Towards the Security of AI-enabled UAV Anomaly Detection

Ashok Raja
Department of CIS
University of Massachusetts Dartmouth
Email: araja1@umassd.edu

Mengjie Jia
Department of CIS
University of Massachusetts Dartmouth
Email: mjia@umassd.edu

Jiawei Yuan
Department of CIS
University of Massachusetts Dartmouth
Email: jyuan@umassd.edu

***Abstract*—Unmanned aerial vehicles (UAVs) are increasingly adopted to perform various military, civilian, and commercial tasks in recent years. To assure the reliability of UAVs during these tasks, anomaly detection plays an important role in today's UAV system. With the rapid development of AI hardware and algorithms, leveraging AI techniques has become a prevalent trend for UAV anomaly detection. While existing AI-enabled UAV anomaly detection schemes have been demonstrated to be promising, they also raise additional security concerns about the schemes themselves. In this paper, we perform a study to explore and analyze the potential vulnerabilities in state-of-the-art AI-enabled UAV anomaly detection designs. We first validate the existence of security vulnerability and then propose an iterative attack that can effectively exploit the vulnerability and bypass the anomaly detection. We demonstrate the effectiveness of our attack by evaluating it on a state-of-the-art UAV anomaly detection scheme, in which our attack is successfully launched without being detected. Based on the understanding obtained from our study, this paper also discusses potential defense directions to enhance the security of AI-enabled UAV anomaly detection.**

I. Introduction

Recent years have witnessed significant growth in the adoption of UAVs, or drones, in various applications, such as intelligence, surveillance, and reconnaissance (ISR), search and rescue, and infrastructure inspection [1]–[3]. Meanwhile, due to the possible software and hardware failures and potential external attacks, the security and reliability of UAVs are increasingly drawing research attention [4]. The abnormal status of UAVs will cause them to fail their missions and even lead to public safety threats, such as crashes in public areas.

To address the growing concerns about UAV security and reliability, recent research has focused on leveraging AI and machine learning techniques to design UAV anomaly detection and recovery schemes [5]–[10]. These schemes typically build a prediction model for UAV flying status using deep neural network architectures, and then compare the prediction result with the actual system measurement to determine whether the UAV status is abnormal or not. While these AI-enabled schemes have shown their effectiveness for the anomaly detection and recovery of UAV flight data, limited attention has been given to the security of these schemes themselves. Specifically, existing AI-enabled prediction models for UAV status rely on time series analysis of UAV's flight data as they are continuously produced during the UAV's operation. However, multiple recent studies have demonstrated that deep learning models used for time series data analysis can be vulnerable to adversarial attacks [11], [12]. If such vulnerabilities also exist in state-of-the-art AI-enabled UAV anomaly detection and recovery designs, they can be leveraged by malicious entities to bypass the detection and launch different attacks towards UAVs. Therefore, it is critical to uncover such potential vulnerabilities and obtain an improved security understanding for the integration of AI and UAVs.

To fill this research gap, in this paper we perform the study on the security property of AI-enabled UAV anomaly detection with a focus on roll angle data. Roll angle data is one of the most important parameters to reflect the safe operation of UAVs and has been widely adopted by recent research for UAV anomaly detection [5]–[7]. In a typical AI-enabled anomaly detection design, a UAV's status is predicted by the AI model, which is then compared with the actual sensor reading or system status using a threshold. If the difference exceeds the threshold, the status is considered as abnormal and recovery procedure will be applied to adjust the error. Therefore, identifying vulnerabilities can be initially transformed as designing adversarial inputs that can change the prediction of AI models for UAV status but keep the difference within threshold to avoid being detected. However, the capability of such attacks constructed using one-time adversarial inputs is restricted, i.e., the changes caused to the UAV status are bounded by the threshold to avoid being detected, which can be eventually adjusted by the UAV system. Therefore, the challenging question here becomes whether it is possible to construct effective attacks that can continuously exploit the potential vulnerabilities to bypass the UAV anomaly detection and cause significant impact on UAV operations.

To answer this question, this paper first explores the possibility of constructing adversarial inputs towards AI-enabled UAV anomaly detection, which confirms the existence of vulnerability. Then, we analyze the upper-bound attack effectiveness of a single attack with adversarial inputs using dynamic analysis. On top of that, we propose an effective iterative attack that is able to gradually change the status of UAVs without being detected. We evaluate our proposed attack on the recently proposed AI-enabled UAV anomaly detection design [5]. Our evaluation results not only validate the security vulnerability in widely adopted "AI prediction + threshold" UAV anomaly detection designs, but also demonstrate the

ICC 2023 - IEEE International Conference on Communications | 978-1-5386-7462-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICC45041.2023.10279224

978-1-5386-7462-8/23/$31.00 ©2023 IEEE

effectiveness of our iterative attack. Specifically, our attack is able to change the roll angle of a UAV up to 46 degrees when appropriately launched, which is not being detected by the UAV anomaly detection design. Based on the understanding obtained from our investigation, this paper also discusses potential defense directions to enhance the security of AI-enabled UAV anomaly detection.

The rest of the paper is organized as follows: In Section II, we review and discuss related works. In Section III, we formulate the problem and propose the detailed construction of our attacks. In Section IV, we evaluate and discuss our proposed attacks and the identified vulnerability. We discuss potential defenses in Section V and conclude our paper in Section VI.

II. Related Work

A. AI-enabled UAV Anomaly Detection

The problem of anomaly detection for UAVs has drawn a lot of research attention given its importance in assuring the safety and security of UAV operations [5]–[10], [13]–[18]. In particular, machine learning and AI techniques are widely adopted in recent research. The learning-based approach monitors the status of a system using a trained machine learning or deep learning model. In [18], neural networks and Extended Kalman Filter (EKF) are integrated to support the detection of faults in UAVs' sensors and actuators. Recently, Zhong *et al*. [6] introduced a spatio-temporal correlation based anomaly detection method for high-dimensional flight data. In this method, an artificial neural network is first used for spatio-temporal correlation analysis to select the most correlated flight parameters with the monitored sensor. Then, these selected parameters are used as the input of a long short-term memory (LSTM)-based regression model for estimation. Wang *et al*. [5] also created a regression model based on LSTM networks with residual filtering to prevent random noise. The model's input is multivariate time series data with several attitude parameters. The outcomes of their study show that their methodology can lessen the impact of random noise in flight data and enhance the sensitivity of the detection to minor faults. In [7], a multi-output convolutional LSTM is proposed for multivariate anomaly detection of UAV flight data. This design combines convolutional neural networks (CNN) and LSTM to achieve sequence-to-sequence prediction.

B. Adversarial Attacks to Neural Network on Time series Data

Adversarial examples make the neural networks vulnerable and this was first pointed out by Szegedy *et al*. [19]. Papernot *et al*. [20] shows that the adversarial attacks on the computer vision domain can be translated to other domains. Since then, a significant amount of research efforts have been spent towards adversarial attacks on time series data. Recently, Karim *et al*. [11] designed an Adversarial Transformation Network (ATN) that takes an input time series sample and generates an adversarial sample to attack the time series classification models. Xu *et al*. [12] proposed a method to use gradient information to generate adversarial attacks for the LSTNet model. Their goal is to maximize the $L1$ loss function so that the model outputs incorrect predictions. They use the signed gradient and the direction of the gradient is determined based on where the loss function yields the maximum value. Mode *et al*. [21] designed adversarial attacks for multivariate time series regression models using popular techniques in the image classification domain like the Fast Gradient Sign Method (FGSM) and Basic Iterative Method (BIM). Both methods leverage gradient information of the loss function to generate adversarial examples.

III. Construction of Attacks

A. Problem Formulation

The AI-enabled UAV anomaly detection can be divided into two major stages as shown in Fig.1, i.e., 1) predicting the next status of UAV by feeding measured sensor and system data into a pre-trained AI-model, and 2) comparing the difference between prediction data and system reading with a predefined threshold to determine the abnormal status of the UAV. We adopt the recently proposed UAV anomaly detection design [5] as the representative to explore the security vulnerability in AI-enabled design. In particular, the anomaly detection design in [5] utilizes a LSTM-based multivariate deep neural network (DNN) as the AI model. The model integrates inputs from multiple related sensors to predict the UAV roll angle, which is then used as an indicator for the abnormal status. LSTM-based design as well as roll angles are also adopted by other state-of-the-art AI-enabled UAV anomaly detection [6], [7].

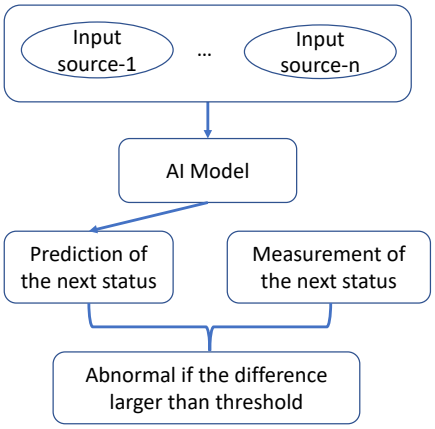

Fig. 1. Overall of AI-enabled UAV Anomaly Detection

We now denote the DNN model for prediction as $f(\cdot)$ and the prediction using $f(\cdot)$ at timestamp t with input X_t as $f(X_t) \rightarrow P_t$. Here X_t is a sliding window of grouped sensor input sequences before t and they are used to predict the sensor value at the next time step. They can be denoted as

$$X_t = \begin{bmatrix} x_{1,t-D+1}, x_{1,t-D+2}, \cdots, x_{1,t-1} \\ x_{2,t-D+1}, x_{2,t-D+2}, \cdots, x_{2,t-1} \\ \cdots \\ x_{n,t-D+1}, x_{n,t-D+2}, \cdots, x_{n,t-1} \end{bmatrix} \quad (1)$$

where $x_{i,t-j}$ is the input from the i-th sensor at j-th timestamp before t. The anomaly detection at timestamp t can be represented as $(g(f(X_t), Y_t) \leq \mathcal{T})$ where $\mathcal{T}$ is a pre-defined threshold and $g(.)$ is the method to compare the prediction value and the actual measurement of the target sensor Y_t. Therefore, the attack toward such an anomaly detection design can be formulated as

$$\beta_t, \delta_t \quad s.t. \quad g(f(X_t + \beta_t), Y_t + \delta_t) \leq \mathcal{T} \tag{2}$$

where δ_t is the modification applied to target sensor's reading at timestamp t and β_t is the adversarial perturbation injected before timestamp t. When β_t can cause the corresponding change of prediction that makes $g(f(X_t + \beta_t), Y_t + \delta_t) \leq \mathcal{T}$, then the abnormal change δ_t injected to the target sensor will not be detected by the UAV anomaly detection system. Different from δ_t that is applied for attacking directly at timestamp t, applying β_t can take multiple timestamps. This is because X_t covers a sliding window of timestamps and adversarial perturbation for a particular timestamp needs to be applied at that particular timestamp. Thus, the design of a successful attack at timestamp t needs to gradually affect sensor readings before t and eventually cause the prediction at t to become close to the attacked sensor reading $Y_t + \delta_t$ to avoid being detected. Based on our initial analysis, we now present the detailed construction of our proposed attack.

B. Detailed Construction of Attack

The construction of our attack is going to leverage the vulnerability of the threshold-checking design in AI-enabled UAV anomaly detection. First, due to the existence of threshold-checking, a single attack can be easily launched at timestamp t with a small δ_t that makes $g(f(X_t), Y_t + \delta_t) \leq \mathcal{T}$. In this kind of single attack, δ_t is likely to be bounded by $\mathcal{T}$ because the predicted value $f(X_t)$ will be close to the normal sensor reading Y_t, which can eventually be tolerated by the UAV system. Although such single attacks cannot directly affect the UAV system, it can be leveraged as the building block to construct effective attacks and exploit the vulnerability.

To be specific, by continuously launching single attacks with small attacking values, we are able to form a sliding window of attacked data $(X_t + \beta_t)$ before timestamp t without being detected. Therefore, the prediction value $f(X_t + \beta_t)$ at t will also be changed. By denoting the prediction difference as $\epsilon_t = f(X_t + \beta_t) - f(X_t)$, the attacking capability of the next single attack at timestamp t is also increased from β_t to $\beta_t + \epsilon_t$. While attacks towards a single sliding window only increase the attacking capability by ϵ_t, it can be accumulated as a large increment if such attacks are continuously launched for sequences of sliding windows as the example shown in Fig.2.

Based on this concept, we propose the detailed construction of our iterative attack in Algorithm 1, in which attacking values are incrementally raised to reach the attacking goal. Specifically, our algorithm first sets an attacking goal δ^*, i.e., at the end of the attack our algorithm aims to modify the value of the target sensor by δ^* as $Y + \delta^*$. Meanwhile, an initial attacking value will be estimated for the first round of attack based on the potential threshold used by the UAV's anomaly detection system. The attacking value β will be adjusted according to the gradient vector pre-computed offline. To generate the gradient vector, we first performed multiple offline attacks over the flight data of different flight paths. Then, the average of gradient time series from different attacks with the same trend (e.g., increasing or decreasing the target sensor value) is adopted from the pre-computed gradient vectors. We use the offline average gradient vectors to estimate online gradient with a similar flying path because the parameters to compute them are typically not available for adversaries during real-time attacks. Our algorithm also uses two additional parameters α, μ to adjust the increment of the attacking value during the attack.

The successful execution of our algorithm also depends on the appropriate estimation of the threshold to assure $g(f(X_t), Y_t + \beta) \leq \mathcal{T}$. Given the fact the actual threshold used by the UAV may not be available to external entities, we estimate the threshold using offline dynamic analysis by applying different thresholds to the UAV anomaly detection scheme. As a small threshold ($\mathcal{T}_s$) can cause high false positive rate and a large threshold ($\mathcal{T}_l$) can cause high false negative rate, a reasonable threshold adopted by the UAV system

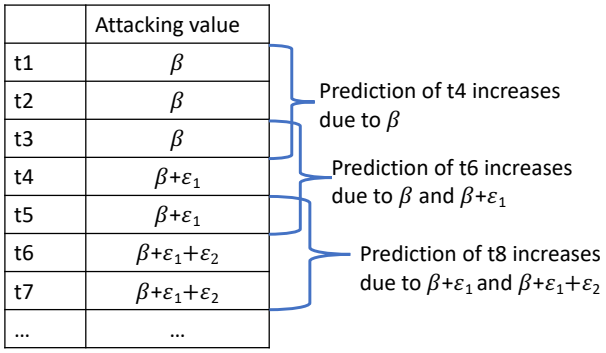

Fig. 2. Example of Accumulated Attacking Capability

Algorithm 1: Construction of Iterative Attack

Input: δ^* : attacking goal,
G: offline gradient vector,
D: Sliding window size,
α, μ: attacking value incremental parameters

Set $j = 0$, $\delta = 0$, $\beta =$initial attacking value;
while $\delta < \delta^*$ **do**
  Set $i = 1$;
  while $i \leq D$ **do**
    Launch single attack with β
  end
  $\delta = \delta + \beta$
  $\beta = \beta + (\alpha * G[j])$
  $\alpha = (1 + \mu)\alpha$
  j = j+1;
end

shall be within $(\mathcal{T}_s, \mathcal{T}_l)$ to achieve high detection accuracy. Therefore, the estimated threshold shall also be chosen from $(\mathcal{T}_s, \mathcal{T}_l)$. The selection of the estimated threshold raises a trade-off between the probability of being detected and the attacking speed. On one hand, a small threshold has a high probability to be smaller than the actual threshold and assures that $g(f(X_t), Y_t + \beta) \leq \mathcal{T}$ when launching the attack with β. On the other hand, a large threshold allows a larger attacking value β in each round and hence can achieve the attacking goal faster.

C. Discussion

In our proposed attack, the attacking value is applied to the target sensor only. While it is the major factor that contributes to the prediction $f(X_t)$ in anomaly detection, $X_t = [X[1]_t, X[2]_t, \cdots, X[n]_t]$ also contains inputs from other related sensors. When applying an attacking value β to the target sensor in X_t (say $X[k]_t$), we have

$$X_t + \beta = [X[1]_t, \cdots, X[k]_t + \beta, \cdots, X[n]_t]$$
$$f(X_t + \beta) - f(X_t) < \beta$$

With the iterative attacks going on, the accumulation of attacking values in the prediction $f(X_t + \beta)$ is slower than that directly applied to the target sensor $Y_t + \delta$, and eventually makes their gap become larger than the threshold and fails the attack. As shown in Fig.3, such a gap keeps increasing while continuously attacking. Therefore, although our attack is able to significantly change the target sensor value, it is still bounded by the threshold of the anomaly detection system at some point. A more detailed evaluation in terms of such limitations is provided in Section IV.

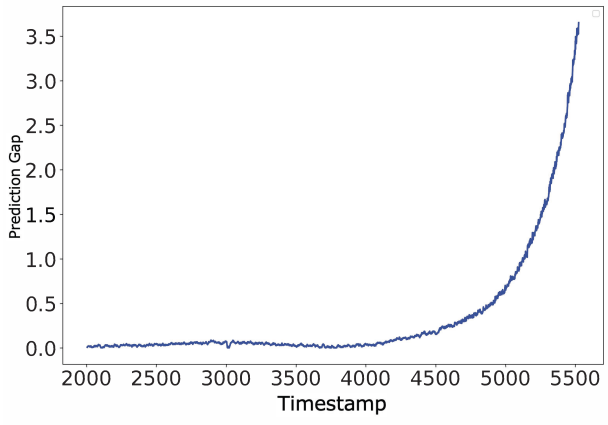

Fig. 3. Example of Prediction Gap

IV. Evaluation

A. Experimental Setup

The evaluation of our proposed attack is performed on the LSTM-based UAV anomaly detection design recently proposed in [5]. This design adopts a neural network with an input layer, one stacked LSTM layer, and an output layer that compiles the data extracted by previous layers to form the final output. 10 sensors are used as inputs as summarized in Table I and the prediction is made toward roll angle for anomaly detection. UAV flight data to train the anomaly detection system is generated using PX4 autopilot [22] and QGroundControl [23] with 10 different flight paths. By setting the sliding window size as 5, our dataset consists of 3,000 sliding windows of sensor data for training after reformulation and normalization. The attacking data is generated in real-time and injected into the simulation of UAV operations in PX4 autopilot. The final estimated threshold is set as $\mathcal{T} = 3.69$, which is consistent with the parameters adopted in the anomaly detection design in [5]. The initial attacking value is set from 2.5 to 3.5. We also set $\mu = 0.01$ to optimize the effectiveness of our attack according to our experiments. All experiments are performed on a desktop computer with i7 8-core CPU, 32GB memory, and one RTX 3070 GPU.

TABLE I
Sensor Inputs

Index	Sensor Data	Unit
1	Roll Angle	degree
2	Pitch Angle	degree
3	Yaw Angle	degree
4	Actuator Roll Angle	degree
5	Actuator Pitch Angle	degree
6	Actuator Yaw Angle	degree
7	Roll Rate	degree/s
8	Pitch Rate	degree/s
9	Yaw Rate	degree/s
10	Airspeed	m/s

B. Experimental Results

In our evaluation, we first validate that our attack can be successfully launched to the AI-enabled UAV anomaly detection. As shown in Fig.4, our attack is able to cause the prediction design in anomaly detection to increase together with the attacked sensor value, and hence avoid being detected by the threshold-checking design. It is also noteworthy that our attack is able to maintain the attacked sensor and prediction value at a certain level when necessary. While our attack fails after changing the roll angle by $46°$ due to the existence of other sensor inputs as discussed in Section III-C, it is already a considerable amount of changes in practical UAV operations and can significantly affect the security and safety of UAVs.

With regard to the attacking speed, Fig.5 shows that our attack can reach the maximum change of roll angle with 28.9 seconds. In particular, the attacking time increases significantly when the attacking goal becomes larger than $40°$. This is because when the attacking goal is close to the maximum attacking capability, the increasing rate of attacking values needs to be slowed down to avoid being detected. As shown in Table II, larger values can be selected for incremental parameter α to achieve a higher attacking speed for small attacking goals. As a comparison, the value of α has to be reduced to small ones when attacking goals are close to the boundary.

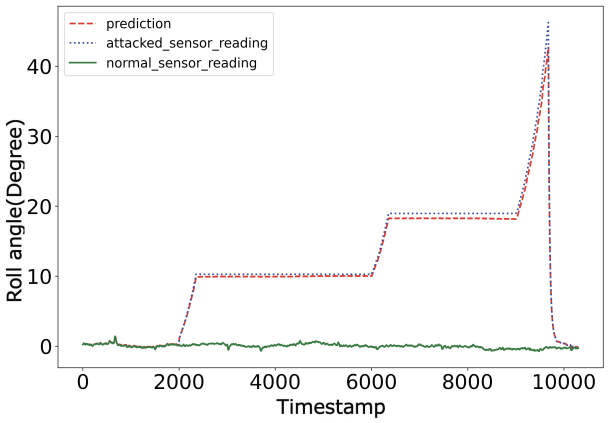

Fig. 4. Successful Attacks to the AI-enabled UAV anomaly detection

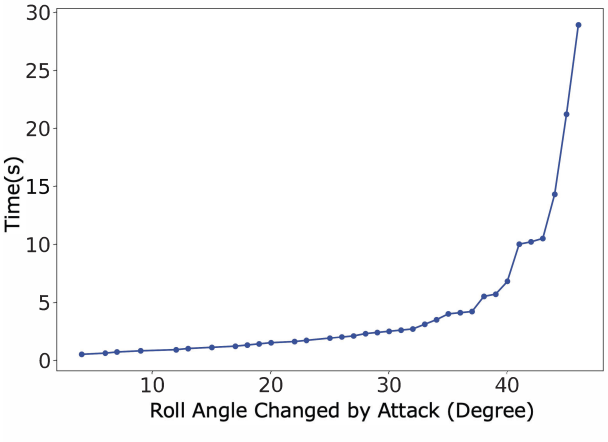

Fig. 5. Attacking Time for Different Attacking Goals

To further understand how the threshold adopted by the anomaly detection system can affect the attacking effectiveness of our attack, we evaluate the maximum roll angle that can be changed by our attack under different thresholds. In particular, we set the range of the estimated threshold from 0.6 to 6, which achieves false positive rate and false negative rate at 0 respectively for anomaly detection without attacks. As presented in Fig.6, the attacking capability of our attack increases with the threshold. This is because a larger threshold allows a bigger gap between the prediction value and target sensor value, which provides more room for our attacks to succeed.

TABLE II
SELECTION OF α

Time(s)	Roll Angle Changed by Attack	Value of α
0.5	4	165
0.6	6	165
0.7	7	165
0.8	9	165
0.9	12	165
1	13	155
1.1	15	155
1.2	17	155
1.3	18	145
1.4	19	130
1.5	20	130
1.6	22	130
1.7	23	130
1.9	25	130
2	26	130
2.1	27	130
2.3	28	130
2.4	29	125
2.5	30	120
2.6	31	120
2.7	32	120
3.1	33	105
3.5	34	90
4	35	80
4.1	36	80
5.5	38	55
6.8	40	45
10	41	25
10.2	42	25
10.5	43	25
14.3	44	25
21.2	45	15
28.9	46	10

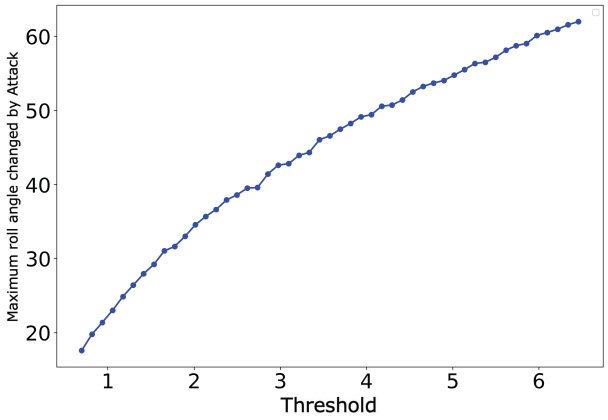

Fig. 6. Threshold verses Maximum Attacking Capability

V. DEFENSE DISCUSSION

In this section, we discuss the potential defense directions against the proposed attack based on the understanding of the identified vulnerability obtained in this paper.

A. Increasing the Weight of Related Sensors in Prediction

As discussed in Section III-C, the involvement of related sensors in the prediction of the target sensor value helps the anomaly detection system to detect our proposed attack when it changes the target sensor too much. Therefore, when increasing the weights of these related sensors in the prediction, attacks performed to the target sensor will only have a lower impact on the prediction. In such a case, the gap between the attacked prediction and the attacked target sensor value will grow fast and make the attack to be detected in the early

stage. However, adjusting the weights of other sensors can also affect the overall accuracy of anomaly detection when there is no attack. Hence, we suggest involving additional sensor inputs or system parameters that are directly related to the target sensor in the prediction, such as roll rate for the roll angle prediction.

B. Using Dynamic Thresholds for Adaptive Detection

As shown in our experimental results, the attacking capability of our attack is directly affected by the threshold. Therefore, instead of adopting the same threshold in the UAV anomaly detection, dynamic thresholds can be leveraged to improve the sensitivity of the detection. For example, if 75% of the threshold $\mathcal{T}$ has been exceeded continuously for certain rounds of detection, the system can temporally set $\mathcal{T} = \mathcal{T} * x$, where $x \in [0.5, 0.8]$. By using dynamic thresholds, the system will perform adaptive detection that enhances the detection sensitivity when the status of the target sensor stays close to abnormal. In addition, adopting dynamic thresholds will also make the adversary difficult to estimate the actual threshold adopted by the detection system and hence limiting its attacking capability.

VI. Conclusion

This paper performs a study on the security property of AI-enabled UAV anomaly detection. Outcomes from this paper contribute to the understanding of vulnerabilities in AI-enabled UAV anomaly detection schemes and will help improve security awareness when designing and deploying such schemes. Specifically, this paper first validates that the "AI prediction + threshold" design mode in typical AI-enabled UAV anomaly detection can be leveraged to launch attacks. Then, we propose an iterative attack and demonstrate the identified vulnerability can be effectively exploited and cause a significant impact on the security and safety of UAV operations. We evaluate our attack on a state-of-the-art AI-enabled UAV anomaly detection scheme, which demonstrates the effectiveness of our attack. Additional evaluation of our attack is also carried out to obtain a better understanding of the security property. Moreover, this paper also discusses the potential defense directions to improve the security and reliability of AI-enabled UAV anomaly detection.

Acknowledge

This work is supported US NSF Awards OAC-2229976 and DGE-1956193 and the UMass Dartmouth Cybersecurity Center Fellowship.

References

[1] M. Bhaskaranand and J. D. Gibson. Low-complexity video encoding for uav reconnaissance and surveillance. In *2011 - MILCOM 2011 Military Communications Conference*, pages 1633–1638, Nov 2011.

[2] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. L. Grixa, F. Ruess, M. Suppa, and D. Burschka. Toward a fully autonomous uav: Research platform for indoor and outdoor urban search and rescue. *IEEE Robotics Automation Magazine*, 19(3):46–56, Sept 2012.

[3] Ashok Raja, Laurent Njilla, and Jiawei Yuan. Blur the eyes of uav: Effective attacks on uav-based infrastructure inspection. In *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 661–665, 2021.

[4] Riham Altawy and Amr M. Youssef. Security, privacy, and safety aspects of civilian drones: A survey. 1(2), nov 2016.

[5] Benkuan Wang, Datong Liu, Yu Peng, and Xiyuan Peng. Multivariate regression-based fault detection and recovery of uav flight data. *IEEE Transactions on Instrumentation and Measurement*, 69(6):3527–3537, 2020.

[6] Jie Zhong, Yujie Zhang, Jianyu Wang, Chong Luo, and Qiang Miao. Unmanned aerial vehicle flight data anomaly detection and recovery prediction based on spatio-temporal correlation. *IEEE Transactions on Reliability*, 71(1):457–468, 2022.

[7] Ahmad Alos and Zouhair Dahrouj. Using mlstm and multioutput convolutional lstm algorithms for detecting anomalous patterns in streamed data of unmanned aerial vehicles. *IEEE Aerospace and Electronic Systems Magazine*, 37(6):6–15, 2022.

[8] Alireza Abbaspour, Kang K. Yen, Shirin Noei, and Arman Sargolzaei. Detection of fault data injection attack on uav using adaptive neural network. *Procedia Computer Science*, 95:193–200, 2016. Complex Adaptive Systems Los Angeles, CA November 2-4, 2016.

[9] Yuqi Chen, Christopher M. Poskitt, and Jun Sun. Learning from mutants: Using code mutation to learn and monitor invariants of a cyber-physical system. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 648–660, 2018.

[10] Benkuan Wang, Zeyang Wang, Liansheng Liu, Datong Liu, and Xiyuan Peng. Data-driven anomaly detection for uav sensor data based on deep learning prediction model. In *2019 Prognostics and System Health Management Conference (PHM-Paris)*, pages 286–290, 2019.

[11] Fazle Karim, Somshubra Majumdar, and Houshang Darabi. Adversarial attacks on time series. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3309–3320, 2021.

[12] Aidong Xu, Xuechun Wang, Yunan Zhang, Tao Wu, and Xingping Xian. Adversarial attacks on deep neural networks for time series prediction. In *2021 10th International Conference on Internet Computing for Science and Engineering*, ICICSE 2021, page 8–14, New York, NY, USA, 2021. Association for Computing Machinery.

[13] Sanmeet Kaur and Maninder Singh. Automatic attack signature generation systems: A review. *IEEE Security Privacy*, 11(6):54–61, 2013.

[14] Man-Ki Yoon, Sibin Mohan, Jaesik Choi, Jung-Eun Kim, and Lui Sha. Securecore: A multicore-based intrusion detection architecture for real-time embedded systems. In *2013 IEEE 19th Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pages 21–32, 2013.

[15] Man-Ki Yoon, Bo Liu, Naira Hovakimyan, and Lui Sha. Virtualdrone: Virtual sensing, actuation, and communication for attack-resilient unmanned aerial systems. In *2017 ACM/IEEE 8th International Conference on Cyber-Physical Systems (ICCPS)*, pages 143–154, 2017.

[16] Dawei Pan, Longqiang Nie, Weixin Kang, and Zhe Song. Uav anomaly detection using active learning and improved s3vm model. In *2020 International Conference on Sensing, Measurement Data Analytics in the era of Artificial Intelligence (ICSMD)*, pages 253–258, 2020.

[17] Huimin Lu, Yujie Li, Shenglin Mu, Dong Wang, Hyoungseop Kim, and Seiichi Serikawa. Motor anomaly detection for unmanned aerial vehicles using reinforcement learning. *IEEE Internet of Things Journal*, 5(4):2315–2322, 2018.

[18] Alireza Abbaspour, Payam Aboutalebi, Kang K. Yen, and Arman Sargolzaei. Neural adaptive observer-based sensor and actuator fault detection in nonlinear systems: Application in uav. *ISA Transactions*, 67:317–329, 2017.

[19] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

[20] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples, 2016.

[21] Gautam Raj Mode and Khaza Anuarul Hoque. Adversarial examples in deep learning for multivariate time series regression. In *2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pages 1–10, 2020.

[22] PX4 Autopilot. https://px4.io/.

[23] QGroundControl. http://qgroundcontrol.com/.