Camera Time Warp: Compensating Latency in Video See-Through Head-Mounted-Displays for Reduced Cybersickness Effects

Jann Philipp Freiwald* University of Hamburg

Nicholas Katzakis† University of Hamburg

Frank Steinicke‡ University of Hamburg

ABSTRACT

Camera Time Warp or "CamWarp" is an extension of current reprojection techniques for video see-through augmented reality (AR), which significantly reduces the registration error between captured real-world videos and rendered virtual images. Experiment participants were asked to report discomfort while moving their head in a Fitts' Law inspired pattern. Results suggest that CamWarp can reduce discomfort and cybersickness symptoms for all tested camera configurations. In a second experiment participants were asked to move physical objects on a projected path as quickly and precisely as possible. CamWarp had a positive effect on speed and accuracy.

Index Terms: Human-centered computing—Human computer interaction (HCI)—Epirical studies in HCI—Cyersickness; Human-centered computing—Interaction paradigms—Mixed / augmented reality

1 Introduction and Related Work

One of the major drawbacks of today's video see-through (VST) head-mounted displays (HMDs) is latency between real-world video images and renderings of the virtual objects. In dynamic scenes, when the user's head, limbs or other real-world objects move, the delay may result in significant registration errors causing discomfort, cybersickness and degraded interaction performance [1, 9]. Despite increases in camera sensor resolution and framerates, a certain amount of latency from recording to display will most likely persist. With this latency being a constant in HMDs, compensating its effects remains a necessity even when the hardware of VST HMDs advance.

Fernandes and Feiner reported a decrease in negative effects of cybersickness by partially overlaying a black ring from the outer borders to the center of the field of view [3].

Van Waveren et al. presented "Asynchronous Time Warp" (ATW) [8]. ATW was a 2D transformation of the rendered image, shifting it on the horizontal and vertical axes depending on the head movement that had occurred immediately after completing the framebuffer. When ran asynchronous to the refresh rate of the HMD, the perceived frame rate of an application was increased, especially if the computer was not able to deliver the maximum frames per second that were supported by the headset by essentially interpolating frames.

Yokokohji et. al [9] implemented a predictive approach that shifts the virtual objects based on an estimation of the user's imminent head movement. They matched the update rate of these transformations to the FPS setting of the AR cameras to reduce registration errors.

Kim et al. [4] proposed "DotWarp", a technique that relies on an additional high frame rate infrared camera. By blending the stereo camera images with the information from the infrared camera, they achieved an improvement in the perceived registration error.

*e-mail: freiwald@informatik.uni-hamburg.de
†e-mail: nicholas.katzakis@uni-hamburg.de
‡e-mail: frank.steinicke@uni-hamburg.de

Bajura & Neumann's [1] premise was that at the time, camera technology had a lower latency than tracking technology. They induced a latency to the video stream by buffering the actual video stream's images. With this, they matched the latency of both technologies, resulting in a reduction of registration errors.

Davis et al. [2] introduced an AR camera stabilization algorithm that freezes the pose of the image plane until a new frame arrives from the stereo cameras. This reduced the negative effect of the discrepancy from physical camera FPS to the HMD's refresh rate.

Orlosky et al. [6] proposed to synchronize the latency of multiple AR camera systems by storing the headset's rotational data in a queue structure each frame. By adjusting the queue length for each individual camera, they compensated the latency from image recording to delivery into the 3D engine.

2 Camera Time Warp

Camera Time Warp or "CamWarp" is an extension of the works of [1, 2, 6]. It is a reprojection technique for VST-AR that significantly reduces the registration error between captured real-world videos and rendered virtual images. We combined the camera stabilization from [2] with the latency compensation of [6]. We induce an additional delay to the tracking system to match the camera latency, but apply it in a separate stream to the image plane that displays the camera images. We store the positional and rotational data of the HMD in a queue and update the plane's pose when a new camera frame arrives. The resulting black border during head movements are masked with an enlarged and burred copy of the current frame.

3 Cybersickness Experiment

The goal of this experiment is to confirm our hypothesis that CamWarp improves subjective comfort and cybersickness effects. We used ATW as baseline and compared it to CamWarp+ATW, similar to [3]. In the experiment, participants had to move their head in a Fitts' Law and ISO 9241-inspired pattern at different speeds, while wearing an HMD with a stereo camera extension. We tested with 30, 60 and 90 FPS to have equidistant conditions while having a low baseline and the highest setting the driver supports. The resolution was set to 480p during all trials for consistency.

One of our hypotheses was that the level of discomfort would increase with higher head movement speeds. Through a short pilot study we derived a speed of 40 °/s as baseline and multiples of it with 80 and 120 °/s for medium and fast movements respectively.

24 participants (19 and 69 years old - $M = 34.39$, $SD = 13.02$) took part in the experiment. To ensure that the data of all participants is comparable, their head movement was restricted by giving them the task to gaze at a virtual red sphere which moved in a Fitts' law inspired pattern [5] at a speed that produced the desired angular head movement. A ring of 9 white spheres was placed 1 meter away from the participant. The diameter of the ring was defined by the distance to the user and the maximum desired head turning angle of 45° from the center. The red sphere starts at the center of the ring and then proceeds to move in a zig-zag pattern [5]. The participants were told to keep the red sphere as close to the center of their FOV as possible while paying attention to the video feed.

After each trial, a panel was displayed in front of the ring that asked the participant to input their subjective level of discomfort caused by the trial, on a scale of 1 (no problems) to 10 (severe discomfort) [3, 7]. This is based on studies by Fernandes [3] and Rebenitsch [7], who used this value in their analysis to derive conclusions about cybersickness related symptoms. We excluded 0 from our scale because of reports of accidental inputs from initial test runs. The experiment had to be a within-subject design due to the nature of discomfort. By testing all configurations, participants could rate their experiences relative to each other.

The results show that CamWarp received better discomfort ratings compared to the baseline in all cases. We recorded the following means for CamWarp for the 30, 60 and 90 FPS settings: 3.58 ($SD = 1.99$), 1.69 ($SD = 0.69$) and 1.61 ($SD = 0.58$). The baseline results read: 6.58 ($SD = 2.27$), 2.99 ($SD = 1.98$) and 2.38 ($SD = 1.65$). We found these mean levels of discomfort for head movement speeds of 40, 80 and 120 °/s: 3.02 ($SD = 0.18$), 3.13 ($SD = 0.19$) and 3.25 ($SD = 0.21$). For the analysis we ran a repeated measures ANOVA test at the 5% significance level for all variables. The results show a significant influence of the technique ($F_{1,23} = 48.73, p < 0.001$) and a significant influence of FPS ($F_{1,23} = 136.34, p < 0.001$) on subjective levels of discomfort each. This corresponds to participants reporting their discomfort lowest for 90 FPS and CamWarp enabled and highest for 30 FPS and CamWarp disabled. The significant interaction effect between technique and FPS on subjective levels of discomfort ($F_{1,23} = 32.56, p < 0.001$) supports this.

The head movement speed condition surprisingly did not have a significant effect on discomfort ($F_{1,23} = 1.23, p = 0.279$). However, we found a significant interaction effect between FPS and head movement speed ($F_{1,23} = 9.59, p = 0.005$). Users reported their highest levels of subjective discomfort during fast head movement speeds and 30 FPS. This indicates that head movement speed has a stronger influence on comfort if the FPS setting causes discomfort by itself, having essentially a super-additive effect.

The mean for 60 FPS with the technique enabled is lower than that of the baseline 90 FPS, which suggests that it is even possible to sacrifice FPS for an increase in camera resolution. That could potentially increase the comfort even further. This suggests that there is a point at which increasing FPS has a smaller impact on comfort than increasing camera resolution, and that these parameters need to be balanced carefully to optimize user experience. Studies in VR [3] that reported a significant effect of their technique on discomfort concluded that their technique reduces cybersickness effects. Since our study showed a similar effect, this subsequently implies that our approach likewise decreases cybersickness effects.

4 User Performance Experiment

The discomfort experiment demonstrated the effectiveness of CamWarp for head movements in scenes with a static real-world environment. We therefore asked the question: Does our approach have similar beneficial effects in a real-world interaction scenario? Like the previous study, we used ATW as a baseline and compared it to CamWarp+ATW. We compared them in regards to execution speed and accuracy in hand-eye coordination tasks. Participants had to move physical objects from one shelf compartment to another through an overlayed 3D-tunnel. This experiment is inspired by the steering law, which predicts movement speed in a tunnel on a 2D-plane. It reflects scenarios like a mechanic receiving AR support during engine maintenance, a surgeon getting instructions for incisions or a warehouse worker sorting the inventory. We opted to use the same camera configurations as in the first experiment (30, 60 and 90 FPS at 480p per eye).

23 participants (20 and 38 years old - ($M = 27.33$, $SD = 4.49$) took part in the experiment. Following calibration, participants were asked to move one of the Vive controllers from the compartment it was located in to an unoccupied compartment through the tunnel as quickly as possible. They were asked to keep the center of the Vive controller's Ring as close as possible to the indicated center of the tunnel. Participants were asked to maintain a balance between speed and accuracy. As soon as the indicated controller reached its destination, the active controller and its destination were picked at random for the next trial.

For accuracy, we measured the average distance from the center of the Vive controller's ring to the closest point on the center line (lower is better). We found the following accuracy means for CamWarp for the 30, 60 and 90 FPS settings: 2.51 ($SD = 1.40$), 2.22 ($SD = 1.36$) and 2.07 ($SD = 1.59$). These are the baseline results: 3.93 ($SD = 2.55$), 3.46 ($SD = 1.77$) and 2.90 ($SD = 1.64$). The results of the ANOVA test show a significant effect of the technique ($F_{1,22} = 27.74, p < 0.001$) and a significant effect of FPS ($F_{1,22} = 13.82, p = 0.002$) on the achieved accuracy.

We tracked the average velocity of the Vive controller as a function of the time from picking it up to laying it down and the length of the current tunnel. The velocity means for CamWarp for the 30, 60 and 90 FPS settings: 0.93 ($SD = 0.30$), 0.98 ($SD = 0.33$) and 1.13 ($SD = 0.39$). Baseline: 0.85 ($SD = 0.36$), 0.87 ($SD = 0.39$) and 0.88 ($SD = 0.52$). The results of the ANOVA test show a significant effect of the technique ($F_{1,22} = 16.70, p < 0.001$) and a significant effect of FPS ($F_{1,22} = 8.16, p = 0.009$) on the velocity of the controllers. We found an interaction effect between technique and FPS, which tells us that CamWarp has a stronger effect at certain FPS configurations ($F_{1,22} = 9.75, p = 0.005$). This is the case at the 90 FPS value, where CamWarp performed even better than expected.

In all tested configurations CamWarp had a positive impact on accuracy and velocity. At the highest available FPS configuration, on average participants moved objects 28.6% faster while being 22.1% closer to the perfect trajectory. The interaction effect between FPS and technique suggests that high FPS have a positive effect on the effectiveness of CamWarp in regards to velocity.

References

[1] M. Bajura and U. Neumann. Dynamic registration correction in video-based augmented reality systems. *IEEE Computer Graphics and Applications*, 15(5):52–60, 1995.

[2] B. A. Davis, K. Bryla, and P. A. Benton. *Oculus Rift in Action*. Manning Publications Company, 2015.

[3] A. S. Fernandes and S. K. Feiner. Combating vr sickness through subtle dynamic field-of-view modification. In *2016 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 201–210, March 2016. doi: 10.1109/3DUI.2016.7460053

[4] P. Kim, J. Orlosky, K. Kiyokawa, P. Ratsamee, and T. Mashita. [poster] dotwarp: Dynamic object timewarping for video see-through augmented reality. In *2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pp. 184–185, Oct 2017. doi: 10.1109/ISMAR-Adjunct.2017.61

[5] I. S. MacKenzie. Fitts' law as a research and design tool in human-computer interaction. *Hum.-Comput. Interact.*, 7(1):91–139, Mar. 1992. doi: 10.1207/s15327051hci0701_3

[6] J. Orlosky, P. Kim, K. Kiyokawa, T. Mashita, P. Ratsamee, Y. Uranishi, and H. Takemura. Vismerge: Light adaptive vision augmentation via spectral and temporal fusion of non-visible light. In *Mixed and Augmented Reality (ISMAR), 2017 IEEE International Symposium on*, pp. 22–31. IEEE, 2017.

[7] L. Rebenitsch and C. Owen. Individual variation in susceptibility to cybersickness. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, UIST '14, pp. 309–317. ACM, New York, NY, USA, 2014. doi: 10.1145/2642918.2647394

[8] J. M. P. Van Waveren. The asynchronous time warp for virtual reality on consumer hardware. In *Proceedings of the 22Nd ACM Conference on Virtual Reality Software and Technology*, VRST '16, pp. 37–46. ACM, New York, NY, USA, 2016. doi: 10.1145/2993369.2993375

[9] Y. Yokokohji, Y. Sugawara, and T. Yoshikawa. Accurate image overlay on video see-through hmds using vision and accelerometers. In *Proceedings IEEE Virtual Reality 2000 (Cat. No.00CB37048)*, pp. 247–254, 2000. doi: 10.1109/VR.2000.840505