2019 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)

Realtime Behavior-Based Continual Authentication of Users in Virtual Reality Environments

Robert Miller, Ashwin Ajit, Natasha Kholgade Banerjee, and Sean Banerjee
Clarkson University, Potsdam, NY
{romille,ajita,nbanerje,sbanerje}@clarkson.edu

***Abstract*—In this work, we present a realtime interactive system that uses the trajectories of the device controllers and headset of a virtual reality (VR) system to perform continual authentication of users interacting in VR. Our system identifies a user within $0.21 \pm .04$ seconds of provision of a trajectory for a ball-throwing application. Given an interaction session authorized for a user within an organization, our system continually authenticates the current user by computing matches for the positions and orientations of the points on trajectories of the right hand controller, left hand controller, and headset for the user against controller and headset trajectories in a library of users. The system combines the position and orientation matches using a neural network pre-trained to predict confidences on matches computed within the library, and provides a speech interface to output the identified user as the library user with the maximum value of predicted confidence. The system uses the identity of the user authorized to use a session and a confidence threshold to detect if the user is genuine, a malicious user internal to the organization, or an external intruder. Our system can be readily ported into virtual reality applications that require realtime behavioral authentication of users for secure access.**

***Index Terms*—virtual reality, VR, biometrics, authentication**

I. Introduction

In this work, we present our approach for continually authenticating a user in a VR application in realtime by utilizing their actions in the virtual space. Our approach provides a speech response in $0.21 \pm .04$ seconds of a user performing an action, enabling seamless authentication without needing the user to stop their activity in VR to provide additional information. Our system can be deployed in an organization where multiple users have varying levels of authorization to access VR systems within the organization. Given a user pre-authorized to use a session using traditional means such as PIN or password, our system performs continual authentication using the trajectories of their headset and hand controllers during interaction, and detects hijacking of the session by impostors. The system uses the approach of Ajit et al. [1] to predict a confidence of matching a user interacting with a session against a library of users comprising the organization. The system uses the identified user identity and a confidence threshold to detect if the user is (a) the genuine user authorized for that session, (b) an unauthorized malicious user within the organization, or (c) an intruder external to the organization.

II. Realtime Interactive User Authentication

Our realtime interactive user authentication system consists of a custom ball-throwing application developed in Unity for an HTC Vive running on an AMD Ryzen 2700X machine with 64GB of RAM, 500GB SSD store, and an NVIDIA GTX 1080 Ti GPU. The application consists of a trajectory recording phase and an authentication phase. The authentication phase can be configured by the system operator to set parameters such as the number of library trajectories used to train the system, the scheme used to align the trajectories, and the match metric used to compare trajectories. Once the trajectory phase records the headset and hand controller trajectories, the authentication phase provides a spoken output conveying the user's identity, and whether they are a genuine user, an impostor within the organization, or an external impostor.

a) Ball-Throwing Application: We have developed a custom ball-throwing interaction in Unity for the HTC Vive, the implementation details of which are discussed in [2]. Using the application, we have collected a dataset of 33 subjects, with each subject providing 10 library throws on a training day, for a total of 330 trajectories from both hand controllers and the headset. Creation of a library of users during a user registration period would be standard operational procedure within an organization using such VR systems. For each trajectory, our system records the position and orientation of the corresponding device. Prior to regular operation, the system first undergoes a training phase discussed in Part (b) below. Once our system has been trained, the user is authenticated based on new interactions with the system by comparing each new throw to the existing library of throws as discussed in Parts (c) to (e).

b) System Training: Our system is trained based on the approach discussed in Ajit et al. [1]. We align each library trajectory using one of the five alignment methods discussed in [1]. We then compute position and orientation matches using one of three match metrics discussed in [1] for the three devices using all N^2 trajectory pairs where $N = 330$. We train a perceptron neural network to output a high confidence score for trajectory pairs from the same user, and a low confidence score for different users. Our interface allows the system operator to choose the alignment method, the match metric, the number of trajectory points used during matching, the features and devices used for training the network, and the number of library throws per user used during training. Our timing results in this work are obtained by using parameters that yield high accuracy in [1]. In particular, we use the bounding box center for alignment, the nearest neighbor metric, 100 trajectory points, the orientation of the right hand controller and headset, and all 10 library throws. While the nearest projection also yields high accuracies in [1], the accuracies are comparable to the nearest neighbor, however, the distance

978-1-7281-5604-0/19/$31.00 ©2019 IEEE
DOI 10.1109/AIVR46125.2019.00058

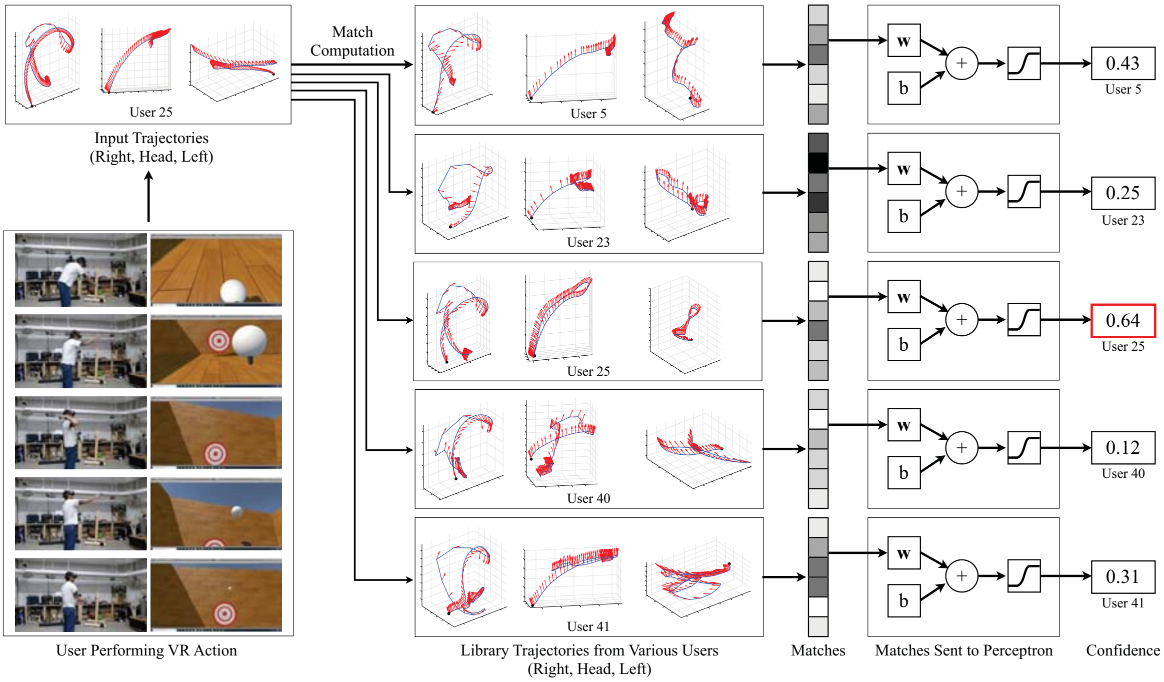

Fig. 1. Overview of user authentication process. As part of a session, the user performs a ball-throwing action in VR, and the VR system records the position (in blue) and orientation (in red vectors) for the trajectories of the right hand controller, headset, and left hand controller. Matches are computed between the positions and orientations of the user trajectory against trajectories for a variety of users in a library set. The computed matches are sent to a perceptron neural network pre-trained to predict confidence of matching. The training user with the maximum confidence (shown by the red box) is tagged as the identified user. If the confidence is above a threshold, the user is returned as genuine if the user's identity matches the user identity authorized to use the session, else is returned as an impostor within the organization. If the confidence falls below a threshold, the user is flagged as an external impostor.

computation time with nearest neighbors is faster than nearest projection during authentication.

c) Known User Authentication: During system user, a user who has provided prior training data is first authorized to access the system using a traditional PIN or password-based approach. As the user interacts with the VR environment, the system continually authenticates the user to guarantee that the user is the genuine user with permission. As demonstrated in Figure 1, once the user performs a ball-throwing action, the trajectory recording phase of the application records the position and orientation of the headset and hand controllers during their interaction. The authentication phase then aligns each device's trajectory, and computes position and orientation matches for the three devices against the 330 trajectories in the library set. The number of library trajectories, number of points per trajectory used for matching, alignment method, and distance metric is the same as pre-specified by the system operator during training. Matches from features and devices used to train the perceptron are passed to the network, and the user for the library trajectory corresponding to the maximum confidence predicted by the network is tagged as the identified user. The system uses a speech interface to convey the identified user and the confidence value.

d) Protection against Session Hijacking by Malicious Internal User: Our system protects a genuine user who has been authorized to use the system, from having their session taken over by an impostor who exists in the training dataset. Such a scenario is common in insider attacks, where a user's identity is stolen by a malicious user within the same organization. At the start of the session, the user's identity is stored by the system, and the identification provided by the network is compared to the stored identity. If the identity does not match, the speech interface announces that the user may be another individual in the organization who lacks access to the session.

e) External Intruder Detection: Our system allows the operator to provide a manual confidence threshold to guard against malicious users who do not belong in the training dataset, i.e., within the organization. If the confidence provided by the network is below a threshold, the speech interface provides a warning signifying that the user may be an impostor external to the organization.

Acknowledgements

This work was partially supported by the National Science Foundation (NSF) grant #1730183.

References

[1] A. Ajit, N. K. Banerjee, and S. Banerjee. Combining pairwise feature matches from device trajectories for biometric authentication in virtual reality environments. In *AIVR*, 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 *MMM*, 2019.