

Mobile Augmented Reality for Retail Environments

Pasi Välikkynen, Alain Boyer, Timo Urhema

VTT Technical Research Centre of Finland

P.O.Box 1300, 33101 Tampere, Finland

{firstname.lastname}@vtt.fi

Riku Nieminen

Tieto Finland Oy / Decision Support Solutions

P.O.Box 449, 33101 Tampere, Finland

riku.nieminen@tieto.com

ABSTRACT

We present a mobile handheld augmented reality system for presenting product information to a consumer. The system allows the user to view a three-dimensional model of product package contents with a mobile phone. The model is displayed over the product packaging, allowing the user to view the package contents without opening the package. Additional digital content related to the product can also be shown. This includes product videos, web-based content and product information from an external data base. The system demonstrates context-awareness by modifying the displayed content according to the location of the user; different content can be shown in a store and at home. In this paper, we describe the overall functionality, interaction and implementation of the system.

Author Keywords

Augmented reality, human-computer interaction.

ACM Classification Keywords

H5.1. Information interfaces and presentation: Multimedia Information Systems.

General Terms

Human Factors.

INTRODUCTION

Augmented reality (AR) is a human-computer interaction paradigm in which images are projected over the real world making virtual and physical realities meet [4]. In mobile handheld AR, the typical approach is to display camera viewfinder image on the mobile device display and augment that image, allowing the user to view the physical world through an “AR lens” [7]. Augmented reality is increasingly becoming an enabling technology for innovative mobile applications. This paper presents some of our design decisions while working on a mobile AR application for retail environments. The application provides the user with relevant information about products, contains some contextual adaptation between the retail and home environments for the same products and utilizes our industrial partner’s database of meta-data. To gather ideas for the further design, a brainstorming workshop was arranged. The main demonstration concepts were decided to be as follows:

- peeking inside the box, for example rendering a 3D model of the contents over the product packaging

creating an illusion that the user is looking inside the box

- showing external digital meta-data related to the product, such as product data, videos and web pages (similar to existing commercial mobile applications such as Foodie.fm¹)
- allowing “scanning” a shelf for products matching some criteria, for example not containing ingredients to which the user is allergic (similar to ShelfTorchlight [5])

As in a typical mobile AR application, the interaction is based on the user viewing the physical environment through the camera view finder. View finder image is augmented with graphics based on what markers are visible. In this context, each product has a unique marker printed on its packaging. This allows for vision-based augmented reality [2] where virtual 3D models can be accurately placed over the physical products. This creates a much more immersive experience compared to position-based AR that relies on GPS and IMU sensors.

One of the initial interaction design decisions was to allow the user to quickly see what meta-data is available for a given product. This was achieved by only showing a group of links to the meta-data over the marker. The interaction was modeled after the “hovering” concept for RFID tags [6] and allows the user to quickly glance into the digital representation of the physical object before activating any further functionality. This approach allows the user to scan a shelf full of products to see data about several products at once.

The main AR functionality is to “peek inside the box”. In this mode, a 3D model of the product, including virtual parts of the packaging, are shown over the product packaging creating an illusion that the user is looking inside the box. This is a different type of AR x-ray system that is better suited for near-field AR as opposed to the conventional far-field x-ray systems [1, 3]. The models used in this prototype are animated, giving richer information to the user. Rotating the marker or viewing the actual product packaging from different angles allows the user to see the model from different perspectives.

¹ <http://en.foodie.fm/>

The other main functionality is to present the user with web and video meta-data related to the product. In addition, product information from an external database is available. In this prototype, only generic data about the product is shown, but it is possible to display anything that is available in the database, such as the carbon footprint data of the product.

To demonstrate the possibility of context-awareness (CA), the application includes a simple location-based CA module, which can determine if the application is being used in a store or at home. The purpose of this differentiation is to provide the user with relevant information based on the current context. For example, advertisement-focused material is displayed in at the store in order to support the user's buying decision. At home, more supportive material, such as user guides or instructions are presented.

INTERACTION

Most of the user interaction is based on the basics of the mobile augmented reality paradigm. The markers are visualized when they enter the view by overlaying an interactive context menu. The context menus consist of buttons with icons. By quickly glancing at the icons, the user can immediately see the available meta-data related to the given product. Pressing a button shows the associated meta-data.

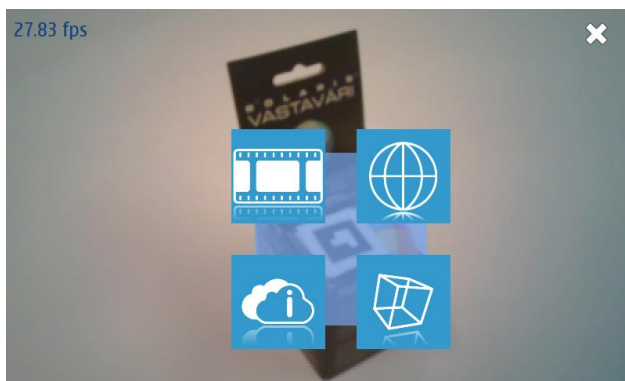


Figure 1: In the planar menu, the menu icons are rendered parallel to the screen.

We implemented two different types of context menus. In the first approach, a planar menu is shown with buttons that are rendered parallel to the device display (Figure 1). In the second approach, a perspective menu is shown with buttons that are rendered parallel to the marker (Figure 2). The main difference between both types of menus is that planar buttons have a more constant size whereas perspective buttons grow and shrink depending on the relative position of the camera to the marker. Planar buttons are arguably easier to press but perspective buttons provide for a more immersive AR experience.



Figure 2: In the perspective menu, the menu icons are rendered parallel to the marker.

The types of meta-data supported include 3D models (for example of the product), videos, web-based data, which can be text, images, videos or anything that can be shown in a browser and product data from a cloud-based database. The specific meta-data is activated by touching its corresponding button. The 3D model is shown in the AR component and the other types of meta-data are shown in the meta-data component, which consists of customized video players and web browsers (Figure 3).



Figure 3: The video player shows an advertisement video of the product.

The augmented reality interaction allows the user to first "peek inside the box" and then to zoom the model by moving the mobile device closer or farther away from the marker (Figure 4). The model can be rotated by physically rotating the product package or viewing the product from different angles. The 3D model can also be animated to make it more interesting or even present basic information about the product that is difficult to explain with words or images.



Figure 4: Augmented product package allows the user to peek inside the box.

ARCHITECTURE

The system uses a client-server based architecture, with the clients running on camera-enabled mobile devices and the server providing cloud-based data to the clients. This paper focuses only on the client application.

The client application consists of two components, an augmented reality component and a meta-data component. Both components provide a full screen view and only one component is active and visible at a time.

The AR component is first displayed when the application launches. After initialization, this component executes the core AR loop which is summarized in the following five steps. *First*, an image is requested from the camera. Appropriate color space conversions are performed and the resulting image is stored in memory. *Second*, markers are detected in the gray scale channel of the image using a version of the Alvar Library² that is optimized for mobile devices. For each visible marker in the image, a 3D transformation between the marker and the camera is computed. *Third*, a list of known markers is updated with the previously computed pose information. If a new marker is detected, the marker id is sent to the meta-data component which replies with a list of required buttons and a model. This information is cached and the state of the marker is initialized. *Fourth*, the image channels are uploaded to the GPU and rendered using OpenGL-ES. *Lastly*, using the marker pose, appropriate virtual content is rendered via OpenGL-ES on top of the marker depending on the state. In the menu state, a context menu consisting of buttons is rendered in a planar mode (parallel to the display plane) or in a perspective mode (parallel to the marker plane). In the model state, a 3D model is rendered on top of the marker. The implementation of these steps is optimized as much as possible in order to achieve a real-time experience at 18-30 frames per second.

While the AR component is executing the augmented reality loop, it also monitors touch events from the screen. If the user presses a menu button, the AR component pauses its rendering loop and informs the meta-data component, which becomes visible to the user. The meta-data component can display contextual information depending on the button pressed. Currently, views to display web-pages, videos and context information from a cloud-based service provider are possible. When the user presses the back button, the AR component is made visible again and its rendering loop is re-started again.

It should be noted that one menu button is used to display a 3D model. When this button is pressed, the view does not switch to the meta-data component, but instead, the state of the marker is switched and the AR component renders a 3D model over the marker instead of the context menu.

The mobile client application was built for the Linux-based Maemo 5 platform and runs on the Nokia N900 device. The core AR technology and the application logic are implemented using the C++ programming language. The user interface is implemented using Qt and the 3D rendering is based on OpenGL-ES 2.0.

DISCUSSION

We performed a small-scale experiment of the planar and perspective context menu implementations. The experiment was conducted while we were demonstrating the prototype. Ten subjects tried out the two menu alternatives and were told to choose which one they preferred and why. We also observed whether they realized that they can zoom in the perspective menu by moving the device closer to the marker.

Five users preferred the perspective menu while the other five preferred the planar menu. Those who preferred the planar version liked the fact that the menu buttons were always the same size and easy to press due to their constant size. Those who preferred the perspective version said they liked to see several packages side by side with small buttons and zoom into the menus manually by moving the mobile device. This initial small-scale study gave us valuable insights into improving the interaction and hints on what to take into account when we organize a real comparative user evaluation of the menu alternatives.

Currently we are visualizing the digital meta-data on top of the marker or product, but we have also been considering visualizing other types of data. For example, showing the carbon footprint of each product, or allergy information matching the user's personal preferences would be useful items to show. The current architecture allows these kinds of additions easily even though they are not implemented in the current prototype.

² www.vtt.fi/multimedia/alvar.html

REFERENCES

1. Avery, B., Sandor, C. and Thomas, B. Improving Spatial Perception for Augmented Reality X-Ray Vision. *Proc. Virtual Reality Conference*, IEEE (2009), pp. 79–82.
2. Azuma, R. A. Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments*, Vol. 6, No. 4, pp. 355–385, 1997.
3. Bane, R. and Höllerer T. Interactive Tools for Virtual X-Ray Vision in Mobile Augmented Reality. *Proc. International Symposium on Mixed and Augmented Reality*, IEEE (2004), pp. 231–239.
4. Dix, A., Finlay, J., Abowd, G. D. and Beale, R. *Human-Computer Interaction*, 3rd ed., Pearson Education Limited, 2004.
5. Löchtefeld, M., Gehring, S., Schöning, J. and Krüger, A. ShelfTorchlight: Augmenting a Shelf Using a Camera Projector Unit. *Proc. 1st Workshop on Personal Projection via Mobile and Wearable Pico Projectors*, (2010), pp. 20–23.
6. Väikkynen, P. Hovering: Visualising RFID hyperlinks in a mobile phone. *Proc. Mobile Interaction with the Real World* (2006), pp. 27–29.
7. Wagner, D., Pintaric, T., Ledermann, F. and Schmalstieg, D. Towards massively multi-user augmented reality on handheld devices. *Proc. Pervasive Computing*, Springer (2005), pp. 208–219.