

The AR-Rift 2 Prototype

Anthony Steed*, Yonathan Widya Adipradana†, Sebastian Friston‡

Department of Computer Science, University College London, United Kingdom

ABSTRACT

Video see-through augmented reality (VSAR) is an effective way of combining real and virtual scenes for head-mounted human computer interfaces. In this paper we present the AR-Rift 2 system, a cost-effective prototype VSAR system based around the Oculus Rift CV1 head-mounted display (HMD). Current consumer camera systems however typically have latencies far higher than the rendering pipeline of current consumer HMDs. They also have lower update rate than the display. We thus measure the latency of the video and implement a simple image-warping method to ensure smooth movement of the video.

Keywords: augmented reality, latency, image-based rendering

Index Terms: H.5.1 [Multimedia Information Systems]: Artificial, augmented, and virtual realities;

1 INTRODUCTION

Video see-through augmented reality (VSAR) is an effective way of creating a real-time graphics system that blends real and virtual images. The key idea is that the user wearing a head-mounted display (HMD) sees a view that is a mix of real-time computer-generated graphics and output from one or more video cameras that are attached to the HMD. These video cameras supply imagery that attempts to match what the user would see if the HMD was see-through. VSAR is often compared to optical see-through augmented reality (OSAR), where computer-generated imagery is overlaid on a direct view of the scene. The advantages of each approach have been discussed at length in previous work [2]. These include that VSAR can more easily deal with occlusion between real and virtual objects, and that OSAR systems can be more lightweight, but can't occlude the real world. An obvious problem with both OSAR and VSAR systems is the latency of the displayed graphics (e.g. [6]).

Currently there is a resurgence of interest in augmented reality. New augmented reality systems are coming to market such as the Microsoft HoloLens and Daqri helmet. While there are commercially available VSAR systems, there are currently no cheap and open system for prototyping VSAR displays. Previously the AR-Rift prototype demonstrated a low cost add-on for the Oculus Rift DK1 system that used two low cost webcams to provide stereo input [8]. This system enabled other groups to build their own prototype VSAR systems.

In this paper we present the AR-Rift 2 prototype (Figure 1). This new prototype is built for the Oculus CV1 system. A first problem is that there is a mismatch between the frame rate of the HMD (90Hz) and the cameras (30Hz). A second problem is that the camera imagery has higher latency than the real-time generated imagery. We thus outline some basic steps we have taken towards better experience by estimating latency and performing image warping.

*e-mail:a.steed@ucl.ac.uk

†e-mail:yonathan.adipradana.15@ucl.ac.uk

‡e-mail:s.friston@ucl.ac.uk



Figure 1: AR-Rift 2

2 RELATED WORK

The concept of head-mounted VSAR has been being explored for over 20 years. An early system was that of Edwards et al., which used a pair of video cameras mounted to a Virtual Research Flight Helmet [3]. They discuss several options for mounting video cameras in order to minimize distortion, and match the field of view of the camera with that of the display. Many other VSAR systems have been built, see [6, 2, 7].

Various commercial VSAR products have been produced. The Vuzix Wrap 1200DXARP provided stereo VGA cameras, but was relatively low FOV [9]. The OVRVision Pro is an add-on for the Oculus Rift CV1 that provides stereo video [5]. Our intention with the AR-Rift 2 is to provide a cheap, simple and open system to allow prototyping of new VSAR systems.

3 DESIGN

3.1 Physical Build

The cameras of the AR-Rift 2 are the same as the AR-Rift. Consumer-grade webcams have not changed in a substantial way since the AR-Rift was built. While high-quality USB3 cameras are becoming available, they are still relatively expensive and beyond the budget for a simple prototype. The cameras are Logitech C310s modified with a 120°FOV lens. The CV1 is slightly lower field of view than the DK1, so this lens is sufficient to allow some overlap and latency offset, see below. The cameras are USB2 cameras. Both cameras are plugged into a small USB3 hub, and then a USB3 extension cable is run to the desktop computer. The cameras work at a steady 30Hz.

The CV1 has a hard curved front and side panels. The tracking system on the CV1 uses infra-red LEDs under the front panel and across other rigid surfaces on the HMD. The camera holders are thus designed to minimize the supporting frame. The mount is designed in two main pieces (two of each are built): a camera holder which the camera board clips in to, a frame piece that is designed to flex a small amount and hold the HMD on the top and bottom surfaces. When the frame is placed on the CV1, small rubber feet are added to hold the frame pieces in place. The frame includes screw holes to add optional bracing strips. These bracing strips can be used to ensure the frames are planar, and also to set the inter-

pupillary distance. The holder and frame are designed so that the cameras are inline with the centre of the lenses of the CV1.

The frame, holder and bracing strips were modelled in SolidWorks. There were designed to be 3D printed on low-cost additive manufacturing systems. We used an Ultimaker 2+, with the standard 0.4mm nozzle, printing in grey PLA. We used Cura 2 to generate the printer code with the default high-quality settings.

3.2 Software

The integrated application including image warping was built in Unity 5.4. Head-tracking is provided by the Oculus CV1's camera. We support the Oculus Touch for interaction with the scene. A default scene including avatar animation, interaction with objects and AR/VR switching is available.

3.3 Latency Measurement

A key property of the system is the that webcam images have a higher latency than those generated by the GPU. We thus had to estimate the end-to-end latency of the video. That is, the time between movement of the cameras, or of objects in their field of view, and the change in pixels on the screen. A review of latency estimation techniques can be found in [4]. As we did not want to modify the system hardware, we opted to use the *frame counting* method. Typically, high speed video is taken showing both the display and the real world. The display reflects the state of the real world. E.g. a tracked target drives a virtual object, or the display itself is tracked, changing the virtual viewpoint. The target/display is moved, and the number of frames between the real-world movement and the corresponding change in the virtual world reveals the latency.

In our situation, we cannot film the screen directly because the lenses are not removable. We designed a sled to hold a 1000Hz camera and the headset, fixed relative to each other. The camera could see both the Oculus Rift lenses and the real world in front of the AR cameras. In the real world, two highly distinctive coloured cards were placed on a wall. The sled was slid along rails. Figure 2 Top shows configuration of the sled, rails and cards. The image on the Oculus Rift's screen is highly distorted, but the contrast between the two cards allows their seam to be easily distinguished and its lateral movements tracked in a small range. To count frames we identified turning points in the motion of the sled as it was moved left and right along the rails. For the coloured cards this is easy, as the extreme frame at one end of the motion can be identified. For the image of the cards through the lenses, we had to identify the image where the change in shape of the colours reverses. Please see our website for videos showing example characteristic frames. Figure 2 Bottom shows how the image on the high speed camera captures the lenses and the coloured cards.

By counting frames, the video latency of the prototype was measured to be 70-80ms with an average of 76ms. We expect a range of times, because each video frame is shown multiple times on the display, see next section.

3.4 Image Warping

A second key property of the system is that the webcams have a fixed rate of 30Hz. We can thus expect that each webcam frame would be displayed 3 times on average. To improve apparent smoothness we can warp each video frame so that the video is not static while real-time generated portions of the image are changing. We use the latency estimate, to predict the head position when the video frame was taken, and draw the video to a flat texture plane that is fixed in world coordinates. This is similar in nature to the asynchronous timewarping technique in the Oculus SDK [1].

4 CONCLUSIONS

The AR-Rift 2 prototype enables experimentation with video augmented-reality. Our prototype highlights challenges for upcom-



Figure 2: Latency estimation rig. Top: Showing HMD and camera strapped to a sled that is mounted on rails. Bottom: Showing the camera on the rig capturing both the red/blue target on the wall and the distorted view of the red/blue target through the lens.

ing hardware that may include stereo video input: the latency and framerate requirements are quite demanding.

We have made the build plans, method for measuring latency and Unity project for the demonstrations available under open source licenses at <http://vr.cs.ucl.ac.uk/ar-rift/>.

REFERENCES

- [1] M. Antonov. Asynchronous Timewarp Examined. <https://developer3.oculus.com/blog/asynchronous-timewarp-examined/>, 2015. [accessed 12/12/2016].
- [2] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre. Recent advances in augmented reality. *IEEE computer graphics and applications*, 21(6):34–47, 2001.
- [3] E. K. Edwards, J. P. Rolland, and K. P. Keller. Video see-through design for merging of real and virtual environments. In *Virtual Reality Annual International Symposium, 1993., 1993 IEEE*, pages 223–233. IEEE, 1993.
- [4] S. Friston and A. Steed. Measuring Latency in Virtual Environments. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Virtual Reality 2014)*, 20(4), 2014.
- [5] OVRVision. <http://ovrvision.com/entop/>, 2016. [accessed 12/12/2016].
- [6] J. P. Rolland and H. Fuchs. Optical versus video see-through head-mounted displays in medical visualization. *Presence: Teleoper. Virtual Environ.*, 9(3):287–309, June 2000.
- [7] D. Schmalstieg and T. Hollerer. *Augmented Reality: Theory and Practice*. Addison Wesley, 2016.
- [8] W. Steptoe, S. Julier, and A. Steed. Presence and discernability in conventional and non-photorealistic immersive augmented reality. In *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on*, pages 213–218. IEEE, 2014.
- [9] Vuzix. <https://www.vuzix.com/Products/LegacyProduct/4>, 2016. [accessed 12/12/2016].