Using External Video to Attack Behavior-Based Security Mechanisms in Virtual Reality (VR)

Robert Miller* Natasha Kholgade Banerjee† Sean Banerjee‡

Clarkson University

Abstract

As virtual reality (VR) systems become prevalent in domains such as healthcare and education, sensitive data must be protected from attacks. Password-based techniques are circumvented once an attacker gains access to the user's credentials. Behavior-based approaches are susceptible to attacks from malicious users who mimic the actions of a genuine user or gain access to the 3D trajectories. We investigate a novel attack where a malicious user obtains a 2D video of genuine user interacting in VR. We demonstrate that an attacker can extract 2D motion trajectories from the video and match them to 3D enrollment trajectories to defeat behavior-based VR security.

Index Terms: Human-centered computing—Human computer interaction (HCI)—Interaction paradigms—Virtual reality; Security and privacy—Security services—Authentication—Biometrics

1 Introduction

VR applications in domains such as healthcare, teleoperation, and education are expected to store sensitive personal data on the users and must be protected from access by malicious agents. A large body of research has emerged on performing identification and authentication of users by tracking the behavior of users, i.e., the 3D trajectories of the VR headset and hand controllers, and using them as a signature to provide security [2–4, 6–8]. Manual methods to defeat behavior-based security have been evaluated [5], however the complexity of human actions renders manual mimicry challenging [8]. We investigate a new attack scenario where a malicious user gains access to 2D video capture of a user performing VR interactions and uses the video for automated defeat of behavior-based security in VR. Web-based closed circuit television monitoring systems or IP cameras are commonplace in schools, offices and clinics, and contain security vulnerabilities. We assess the potential of performing a video-based attack, by matching 2D video performance of a user to 3D trajectories for a set of users acquired during an enrollment phase. The enrollment trajectories would ordinarily be used by a genuine matcher to match against runtime 3D trajectories from a VR device. Our approach makes the assumption that the malicious agent has no physical access to the user's environment. The malicious agent can only perform remote attacks and can inject a rogue matching algorithm into the system. Using the videos and 3D trajectories in the 41-subject multi-system dataset of Miller et al. [6, 7], we demonstrate maximum accuracies of 82.2%, 43.2%, and 34.4% at same-system/same-session, same-system/cross-session, and cross-system attack. While accuracies are still low, they indicate potential for rogue matching algorithms to circumvent behavior-based security mechanisms in VR.

*e-mail: romille@clarkson.edu

†e-mail:nbanerje@clarkson.edu

‡e-mail:sbanerje@clarkson.edu

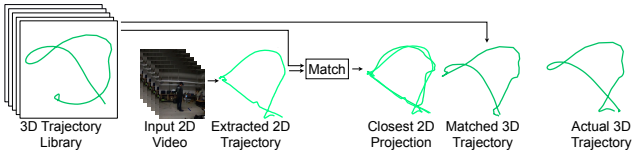

Figure 1: We use external video to attack on behavior-based VR security mechanisms by extracting 2D trajectories from the video and matching to 3D library trajectories.

2 Method

We use the Miller et al. [6, 7] dataset, which consists of 41 subjects performing a ball throwing task using three VR systems—the HTC Vive and Vive Cosmos, and the Oculus Quest. Each user provides 2 sessions per system, with 10 throws per session. The dataset contains 3D trajectory tracks for the headset and hand controllers, and external video recorded at 240 FPS and 1280×960 resolution using a GoPro Hero 7 camera. The GoPro video acts as the runtime data that has been obtained by a malicious agent. We eliminate data from the first Quest session as GoPro videos were incorrectly captured for 4 users. Our matching approach for the attacker estimates a projection matrix that best aligns the 3D trajectory of the right controller to the motion of the user's right hand in the video obtained by tracking the right controller over video frames. We perform the matching using Cosmos video, as the bright Cosmos controller enables color-based tracking, in comparison to the Quest and Vive controllers whose black color resembles dark hair, clothing, or background elements.

We automatically extract a single point representing an anchor for the controller in each Cosmos video frame. We initialize a search location for the controller by automatically aligning a skeleton to the user using OpenPose [1], extracting the positions of the right wrist and elbow, and aligning rectangular regions as shown in Figure 2 by comparing the wrist and elbow confidences to a threshold. To extract the controller, we retain search region pixels with hue $\in [160^\circ, 300^\circ]$, saturation $\in [0, 145]$, and value > 195. To remove non-controller pixels with similar intensities, if the variance of the pixel coordinates exceeds 12,500, we cluster the pixel coordinates using k-means into two groups, and pick the group with cluster mean closest to the prior frame anchor as the current frame anchor. Otherwise, we assume the region lacks non-controller pixels, and return the mean of the pixel coordinates as the current frame anchor. Figure 2(e) shows cluster centers for an example search region in Figure 2(d). We concatenate all anchors into a single 2D trajectory as shown in Figure 2(f).

The 3D trajectories may lack perfect alignment 2D video from the user due to start frame misalignments and variable start times. We upsample the 3D and 2D trajectories to have 270 points per trajectory, set up a sliding window of size $F = 240$ frames over the 2D trajectory, and match the sliding window to a fixed window extracted from the 3D trajectory. To address variable start times, we use two fixed windows from the 3D trajectory—one that starts at the first frame, and one that starts 20 frames into the motion. Given a fixed 3D window and a sliding 2D window, we perform random sample consensus (RANSAC) over 100 iterations to estimate a projection matrix $\mathbf{P}$ of

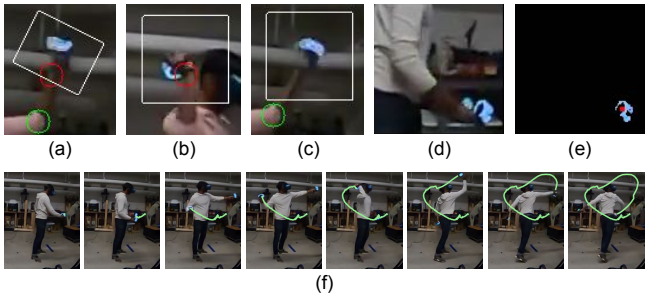

Figure 2: Search regions generated when OpenPose provides high confidence detections for (a) both wrist and elbow, (b) wrist only, and (c) elbow only. The confident wrist and elbow are marked in red and green respectively. (d) Extracted search region and (e) cluster pixels in white and optimal cluster center in red. (f) Film strip with extracted trajectory plotted as evolving over task.

size 3×4 that optimally lines the 3D window with the 2D window. Within each RANSAC iteration, we randomly sample a set S of 6 points to estimate $\mathbf{P}$, where $S \subset \{1, 2, \cdots, F\}$. We estimate $\mathbf{P}$ as the values for its rows $\mathbf{p}_1^T$, $\mathbf{p}_2^T$, and $\mathbf{p}_3^T$ that optimize the algebraic error $\sum_{i \in S} \left(u_i \mathbf{p}_3^T \mathbf{X}_i - \mathbf{p}_1^T \mathbf{X}_i\right)^2 + \left(v_i \mathbf{p}_3^T \mathbf{X}_i - \mathbf{p}_2^T \mathbf{X}_i\right)^2$, where $\mathbf{X}_i$ represents the homogeneous coordinates for the i^{th} 3D point, u_i and v_i represent the coordinates and for the corresponding 2D point $\mathbf{x}_i$. We project each point $\mathbf{X}_i, i \in \{1, F\}$ in the 3D trajectory into 2D by computing the 2D coordinates $\left(\mathbf{p}_1^T \mathbf{X}_i / \mathbf{p}_3^T \mathbf{X}_i, \mathbf{p}_2^T \mathbf{X}_i / \mathbf{p}_3^T \mathbf{X}_i\right)$. We retain the value of $\mathbf{P}$ that provides the lowest inlier error over all RANSAC iterations containing n or more inliers, i.e., points for whom the re-projection error $e_i = \left(\mathbf{p}_1^T \mathbf{X}_i / \mathbf{p}_3^T \mathbf{X}_i - u_i\right)^2 + \left(\mathbf{p}_2^T \mathbf{X}_i / \mathbf{p}_3^T \mathbf{X}_i - v_i\right)^2$ falls below a threshold of 50 pixels. For each inlier set, we re-estimate $\mathbf{P}$ by optimizing the algebraic error over the inliers, and obtain the inlier error by summing e_i over all inliers. The lowest inlier error over all RANSAC iterations with inlier count above 220, all 2D sliding windows, and all fixed 3D windows forms the distance between the 3D trajectory and the 2D video. The typical use of RANSAC requires a precise underlying geometric model which does not exist for trajectories where the motion is mismatched from the video. In several cases of non-matching trajectories, RANSAC is unable to obtain return a projection matrix where the error for 220 or more points falls below 50 within 100 iterations. In this case, we set the projection error to be a large value at 10^{20}.

3 Results

Table 1 shows results by matching 2D videos from the two Cosmos sessions against 3D enrollment trajectories from the same and different Cosmos sessions, from two Vive sessions, and from the second Quest session. 'Cn', 'Vn', and 'Qn' represent data from the n^{th} session of the Cosmos, Vive, and Quest respectively. Enrollment and query rows represent sessions from which 3D enrollment trajectories and 2D query videos are obtained. We analyze success of defeating a VR application when an attacker replaces a behavior-based security mechanism with the proposed matching approach in order to present videos that enable the attacker to masquerade as the genuine user for identification, and to remain verified for authentication. We perform identification by labeling the trajectory with the identity of the user with the nearest matching 3D trajectory in the enrollment set. We perform authentication of a 2D video against a particular 3D trajectory by comparing all video-trajectory distances against a threshold. We show identification accuracies and equal error rate (EER) for authentication obtained by computing false accept and false reject rates using the best matches over all throws for each enrollment user. We obtain the highest accuracy of 82.2% and lowest EER of 11.4% when Cosmos video is matched to Cosmos trajectories in the second session. The starred columns in Table 1 provide results when we remove the trajectory corresponding to each presented query video from the enrollment set, in which case accuracy drops to 54.6% and 59.5% for same day Cosmos matching, and EER rises to minimum of 15.4%. Cross-day Cosmos accuracies drop to slightly over 37-43%, and EER rises to 23.6%. Cross-system accuracies drop to 28-34% for the Vive-Cosmos pair and 22-24% using the Quest-Cosmos pair. In cross-system matching, lowest EER of 26.0% is obtained using session 2 Cosmos and Vive data, with Quest-Cosmos EER values being higher, mirroring identification.

Enr.	C1	C1⋆	C2	C2⋆	C1	C2	V1	V1	V2	V2	Q2	Q2
Query	C1	C1	C2	C2	C2	C1	C1	C2	C1	C2	C1	C2
Acc.	79.0	54.6	82.2	59.5	43.2	37.3	31.7	28.3	32.7	34.4	24.7	22.9
EER	14.1	21.2	11.4	15.4	23.6	23.6	27.1	26.3	25.9	26.0	31.7	30.7

Table 1: Results in percentages showing average accuracy (Acc.) and EER. ⋆We remove the precise trajectory corresponding to the query video from the enrollment (Enr.) set to analyze success by comparing against other trajectories provided by the user on the same day.

4 Discussion

Overall, our results indicate that behavior-based security systems that rely on a single system during enrollment and use are more vulnerable. Vulnerability is increased when enrollment and use occur on the same day, for instance if an attacker is able to determine when a new user is being enrolled or if library trajectories are being updated due to user profile changes. In future, we will explore learning-based methods to improve accuracy of cross-session and cross-system matching of 3D trajectories to features extracted from 2D video. We will tie object detection and tracking approaches to localize and track the VR hand controllers for 2D trajectory extraction. Our approach assumes that the attacker can inject a rogue matching algorithm. Our future work will provide automated methods to synthesize 3D trajectories from 2D videos to enable attacks independent of the matching method.

References

[1] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *Trans. on Pattern Analysis and Machine Intelligence*, 43(1):172–186, Jan 2019.

[2] A. Kupin, B. Moeller, Y. Jiang, N. K. Banerjee, and S. Banerjee. Task-Driven Biometric Authentication of Users in Virtual Reality (VR) Environments. In *Proc. MMM*. Springer, Berlin, Germany, 2019.

[3] J. Liebers, M. Abdelaziz, L. Mecke, A. Saad, J. Auda, U. Grünefeld, F. Alt, and S. Schneegass. Understanding user identification in virtual reality through behavioral biometrics and the effect of body normalization. In *Proc. CHI*. ACM, New York, USA, 2021.

[4] F. Mathis, H. I. Fawaz, and M. Khamis. Knowledge-driven biometric authentication in virtual reality. In *Proc. CHI Extended Abstracts*. ACM, New York, USA, 2020.

[5] F. Mathis, J. Williamson, K. Vaniea, and M. Khamis. Rubikauth: Fast and secure authentication in virtual reality. In *Proc. CHI Extended Abstracts*. ACM, New York, USA, 2020.

[6] R. Miller, N. K. Banerjee, and S. Banerjee. Within-system and cross-system behavior-based biometric authentication in virtual reality. In *Proc. VRW*. IEEE, New York, USA, 2020.

[7] R. Miller, N. K. Banerjee, and S. Banerjee. Using siamese neural networks to perform cross-system behavioral authentication in virtual reality. In *Proc. VR*. IEEE, New York, USA, 2021.

[8] I. Olade, C. Fleming, and H.-N. Liang. Biomove: Biometric user identification from human kinesiological movements for virtual reality systems. *Sensors*, 20(10):2944, May 2020.