Instruction Sender Authentication Utilizing Touch Screen Operation Action Fingerprint for UAV Systems

Guozhu Zhao[1], Pinchang Zhang[2],
[1]School of Systems Information Science, Future University Hakodate,
116-2 Kamedanakano-cho, Hakodate, Hokkaido, 041-8655, Japan
[2]School of Computer, Nanjing University of Posts and Telecommunications, Nanjing, Jiangsu, 210023, China

Instruction sender authentication of UAV (Unmanned Aerial Vehicle) control is critical for the secure operation of UAV systems. By exploring user touch action characteristics of the user interaction with the UAV control mobile terminal, this paper proposes a novel instruction sender continuous authentication framework for UAV control systems. In particular, we first obtain the touch screen data in real time from the interaction with the mobile device screen of a user when he controls the UAV. Then we create a polynomial to fit the touch trajectory on the screen of the user, and use the least square method to obtain the optimal estimate of the polynomial coefficients. Finally, we mark the grid area covered by the polynomial to extract the user's touch screen operation action fingerprint(OAF), and employ the fuzzy extractor method to realize the identity authentication of the sender with certain error tolerance. Extensive experiments are conducted to illustrate the authentication performance of the proposed authentication framework in terms of false acceptance rate, false rejection rate and equal-error rate.

Index Terms—**Continuous authentication, UAV, fuzzy extractor, behavioral biometric, operation action fingerprint.**

I. Introduction

UAV (Unmanned Aerial Vehicle) system is a significant network architecture that encapsulates flight control, UAV monitoring and UAV network to enable the precise, efficient and real-time control to be conducted in UAV systems. With the rapid development of 5G network and cloud computing technologies, the UAV system becomes highly promising to boost the long-distance unmanned operation, high-precision real-time monitoring and drone-assisted decision-making in the global related fields. Notice that the UAV systems generally target at some critical fields, like the military, smart manufacturing, agricultural production and smart grid, so the security guarantee is of great importance for the secure control of UAV systems. However, the current UAV systems are facing various security challenges, both from the network and UAV control layer that the UAV relies on for normal operation. Among these challenges, instruction sender authentication serves as a critical one since such systems usually involve a lot of users with highly diverse authority rights.

Instruction sender authentication in a UAV system is used to verify the identities of a sender (user) who is attempting to send control instruction to the UAV, and thus to prevent unauthorized users from gain control of the UAV system. Depending on whether the authentication process is continuous or not, the instruction sender authentication in UAV systems can be roughly classified as one time sender authentication and continuous sender authentication. One time sender authentication usually requires instruction sender to be authenticated to provide credentials that can prove legitimacy of identity at the entrance to the access UAV control system, and it is generally used for the one time authentication system where the continuous monitoring of user identity legitimacy is not necessary once the user is successfully authenticated as a legitimate one [1], [2], [3], [4]. In contrast, the continuous sender authentication mainly explores the intrinsic properties related to the sender inherent activities and behaviors to carry out user authentication, so it does not need additional actions from a user for authentication purpose and can monitor construction sender identity continuously.

Continuous sender authentication is particularly attractive for the efficient and secure control of UAV systems. First, continuous sender authentication conducts sender identity verification in a non-intrusive manner, then instruction senders do not need additional frequent actions for identity authentication. Second, UAV systems are easy to be interfered, intercepted and illegally accessed by malicious attackers in practical applications, continuously verifying the identity of the instruction sender of UAV commands is essential. Thus, we are motivated to design a flexible and cost-effective continuous instruction sender authentication approach for continuous and non-intrusive user authentication in UAV systems.

By now, some research efforts have been devoted to the study of continuous sender authentication in UAV related fields. In [5], the authors demonstrate that users' interaction actions with the UAV control terminals can be defined as a sequence of flight commands using a standard radio control transmitter, and the operation action features are extracted to determine user identities using machine learning methods. The authors in [6] proposes a lightweight mutual authentication protocol to secure communications between UAVs and stations. Using the challenge-response pairs of PUF (physical unclonable function), the protocol randomly shuffle a message which piggybacks a seed for generating a secret session key.

Notice that in UAV control process, the touch screen actions from different users (e.g., slid down, slid up and slid left) exhibit unique behavioral biometric characteristics due to the difference of individual operation behavior habits, and the

Manuscript received December 20, 2021; revised January 24, 2022. Corresponding author: Guozhu Zhao (email: gzzhao2015@126.com).

touch screen characteristics [7], [8] can be represented as touch screen operation action fingerprint to identity users. To the best of our knowledge, there is no existing approach for continuous sender authentication based on touch screen operation action fingerprint. Therefore, in this paper we attempt to extract user OAF to design a continuous user authentication approach for UAV scenarios. The main contributions of this paper are summarized as follows:

- By sampling user touch screen operation action trajectories from the UAV control process of a user and adopting Least squares polynomial fitting to model these trajectories, we develop a new grid-based method to characterize the user OAF to verify user identities.
- Based on the extracted OAF, we proposed a fuzzy extractor-based approach to realize UAV instruction sender authentication continuously with certain error tolerance during UAV control processes.

The remainder of this paper is organized as follows. Section II introduces the UAV system and threat model. Section III presents the proposed instruction sender authentication approach. The experiment results and analysis are provided in Section IV. Finally, Section V concludes this paper.

II. UAV System and Threat Model

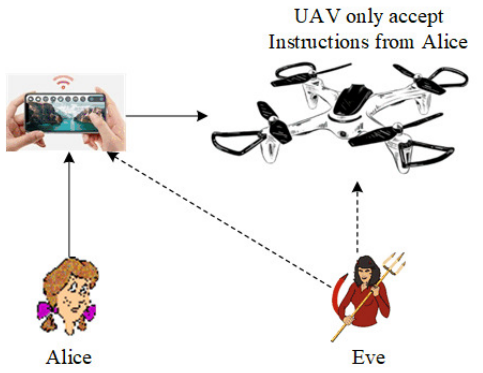


Fig. 1. Attack model for UAV control scenarios

Consider a UAV system consisting of a UAV and a legitimate user Alice who sends instruction to control the UAV using a mobile control terminal, and a potential attacker Eve, as shown in Fig. 1. In the UAV system, Alice holding/taking mobile control terminal always interact with UAV system by performing some common operation actions (e.g., slide up, slide down and slide left) on the terminal during the instruction sending processes. The potential attacker Eve has physical access to Alice's control terminal of the UAV, and is already in possession of passcodes (e.g., PIN or fingerprints) to unlock the control terminal. Thus Eve can impersonate as Alice to launch a spoof attack by implementing series of control operation actions on the mobile control terminal or illegal access to the UAV network to send control instructions to the UAV, and hope to gain control of the UAV.

As a result, an illegal instruction sender Eve can send control instructions to the UAV through alice's terminal control, and control of the UAV may be obtained by Eve. Hence, the goal of our work is to design a continuous instruction sender authentication approach for the UAV system, which discriminates sender identities continuously and non-intrusively through the tiny difference of sender's touch screen on mobile control terminals.

III. Proposed Instruction Sender Authentication Approach

In this section, we develop a flexible and cost-effective continuous sender authentication approach to determine user (sender) identities for the UAV system, which exploits the behavioral biometric characteristics from their routine UAV control processes.

As illustrated in Fig. 2, the proposed authentication approach consists of three processes: 1) OAF extraction based on user touch screen operation actions; 2) The fuzzy extractor scheme based on OAF; 3) Instruction sender authentication.

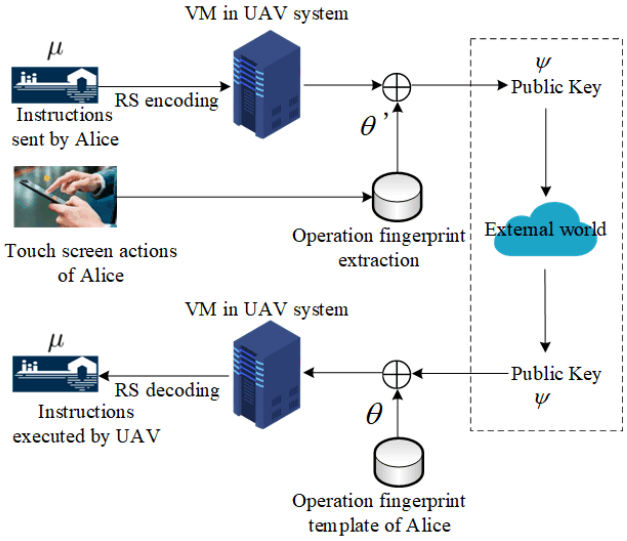


Fig. 2. The processes of the proposed instruction sernder authentication approach for UAV control scenarios.

A. OAF extraction based on user touch screen operation actions

1) Touch screen trajectory in UAV control process

A user holding a UAV control terminal always performs some common operation actions (e.g., slide up, slide down, and slide left) on the mobile device touch screen during the operation of the control UAV. The touch screen of the mobile device is typically equipped with various sensors and provide API for users to obtain the touch screen data in real time. Common operation actions a user performs generally consist of slide up for UAV rising (SUR), slide up for UAV forward (SUF), slide down for UAV down (SDD), slide down for UAV backward (SDB), slide left for UAV left flight (SLL), slide left for rotating counterclockwise (SLC), slide right for UAV right flight (SRR) and slide right for rotating clockwise (SRC), as shown in Fig. 3. We use $O = \text{SUR}, \text{SUF}, \text{SDD}, \text{SDB}, \text{SLL}, \text{SLC}, \text{SRR}, \text{SRC}$ to denote the set

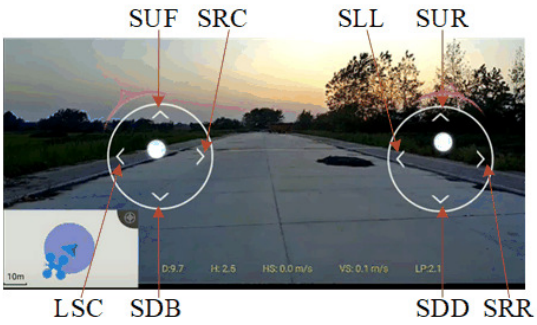


Fig. 3. Touch screen operation action during the operation of the control UAV

of operation actions from the user. An arbitrary operation action $O_k \in O$ ($k = 1, 2, \cdots, N$, N is the number of elements in O) can be regarded as a stroke, which is a sequence of touch data denoted by S_k (on the touch screen of the mobile device) that begins with the user's finger touching the screen and finishes with the user's finger leaving the screen. S_k is the trajectory of the user sliding across the screen with his finger, which is represented as $S_k = ((x_1, y_1), \cdots, (x_r, y_r))$, where (x_r, y_r) is the x-axis and y-axis of the screen coordinates of the mobile device respectively, $r \in N_k$ and N_k is the sum number of points that the trajectory S_k contains.

Generally, the trajectory of a user may contain hundreds or more points consist of (x_r, y_r) corresponding x-axis and y-axis. In principle, we should use all touch points generated during a user's control processes in the extraction of OAF to obtain better results on the authentication performance. However, when the number of points grows, multiple points very close to each other may repeatedly mark the area covered by the user's touch trajectory. For example, a trajectory contains points (238.25,812.75), (238.50,812.75) and (238.50,812.25), and these points represent almost the same area on the touchscreen of mobile terminals. As a consequence, it might not be a good choice to extract OAF utilizing all the touch screen data during UAV control process. This is because including these duplicate data in the construction of user OAF yields high computational and storage complexity without much real benefit in terms of accurate fingerprint construction and authentication performance. Thus, we proposed touch screen trajectory extraction (TTE) algorithm based on grid and polynomial fitting for OAF construction.

2) TTE algorithm

In this section, we describe the OAF that we extract from the operation action record of users' interacting with the mobile device and demonstrate its uniqueness. We develop a flexible and cost-effective TTE algorithm to obtain user touch screen trajectory for the authentication system in UAV scenario, which exploits touch screen data during the UAV control process of a user. The proposed algorithm consists of three processes: 1) Grid creation and coordinate axis conversion; 2) User touch trajectory fitting; 3) User touch trajectory extraction.

3) Grid creation and coordinate axis conversion

To accurately and systematically describe the area covered by the trajectory from a user, we create a grid with m rows and n columns, which is denoted by $\aleph$. Taking the upper right corner of grid $\aleph$ as the origin, we then establish the coordinate system of the grid. Thus, we can see that the maximum values of abscissa and ordinate of the coordinate system $\aleph$ are $\aleph abs$ and $\aleph ord$, respectively.

In the mobile device touch screen coordinate system, taking the upper right corner of the touch screen as the origin, we define the maximum coordinate values that the touch screen can represent in the horizontal and vertical directions as $H_{\max}$ and $V_{\max}$, respectively. An arbitrary point on the touch screen $(x_r, y_r))$ can be converted into the grid coordinate system, namely (α_r, β_r) and (α_r, β_r), and they can be calculated by

$$\alpha_r = (\aleph abs - 0) * (x_r - 0)/(V_{\max} - 0), \tag{1a}$$
$$\beta_r = (\aleph ord - 0) * (y_r - 0)/(H_{\max} - 0). \tag{1b}$$

According to (1), a touch trajectory $S_k = ((x_1, y_1), \cdots, (x_r, y_r))$ is mapped to the grid coordinate system and is rewritten as $\Upsilon_k = ((\alpha_1, \beta_1), \cdots, (\alpha_r, \beta_r))$.

4) User touch trajectory fitting

To accurately and efficiently describe user touch trajectory in coordinate system $\aleph$, we first sample the Υ_k according to the time T_k when the touch trajectory occurs. To ensure that the sampling points can cover the entire touch trajectory as completely as possible, we then divide T_k into d parts evenly, that is, the sampling interval is $\frac{T_k}{d}$, so that the sum number of sampling points is $\zeta = \frac{T_k}{d} + 1$. Finally, for a given touch trajectory Υ_k, the corresponding sampling result denoted by v_k is written as $v_k = ((\alpha_1, \beta_1), \cdots, (\alpha_q, \beta_q))$, $q = 1, 2, \cdots, \zeta$.

5) User touch trajectory extraction

To obtain user touch trajectory based on the points in v_k, we leverage k-order polynomial $f(\alpha)$ to fit the user's touch trajectory, and $f(x)$ is given by

$$f(\alpha) = a_0 + a_1\alpha, +a_2\alpha^2 +, \cdots, +a_k\alpha^k, \tag{2}$$

where $a_0, a_1, \cdots, a_k$ are the coefficients of the polynomial $f(\alpha)$. Because the ζ points are on the curve represented by the polynomial $f(\alpha)$, we can get ζ equations, i.e.,

$$a_0 + a_1\alpha_1, +a_2\alpha_1^2 +, \cdots, +a_k\alpha_1^k = \beta_1, \tag{3a}$$
$$a_0 + a_1\alpha_2, +a_2\alpha_2^2 +, \cdots, +a_k\alpha_2^k = \beta_2, \tag{3b}$$
$$\vdots \tag{3c}$$
$$a_0 + a_1\alpha_\zeta, +a_2\alpha_\zeta^2 +, \cdots, +a_k\alpha_\zeta^k = \beta_\zeta. \tag{3d}$$

These equations are not necessarily solvable, that is, the curve of the k-order polynomial may not fit every point of v_k. Hence, we use Least Square Method to obtain the optimal parameters of the polynomial $f(\alpha)$. Let $L(f(\alpha), \beta)$ be the sum of squares of errors between the fitted data $f(\alpha)$ and the actual data β, and $L(f(\alpha), \beta)$ is given by

$$L(f(\alpha), \beta) = \sum_{i=1}^{\zeta} (f(\alpha) - \beta)^2. \tag{4}$$

Then we can obtain Optimal estimation of parameters for the polynomial $f(\alpha)$ under Min$(L(f(\alpha), \beta))$.

Employing the intersection of polynomial $f(\alpha)$ and lines around the grid, we can accurately describe the user's touch screen operation fingerprint. Specifically, For each time the user touches the screen, we first obtain a square area denoted by $\S = \alpha \in [\text{Min}(\alpha_r), Max(\alpha_r)], \beta \in [\text{Min}(\beta_r), Max(\beta_r)]$, where $r = 1, 2, \cdots, \zeta$. Let the intersection of the curve $f(\alpha)$ and the vertical line of the grid be $\Phi = f(\alpha), \alpha = i$, $i \in \S$ and $i \in N$, and the intersection of the curve and the horizontal line of the grid be $\Psi = f(\beta), \beta = j$, $j \in \S$ and $j \in N$, respectively. According to grid mark rules defined in (5), we mark the grids involved in these intersections and accumulate the number of times each grid is marked to obtain user OAF in a touch screen action.

$$\text{Grid mark rules} \in \begin{cases} if\ f(\alpha) > g - 1 \text{ and } f(\alpha) < g: \\ \text{Mark ceil } \{\text{grid}(i, g), \text{grid}(i + 1, g)\}, \\ if\ \text{PloySolve}(j) > h - 1 \text{ and} \\ \text{PloySolve}(j) < h: \\ \text{Mark ceil } \{\text{grid}(h, j), \text{grid}(h + 1, j)\}, \\ if(\alpha) == j: \\ \text{Mark ceil } \{\text{grid}(i, j), \text{grid}(i + 1, j), \\ \text{grid}(i + 1, j), \text{grid}(i + 1, j + 1)\}, \end{cases} \tag{5}$$

where $(i, f(\alpha)) \in \Phi$ and $(PloySolve(j), j) \in \Psi$.

6) The construction of OAF based on user touch screen operation actions

We can see that a trajectory of user touch screen action is represented as a polynomial, and the mark rules in (5) are used to mark the precise path covered by a user's touch action. Hence, we use χ to denote the user's one touch, and χ is written as $\chi = (i, j)$ where grid(i,j) is marked.

To capture the user's touch motion characteristics accurately and stably, we employ a user's κ touch screen actions to construct the user's OAF, and we have OAF $= [\chi_1, \cdots, \chi_\kappa]$. We provide in Fig. 4 the differences of operation action sequences' time-varying properties between User 1 and User 2.

B. The fuzzy extractor scheme based on OAF

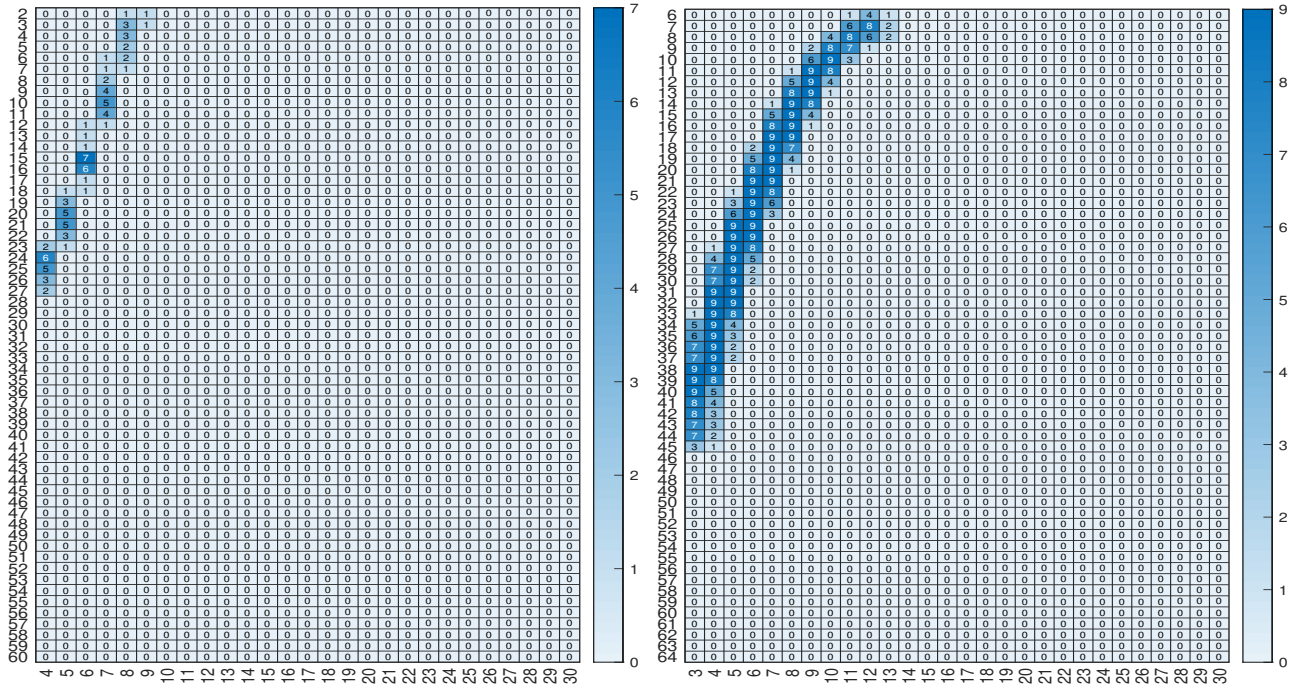
Traditional cryptography requires precisely reproducible random strings for secrets. However, the secret strings are generally uniformly distributed and are difficult to create, store, and reliably retrieve. Moreover, the interaction between UAV sender and receiver is very frequent, and the production, management and update of passwords for securing the instructions during the UAV control processes need to consume more UAV storage and network resources. It may bring a large burden of storage and transmission to the UAV system, and even lead to problems such as decreased UAV control efficiency and excessive command delay. Notice that strings generated from human biometric features generally are neither uniformly random nor reliably reproducible [9], making it suitable for securing the UAV system. For example, a random person's fingerprint or iris scan is clearly not a uniform random string, nor does it get reproduced precisely each time it is measured. Similar to some human biometric features, such as iris, face and fingerprint, the OAF is clearly not a uniform random string, nor does it get reproduced precisely each time it is measured.

We use extractor theory proposed in [10] to realize the secure transmission and reproduction of UAV instructions. The fuzzy extractor in this paper includes two processes: (1) The UAV instruction is encoded as a public key ψ, and $\psi = \text{Encoder}(\theta, \mu)$, where μ is the instruction that will be sent to a specific UAV and θ is the behavioral biometric from the user (UAV instruction sender). (2) The UAV instruction $\mu^{'}$ is restored from the public key ψ by using user behavioral biometric feature $\theta^{'}$, and $\mu^{'} = \text{Decode}(\psi, \theta^{'})$. In UAV control processes, only the public key ψ is stored and transmitted in internal cloud platform and external network space, and user behavioral biometric (θ and $\theta^{'}$) and the UAV instruction (μ and $\mu^{'}$) are only generated locally and stored briefly. As shown in Fig. 2, the UAV instructions are bound to the behavioral biometrics of the instruction sender, and we can see that the UAV instruction transmission and reproduction process is much more complicated and security than the traditional instruction sending and transmission mechanism.

In particular, let θ and $\theta^{'}$ be the template OAF stored in the UAF firmware and the real-time OAF extracted from user touch screen operation actions [9], [10]. For a UAV control instruction denoted by μ (here, μ can be the plaintext of the UAV control instructions, or the encrypted ciphertext of the UAV control instructions) with a length of k bits, the fuzzy extractor scheme first creates a Galois field [10] array from data μ, θ and $\theta^{'}$ denoted by $\Theta_{message} = G(\mu, 2^m)$, $\Theta_\theta = G(\mu, m)$ and $\Theta_{\text{message}} = G(\theta^{'}, m)$, and we can see the length of θ is $2^m - 1$ [11], [12]. Then, we use Reed-Solomon codes to obtain the corresponding code RS_{message} of Θ_{message}, and a public key ψ is generated from the exact key RS_{message} and real time OAF $\theta^{'}$ extracted from user operation action during UAV control processes, i.e., $\psi = C(\theta^{'}, RS_{\text{message}})$, where $C(\cdot)$ is XOR operator. Finally, we obtain the UAV operation instructions of the sender by Reed-Solomon decoding denoted by $\mu^{'}$ utilizing user real time OAF, and we have $\mu^{'} = \text{RS-Decoding}(C(\psi, \theta))$ [13].

C. Instruction sender authentication

The processes of instruction sender authentication include two phases: (1) User OAF enrollment. Initially, we enroll OAF of all legitimate users into the UAV local storage. For a specific UAV flight business, we require each user to operate the UAV within the line-of-sight range to complete this specific UAV flight business, and extract the user's OAF according to the proposed method in section III-A. We can see that the enrolled OAF is represented as θ in Fig. 2. (2) Instruction sender authentication. For a user who perform operation action on the UAV control terminal, we extract his/her real-time OAF $\theta^{'}$ to generate public key ψ. Using $\mu^{'} = \text{RS-Decoding}(C(\psi, \theta))$, we can obtain the UAV instruction $\mu^{'}$. It can be seen that the task of constructing a fuzzy extractor is much more complicated than the traditional authentication problem since it requires the construction of a non-binary function $\rho \in 0, 1$, and the



(a) OAF of User1 (b) OAF of User2

Fig. 4. The difference of OAF from two users under the same control business (i.e., the two user complete the same task using a UAV system).

decoding process $\mu^{'} = \text{RS-Decoding}(C(\psi, \theta))$. Hence, we can verify user identity employing above fuzzy extractor process, and we have

$$\rho \in (\theta, \theta^{'}) \in \begin{cases} 0 \longrightarrow \mu = \mu^{'}, \text{a legitimate sender} \\ 1 \longrightarrow \mu \neq \mu^{'}, \quad \text{an attacker.} \end{cases} \tag{6}$$

IV. Authentication Performance Analysis

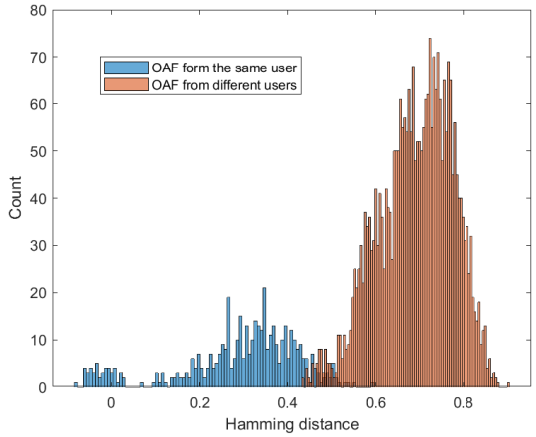


Fig. 5. Hamming distance statistical analysis

A. The robustness of OAF

We select OAF from the same user's touch screen actions to compute the Hamming distances from the same user and chose OAF from different OAF to compute the Hamming distances. We perform 2900 comparisons between different OAF from the same user and 450 comparisons for the OAF from different users. The results are shown in Fig. 5. We can see that most of the Hamming distance of OAF from the same user is less than 0.56. However, the Hamming distance between different users is usually greater than 0.45. This indicates that by setting different thresholds of Hamming distance, we can obtain better user authentication results using fuzzy extractor scheme in practical instruction sender authentication of UAV system.

B. Performance metric

To evaluate the performance of the proposed UAV sender authentication approach, we apply three typical metrics, namely the false acceptance rate (FAR), false rejection rate (FRR) and equal-error rate (EER) [14]. We recruit 10 volunteers (users) to complete a specific UAV control tasks for extracting their OAF. The 10 users under the same UAV control task are evenly divided into two groups A and B to construct 2 sub-datasets denoted by $\wp_1$ and $\wp_2$. In $\wp_1$, we require users in group A and group B perform their UAV control operation action without impersonating each other and collect their operation actions on UAV control terminals 10 times (the user completes the same UAV control task each time) for each user. In $\wp_2$, we first conduct the one-to-one randomly pairing between users in group A and group B. We then require users in group A to impersonate the operation actions of his corresponding pair in the group B, and collect the operation actions 5 times for each user in group A. Finally, we collect the UAV control operation actions from users in group B 5 times for each user. We can see that the dataset $\wp_1$

is generated by user normal operations on the UAV control terminal and the dataset $\wp_2$ contains impersonation attack during the UAV control processes.

We then perform instruction sender authentication utilizing the proposed UAV sender authentication approach based on datasets $\wp_1$ and $\wp_2$. We calculate three typical metrics to obtain FAR=0.041, FRR=0.035 and EER =0.038 for dataset $\wp_1$, and FAR=0.047, FRR=0.05 and EER =0.049 for dataset $\wp_2$. The experiment results show that the proposed approach can achieve better authentication performance and has a certain ability to resist impersonation attacks. Hence, the proposed instruction sender authentication approach is promising to adapt various complicated UAV application environments for securing the UAV system.

V. Conclusion

By exploiting OAF caused by user's touch screen operation actions during UAV control processes, this paper proposed a novel instruction sender authentication approach for UAV systems. We demonstrated that the new approach enables a flexible and efficient authentication performance for satisfying different authentication performance requirements across various UAV control scenarios. Moreover, it is expected that the new authentication approach can serve as a good enhancement and complementary to the traditional authentication solutions for UAV systems.

References

[1] A. Roy, N. Memon, and A. Ross, "Masterprint: Exploring the vulnerability of partial fingerprint-based authentication systems," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 9, pp. 2013–2025, Sep. 2017.

[2] S. Li and A. C. Kot, "Fingerprint combination for privacy protection," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 2, pp. 350–360, Feb. 2013.

[3] D. F. Smith, A. Wiliem, and B. C. Lovell, "Face recognition on consumer devices: Reflections on replay attacks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 736–745, Apr. 2015.

[4] S. Thavalengal, I. Andorko, A. Drimbarean, P. Bigioi, and P. Corcoran, "Proof-of-concept and evaluation of a dual function visible/nir camera for iris authentication in smartphones," *IEEE Transactions on Consumer Electronics*, vol. 61, no. 2, pp. 137–143, May. 2015.

[5] A. Shoufan, "Continuous authentication of uav flight command data using behaviometrics," in *2017 IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC)*, 2017, pp. 1–6.

[6] C. Pu and Y. Li, "Lightweight authentication protocol for unmanned aerial vehicles using physical unclonable function and chaotic system," in *2020 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN*, 2020, pp. 1–6.

[7] H. Liu, Y. Wang, J. Liu, J. Yang, Y. Chen, and H. V. Poor, "Authenticating users through fine-grained channel information," *IEEE Transactions on Mobile Computing*, vol. 17, no. 2, pp. 251–264, Feb. 2018.

[8] R. Alazrai, A. Awad, B. Alsaify, M. Hababeh, and M. I. Daoud, "A dataset for Wi-Fi-based human-to-human interaction recognition," *Data in Brief*, vol. 31, p. 105668, Aug. 2020.

[9] E. Zainulina and I. Matveev, "Methods of using fuzzy extractors on the iris data *," in *Situation, Language, Speech, 2019At: Rome, Italy*, 2019, pp. 1–15.

[10] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM Journal on Computing*, vol. 38, no. 1, pp. 97–139, 2008.

[11] N. Li, F. Guo, Y. Mu, W. Susilo, and S. Nepal, "Fuzzy extractors for biometric identification," in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, 2017, pp. 667–677.

[12] F. Hao, R. Anderson, and J. Daugman, "Combining crypto with biometrics effectively," *IEEE Transactions on Computers*, vol. 55, no. 9, pp. 1081–1088, 2006.

[13] B. Fuller, L. Reyzin, and A. Smith, "When are fuzzy extractors possible?" *IEEE Transactions on Information Theory*, vol. 66, no. 8, pp. 5282–5298, 2020.

[14] C. Shen, Y. Li, Y. Chen, X. Guan, and R. A. Maxion, "Performance analysis of multi-motion sensor behavior for active smartphone authentication," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 1, pp. 48–62, Jan. 2018.