

Augmented Reality Aided Operation and Maintenance System for Indoor Environments^{*)}

Yuichi TAMURA, Tomohiro UMETANI and Shin KUBO¹⁾

Konan University, Kobe 658-8501, Japan

¹⁾*National Institute for Fusion Science, Toki 509-5292, Japan*

(Received 22 November 2012 / Accepted 14 May 2013)

This paper proposes an Augmented Reality (AR) system to assist operation and maintenance tasks in an indoor environment, such as a nuclear fusion reactor and its building. AR is a technology that enhances real information by adding 3D virtual objects, images, sounds, or movies via a web camera. The AR system often uses “markers” such as QR code to detect the place where the virtual content should appear. However, these markers are unnatural and they can disturb the scenery. We propose an AR system that can detect natural markers, which provides AR content via a network. This system stores the information related to markers and virtual objects on a server. A device connected to this system automatically downloads this content so that the user can watch the AR content via a web camera. We add a real-time numerical simulation function that allows us to simulate physical phenomena by touching AR contents. It also enables us to observe simulation results by downloading a movie of numerical simulation results from the simulation server. Overall, this system allows us to watch the same content with multiple devices and to simulate physical phenomena using various parameters.

© 2013 The Japan Society of Plasma Science and Nuclear Fusion Research

Keywords: augmented reality, virtual reality, natural marker, real-time simulation

DOI: 10.1585/pfr.8.2406102

1. Introduction

Augmented Reality (AR) enhances reality by adding virtual information and it is used in various fields. Some of the most effective applications of AR technology are maintenance and operation tasks for machines and devices. The basic concept is that AR can assist technicians by providing information about the machine and how to use it, as well as detailed documents or structural information about machines and devices.

Many applications and development libraries have been proposed for using AR technology. In particular, the ARToolkit [1] is an excellent development library. A programmer with no knowledge of AR and image processing can easily produce AR applications and environments. However, if the user wants to produce this environment on multiple computers, it is necessary to install this library on all computers before implementing AR applications that use this library. It is difficult to maintain the same environment on different computers. Therefore, we propose a system that can easily generate and share AR environments. Moreover, we add a real-time numerical simulation function that enables users to do numerical simulations by touching AR contents and to watch a movie consisting of simulation results that have been automatically converted from output numerical data to movie form in a simulation server. This means that we can simulate physical phenom-

ena associated with AR contents. For example, if an AR marker is tagged to a waveguide image, we can simulate about this waveguide by touching the AR marker and inputting parameters associated with it for numerical simulation.

2. Related Work

Many studies on AR systems have been proposed (e.g., [2, 3]). However, all of these systems use artificial markers (Fig. 1 (a)) to detect the position and pose of the AR content, which can ruin the scenery or a manual document that contains the AR content. AR systems without artificial markers have also been proposed (e.g., [4–6]). Systems without markers preserve the scenery but the accuracy is lower than an AR system with artificial markers.

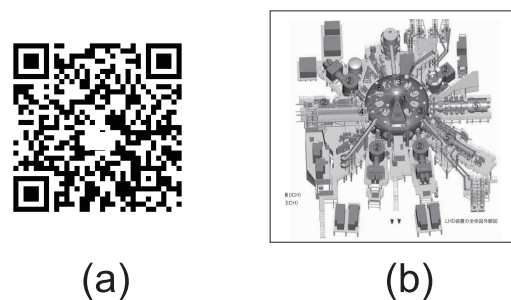


Fig. 1 Markers (a) artificial marker (b) natural marker.

author's e-mail: tamura@konan-u.ac.jp

^{*)} This article is based on the presentation at the 22nd International Toki Conference (ITC22).

We propose the use of natural markers (Fig. 1 (b)) using “junaio” that does not ruin the scenery or documents. The junaio has a highly accurate natural marker detection engine. In practical use, it easily and accurately detects images, except for thin line images and images that are blurry, too dark, or too bright.

Prototype applications have been proposed that support the maintenance and operation using AR technology (e.g., [7]). However, it is not easy to produce the same environment on tablet devices and smartphones.

2.1 The junaio framework

To overcome the aforementioned problems, we use the junaio [8] framework. The first advantage of this framework is that its front-end is open to the public and anyone can access it via the Internet. The second advantage is that the junaio software, which is an application for end-users, is distributed by commercial networks such as Google Play and Apple Store. This means that it is easy to install the application and set up the same environment for end-users. Thus, we use this framework as a front-end. The junaio also provides an image processing engine that detects natural markers as well as other AR libraries.

3. Data Flow

Figure 2 shows the data flow of the proposed system.

First, an end-user scans an artificial code (QR code) image using a web camera on a tablet device or smartphone. This artificial code contains information about the junaio web server and a “channel.” The channel is a personal space on the junaio web server where information is located about the URL of the personal web server containing the AR content. This means that the end-user does not need to input the URL directly and the developer can easily modify the URL and web site structure. Ideally, the connection to the junaio web server should be made simply by scanning the natural marker; however, no standards exist for natural markers and so it is difficult to connect to

the junaio web server by only scanning the natural marker since there is no data relevant to the AR contents in the device prior to it connecting to the junaio server. In contrast, QR code is standardized by the International Organization for Standardization; thus, the user can connect to a URL simply by scanning this code. Therefore, the user needs to scan a QR code to connect to the junaio web server. However, once the device has connected to the server, it is not necessary to scan the QR code again. This is because information about which image is the natural marker and which image is tagged to the AR content are subsequently downloaded to the device automatically. As a result, there is no need for the user to scan the QR code after the initial scan.

Next, the end-user’s device is connected automatically to the personal web site and it starts to download the AR contents, AR markers, and some control programs written in JavaScript. These AR contents were previously linked to natural markers that were produced and located on the personal server. After downloading is complete, the user scans these natural markers via a web camera and the junaio-api detects the natural markers that are already downloaded on the user’s device, before calculating the position and pose of the marker, which shows the object linked to the marker. Finally, the user can watch the virtual object, movie, or image, and information related to the natural markers.

Figure 3 shows an example of the AR movie content while Fig. 4 shows the 3D virtual model. The user can observe the 3D virtual model from any point by moving or inclining the marker image.

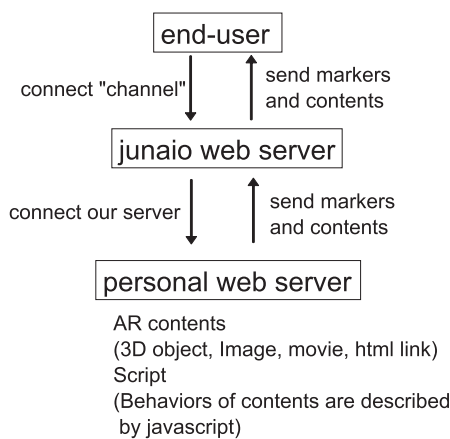


Fig. 2 Data flow.

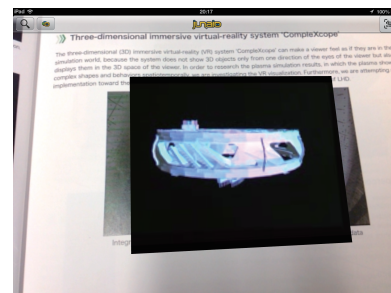


Fig. 3 AR movie content about an assembling task in LHD.

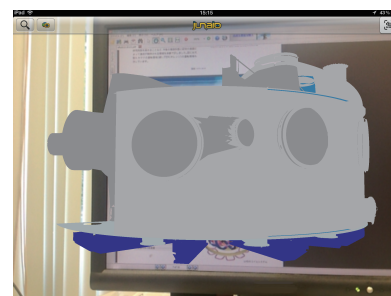


Fig. 4 3D virtual LHD model.

4. Real-Time Simulation Function and Example

Figure 5 shows the data flow of the real-time simulation function.

If the user scans a specific natural marker and taps its image (for example, the corrugated waveguide shown in Fig. 6), a message board pops up (Fig. 7). If the user pushes the button on the message board, the device connects automatically to the personal web page.

After connecting, the user is required to input the initial conditions and parameters for the numerical simulation via this web page (Fig. 8). In this example, we simulate a corrugated waveguide with a structural error using the finite-difference time-domain (FDTD) method [9]. Figure 9 shows the simulation model while the shape error model

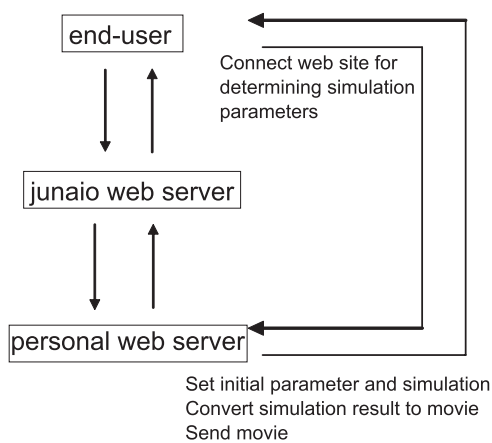


Fig. 5 Data flow of the real-time simulation system.



Fig. 6 A specific image that links to a URL.

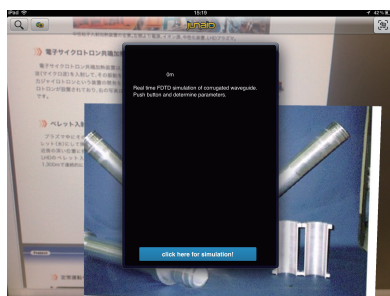


Fig. 7 Message board.

is described in eq. (1).

$$d^{error} = d + a \sin\left(\frac{2\pi}{p^{error}}z\right), \tag{1}$$

where d is the depth of a groove, a is the amplitude of the error, and p^{error} is the pitch of the error.

Using the FDTD method, the 3D Maxwell equation is solved numerically according to the perfect matched layer (PML) [10] boundary condition for the same situation as the experimental configuration. However, this equation is difficult to solve directly in the 3D environment because the data size becomes very large and the simulation period is quite long. It is necessary to perform the calculation rapidly in this system, so we use a 2D cylindrical symmetry model. If it is necessary to reduce the calculation time further, the user can change the simulation parameters, such as the total simulation time, output interval, and lattice size. However, the accuracy of the simulation results may also be reduced if the simulation time is reduced.

After completing the numerical simulation, the output data is converted to gif images by the certain time interval determined by inputting the simulation parameters. The user can watch the resulting movie simply by pushing the “show result” button. It is presented as a motion gif movie made from the output images on the personal web page. The resulting movie does not provide as the AR content because all of the AR content is downloaded when the user scans the QR code image and it is difficult to renew the AR content. Furthermore, no special movie decoder is required to watch the motion gif movie. Figure 10 shows the results

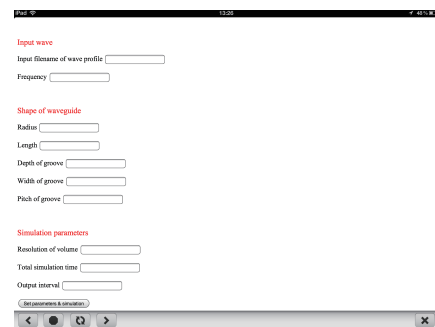


Fig. 8 Initial conditions and simulation parameters.

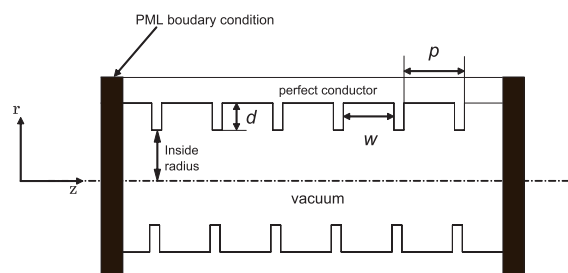


Fig. 9 Simulation model of a corrugated waveguide. w is the width of a groove, d is the depth of a groove, and p is pitch.

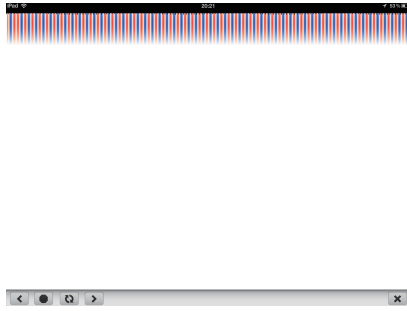


Fig. 10 Simulation result, $d = 0.75$ mm, $a = 0.3$ mm, and $p = 50$ mm.

of the FDTD simulation of the corrugated waveguide.

There is no need to produce specific simulation code to apply this system. This is because that the real-time simulation function simply overwrites the input file containing the FDTD simulation parameters and the initial condition. Therefore, this simulation code can be used in both offline and online without changing the code.

5. Conclusion

We propose a prototype AR system that supports operation and maintenance tasks using the junaio framework. The proposed AR system can be used simply by installing the application via a digital application distribution platform. Therefore, the user can use this application without

installing specific graphic libraries for image processing. This also reduces the cost when producing the environment on various devices. This system allows the user to view information about the device they are using or the document they are viewing. We also developed a real-time simulation environment that facilitates the simulation of complex physical phenomena using tablet devices, which provides the simulation results as a movie.

- [1] H. Kato, M. Billingham, I. Poupyrev, K. Imamoto and K. Tachibana, Proc. Int. Symp. on Augmented Reality (2000) p.111.
- [2] L. Naimark and E. Foxlin, Proc. IEEE/ACM Int. Symp. on Mixed and Augmented Reality (2002) p.27.
- [3] S. Saito, A. Hiyama, T. Tanikawa and M. Hirose, Proc. IEEE Virtual Reality (2007) p. 67.
- [4] V. Lepetit, L. Vacchetti, D. Thalmann and P. Fua, Proc. Int. Symp. on Mixed and Augmented Reality (2003) p.93.
- [5] I. Skrypnik and D.G. Lowe, Proc. Int. Symp. on Mixed and Augmented Reality (2004) p.110.
- [6] M. Oe, T. Sato and N. Yokoya, Proc. 14th Scandinavian Conf. on Image Analysis (2005) p.171.
- [7] J.R. Lia, L.P. Khoob and S.B. Torc, Comput. in Industry **52**, 2, 109 (2003).
- [8] L. Madden, *Professional Augmented Reality Browsers for Smartphones: Programming for junaio, Layar and Wikitude* (Wrox, 2011) p.189.
- [9] K. Kunz and R. Luebbers, *The Finite Difference Time Domain Method for Electromagnetics* (CRC Press, 1993).
- [10] J.P. Berenger, J. Comput. Phys. **114**, 2, p.185 (1994).