Continuous Operator Authentication for Teleoperated Systems Using Hidden Markov Models

JUNJIE YAN, University of Washington

KEVIN HUANG, Trinity College

KYLE LINDGREN, University of Washington

TAMARA BONACI, Northeastern University

HOWARD J. CHIZECK, University of Washington

In this paper, we present a novel approach for continuous operator authentication in teleoperated robotic processes based on Hidden Markov Models (HMM). While HMMs were originally developed and widely used in speech recognition, they have shown great performance in human motion and activity modeling. We make an analogy between human language and teleoperated robotic processes (i.e. words are analogous to a teleoperator's gestures, sentences are analogous to the entire teleoperated task or process) and implement HMMs to model the teleoperated task. To test the continuous authentication performance of the proposed method, we conducted two sets of analyses. We built a virtual reality (VR) experimental environment using a commodity VR headset (HTC Vive) and haptic feedback enabled controller (Sensable PHANToM Omni) to simulate a real teleoperated task. An experimental study with 10 subjects was then conducted. We also performed simulated continuous operator authentication by using the JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS). The performance of the model was evaluated based on the continuous (real-time) operator authentication accuracy as well as resistance to a simulated impersonation attack. The results suggest that the proposed method is able to achieve 70% (VR experiment) and 81% (JIGSAWS dataset) continuous classification accuracy with as short as a 1-second sample window. It is also capable of detecting an impersonation attack in real-time.

CCS Concepts: • **Security and privacy** → **Biometrics**; **Intrusion detection systems**; Usability in security and privacy; • **Human-centered computing** → *Haptic devices*; *Haptic devices*; • **Computer systems organization** → *Robotic control*; Robotic control.

Additional Key Words and Phrases: Authentication, Hidden Markov models, Telerobotics

1 INTRODUCTION

Teleoperated robotic systems have become an emerging and popular technology, largely due to several salient benefits. Critically, teleoperation provides a means to extend human capability to spaces that are otherwise inaccessible by human beings. The task may, for example, be too dangerous, such as in radioactive or chemically caustic environments, or even disaster scenarios. Furthermore, the task may be at a scale too large or too small for a human to physically accomplish. Consider the case where a human's particular expertise is not locally accessible or available, e.g. a specialized surgeon is needed on another continent. In these cases, the use of a remote robot-controlled at a distance, via a human operator, provides a practical solution. However, the benefit of having teleoperators comes with its own set of problems: *what if the security of teleoperated robotic systems is compromised?* In many envisioned high-reward scenarios, basic

infrastructure may be limited. Remote robots may have to operate in a harsh environment. The open and relative uncontrollable nature of the communication link between the operator and the robot potentially makes the teleoperated robotic system more vulnerable to various kinds of attacks. In our prior work [7, 8], we discovered that the tension between real-time operation (usability) and security for teleoperated robotic systems may render many existing security techniques infeasible. Existing methods, such as data encryption and commands signature verification, potentially generate an extra burden on the communication between the operator and the remote robot, which introduces delay and thus degrades the usability of the teleoperated system. Many teleoperated robotic systems deal with delicate or critical tasks. It is, therefore, crucial to make them secure without affecting their usability. The specific solution we propose exploits the fact that there is a human operator in the loop. Human operators have unique ways of interacting with teleoperated robotic systems[47], and this unique operating signature can be used to identify and authenticate the operator, thus enhancing the security and privacy properties of teleoperated robotic systems. In [47], a new password system based on the user's behavior biometric is developed to fulfill an initial login authentication task. However, if the malicious party targets the post authenticated session, such as through communication channel jamming or 'hijacking', initial authentication may not be sufficient.

Therefore, continuous authentication needs to be developed to secure the teleoperated robotic system. Behavior biometric-based continuous authentication has emerged recently to mitigate the security problems and attacks that target the post-authenticated session after the initial login for computer systems[5, 26, 43] and mobile devices [6, 11, 13, 22, 36, 38, 39]. In these works, instead of authorizing a user through a one-time login challenge, the authentication system continuously examines the user's behavior biometrics (i.e. keystroke/mouse dynamics, touch screen usage, device dynamics, etc.) in order to guarantee the identity of the initially authenticated user.

In the aforementioned works, the authentication results are based on the analysis of relatively simple user actions. In [5] and [43], features of user keystroke action, such as key code, press time and interval time between strokes when a user interacts with a desktop computer, are analyzed. Touch actions on mobile devices, such as tapping, scrolling, and flinging, are used for authentication purposes. Using single actions to fulfill the authentication is highly volatile. In most cases, to increase the robustness of the authentication method, multiple consecutive actions are used for the final decision. However, the operator's motions and actions during a teleoperated procedure are far more complex than a single simple action. The methods developed in the above works are not suitable in this case.

In [4], the author proposed a real-time detection method based on the teleoperated robots' dynamic properties. They focused on the case where the attackers are able to gain robot log access, analyze the teleoperation process, and inject malicious commands into the robot control system at the desired critical time. The detector estimates the robot motor, joints, and end-effector position and orientation after executing the given commands. The detector will then raise alerts whenever the velocity/acceleration exceed a pre-defined threshold. However, once the attackers gain full access to the remote robot, they will be able to cause damage without triggering the detector, such as cutting benign tissues with steady motion during a teleoperated surgery, or prematurely trigger an explosion in an unsafe location during a teleoperated bomb disposal task.

Moreover, in most of these approaches, conventional classifiers, such as k-Nearest Neighbour, Support Vector Machine, Neural Network or Random Forests, are implemented. *The major limitation of these classification algorithms is that they heavily rely on the choices of both positive and negative samples during the training process.*

Although the choices of positive samples are straightforward in our case, as we can use data obtained from the genuine operator, negative samples are not so easily acquired. The relevant teleoperated task data are nonlinear, dynamic and high dimensional. Thus, creating negative samples from human-generated spoofs of teleoperated tasks would not

tractably produce reliably complex sets of data that model the real-world. On the other hand, using other operators' data as negative samples for training is volatile, as classification performance will be very sensitive to the choice of these samples.

In this work, by making the analogy between human language and the motion commands of the remote operator, we present a Hidden Markov Model (HMM) based method for continuous teleoperator authentication. The HMM can be trained with only the operator's data (positive samples)[36] and it has been widely used in speech recognition[17] and human motion modeling [37, 40], which fits the teleoperator continuous authentication task well. The operator behavior-based continuous authentication is accomplished on the remote robot side with the operator's kinematic data. It gurantees that the operation performed by the remote robot is authenticated. The authentication process can be fulfilled in parallel with the teleoperation process, which minimizes its effect on the usability of the teleoperated system.

To determine the feasibility of our approach, we performed an experimental study with 10 participants. All subjects carried out a simplified Fundamentals of Laparoscopic Surgery (FLS) block transfer task. It is a standard test used to train and test surgeons[25]. We developed a simulated virtual reality environment with haptic feedback and virtual fixtures enabled for this task. We also explored the performance of our approach on the da Vinci surgical robotic platform, as we performed a simulated continuous operator authentication task by using JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS) [16].

In summary, the main contributions of this work are:

(1) The development of a continuous teleoperator authentication method that uses Left-Right HMM[49] to model an operator's gestures followed by a Token Passing algorithm [50] that concatenates gesture models.
(2) The development and demonstration of a VR simulated teleoperation environment and the experimental user study evaluation.
(3) Experimental demonstration that the proposed continuous teleoperator authentication is able to achieve high accuracy and impersonation attack resistance.

2 RELATED WORK

HMMs have been extensively used in surgical skill assessment [3, 30–34]. In the majority of these efforts, it is assumed that the entire surgical process is generated from a single HMM model while each surgical gesture is represented by a single state in the HMM. In [34], it is assumed that each surgical gesture can be represented by samples from a Gaussian distribution. In [30], the Short Time Fourier Transform (STFT) followed by K-Means is used to discretize the surgical process data into gestures (states in HMM). Discrete HMMs are then trained to fulfill the skill evaluation. Linear discriminant analysis (LDA) is applied to the surgical data to perform dimension reduction in [45] before HMMs are trained. In [41], a Sparse HMM approach is proposed, where a sparse dictionary learning technique called K-SVD[2] is used to model the surgical gesture states in the HMM. Representing a surgical gesture by a single state in HMMs has limitations with regard to fully capturing the dynamic and complex properties of each gesture. In [3, 45], the authors proposed to represent each surgical gesture as an HMM instead of a single state within an HMM.

In all aforementioned work, the analysis is performed offline given the entire kinematic data of the surgical procedure. However, as shown in Figure 1, continuous authentication is an online process, and instead of the whole offline data set, we need to rely on the data from some sample window to perform analysis and authenticate an operator. In this case, unlike the offline scenario, the data in the sample window will contain only partial gestures. The kinematic properties of a partial gesture are different from the complete gesture. Therefore, using a single state in the HMM to represent the
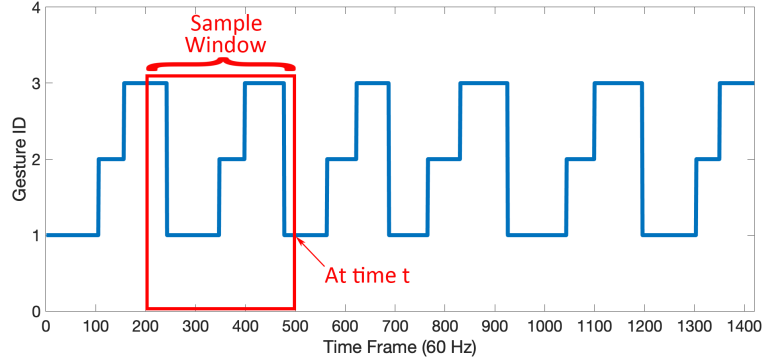
Fig. 1. Comparison Between Offline Analysis and Continuous Authentication

operator's gesture is not applicable for continuous operator authentication. Representing each operator's gesture as an HMM better fits our application.

In [48], the author proposed Recurrent Neural Networks (RNN) and Long short-term memory (LSTM) networks to model surgical gestures among multiple surgeons and achieved real-time unsafe events detection in robot-assisted surgeries. In [3, 19–21, 23, 42], automatic sugical gesture segmentation has been extensively investigated and explored.

In this paper, our scope is to use gesture labeled teleoperation process sequences to train operators' gesture models. We then focus on using the trained operators' gesture models to continuously authenticate the corresponding operator given the unlabeled real-time teleoperation process sequences.

3 THREAT MODEL AND DETECTION STRATEGY

In this paper, we focus on the detection and mitigation of **impersonation attacks**. Impersonation attacks against teleoperated systems represent an advanced, persistent threat against complex teleoperated systems, especially those used in safety-critical missions.

We assume an attacker with enough computational resources and knowledge about the system to gain access to a remote robot. The attacker can gain such an access by exploiting the vulnerable software/hardware supply channel[1], by exploiting exiting software vulnerabilities [10, 18, 24, 44], by targeting vulnerable communication channels, by using stolen credentials, or through insider attacks.

Once gained access, the attackers' goal is to stay stealthy within the system, and cause damage/loss to the system users over an extended period of time. In doing so, an attacker is likely to want to impersonate a legitimate, already authenticated teleoperator, and perform operations using their credentials.

In order to detect such an attack, we focus on the kinematic motion commands received by the remote device, presumably sent from the operator. By continuously analyzing the motion commands, the proposed detection method is able to determine whether the commands are sent from the authorized operator in real-time. It guarantees that the operation performed by the remote robot is authenticated.

4 CONTINUOUS OPERATOR AUTHENTICATION FOR TELEOPERATED SYSTEMS

In order to achieve continuous operator authentication for teleoperated systems and overcome the constraints present in existing methods, we develop a novel scheme as shown in Figure 2 (training process) and 3 (continuous authentication
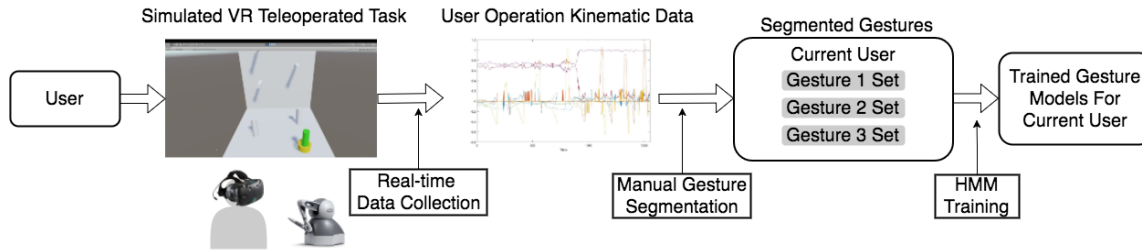
Fig. 2. Continuous Operator Authentication Training Phase. The user performs a simulated teleoperation in the VR environment with a haptic input device. Real-time kinematic data of the entire teleoperation process is collected. We manually segment the teleoperation process into several basic gestures. The corresponding user's gesture HMM models are trained based on the segmented gesture pieces.
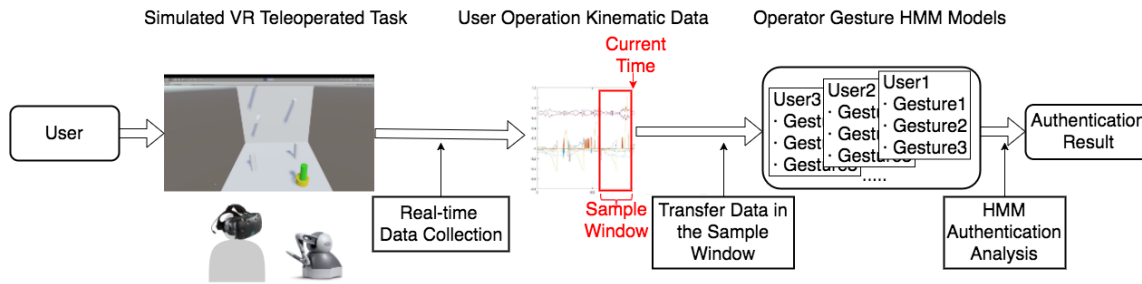

Fig. 3. Continuous Operator Authentication Testing Phase. The user performs the simulated teleoperation task while we use a moving sample window to collect kinematic data. Given the operator gesture HMM models obtained in the training phase, an HMM likelihood analysis is performed on the kinematic data within the sample window. The continuous authentication result is based on the likelihood analysis

process). In the training phase, a user performs a simulated teleoperation in the VR environment by using a haptic input device. Real-time kinematic data (i.e. velocity, orientation, and force applied) of the entire teleoperation process is collected. We then manually segment the teleoperation process into several basic gestures. The corresponding user's gesture HMM models are trained based on the segmented gesture pieces. In the testing phase, a user performs the simulated teleoperation task while we use a moving sample window with width T to collect kinematic data from $t - T$ to t, where t is the current time. An HMM likelihood analysis is then performed on the data within the sample window based on the trained operator gesture HMM models and this generates the continuous authentication result.

5 EXPERIMENT

5.1 Experimental Setup

In this work, we first built a VR environment within the Unity Game Engine[9] to let subjects perform a simplified FLS block transfer task. As shown in Fig. 4, the user was asked to use the Sensable PHANToM Omni[27] to control the 6 degree of freedom (DOF) configuration of a virtual ring in order to transfer it through the virtual pegs on the board in a predefined sequence.

(a) VR Environment   (b) PhantomOmni Controller
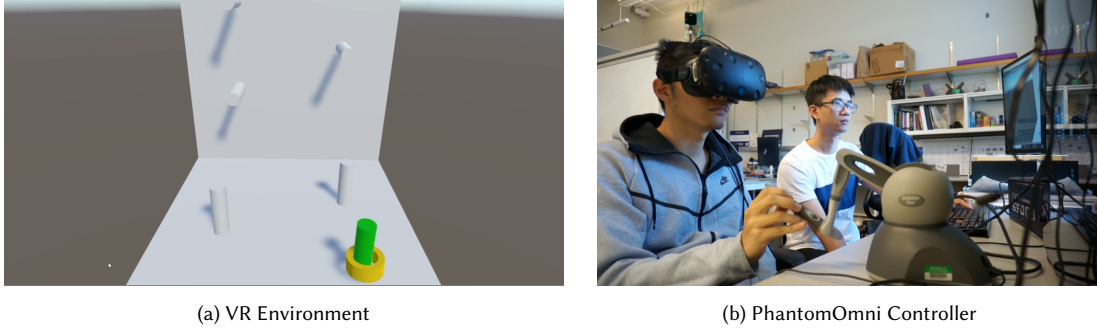
Fig. 4. Experiment: The VR Environment (a) and The user and PhantomOmni Controller (b)

To assess the applicability of our proposed approach to real-time teleoperator authentication, we create a simplified Fundamentals of Laparoscopic Surgery (FLS) block transfer task, a standard test used to train and test surgeons[25]. The FLS block transfer task is a well-established surgical task, which is simple enough for surgical non-experts to execute, yet complex enough to segment into meaningful surgical gestures.

In the experiment, each user wore the HTC Vive headset for 3D visual feedback from the VR simulated task. Meanwhile, haptic feedback via virtual fixtures[35] was enabled during the entire operation as the user-provided motion commands with the Sensable PHANToM Omni. Two types of virtual fixtures were introduced: 1) Forbidden region around the pegs and baseboard and 2) Guidance toward the next peg tip. The guidance virtual fixture was only activated when the ring was out of the peg and being transferred toward the next peg. The haptic feedback offers the user a sense of touch and helps improve the operational performance. Moreover, in [47], we found that humans have unique ways of interacting with haptic interfaces and that haptic feedback can be used for continuous authentication.

In this experiment, subjects were first asked to explore the VR environment to get used to the interface and practice the simulated teleoperated task 10 to 15 times until they gained enough familiarity with the task. The goal of this process is to eliminate any learning effects on the continuous authentication performance.

Each subject was then asked to perform the task 5 times, while the following data were collected in real-time.

(1) Position of the center of the ring (x, y, z)
(2) Orientation of the ring $(Q_x, Q_y, Q_z, Q_w,$ in quaternion)
(3) Force applied (f_x, f_y, f_z)

All data were recorded at 60 Hz. In each trial, the app starts recording data when the user starts moving the ring and stops when the user pulls the ring out of the last peg.

Since the position and force data are highly correlated due to the influence of the virtual fixtures, we also collected velocity information from the gathered position data and use it to train the model.

The state vector at time t is then constructed as $s = (v_x, v_y, v_z, Q_x, Q_y, Q_z, Q_w, f_x, f_y, f_z)$.

5.2 Gesture Segmentation

In order to make the analogy between human motion and language, we first need to identify what corresponds to "words" in the teleoperation process, which is the operator's gesture. In this work, we segmented the entire process into 3 basic gestures as listed below.

Table 1. Subjects Demographics

Sample Size	10
Sex	6 Females; 4 Males
Age Range	18 to 28
Handedness	2 Left; 8 Right

1) Gesture 1: Transfer the ring toward next peg tip;

2) Gesture 2: Insert the ring;

3) Gesture 3: Pull out the ring.

The segmentation is based on the position of the ring and the corresponding status of the teleoperation process.

5.3 Subject Demographics

Our experimental results are based on our study involving 10 participants. The demographics of all participants are shown in Table 1. All experiments were conducted with the approval of the UW Institutional Review Board.

6 HMM-BASED CONTINUOUS OPERATOR AUTHENTICATION

In this paper, we use a Left-Right HMM[29] to model operators' gestures followed by a Token Passing algorithm [50] to concatenate the gesture models, thus achieving continuous authentication. The major advantage of using HMMs is that in the training phase, only positive samples are required. Also, HMMs are able to capture local dynamic properties of the operator's gestures[36] which serves the purpose of continuous operator authentication. In the following sections, we will briefly review the Left-Right HMM Model and Token Passing Algorithm and demonstrate how we implement them for the proposed continuous authentication task.

6.1 Left-Right HMM Model

Left-Right HMMs have been widely used in speech recognition[14, 28, 49]. They are used to model each word/phoneme separately. Compared to conventional HMMs, the Left-Right HMM offers non-emitting entry and exit states. They provide benefits by concatenating different words/phonemes together in real-time to achieve speech recognition.

In this paper, we use left-right HMM to represent each gesture from an individual operator. The proposed Left-Right HMM structure is shown in Figure 5. Each state i is associated with an emission probability distribution $b_i(o_t)$, which defines the probability of generating observation o_t at time t. Additionally, the transition probability between each pair of states i and j is determined by transition probability $\{a_{ij}\}$. Furthermore, the entry (first) and exit (last) states of the proposed HMM are non-emitting. These two states are used to facilitate the concatenation between surgeme models as explained in more detail later. The other states are emitting states associated with emission probability distributions. The transition matrix is $N \times N$, where N is the number of states. The sum of each row will be one except for the last row which is zero since no transition is allowed from the final state.

We assume that for each emitting state i that the emission probability distribution is a Gaussian mixture as similar assumptions are frequently made with motion detection and speech recognition [14, 28, 36, 49]. For state i, the probability
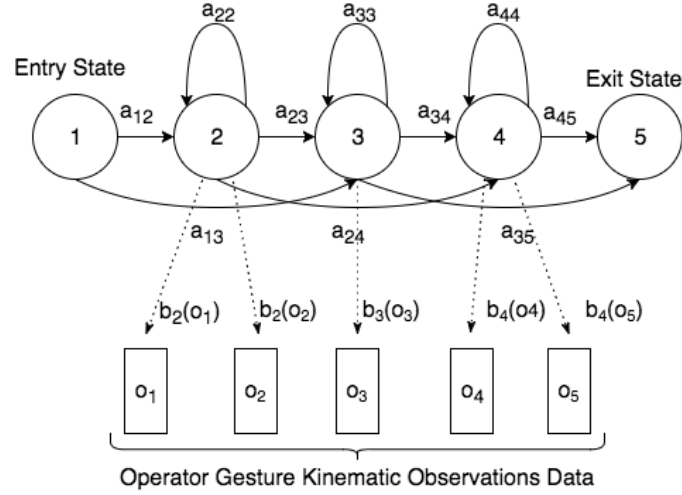
Fig. 5. Left-Right Hidden Markov Model with Non-emitting State

$b_i(o_t)$ of generating observation o_t is given by

$$b_i(o_t) = \sum_{m=1}^{M_i} c_{im} \mathcal{N}(o_t; \mu_{im}, \Sigma_{im}) \tag{1}$$

where M_i is the number of Gaussian mixtures in state i, c_{im} is the weight of the mth mixture and $\mathcal{N}(\cdot; \mu_{im}, \Sigma_{im})$ is the probability density function of a multivariate Gaussian distribution with mean μ_{im} and covariance matrix Σ_{im}.

In the training phase, Baum-Welch re-estimation [29] is used. Segmented pieces of gesture sequences from each operator (as mentioned in section 5.2) are used as ground truth and the Baum-Welch algorithm is applied to obtain the maximum likelihood estimation of the model parameter state transition probability matrix $\{a_{ij}\}$ and emission probability distribution $b_i(o_t)$ for $i, j = 1, ..., N$.

6.2 Token Passing Algorithm

Given the observation sequence, to achieve gesture recognition and continuous authentication, the first step is to determine the hidden state sequence. This can be done using Viterbi Decoding Algorithm [12]. In this work, an alternative formulation of the Viterbi Algorithm called the Token Passing Algorithm[50] is used. It is able to realize single gesture recognition while simplifying concatenating gesture models for continuous operator authentication.

First, for the base case of single gesture recognition, the Token Passing algorithm works as follows. At each time frame t, the following algorithm is executed:

for t= 1 **to** T do

    for each state i **do**

        Pass a copy of the token in state i to each connecting state j:

        $\psi_{i \to j}(t) \leftarrow \psi_i(t-1) + log(a_{ij}) + log(b_j(o(t)))$;

    end

Discard the original tokens;

for each state j **do**

$$\psi_j(t) = \max_i\{\psi_{i\rightarrow j}(t)\} \text{ for each state } i \text{ connected to state } j$$

end

end

where:

$\psi_i(t)$: the maximum log-likelihood of observing operation signal $o_{1:t}$ and being in state i at time t;

$\psi_{i\rightarrow j}(t)$: the log-likelihood of observing operation signal $o_{1:t}$ and a state transition from i to j at time t;

a_{ij}: the state transition probability from state i to state j;

$b_i(o_t)$: the emission probability of state i given observation o_t.

In the Token Passing algorithm, it is assumed that each state j of an HMM at time t holds a single movable token that contains partial log-likelihood $\psi_j(t)$.

Let $o_{1:T}$ denote the gesture observation sequence with length T. $\psi_{max}(T|o_{1:T}, \text{ operator } i, \text{ gesture } j)$ denotes the log likelihood held by the remaining token at time T, given that the gesture model is from operator i and gesture j. The gesture observation sequence $o_{1:T}$ can be recognized as operator i's gesture j, given:

$$(i, j) = arg \max_{i,j}\{\psi_{max}(T|o_{1:T}, \text{ operator } i, \text{ gesture } j)\} \tag{2}$$

Next, in order to realize continuous operator authentication, individual gesture models need to be concatenated. Similar to human language, there is *grammar* in the teleoperation task as well, which defines how the gestures can be connected. During the experiment, each subject first transfers the ring towards next peg tip (Gesture 1), inserts the ring until it reaches the back board (Gesture 2), pulls out the ring (Gesture 3) and then repeats these operations until the subject finishes the task. Therefore, in our proposed simulated teleoperated task, the grammar is defined as in Figure 6.

Fig. 6. Gesture Grammar

The structure of concatenating the gestures based on the grammar is shown in Figure 7. The non-emitting entry and exit states now work as *glue* to join gesture models together. At each time frame t, the following algorithm is executed for arbitrary observation sequence that potentially contains multiple gestures:

for t= 1 **to** T do

for each state i of gesture k **do**

Pass a copy of the token in the state i of the gesture k to each connecting state j of the gesture k :

$\psi_{(g_k,s_i)\rightarrow(g_k,s_j)}(t) \leftarrow \psi_{(g_k,s_i)}(t-1) + log(a_{(k,ij)}) + log(b_{(k,j)}(o(t)));$

Pass a copy of the token in the state i of the gesture k to state j of the gesture l via non-emitting exit and entry state based on the gesture grammar:

$\psi_{(g_k,s_i)\rightarrow(g_l,s_j)}(t) \leftarrow \psi_{(g_k,s_i)}(t-1) + log(a_{(k,i\rightarrow exit)}) + log(a_{(l,entry\rightarrow j)}) + log(b_{(l,j)}(o(t)));$

end

Discard the original tokens;

for each state i of gesture k **do**

$$\psi_{(g_k,s_i)} = \max_{j,l}\{\psi_{(g_l,s_j)\to(g_k,s_i)}(t)\} \text{ for each gesture } l \text{ connected to gesture } k \text{ based on the gesture gram-}$$

mar,

and each connected state j.

end

end

where:

$\psi_{(g_k,s_i)}(t)$: the maximum log-likelihood of observing operation signal $o_{1:t}$ and being in the state i of the gesture k at time t.

$\psi_{(g_k,s_i)\to(g_l,s_j)}(t)$: the log-likelihood of observing operation signal $o_{1:t}$ and a transition from the state i of the gesture k to the state j of the gesture l at time t;

$a_{(k,ij)}$: the state transition probability for the gesture k from the state i to the state j;

$a_{(k,i\to exit)}$: the state transition probability for the gesture k from the state i to the non-emitting exit state;

$a_{(k,entry\to j)}$: the state transition probability for the gesture k from the non-emitting entry state to the state j;

$b_{(k,i)}(o_t)$: the emission probability of the gesture k, state i given observation o_t.

For connected gesture recognition, besides the overall log-likelihood, we also want to know the best matching gesture sequence. Tokens are assumed to hold a path identifier as well as the path log-likelihood. The path identifier is used to record gesture boundary information which will be called Gesture Link Record (GLR). At each time t, extra steps shown as follows are taken in addition to the individual gesture recognition algorithm listed above:

for each token entered EXIT state at time t do

create a new GLR containing:

<token contents, t, identity of emitting gesture>;

change the path identifier of the token to point to this

new record

end

By doing so, potential gesture boundaries are recorded in a linked list, and on completion at time T, the path identifier held in the token with the largest log-likelihood can be used to trace back through the linked list to find the best gesture sequence and the corresponding gesture boundary locations.

6.3 Continuous Operator Authentication

To accomplish continuous operator authentication, we put an additional constraint on the aforementioned gesture recognition scheme by mandating that consecutive gestures must come from the same operator.

At time t, given the observation from the sample window with width T as $O_{t-T:t}$, operator recognition is done by solving $\psi_{max}(t)$ and checking the gesture labeling $l_{t-T:t}$. The observation sequence will be recognized as operated by

Fig. 7. Token Passing Algorithm

user i if

$$l_{t_0} \in L_i, t_0 = t - T, ..., t \tag{3}$$

where i is the operator ID and L_i is the corresponding gesture label set for the ith user.

7 RESULTS

7.1 Authentication Result

In this work, we used the leave-one-trial-out cross-validation strategy to train and test the proposed method.

First, in the training phase, the gesture models from each subject are trained based on the segmented individual gesture pieces in those training trials. In this way, 3N subject gesture models were obtained where 3 is the number of gestures and N is the number of the subjects.

We varied the hyperparameters of the HMM to test the corresponding continuous authentication performance. We varied the number of states from 3 to 6 and the number of mixtures in each state from 1 to 3. We also tested moving sample windows with widths of 5 seconds, 3 seconds, and 1 second.

Let L_{window} denote the number of observations contained by a moving sample window. As the kinematic data for the VR experiment is recorded at 60 Hz, the tested 5s, 3s, 1s moving sample windows contain 300, 180, and 60 observations respectively. $L_{i,j}$ denote the total number of observations contained by the jth trial from subject i. The continuous

(a) 1-Second Sample Window   (b) 3-Second Sample Window   (c) 5-Second Sample Window

Fig. 8. Continuous Authentication Accuracy with 1-Second(a), 3-Second(b) and 5-Second(c) Sample Window

authentication starts when there is at least L_{window} observations. Each time the moving sample window is shifted by 1 observation, hence the size of the overlap between two consecutive sample windows is $L_{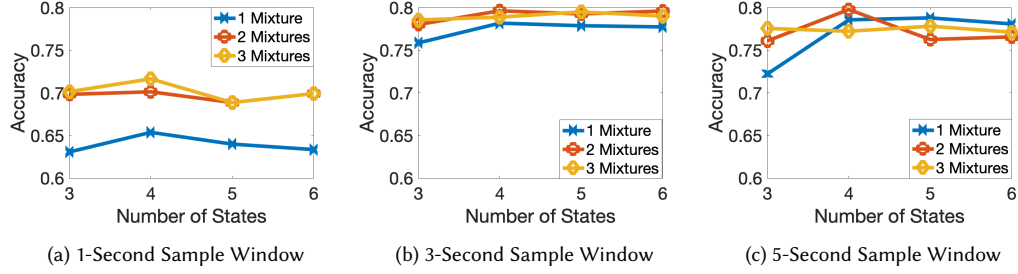window} - 1$. As a consequence, there are $L_{i,j} - L_{window} + 1$ sample windows evaluated for the testing trial. The HMM models generate a predicted subject label for each sample window. Therefore, the performance analysis is equivalent to a multi-class classification on the sample windows from all the trials. The accuracy is calculated as:

$$Accuracy = \frac{N_{hit}}{N_{tot}} \tag{4}$$

where:

N_{hit}: the total number of sample windows where the subject is correctly classified in all trials.

N_{tot}: the total number of sample windows in all trials.

The corresponding results with different sample window width and hyperparameters are shown in Figure 8.

We found that when we choose the hyperparameter of the HMM which models each gesture as 4 states with 2 Gaussian mixtures for each state, we were able to achieve the best authentication accuracy. We set the hyperparameter as 4 states with 2 mixtures in the following context. The detailed confusion matrices for the VR experiment with 5s, 3s, and 1s sample window are shown in Table 9-11 in the Appendix respectively. The authentication accuracy, macro average precision, and macro average recall are listed in Table 2.

Table 2. Continous Authentication Performance with Multiple Sample Window Width for the VR Experiments

Window Width	5 sec	3 sec	1 sec
Accuracy	79.78%	79.63%	70.12%
Avg. Precision	79.48%	79.75%	70.71%
Avg. Recall	82.54%	82.12%	73.12%

When we used a sample window width of 5 seconds, we were able to authenticate the operator in realtime with almost 80% accuracy, average precision, and recall. With a 1-second sample window, we achieved above 70% continuous authentication accuracy, average precision, and recall. It shows that this method works and is promising for continuous authentication performance.

7.2 Simulated Impersonation Attack Resistance

We simulated an impersonation attack in the following steps. First, we picked two subjects (represented as User 1 and User 2 in the following context) and used the leave-one-trial-out strategy to train the model for the gestures of each subject. In the testing phase, instead of using the remaining testing trial to examine the authentication performance, we split the test trial from User 1 and User 2 into 2 half pieces and concatenate User 1's first piece to User 2's second piece and vice versa. In this way, we generated two artificial teleoperation process observation sequences, where User 2 impersonates User 1 during the second half of the teleoperation task and vice versa. Compared to actual malicious impersonation attacks, we anticipate that the simulated impersonation attack is more difficult to detect. As in the simulated impersonation attack, both users were performing benign operations, hence the operations were similar. We anticipate the differences between malicious operations and benign operations are more significant than the differences among benign operations.

Figure 9 shows one of the results in detail. The blue and red lines represent the likelihood that the data within the corresponding sample window is operated by User 1 or User 2 respectively. The orange dashed line is the point where the simulated impersonation attack takes place. First, this method is able to detect the impersonation attack as the likelihood of the original user drops significantly after the simulated attack launches. Moreover, we notice that with a longer sample window, it is easier to distinguish two operators as the differences between the likelihood of the two users are significantly larger at various points when the size of the sample window increases. On the other hand, the response time for the continuous authentication system to detect the impersonation attack increases when the sample window size becomes wider. Denoting the response time as the time between the time of the impersonation attack and the likelihood cross point between two users, the average response time is shown in Table 3.

Fig. 9. Simulated Impersonation Attack For the VR Experiment

Table 3. Average Response Time to Impersonation Attack with Multiple Sample Window Widths

Window Width	5 sec	3 sec	1 sec
Response Time	2.35 sec	1.51 sec	0.49 sec

Therefore, when choosing the size of the sample window, there is a tradeoff between the accuracy of the continuous authentication and response time to attacks. A wider sample window is able to generate more stable continuous authentication accuracy, however, it takes more time to respond to the attack. On the other hand, shorter sample windows offer the ability to react quickly to impersonation attacks, but the authentication accuracy is less stable.

7.3   Authentication Performance on Surgical Robotic System

We tested the continuous operator authentication method on JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS)[16] and explored the authentication performance of the developed method on a surgical telerobotic system. In JIGSAWS, the dataset is obtained through the da Vinci Surgical Robot, where subjects were asked to perform several surgery tasks. Kinematic data of three basic surgical tasks (suturing, needle passing, and knot tying) performed by 8 study subjects are included. The kinematic data were recorded at 30 Hz. Each task was performed 3-5 trials for each subject. Each teleoperation process is manually labeled as a sequence of surgical gestures. The definition of each gesture[16] is presented in Table 4.

Table 4. Surgical Gesture Definition[16]

Gesture Index	Surgeme Definition
G1	Reaching for needle with right hand
G2	Positioning needle
G3	Pushing needle through tissue
G4	Transferring needle from left to right
G5	Moving to center with needle in grip
G6	Pulling suture with left hand
G7	Pulling suture with right hand
G8	Orienting needle
G9	Using right hand to help tighten suture
G10	Loosening more suture
G11	Dropping suture at end and moving to end points
G12	Reaching for needle with left hand
G13	Making C loop around right hand
G14	Reaching for suture with right hand
G15	Pulling suture with both hands

Given the observation kinematic data and gesture labeling, we segmented the teleoperated surgical process into gesture pieces and use that as ground truth to train the corresponding subject's gesture HMM models. To unify the data properties for each subject, we focused on those subjects with complete 5 trial kinematic and labeling data for each surgical task. By doing so, we obtained 7 valid subjects for the suturing task (indexed as "B", "C", "D", "E", "F", "G", "I" in JIGSAWS), 2 valid subjects for the needle passing task (indexed as "C", "D" in JIGSAWS) and 5 valid subjects for the knot tying task (indexed as "C", "D", "E", "F", "G" in JIGSAWS).

For each surgical task, we kept the leave-one-trial-out setting to train and test the continuous operator authentication. In the training phase, we obtained MN gesture models, where M is the number of gesture types and N is the number of subjects we tested. Table 5 shows the detailed subject and gesture indices for all 3 surgical tasks. We tested the sample window with a width of 5 seconds, 3 seconds, and 1 second. We use the same setting as discussed in the previous section to evaluate the continuous authentication accuracy. The grammar of the surgical gesture[3] for each task is presented in Figure 10, which defines how the gesture models are connected for each surgical task.

Table 5. Subjects Evaluated and Gestures In the Corresponding JIGSAWS Tasks

	Number of Subjects	Subject Indices	Number of Gestures	Gesture Indices
Suturing	7	B,C,D,E,F,G,I	9	G1,G2,G3,G4,G5,G6,G8,G9,G11
Needle Passing	2	C,D	8	G1,G2,G3,G4,G5,G6,G8,G11
Suturing	5	C,D,E,F,G	6	G1,G11,G12,G13,G14,G15

Fig. 10. Grammar graph for suturing (left), needle passing (center), and knot tying (right)[3]

We then obtain the following continuous authentication result as shown in Table 6. The detailed confusion matrices for all sample window widths and tasks are listed in Appendix A.2.

From these results, we found that even with 1-second observation sequence, the continuous operator authentication accuracy to detect the subject is above 80% for all three tasks. This shows that the developed continuous authentication method also works for teleoperated robotic surgeries.

We also conducted a simulated impersonation attack using the same setup as discussed in section 6.2. We performed the analysis on Subjects C and D as they are included in all 3 surgical tasks.

Table 6. Continous Authentication Performance with Multiple Sample Window Width for the JIGSAWS dataset

Task		5s	3s	1s
Suturing	Accuracy	94.79%	94.29%	92.78%
	Avg. Precision	94.31%	93.78%	92.35%
	Avg. Recall	95.10%	94.52%	92.76%
Needle Passing	Accuracy	91.36%	91.95%	91.75%
	Avg. Precision	91.77%	92.35%	92.13%
	Avg. Recall	91.44%	92.03%	91.83%
Knot Tying	Accuracy	85.00%	84.11%	81.13%
	Avg. Precision	87.18%	85.98%	82.37%
	Avg. Recall	85.88%	84.76%	81.44%

Figure 11-13 show one of the results of each surgical task in detail. The average response time for each surgical task is shown in Table 7. The results confirm that for all 3 surgical tasks, the proposed method is able to detect the simulated impersonation attack. Additionally, similar to the VR experiment, the tradeoff between accuracy and response time is demonstrated as well. A larger sample window can generate more stable authentication accuracy with a longer response time to the attack, while a smaller sample window can generate less stable authentication accuracy with a faster response to the attack.

Table 7. Average Response Time to Impersonation Attack for All JIGSAWS Surgical Tasks

Window Width	5 sec	3 sec	1 sec
Suturing	2.99 sec	1.73 sec	0.53 sec
Needle Passing	2.55 sec	1.56 sec	0.62 sec
Knot Tying	2.35 sec	1.26 sec	0.42 sec

8 DISCUSSION

In this section, we will discuss several limitations of this paper and also raise possible extensions of our approach.

Inexperienced Experimental Subjects In the VR simulated teleoperation experiment, most subjects had never interacted with VR and/or haptic input devices. Although we conducted a training session to familiarize them with the system, we still noticed that there was a learning effect during the data collection, whereby the subject became better in handling and operating the system. Also, in some cases, subjects were less patient towards the end of the experiment, which influenced their motion to deviate from the original model. All these factors might have undermined the authentication performance result in our experiment. However, in most real-life applications, genuine operators are usually well trained and have sufficient familiarity and experience with the teleoperated system. Therefore, it is more
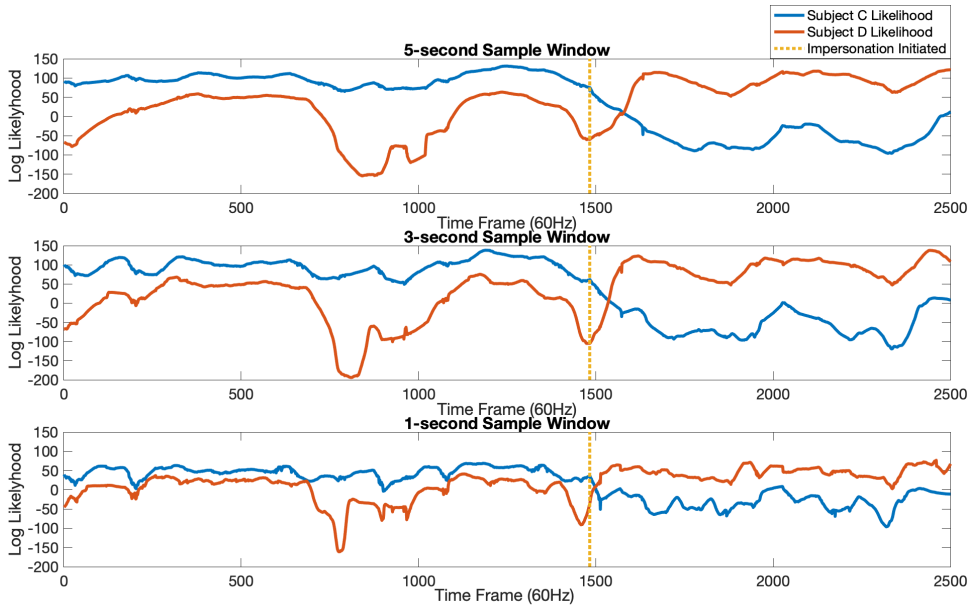
Fig. 11. Simulated Impersonation Attack on the Suturing Task
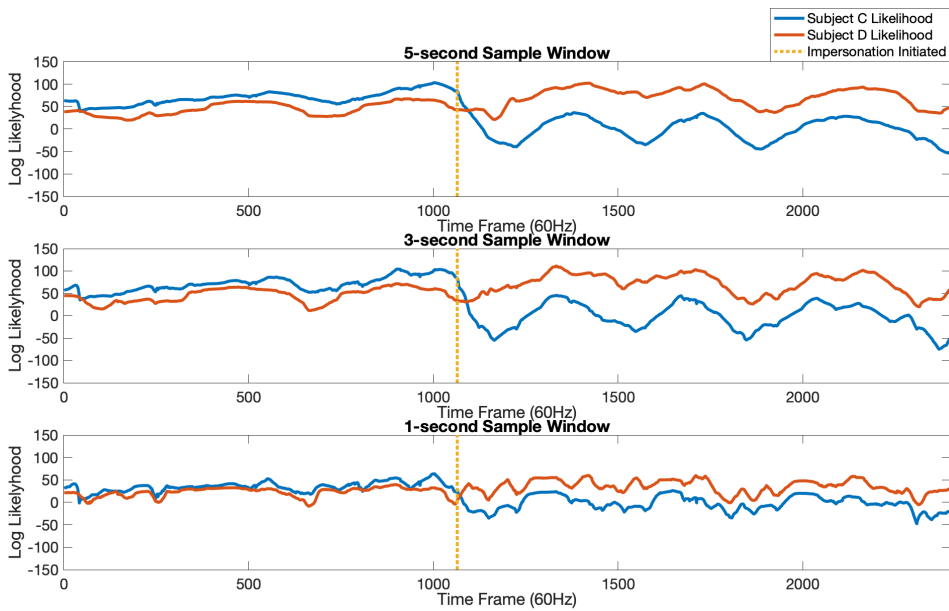

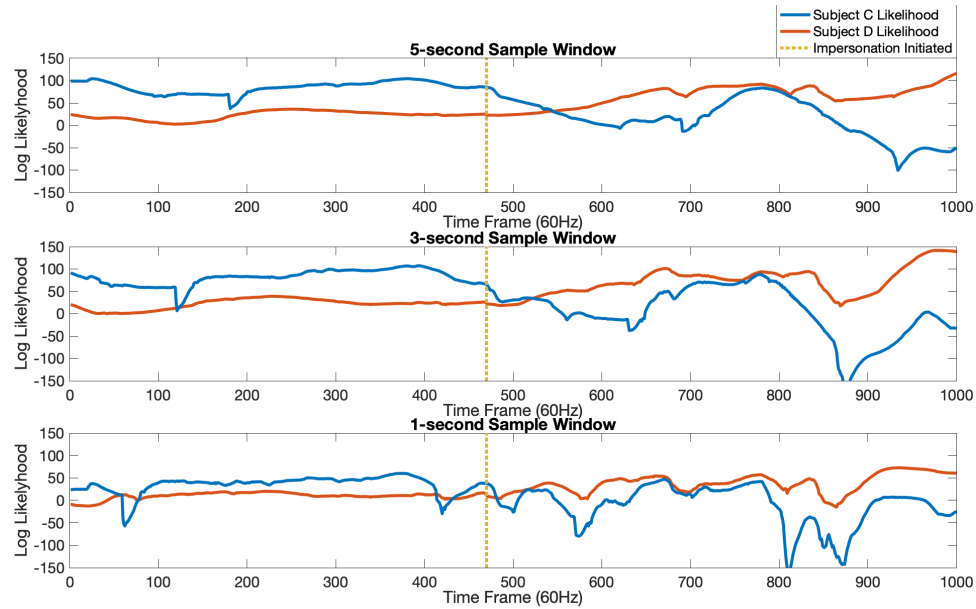Fig. 12. Simulated Impersonation Attack on the Needle Passing Task

Fig. 13. Simulated Impersonation Attack on the Knot Tying Task

likely that the operator has a unique operating 'pattern' which makes it easier to accomplish the continuous operator authentication.

On the other hand, when dealing with the JIGSAWS dataset, we were able to achieve better continuous operator authentication accuracy compared to the VR simulated experiment as some of the subjects have previously trained to operate the surgical robot. Moreover, in the original JIGSAWS dataset, all the subjects are categorized as 3 skill levels based on their robotic surgical experience (Expert: more than 100 hours, Intermediate: between 10 and 100 hours, Novice: less than 10 hours). Table 8 represents the average continuous authentication accuracy among experts and non-experts for all 3 surgical tasks. The result shows that our approach can achieve better continuous authentication performance when dealing with expert subjects. It confirms our assumption that it would be easier to extract more unique features from the operator and achieve better performance when the operator has sufficient experience with the teleoperated system.

Teleoperation Tasks Involve Intended Operator Switches In some teleoperation tasks, such as robotic-assisted surgeries, there are intentional operator switches. A single operator model is not sufficient to continuously authenticate all the collaborating operators. In this case, instead of using a single operator model, we can develop an allowlist of operator models to capture all the authenticated operators. During the continuous authentication process, the system can authenticate the current operator as long as data within the current sample window gets authenticated by one of the operator models in the allow list.

Teleoperation Task Complexity In this work, our experimental results were based on relatively simple (VR simulation) and well-structured (basic surgical tasks) teleoperation tasks. In real-life scenarios, some teleoperation tasks are more complex and less structured. It would be interesting to explore whether our approach is able to achieve good

Table 8. Continous Authentication Performance with Different Skill Level

Window Size	Expert		Non-expert	
	Avg. Precision	Avg. Recall	Avg. Precision	Avg. Recall
5 sec	91.48%	93.51%	91.35%	90.04%
3 sec	90.93%	92.00%	90.71%	89.94%
1 sec	89.24%	88.52%	88.49%	88.66%

continuous authentication performance in these applications. We anticipate that better gesture segmentation, as well as gesture grammar, is needed to better generalize these types of teleoperation processes.

Authentication Performance Under Various Conditions In the proposed experimental study, we were only able to use data collected within a single-day for each subject and the subjects all exhibit normal cognitive condition. In[15][46], how teleoperators' behavior patterns vary over time and under different cognitive condition were investigated. Therefore, teleoperators' gestures vary over the long-term and different cognitive conditions may affect the continuous authentication performance.

For example, within each teleoperation procedure, the operator's fatigue and pressure under time constrain may change an operator's behavior pattern. Additionally, the operators' experise level will improve over time and thus change how they interact with the remote robot. It is worth further investigation on how these factors will affect the continuous operator authentication accuracy.

A possible way to address this issue is to develop a mechanism to adaptively update the operator's authentication model based on the new test data. Every time a verified genuine operation process is conducted, gesture sequences can be obtained either manually or by using automatic surgical gesture segmentation as proposed in [3, 19–21, 23, 42]. The model can then be updated by using the new set of gestures. This will help the authentication model to capture the operator's behavior pattern under different operation conditions, such as fatigue or high-stress level. It also allows the model to accommodate for the operator's long term variation due to evolving skill level or aging.

Replay Attacks One possible way for an attacker to bypass the proposed continuous authentication is through replay attacks. If an attacker can record a benign kinematic operation command sequence from an authorized operator, it is possible for the attacker to edit sequences to smoothly transition into and between replay attacks. Such types of attacks can potentially cause damage as well. However, as the operation command sequence is generated by an authorized genuine operator, and the transition between each replay has been smoothed out, the proposed continuous authentication will not be able to detect them. Nevertheless, the human-in-the-loop nature of teleoperation processes allows us to mitigate this problem. Since humans are not perfect machines, it is almost impossible for any human to generate exactly the same kinematic operation command via a control console. A command sequence history can be monitored to raise an alert when an exact or near-duplicate command sequence (i.e. attackers may add small noise to each replay) is observed. This will allow us to overcome the problem of replay attacks against the proposed continuous authentication method.

9 CONCLUSION

In this paper, we develop a continuous and real-time operator authentication method by making an analogy between human motion and human language (gesture to word and operation process to sentence). We use HMMs to model each operator's gestures and then concatenate them by using the Token Passing Algorithm based on a predefined operation grammar to achieve continuous authentication. We built a VR simulated environment and conducted a human subject experiment where the subjects conducted a simulated teleoperation task within the VR environment. We also tested our approach on a teleoperated surgical process as we used the JIGSAWS dataset and explored its continuous authentication performance. Our experimental results indicate that the developed continuous teleoperator authentication method works and is able to achieve above 70% accuracy rate for the VR simulated teleoperation task and 81% accuracy rate for JIGSAWS surgical tasks with as short as a 1-second sample window. Moreover, we further examined the continuous authentication system resistance to impersonation attack and demonstrated that our approach is able to detect impersonation attacks with short response time.

ACKNOWLEDGMENTS

REFERENCES

[1] 2021. Form 8-K Solarwinds Corp Current Report. *[Online][Cited: April 18, 2021.] https://sec.report/Document/0001739942-21-000015/* (2021).

[2] Michal Aharon, Michael Elad, and Alfred Bruckstein. 2006. SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. *Signal Processing, IEEE Transactions on* 54, 11 (2006), 4311–4322.

[3] Narges Ahmidi, Lingling Tao, Shahin Sefati, Yixin Gao, Colin Lea, Benjamin Bejar Haro, Luca Zappella, Sanjeev Khudanpur, René Vidal, and Gregory D Hager. 2017. A dataset and benchmarks for segmentation and recognition of gestures in robotic surgery. *IEEE Transactions on Biomedical Engineering* 64, 9 (2017), 2025–2041.

[4] Homa Alemzadeh, Daniel Chen, Xiao Li, Thenkurussi Kesavadas, Zbigniew T Kalbarczyk, and Ravishankar K Iyer. 2016. Targeted attacks on teleoperated surgical robots: Dynamic model-based detection and mitigation. In *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 395–406.

[5] Lívia CF Araújo, Luiz HR Sucupira, Miguel Gustavo Lizarraga, Lee Luan Ling, and Joao Baptista T Yabu-Uti. 2005. User authentication through typing biometrics features. *IEEE transactions on signal processing* 53, 2 (2005), 851–855.

[6] Cheng Bo, Lan Zhang, Xiang-Yang Li, Qiuyuan Huang, and Yu Wang. 2013. Silentsense: silent user identification via touch and movement behavioral biometrics. In *Proceedings of the 19th annual international conference on Mobile computing & networking*. ACM, 187–190.

[7] Tamara Bonaci, Jeffrey Herron, Tariq Yusuf, Junjie Yan, Tadayoshi Kohno, and Howard Jay Chizeck. 2015. To make a robot secure: An experimental analysis of cyber security threats against teleoperated surgical robots. *arXiv preprint arXiv:1504.04339* (2015).

[8] Tamara Bonaci, Junjie Yan, Jeffrey Herron, Tadayoshi Kohno, and Howard Jay Chizeck. 2015. Experimental analysis of denial-of-service attacks on teleoperated robotic systems. In *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems*. ACM, 11–20.

[9] Unity Game Engine. 2008. Unity game engine-official site. *[Online][Cited: October 9, 2008.] http://unity3d. com* (2008).

[10] Nicolas Falliere, Liam O Murchu, and Eric Chien. 2011. W32. stuxnet dossier. *White paper, Symantec Corp., Security Response* 5, 6 (2011), 29.

[11] Tao Feng, Ziyi Liu, Kyeong-An Kwon, Weidong Shi, Bogdan Carbunar, Yifei Jiang, and Nhung Nguyen. 2012. Continuous mobile authentication using touchscreen gestures. In *Homeland Security (HST), 2012 IEEE Conference on Technologies for*. IEEE, 451–456.

[12] G David Forney Jr. 1973. The viterbi algorithm. *Proc. IEEE* 61, 3 (1973), 268–278.

[13] Mario Frank, Ralf Biedert, Eugene Ma, Ivan Martinovic, and Dawn Song. 2013. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. *IEEE transactions on information forensics and security* 8, 1 (2013), 136–148.

[14] Mark JF Gales. 1998. Maximum likelihood linear transformations for HMM-based speech recognition. *Computer speech & language* 12, 2 (1998), 75–98.

[15] Anthony G Gallagher, Emily Boyle, Paul Toner, Paul C Neary, Dana K Andersen, Richard M Satava, and Neal E Seymour. 2011. Persistent next-day effects of excessive alcohol consumption on laparoscopic surgical performance. *Archives of Surgery* 146, 4 (2011), 419–426.

[16] Yixin Gao, S Swaroop Vedula, Carol E Reiley, Narges Ahmidi, Balakrishnan Varadarajan, Henry C Lin, Lingling Tao, Luca Zappella, Benjamın Béjar, David D Yuh, et al. [n.d.]. JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS): A Surgical Activity Dataset for Human Motion Modeling. ([n. d.]).

[17] Xuedong D Huang, Yasuo Ariki, and Mervyn A Jack. 1990. Hidden Markov models for speech recognition. (1990).

[18] Paul Kocher, Jann Horn, Anders Fogh, Daniel Genkin, Daniel Gruss, Werner Haas, Mike Hamburg, Moritz Lipp, Stefan Mangard, Thomas Prescher, et al. 2019. Spectre attacks: Exploiting speculative execution. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1–19.

[19] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. 2017. Temporal convolutional networks for action segmentation and detection. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 156–165.

[20] Colin Lea, Austin Reiter, René Vidal, and Gregory D Hager. 2016. Segmental spatiotemporal cnns for fine-grained action segmentation. In *European Conference on Computer Vision*. Springer, 36–52.

[21] Colin Lea, Rene Vidal, Austin Reiter, and Gregory D Hager. 2016. Temporal convolutional networks: A unified approach to action segmentation. In *European Conference on Computer Vision*. Springer, 47–54.

[22] Yantao Li, Hailong Hu, Zhangqian Zhu, and Gang Zhou. 2020. SCANet: sensor-based continuous authentication with two-stream convolutional neural networks. *ACM Transactions on Sensor Networks (TOSN)* 16, 3 (2020), 1–27.

[23] Henry C Lin, Izhak Shafran, David Yuh, and Gregory D Hager. 2006. Towards automatic skill evaluation: Detection and segmentation of robot-assisted surgical motions. *Computer Aided Surgery* 11, 5 (2006), 220–230.

[24] Moritz Lipp, Michael Schwarz, Daniel Gruss, Thomas Prescher, Werner Haas, Stefan Mangard, Paul Kocher, Daniel Genkin, Yuval Yarom, and Mike Hamburg. 2018. Meltdown. *arXiv preprint arXiv:1801.01207* (2018).

[25] Mitchell JH Lum, Diana CW Friedman, Ganesh Sankaranarayanan, Hawkeye King, Andrew Wright, Mika Sinanan, Thomas Lendvay, Jacob Rosen, and Blake Hannaford. 2008. Objective assessment of telesurgical robot systems: Telerobotic FLS. *Studies in health technology and informatics* 132 (2008), 263.

[26] John V Monaco, Ned Bakelman, Sung-Hyuk Cha, and Charles C Tappert. 2012. Developing a keystroke biometric system for continual authentication of computer users. In *Intelligence and Security Informatics Conference (EISIC), 2012 European*. IEEE, 210–216.

[27] Omni PHANTOM. [n.d.]. SensAble Technologies. *Inc., http://www.sensable.com* ([n. d.]).

[28] Lawrence Rabiner and Biing-Hwang Juang. 1993. Fundamentals of speech recognition. (1993).

[29] Lawrence R Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* 77, 2 (1989), 257–286.

[30] Carol E Reiley and Gregory D Hager. 2009. Task versus subtask surgical skill evaluation of robotic minimally invasive surgery. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2009*. Springer, 435–442.

[31] Carol E Reiley, Henry C Lin, Balakrishnan Varadarajan, B Vagvolgyi, S Khudanpur, DD Yuh, and GD . 2008. Automatic recognition of surgical motions using statistical modeling for capturing variability. *Studies in health technology and informatics* 132 (2008), 396.

[32] Jacob Rosen, Jeffrey D Brown, Lily Chang, Mika N Sinanan, and Blake Hannaford. 2006. Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete Markov model. *Biomedical Engineering, IEEE Transactions on* 53, 3 (2006), 399–413.

[33] Jacob Rosen, Blake Hannaford, Christina G Richards, and Mika N Sinanan. 2001. Markov modeling of minimally invasive surgery based on tool/tissue interaction and force/torque signatures for evaluating surgical skills. *Biomedical Engineering, IEEE Transactions on* 48, 5 (2001), 579–591.

[34] Jacob Rosen, Massimiliano Solazzo, Blake Hannaford, and Mika Sinanan. 2002. Task decomposition of laparoscopic surgery for objective evaluation of surgical residents' learning curve using hidden Markov model. *Computer Aided Surgery* 7, 1 (2002), 49–61.

[35] Louis B Rosenberg. 1993. Virtual fixtures: Perceptual tools for telerobotic manipulation. In *Virtual Reality Annual International Symposium, 1993., 1993 IEEE*. IEEE, 76–82.

[36] Aditi Roy, Tzipora Halevi, and Nasir Memon. 2014. An hmm-based behavior modeling approach for continuous mobile authentication. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 3789–3793.

[37] Guangyi Shi, Yuexian Zou, Yufeng Jin, Xiaole Cui, and Wen J Li. 2009. Towards HMM based human motion recognition using MEMS inertial sensors. In *Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference on*. IEEE, 1762–1766.

[38] Weidong Shi, Jun Yang, Yifei Jiang, Feng Yang, and Yingen Xiong. 2011. Senguard: Passive user identification on smartphones using multiple sensors. In *Wireless and Mobile Computing, Networking and Communications (WiMob), 2011 IEEE 7th International Conference on*. IEEE, 141–148.

[39] Zdeňka Sitová, Jaroslav Šeděnka, Qing Yang, Ge Peng, Gang Zhou, Paolo Gasti, and Kiran S Balagani. 2015. HMOG: New behavioral biometric features for continuous authentication of smartphone users. *IEEE Transactions on Information Forensics and Security* 11, 5 (2015), 877–892.

[40] Cristian Sminchisescu, Atul Kanaujia, and Dimitris Metaxas. 2006. Conditional models for contextual human motion recognition. *Computer Vision and Image Understanding* 104, 2-3 (2006), 210–220.

[41] Lingling Tao, Ehsan Elhamifar, Sanjeev Khudanpur, Gregory D Hager, and René Vidal. 2012. Sparse hidden markov models for surgical gesture classification and skill evaluation. In *Information Processing in Computer-Assisted Interventions*. Springer, 167–177.

[42] Lingling Tao, Luca Zappella, Gregory D Hager, and René Vidal. 2013. Surgical gesture segmentation and recognition. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 339–346.

[43] Charles C Tappert, Mary Villani, and Sung-Hyuk Cha. 2010. Keystroke biometric identification and authentication on long-text input. In *Behavioral biometrics for human identification: Intelligent applications*. IGI Global, 342–367.

[44] Microsoft 365 Defender Threat Intelligence Team. 2021. Analyzing attacks taking advantage of the Exchange Server vulnerabilities. *[Online][Cited: April 18, 2021.] https://www.microsoft.com/security/blog/2021/03/25/analyzing-attacks-taking-advantage-of-the-exchange-server-vulnerabilities/* (2021).

[45] Balakrishnan Varadarajan, Carol Reiley, Henry Lin, Sanjeev Khudanpur, and Gregory Hager. 2009. Data-derived models for segmentation with application to surgical assessment and training. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2009*. Springer, 426–434.

[46] Lee Woodruff White. 2013. *Quantitative objective assessment of preoperative warm-up for robotic surgery*. Ph.D. Dissertation.

[47] Junjie Yan, Kevin Huang, Tamara Bonaci, and Howard J Chizeck. 2015. Haptic passwords. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 1194–1199.

[48] Mohammad Samin Yasar and Homa Alemzadeh. 2020. Real-Time Context-aware Detection of Unsafe Events in Robot-Assisted Surgery. *arXiv preprint arXiv:2005.03611* (2020).

[49] Steve Young, Gunnar Evermann, Mark Gales, Thomas Hain, Dan Kershaw, Xunying Liu, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, et al. 1997. *The HTK book*. Vol. 2. Entropic Cambridge Research Laboratory Cambridge.

[50] Stephen John Young, NH Russell, and JHS Thornton. 1989. *Token passing: a simple conceptual model for connected speech recognition systems*. Cambridge University Engineering Department Cambridge, UK.

A CONFUSION MATRICES

A.1 VR Experiements

Table 9. Confusion Matric for the VR Experiment with 5-second Sample Window

		Predicted Label										
		Subj 1	Subj 2	Subj 3	Subj 4	Subj 5	Subj 6	Subj 7	Subj 8	Subj 9	Subj 10	Recall
Actual Label	Subj 1	4869	0	0	862	76	45	69	0	16	0	82.01%
	Subj 2	0	4934	0	0	0	0	0	0	0	0	100%
	Subj 3	0	0	3100	107	0	0	41	3	13	0	94.98%
	Subj 4	636	0	102	4088	39	0	384	0	67	0	76.90%
	Subj 5	404	1106	0	107	8969	1004	465	299	290	0	70.93%
	Subj 6	686	519	0	76	1312	5342	0	0	3	0	67.30%
	Subj 7	43	0	607	119	0	0	6156	164	282	0	83.52%
	Subj 8	0	36	30	9	0	0	483	5659	0	0	91.02%
	Subj 9	120	0	835	677	11	0	514	0	4417	341	63.88%
	Subj 10	0	0	0	0	0	0	0	0	259	4794	94.87%
	Precision	72.05%	74.81%	66.32%	67.63%	86.18%	83.59%	75.89%	92.39%	82.61%	93.36%	

Table 10. Confusion Matric for the VR Experiment with 3-second Sample Window

		Predicted Label										
		Subj 1	Subj 2	Subj 3	Subj 4	Subj 5	Subj 6	Subj 7	Subj 8	Subj 9	Subj 10	Recall
Actual Label	Subj 1	5201	17	0	1061	27	49	74	0	108	0	79.56%
	Subj 2	10	5342	0	0	72	54	56	0	0	0	96.53%
	Subj 3	0	0	3255	127	0	0	192	145	145	0	84.24%
	Subj 4	555	0	121	4887	0	0	258	0	95	0	82.61%
	Subj 5	640	1468	2	346	8214	1299	694	275	306	0	62.02%
	Subj 6	617	313	0	150	897	6561	0	0	0	0	76.84%
	Subj 7	11	0	226	372	23	0	6603	184	552	0	82.84%
	Subj 8	0	61	61	86	26	0	702	5825	56	0	85.45%
	Subj 9	94	0	798	332	6	0	261	0	5834	190	77.63%
	Subj 10	0	0	0	0	27	0	0	0	343	5283	93.45%
	Precision	72.97%	74.18%	72.93%	66.39%	88.40%	82.39%	74.69%	90.61%	78.42%	96.53%	

A.2 JIGSAWS Dataset

Table 11. Confusion Matric for the VR Experiment with 1-second Sample Window

		Subj 1	Subj 2	Subj 3	Subj 4	Subj 5	Subj 6	Subj 7	Subj 8	Subj 9	Subj 10	Recall
	Subj 1	5478	67	1	948	68	255	111	0	182	27	76.75%
	Subj 2	20	5759	4	30	158	92	64	0	7	0	93.89%
	Subj 3	0	0	3263	288	0	0	425	152	327	9	73.10%
	Subj 4	645	0	216	4742	18	29	565	0	301	0	72.77%
Actual Label	Subj 5	583	2033	50	765	6525	1451	705	1105	627	0	47.13%
	Subj 6	777	849	0	214	933	6126	0	50	189	0	67.03%
	Subj 7	125	0	352	362	136	0	5841	569	1186	0	68.15%
	Subj 8	0	131	373	242	295	0	772	5366	238	0	72.35%
	Subj 9	244	19	984	301	88	10	543	4	5651	271	69.67%
	Subj 10	51	0	0	0	82	0	0	0	468	5652	90.39%
	Precision	69.14%	65.01%	62.24%	60.09%	78.59%	76.93%	64.71%	74.05%	61.58%	94.85%	

Table 12. Confusion Matric for the JIGSAWS Suturing Task with 5-second Sample Window

		B	C	D	E	F	G	I	Recall
	B	14232	895	2397	68	135	140	0	79.66%
	C	0	13681	0	233	0	0	0	98.33%
Actual Label	D	0	59	11002	41	0	0	0	99.10%
	E	0	450	737	12618	0	0	0	91.40%
	F	0	0	0	0	11088	52	0	99.53%
	G	321	0	0	0	95	21519	21	98.01%
	I	0	0	0	0	52	13	19700	99.67%
	Precision	97.79%	90.69%	77.83%	97.36%	97.52%	99.06%	99.89%	

Table 13. Confusion Matric for the JIGSAWS Suturing Task with 3-second Sample Window

		B	C	D	E	F	G	I	Recall
	B	14531	1048	1996	86	170	336	0	79.99%
	C	0	13903	36	275	0	0	0	97.81%
Actual Label	D	0	177	11073	152	0	0	0	97.11%
	E	0	447	818	12840	0	0	0	91.03%
	F	1	0	0	0	11317	122	0	98.92%
	G	405	0	0	0	129	21654	68	97.29%
	I	5	0	0	0	85	16	19959	99.47%
	Precision	97.25%	89.26%	79.53%	96.16%	96.72%	97.86%	99.66%	

Table 14. Confusion Matric for the JIGSAWS Suturing Task with 1-second Sample Window

		B	C	D	E	F	G	I	Recall
	B	15013	1298	957	385	85	729	0	81.30%
	C	0	13881	91	502	0	40	0	95.64%
Actual Label	D	0	351	10914	437	0	0	0	93.27%
	E	0	789	846	12770	0	0	0	88.65%
	F	80	0	0	0	11290	322	48	96.17%
	G	565	0	0	0	231	21669	91	96.07%
	I	23	0	0	0	128	213	20001	98.21%
	Precision	95.74%	85.06%	85.21%	90.61%	96.22%	94.32%	99.31%	

Table 15. Confusion Matric for the JIGSAWS Needle Passing Task with 5-second Sample Window

		Predicted Label		
		C	D	Recall
Actual Label	C	13574	2162	86.26%
	D	516	14738	96.62%
	Precision	96.34%	87.21%	

Table 16. Confusion Matric for the JIGSAWS Needle Passing Task with 3-second Sample Window

		Predicted Label		
		C	D	Recall
Actual Label	C	13940	2096	86.93%
	D	446	15108	97.13%
	Precision	96.90%	87.82%	

Table 17. Confusion Matric for the JIGSAWS Needle Passing Task with 1-second Sample Window

		Predicted Label		
		C	D	Recall
Actual Label	C	14196	2140	86.90%
	D	515	15339	96.75%
	Precision	96.50%	87.76%	

Table 18. Confusion Matric for the JIGSAWS Knot Tying Task with 5-second Sample Window

		Predicted Label					
		C	D	E	F	G	Recall
Actual Label	C	5098	18	35	0	0	98.97%
	D	495	4929	157	16	0	88.07%
	E	404	69	5707	0	0	92.34%
	F	0	0	0	4629	2954	61.04%
	G	0	0	0	1310	10555	88.96%
	Precision	85.01%	98.27%	96.75%	77.73%	78.13%	

Table 19. Confusion Matric for the JIGSAWS Knot Tying Task with 3-second Sample Window

		Predicted Label					
		C	D	E	F	G	Recall
Actual Label	C	5360	9	82	0	0	98.33%
	D	600	4946	275	76	0	83.87%
	E	444	150	5886	0	0	90.83%
	F	0	0	0	4922	2961	62.44%
	G	0	0	0	1423	10742	88.30%
	Precision	83.70%	96.89%	94.28%	76.65%	78.39%	

Table 20. Confusion Matric for the JIGSAWS Knot Tying Task with 1-second Sample Window

		C	D	E	F	G	Recall
				Predicted Label			
Actual Label	C	5420	151	142	38	0	94.24%
	D	831	4968	298	100	0	80.17%
	E	717	410	5653	0	0	83.38%
	F	0	0	0	5203	2980	63.58%
	G	0	0	0	1764	10701	85.84%
	Precision	77.78%	89.85%	92.78%	73.23%	78.22%	