A Novel Technique to Assess UAV Behavior Using PCA-based Anomaly Detection Algorithm

Ahmad M. Alos, Z. Dahrouj and M. Dakkak
Higher Institute for Applied Sciences and Technology/ Informatics Department, Damascus, Syria
Email: Ahmad.alos@hiast.edu.sy, Zouhair.dahrouj@hiast.edu.sy, Mustapha.dakkak@hiast.edu.sy

***Abstract*— This paper focuses on assessing the behavior of the Unmanned Aerial Vehicle (UAV) through its previous flights as a response to an incident. The technique proposed in this paper helps in determining the abnormal flights, and the contribution of the variables in potential faults, in order to ensure the UAV safety. We statistically represent the behavior of the UAV through a flight by using the values of three features: The values of the Pearson Correlation Coefficient, the Y-Intercept, and the slope of the linear regression for each pair of the UAV variables; then, (Principal Components Analysis) PCA-based anomaly detector is used to extract the abnormal flights and the contributed variables in the potential faults. To test the algorithm's efficiency, we used the MKAD synthetic dataset (Multiple Kernel based Anomaly Detection). This dataset is published for public use and includes discrete and continuous variables, which are previously injected by different types of faults. The conducted experiments showed similar results as the results of the well-known MKAD algorithm, where our approach detected 100% of the abnormal flights, with no false alarms. The advantage of our algorithm is that it is an unsupervised algorithm, so it did not require the massive training dataset as the MKAD method did.**

***Index Terms*—UAV, pearson correlation, linear regression, anomaly detection, principal components analysis, potential faults**

I. Introduction

Unmanned Aerial Vehicle (UAV) is an aircraft piloted by remote control or onboard computers. Its applications have been increasing in recent years, including surveillance, reconnaissance, military missions, aerial photography, and disaster monitoring [1]. It is a very complex system operated by Control, Aerodynamics, Communications, and Informatics. The complexity of the UAV system raises the chances of its failure. Anomaly detection algorithms predict system failure by finding patterns in data that do not conform to expected behavior [2]. These algorithms operate in three modes [2][3]: Supervised, semi-supervised, and unsupervised. The supervised anomaly detection algorithms assume the availability of training data with given labels for the normal class as well as for the anomalous class. The semi-supervised anomaly detectors assume the training data has labeled instances for the normal class only. The unsupervised anomaly detectors do not require any labeled training data. They make the implicit assumption that normal instances are far more frequent than anomalies in the test data.

The history of the UAV flight missions is stored in multiple files. Each file contains the values of many input variables (Commands), and sensor readings (UAV State). Suppose that the UAV is in a critical state on a low altitude, and the pilot decides to increase the altitude to save the UAV from crashing. If he sends the proper altitude command to the UAV, but the response is unusually not effective, then the UAV will undoubtedly crash, which raises an issue about the variables pair: (the altitude command and the altitude). The linear relationship between these two variables would help in predicting the potential crash. Our contribution is a new technique to detect irregularities in the UAV behavior by monitoring the features that characterize the changes in the linear relationship between each pair of variables. This technique helps to assess the chances of system failure based on data from previous missions. The proposed algorithm builds the variable pairs; then, for each pair, it constructs a list of linear relationship features (Pearson Correlation Coefficient, Slope, and the Y-intercept). Finally, it detects the behavior irregularities in each flight by using a PCA-based anomaly detector. The rest of the paper is organized as follows. Section II provides a brief background and a review of related works. Problem description and the tools used in it, such as (The Pearson Correlation, the linear regression is explained in Section III. Section IV explains our algorithm and the PCA-based anomaly detection approach. The used dataset, the results of the experiments, and a comparison with the well-known MKAD method (Multiple Kernel based Anomaly Detection) are explained in Section V. Finally; Section VI describes the conclusion and future suggestions.

II. Related Work

Anomaly detection applications include intrusion, fraud detection, medical applications, and robot behavior. These live applications motivated many researchers to work in this area over recent years. System faults can be predicted by discovering anomalies in the system data.

Manuscript received September 22, 2019; revised December 2, 2019; accepted December 2019. corresponding author: Ahmad M. Alos. Email: Ahmad.alos@hiast.edu.sy

There are three types of fault detection algorithms [4]: Model-based, Knowledge-based, and Data-driven-based algorithms. Cork et al. [5] used a model-based fault detection algorithm, where they estimated the state of the UAV using a nonlinear aircraft dynamic model and used the divergence of the estimated state from its actual value to detect system faults. The knowledge-based algorithms depend on predefined rules (if-then) sentences. Bu et al. [6]developed a method for detecting the UAV sensor's faults using a fuzzy logic model. The data-driven algorithms depend on the statistical information to detect outliers and label them as faults. Lin et al. [7] designed an online algorithm to detect UAV sensor faults based on a statistical analysis of sensor readings and navigation data. Sun et al. [8] were interested in data-driven algorithms because model-based and knowledge-based algorithms have high model and rule dependencies; therefore, knowledge-based algorithms are unable to detect unknown or non-modeled faults, while data-driven algorithms showed flexibility due to the model-free analysis. Lin. [7], Khalastchi. [9], and Pokrajac [10] used statistical methods that produced an anomaly score for each given point of time. Each anomaly score depends on a sliding window of monitored readings. Their methods considered the point density using Mahalanobis Distance, or K-Nearest Neighbor (KNN). Principal Components Analysis (PCA) was widely used in many anomaly-detection algorithms. PCA is a fast technique used for dimensionality reduction. It reduces the dimensionality of a multivariate data set into two or three attributes. Paffenroth et al. [11] developed a PCA-based anomaly detector to predict cyber-network attacks. Yong et al. [12] used PCA to detect anomalies in large-sized samples and complicated relationships of UAV sensor data.

III. Methods

A. Definitions

Let a flight mission denoted by $F^n: n \in \{1, 2, \dots, N\}$, where N is the total number of the flights performed by the UAV. The variables of a flight are either commands or measured readings. The commands could be an Altitude command, Rudder Command, Aileron Command, Throttle..., and the measured readings could be pitch, roll, yaw, longitude, latitude, altitude, and so on. The UAV system collects the values of these variables at each step of time (t). Suppose that *ep* denotes the end period of the mission flight, where it is assumed that each row of data will always be fully recorded, with no empty values [7]. Let $I = \{j: 0 \leq j < |I|\}$ denotes the set of all variables, and $|I|$ is the total number of variables. Thus, in a flight F^n, the values of the variable $j \in I$ are stored in the vector

$$V_j^n = \{v_{j,0}, \dots, v_{j,t}, \dots, v_{j,ep}\}. \tag{1}$$

To analyze the UAV behavior in a flight, we evaluate the features that describe the linear relationship between each two variable$s\, j, k \in I$. These features are: (1) The absolute value of the Pearson Correlation Coefficient, (2) The slope, (3) The Y-intercept of the linear regression.

B. Pearson Correlation and Linear Regression

Pearson correlation coefficient is the most widely used statistical analysis to measure the degree of linear relationships between two variables, because it is based on the method of covariance, and it gives information about the magnitude of the association [13]. The linear regression statistical method helps to characterize the relationship between two continuous variables. It finds the best-fitting line through the points of the two variables [14].

In complex systems such as a UAV, the relationships between variables change during parts of the flight. Thus, the correlation coefficient and the linear regression slope and intercept would change too. Also, in many cases, there will be phase lags and latencies between inputs and outputs (The reader is referred to a brief example of this issue in Khalstchi et al. [4]:). However, the proposed approach is not meant for parts of the flight, but it is meant for the whole flight. In each of multiple normal flights (without known faults), the correlation and the linear regression features tend to be stable. The correlated variables stay correlated, and the unrelated variables stay unrelated. In case of a fault, the values of some of the features will change relentlessly (either increasing or decreasing), as we will see later in the experiments section. The algorithm does not look for similarities, but it searches for flights that exhibit severe changes in the values of these features in order to label them as abnormal flights.

Suppose that $X = \{x_1, \dots, x_n\}$, $Y = \{y_1, \dots, y_n\}$ are two datasets. Let $\bar{x}$, $\bar{y}$ be the mean values of X, Y, respectively. Pearson Correlation Coefficient can be calculated using the following formula

$$c_{X,Y} = \left| \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt[2]{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(y_i-\bar{y})^2}} \right|, \tag{2}$$

$c_{X,Y}$ has a real value in the range [0,1]. On the other hand, the formulas for calculating the regression line are

$$\acute{Y} = bX + a, \tag{3}$$

$$b = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt[2]{\sum_{i=1}^{n}(x_i-\bar{x})^2}}, \tag{4}$$

$$a = \bar{y} - \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt[2]{\sum_{i=1}^{n}(x_i-\bar{x})^2}}\,\bar{x}, \tag{5}$$

$\acute{Y}$ is the regression prediction, b is the slope, and a is the y-intercept of the regression line, therefore, using formulas (1), and formula (2), we calculate the Pearson correlation coefficient for a specified flight F^n, and a pair of two variables $j, k \in I$ by

$$C_{j,k}^n = \left| \frac{\sum_{t=1}^{ep}(v_{j,t}-\overline{v_j})(v_{k,t}-\overline{v_k})}{\sqrt[2]{\sum_{t=1}^{ep}(v_{j,t}-\overline{v_j})^2 \sum_{i=1}^{n}(v_{k,t}-\overline{v_k})^2}} \right|. \tag{6}$$

We calculate the slope and the y-intercept of the linear regression by manipulating formulas (1), (4) and (5) as follows

$$b_{j,k}^{n} = \frac{\sum_{t=1}^{ep}(v_{j,t}-\overline{v_j})(v_{k,t}-\overline{v_k})}{\sqrt[2]{\sum_{t=1}^{ep}(v_{j,t}-\overline{v_j})^2}}. \quad (7)$$

$$a_{j,k}^{n} = \overline{v_k} - \frac{\sum_{t=1}^{ep}(v_{j,t}-\overline{v_j})(v_{k,t}-\overline{v_k})}{\sqrt[2]{\sum_{t=1}^{ep}(v_{j,t}-\overline{v_j})^2}}\overline{v_j}. \quad (8)$$

IV. Algorithm

A. Construct Feature Datasets

The goal of the proposed technique is to detect the abnormal flight missions, predict potential failure, and the contribution of the different variables in detecting the faults. The algorithm starts by extracting the values of the features that describe the relationship between every two non-identical variables. Algorithm 1 constructs the features datasets.

```
Algorithm 1 Construct_Feature_Datasets()
for each variable j ∈ I
    for each variable k < j
        R_(j,k) ← ∅
        for each F^n do
            calculate C^n_{j,k}
            calculate b^n_{j,k}, a^n_{j,k}
            add [C^n_{j,k}, b^n_{j,k}, a^n_{j,k}] to R_(j,k)
        add R_(j,k) to R
return R
```

This algorithm iterates through the pairs of variables $\{ (j,k): j,k \in I, k < j, \}$ and the flights $\{F^n: n \leq N\}$. During iterations, it calculates the Pearson Correlation coefficient $C_{j,k}^{n}$ using formula (6), the slope $b_{j,k}^{n}$, and the y-intercept $a_{j,k}^{n}$ of the linear regression using formula (7), and formula (8), then it adds the three values $[C_{j,k}^{n}, b_{j,k}^{n}, a_{j,k}^{n}]$ as a row to the dataset $R_{(j,k)}$. The algorithm composes the set $R = \{R_{(j,k)}\}$, where each dataset $R_{(j,k)}$ is of size N instances. To classify the instances of each $R_{(j,k)}$ we use Principal Component Analysis (PCA-based) anomaly detector. The PCA-based anomaly detector is composed of two steps: Dimensionality Reduction, and Threshold Classification (see Algorithm 2).

B. PCA Dimensionality Reduction

Principal Component Analysis is a well-established technique for dimensionality reduction [15]. It is a projection method that maps a given set of data points onto a set of uncorrelated variables. These variables are called Principal axis or *Principal Components* (PC), which are ordered by the amount of data variance in descending order [16]. The principal components are linear combinations of the original variables [17]. Generally, applying PCA to the normalized data matrix Y in $\mathbb{R}^n$ generates $\{v_i\}_{i=1}^{p}$ which is a set of p principal components. The first principal component v_1 is the vector that corresponds to the direction of maximum variance [18], and it is denoted by

$$v_1 = arg\ max_{\ \|x\|=1}\|Yx\|, \quad (9)$$

$\|x\|$ is the 2-norm of x, and $\|Yx\|$ is proportional to the variance of the data distributed along x. The second principal component is the linear combination of the original variables with the second-largest variance and orthogonal to the first principal component [15]. Proceeding iteratively, if the previous $i-1$ principal axes have been selected, the residual is the difference between the original samples and the samples corresponding to these $i-1$ principal axes. Therefore, the $i-th$ principal component [18] is defined as

$$v_i = arg\ max_{\|x\|=1}\left\|(Y - \textstyle\sum_{j=1}^{i-1} Y v_j v_j^T)x\right\|. \quad (10)$$

The data matrix Y is of order $(m \times n)$, m is its length, and n is the count of the original variables. v_i is the $i-th$ eigenvector of the estimated covariance matrix

$$A = \frac{1}{m}Y^T Y\ . \quad (11)$$

PCA in the feature space calculates the eigenvalues and eigenvectors according to the following equation [12]:

$$\lambda v_i = A v_i, \quad (12)$$

and the projection of the data onto each principal component is given by

$$u_i = v_i^T, i = 1,2,\dots,p, \quad (13)$$

C. Threshold Classification

Hawkins. [19] proved that if the projections of instance x on the principal components are $u1, u2, \dots, u_n$ where the corresponding Eigen-values are $\lambda_1, \lambda_2, \dots, \lambda_n$ then $\sum_{i=1}^{k}\frac{u_i^2}{\lambda_i}, k \leq n$ has a chi-square distribution. Therefore for a given significance level α, the observation x is an anomaly if

$$\sum_{i=1}^{k}\frac{u_i^2}{\lambda_i} > X_k^2(\alpha). \quad (14)$$

If u_{k+1} passes a defined threshold, the first k principal components are regarded as normal, and the rest will be abnormal [2] [16]. We use PCA to reduce the dimensionality of each feature dataset $R_{(j,k)}$ (refer to algorithm 1). The result of dimensionality reduction is a new dataset $D_{(j,k)}$ of N scores and one dimension, where N is the total number of the flights. Choosing one dimension for the results is necessary to apply a fixed threshold, which classifies the new instances in $D_{(j,k)}$ into either normal or abnormal. Considering that the scores for each dataset $D_{(j,k)}$ have different scales, we can apply the min-max normalization to normalize the scores of each dataset as follows

$$(score_{normalized} = \frac{score_i - score_{min}}{score_{max} - score_{max}}). \quad (15)$$

The resulted normalized scores are stored in $ND_{(j,k)}$(see Algorithm 2), which its values range between zero and One. Defining a suitable threshold for each dataset $ND_{(j,k)}$ can be done with the help of domain expertise, visualization, and the previous knowledge of the percent of the abnormal flights. The scores that exceed the defined threshold are considered abnormal flights. Algorithm 2 summarizes the PCA-based anomaly detection for each generated features dataset $R_{(j,k)}$.

Algorithm 2 PCA_Based_Anomaly_Detector($R_{(j,k)}$)

$D_{(j,k)}$ ←Reduce the dimensionality of $R_{(j,k)}$
$ND_{(j,k)}$ ←Normalize($D_{(j,k)}$)
ThresholdClassification($ND_{(j,k)}$)
return (abnormal flights)

To test the approach proposed in this paper, we used the MKAD synthetic dataset [20], which includes various types of seeded faults.

V. Experimental Results

To conduct experiments, we implemented a tool to test our approach using C# language and Accord.net libraries. We used the MKAD synthetic dataset, which is published for public use. This dataset was used to test the MKAD (Multiple Kernel based Anomaly Detection) method robustness [20], which helped us to compare the results of our approach with the results of the MKAD method. The MKAD method is a state of the art algorithm that builds different multiple kernel functions for both discrete and continuous values. It uses the One-class Support Vector Machine (SVM) to separate the abnormal flights from the normal ones. The One-Class SVM model is learned from a large training dataset. The MKAD synthetic dataset includes 150 testing flights. The flights are labeled sequentially from Flight00201, Flight00202 to Flight00350. Each flight includes 1000 rows and consists of 15 variables. The variables do not have labels, so we labeled them sequentially (A1, A2…, A15). The first ten variables are of discrete values, and the last five are of continuous values. The MKAD dataset is injected with four types of faults. (1) Fault type I: include missing expected values in discrete data (see Fig.1). (2) Fault type II: involves extra unexpected values in discrete data (see Fig.2). (3) Fault type III: involves out of order sequences of values in the discrete data (see Fig.3). (4) Fault type IV: includes abnormal patterns in continuous data (see Fig.2). Three examples of each fault were injected into the flights.

By implementing the Algorithm, 105 Features dataset $R_{(j,k)}$ were generated, where every dataset included the values that describe the changes of the linear relationship between every two variables through all the 150 MKAD flights.

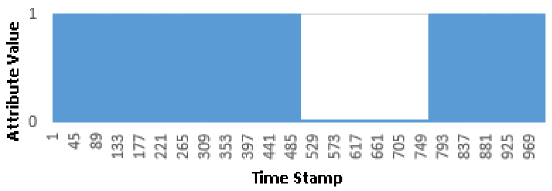

Figure 1. Fault type I (missing expected values).

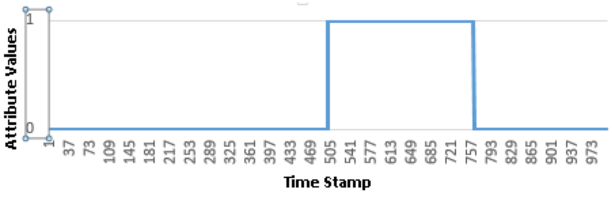

Figure 2. Fault type II (extra unexpected values).

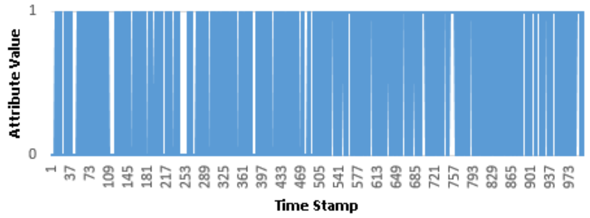

Figure 3. Fault type III: (out of order sequences of values).

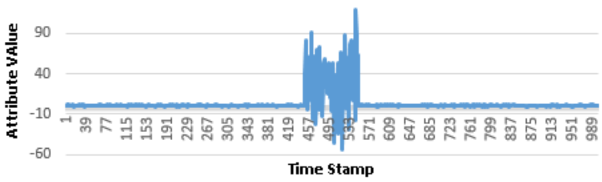

Figure 4. Fault type IV (abnormal patterns).

Applying PCA dimensionality reduction and threshold classification produced 105 score datasets. For each score dataset, the algorithm detected several abnormal flights. By inspecting the results, we could extract the percent of the contributed variable pairs for each abnormal flight. Table 1 shows the result of the experiment. The proposed approach detected 100% of the abnormal flights, with no false alarms. The threshold was defined using the score visualization, and the previous knowledge of the percent of the abnormal flights, which is 4-8% of the total number of flights. This percent value can be extracted from the meta-data of the MKAD synthetic dataset. Our approach had similar results with the MKAD method (The reader is referred to [20] to see the results of the MKAD method). However, our approach did not need the training dataset since it is an unsupervised algorithm. Conversely, the MKAD method requested a large training dataset for building the kernel functions and the SVM model, which is considered time-consuming and memory demanding [21].

TABLE I. Abnormal Flights and Percent of Contributed Variables

Fault Type	Flights	Percent of contributed Variable Pairs
I	Flight00230	24%
	Flight00298	32%
	Flight00314	26%
II	Flight00237	48%
	Flight00260	45%
	Flight00336	48%

Fault Type	Flights	Percent of contributed Variable Pairs
III	Flight00214	9%
	Flight00238	9%
	Flight00325	9%
IV	Flight00233	40%
	Flight00269	40%
	Flight00295	38%

An example of applying the proposed algorithm for the two variables (A2, A7) of the MKAD dataset is shown in Fig.5 and Fig.6.

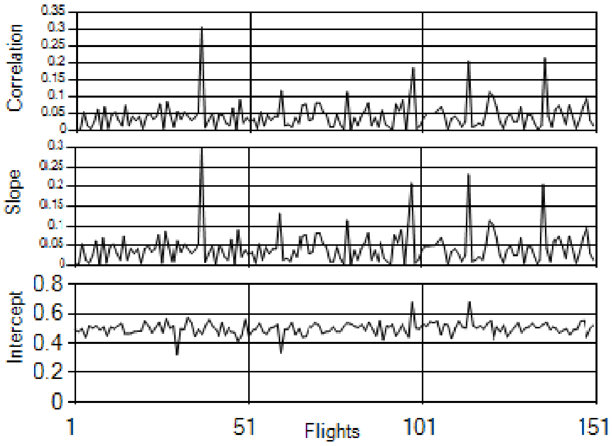

Figure 5. The values of the features of the relationship between the two variables (A2, A7).

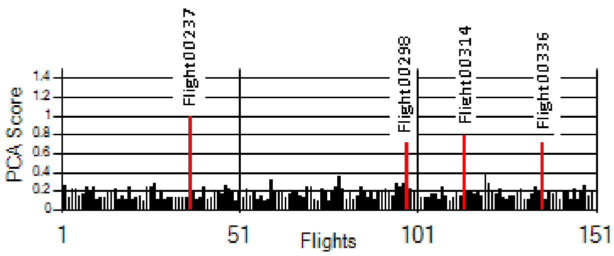

Figure 6. The detected abnormal flights using the features of the relationship between the two variables (A2, A7).

Fig.5 shows the values of the features (Correlation, Slope, and the Y-intercept) of the variables (A2, A7) linear relationship, while Fig.6 shows the scores of the PCA and the detected abnormal flights. It is visually apparent that the used threshold is 0.4. Note that the Four detected abnormal flights in Fig.6 are of either Fault Type I or Fault Type II (see Table I), which means that the pair (A2, A7) contributed to detect the abnormal Flight00237, Flight00298, Flight00314, and Flight00336.

VI. Conclusion

We propose an effective algorithm to assess UAV behavior as a method to ensure UAV safety. The algorithm extracts the values of the features that characterize the relationship between each pair of UAV variables. The used features are the Pearson Correlation Coefficient, the Y-intercept, and the slope of the linear regression. Then, the algorithm uses a PCA-based anomaly detector to extract the abnormal flights and the contributed variables in the potential faults. The proposed algorithm showed similar results as the results of the MKAD method, but on the other hand, it did not need the vast training set as the MKAD method did. Future enhancements could be applied to decrease the number of the used variable pairs in order to make the algorithm faster. In addition, an enhanced method could be applied to find the best threshold to extract abnormal flights.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Ahmad M. Alos designed and implemented the proposed algorithm, conducted the experiments, analyzed the results, and wrote the paper. Z. Dahrouj supervised the entire research, provided valuable advice, and made vital modifications to the paper. M.Dakkak guided the project and participated in analyzing the results. The final version of the paper was revised and approved by all authors.

Acknowledgment

This work is supported by a grant from Higher Institute for Applied Sciences and Technology, Damascus, Syria. The authors appreciate the valuable comments provided by the anonymous referees.

References

[1] I. Colomina and P. Molina, "Unmanned aerial systems for photogrammetry and remote sensing: A review," *ISPRS J. Photogramm. Remote Sens.*, vol. 92, pp. 79–97, 2014.

[2] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, 2009.

[3] I. Renckens, "Artificial intelligence automatic detection of suspicious behaviour," p. 59, 2014.

[4] E. Khalastchi, M. Kalech, G. A. Kaminka, and R. Lin, "Online data-driven anomaly detection in autonomous robots," *Knowl. Inf. Syst.*, vol. 43, no. 3, pp. 657–688, 2015.

[5] L. Cork and R. Walker, "Sensor fault detection for UAVs using a nonlinear dynamic model and the IMM-UKF algorithm," in *Information, Decision and Control*, 2007, pp. 230–235.

[6] J. Bu *et al.*, "Integrated method for the UAV navigation sensor anomaly detection," *IET Radar, Sonar Navig.*, vol. 11, no. 5, pp. 847–853, May 2017.

[7] R. Lin, E. Khalastchi, and G. A. Kaminka, "Detecting anomalies in unmanned vehicles using the Mahalanobis distance," in *Proc. - IEEE Int. Conf. Robot. Autom.*, no. June 2014, pp. 3038–3044, 2010.

[8] R. Sun, Q. Cheng, G. Wang, and W. Y. Ochieng, "A novel online data-driven algorithm for detecting UAV navigation sensor faults," *Sensors (Switzerland)*, vol. 17, no. 10, 2017.

[9] E. Khalastchi, G. a. Kaminka, M. Kalech, and R. Lin, "Online anomaly detection in unmanned vehicles," *10th Int. Conf. Auton. Agents Multiagent Syst. 1*, no. Aamas, pp. 115–122, 2011.

[10] D. Pokrajac, L. J. Latecki, and A. Lazarevic, "Incremental local outlier detection for data streams," *IEEE Symp. Comput. Intell. Data Min.*, no. April, pp. 504–515, 2007.

[11] R. Paffenroth, K. Kay, and L. Servi, "Robust PCA for Anomaly detection in cyber networks," Jan. 2018.

[12] D. Yong, X. Yaqing, Z. Yuanpeng, P. Yu, and L. Datong, "Unmanned aerial vehicle sensor data anomaly detection using

kernel principal component analysis," in *ICEMI 2017 - Proceedings of IEEE 13th International Conference on Electronic Measurement and Instruments*, 2017, vol. 2018-Janua, pp. 241–246.
[13] J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," in *Noise reduction in speech processing*, Springer, Berlin, Heidelberg, 2009, pp. 1–4.
[14] S. Weisberg, *Applied Linear Regression*. Wiley-Interscience, 2005.
[15] A. S. Deepthi, "Anomaly detection using principal component analysis 1 1 2," *Int. J. Comput. Sci. Technol.*, vol. 8491, no. 4, pp. 124–126, 2014.
[16] M. Ding and H. Tian, "PCA-based network traffic anomaly detection," *Tsinghua Sci. Technol.*, vol. 21, no. 5, pp. 500–509, Oct. 2016.
[17] L. Huang, X. Nguyen, M. Garofalakis, M. I. Jordan, A. Joseph, and N. Taft, "In-network PCA and anomaly detection," *Adv. Neural Inf. Process. Syst. 19*, vol. 19, no. July 2014, pp. 617–624, 2007.
[18] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 4, p. 219, 2004.
[19] D. M. Hawkins, "The detection of errors in multivariate data using principal components," *J. Am. Stat. Assoc.*, vol. 69, no. 346, pp. 340–344, June 1974.
[20] S. Das, B. L. Matthews, A. N. Srivastava, and N. C. Oza, "Multiple kernel learning for heterogeneous anomaly detection," in *Proc. of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '10*, 2010, p. 47.
[21] V. M. Janakiraman and D. Nielsen, "Anomaly detection in aviation data using extreme learning machines," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2016-October, pp. 1993–2000, 2016.