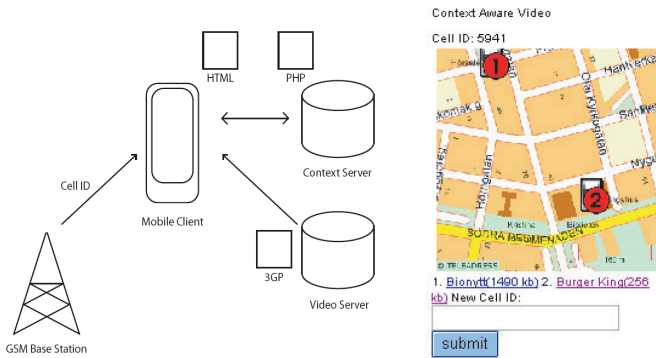


Figure 2: Context Aware Video Architecture based on UMAR Framework

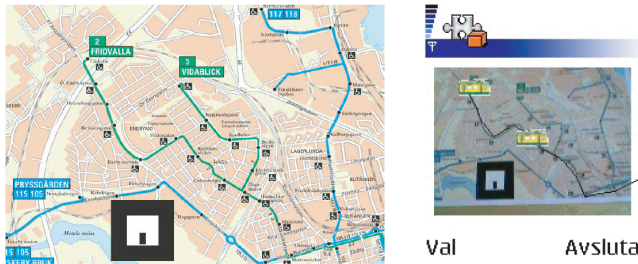


Figure 3: Map with marker and screenshot from mobile device.

was a Sony Ericsson P800 smartphone and our implementation involved integrating three commercial applications: Psiloc miniGPS, Opera browser and PacketVideo pvPlayer; plus our own research software. Context retrieval is performed by miniGPS, an application that displays the cell ID number of the GSM base station used. The application can alert the user when a cell is entered or exited. The GUI consists of a web page displaying an approximal map of the current cell. It has a form where the cell ID is entered manually. Available videos are marked with numbered legends and are presented as hyperlinks. When clicked, the user is presented a dialog box where she or he can choose to open the video file. Lacking a compatible video server we settled with download of the entire clip prior to playing. If the video is downloaded it will automatically be played by pvPlayer. When the video is finished the user can return to the GUI with the click of a button. The client communicates with a context server which is a web server running a PHP script to generate the HTML document to be displayed. In our current implementation the cell IDs, maps and legends are static. The cell ID:s were obtained by walking around the town of Norrköping to which the application is limited and the maps have been obtained from Gula Sidorna [3]. Thirdly, as a hybrid between the two applications we also implemented the tram animation using a bitmap route map as background image, i.e. a context-aware video rendered on the client. Since we assumed no real-world map to augment in this case, we have a weaker spatial relationship between the animation and the users context and are thus using a simpler visualization technique.

A closer look at the implementation reveals that the AR-Toolkit uses floating points with double precision for the representation and calculation of the camera matrix. Since smartphones lack FPU all floating-point arithmetic is emulated in software with a big computational overhead compared to hardware (performance is hundreds of times slower than integer performance). Because of this the full 6DOF tracking could not be done at interactive frame rate but closer to about one frame per second. However 3DOF tracking (2D translation + area to estimate z-value) and marker identification can be done at interactive frame rates with no perceived performance degradation compared to the video frame rate provided by the camera. We experimented with fixed point implementations based on the analysis in [15]. However since there is no corresponding fixed-point library freely available for the Symbian platform we were not able to achieve satisfying results using simple implementations. The maximum range for marker detection using our setup was close to 1.5 m and the Nokia 6600 performed better than previously tested smartphones [13]. The limited range is not a big problem since the quality of the real-time video stream is fairly low and the screen size is small which makes long range tracking less attractive.

5. CONCLUSIONS AND FUTURE WORK

We have successfully ported the ARToolkit to the Symbian platform though some performance issues remain to be solved. We have shown that smartphones can be used for AR without server assistance. We have presented UMAR, a conceptual framework for further research in Mobile Augmented Reality in Context Aware Pervasive Environments. We have implemented simple UMAR applications to show different visualization techniques depending on the level of spatial relationship between information and context. The current platform is limited to the close proximity of a marker in order to provide AR. We would like to expand the platform to incorporate wide area tracking using GPS or cell ID. To extend the marker tracking we would like to look at feature tracking such as [17]. It would also be interesting to study to what extent optical flow measurements could assist the optical tracking. On the rendering side we need to incorporate an efficient 3D renderer (we need to be able to define the view and perspective matrices) or an implementation of OpenGL ES [6]. It would be desirable to have a standard for content representation. We need to implement fixed-point arithmetic with variable precision in order to solve the performance issues. We will also see if we can further adapt the ARToolkit to the smartphone platform. The next generation of smartphones features in some cases both GPS and megapixel cameras. There are also phones with a tilt sensor and digital compass which will open up new possibilities when it comes to tracking and will be an obvious research platform. Over the next phases of the research, field trials and user evaluations will take place, giving us both qualitative and quantitative results.

6. ACKNOWLEDGEMENTS

The first author is supported by a department grant from the Department of Science and Technology at Linköping University. This project was partially funded by a grant from HomeCom. We also acknowledge the support of Prof. Anders Ynnerman.

7. REFERENCES

- [1] Artoolkit. www.hitl.washington.edu/artoolkit/.
- [2] Capnet. www.mediateam oulu.fi/projects/capnet/.
- [3] Gulasidorna. www.gulasidorna.se.
- [4] High energy magic. www.highenergymagic.com.
- [5] Mpeg/7 overview. www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm.
- [6] Opengles. www.khronos.org/opengles.
- [7] Popri. www.popri.jp/real_in_real/PopRi.htm.
- [8] Rdf. www.w3.org/TR/REC-rdf-syntax/.
- [9] Real in real. www.contents4u.com.
- [10] Series 60. www.series60.com.
- [11] Symbian. www.symbian.com.
- [12] Virtual mosquito hunt. w4.siemens.de/en2/html/press/newsdesk_archive/2003/foe03111.html.
- [13] H. Anders and O. Mark. Augmented reality on smartphones. In *2nd IEEE International Augmented Reality Toolkit Workshop*, 2003.
- [14] D. J. C. Dan Cutting, Mark Assad and A. Hudson. Ar phone: Accessible augmented reality in the intelligent environment. In *OZCHI2003*, Brisbane, 2003.
- [15] W. Daniel and S. Dieter. Artoolkit on the pocketpc platform. In *2nd IEEE International Augmented Reality Toolkit Workshop*, Waseda University, Tokyo, Japan, 2003.
- [16] C. Geiger, B. Kleinnjohan, C. Reiman, and D. Stichling. Mobile ar4all. In *2nd IEEE and ACM International Symposium on Augmented Reality (ISAR 2001)*, New York, USA, 2001.
- [17] M. B. Hirokazu Kato, Keihachiro Tachibana and M. Grafe. A registration method based on texture tracking using artoolkit. In *2nd IEEE International Augmented Reality Toolkit Workshop*. Waseda Univ., Tokyo, Japan, 2003.
- [18] C. L. Mathias Möhring and O. Bimber. Video see-through ar on consumer cell-phones. In *In proceedings of International Symposium on Augmented and Mixed Reality (ISMAR'04)*, 2004.
- [19] T. Ojala. Smartrotuaari - context-aware mobile multimedia services. In *2nd International Conference on Mobile and Ubiquitous Multimedia*, Norrköping, Sweden, December 2003. ACM Press.
- [20] T. H. S. Feiner, B. MacIntyre and T. Webster. A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. In *Proc. ISWC '97 (First IEEE Int. Symp. on Wearable Computers)*, Cambridge, MA, 1997.
- [21] K. T. Tom Yeh and T. Darrell. Searching the web with mobile images for location recognition. In *CVPR 2004.*, 2004. To appear.
- [22] T. Vainio and O. Kotala. Developing 3D information systems for mobile users: some usability issues. In *Proceedings of the second Nordic conference on Human-computer interaction*, pages 231–234. ACM Press, 2002.