

Data-driven unsupervised anomaly detection and recovery of unmanned aerial vehicle flight data based on spatiotemporal correlation

YANG Lei¹, LI ShaoBo^{1,2*}, LI ChuanJiang¹, ZHU CaiChao^{1,3},
ZHANG AnSi^{1,2} & LIANG GuoQiang¹

¹ School of Mechanical Engineering, Guizhou University, Guiyang 550025, China;

² State Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China;

³ State Key Laboratory of Mechanical Transmission, Chongqing University, Chongqing 400044, China

Received November 1, 2022; accepted January 5, 2023; published online April 10, 2023

Anomaly detection is crucial to the flight safety and maintenance of unmanned aerial vehicles (UAVs) and has attracted extensive attention from scholars. Knowledge-based approaches rely on prior knowledge, while model-based approaches are challenging for constructing accurate and complex physical models of unmanned aerial systems (UASs). Although data-driven methods do not require extensive prior knowledge and accurate physical UAS models, they often lack parameter selection and are limited by the cost of labeling anomalous data. Furthermore, flight data with random noise pose a significant challenge for anomaly detection. This work proposes a spatiotemporal correlation based on long short-term memory and autoencoder (STC-LSTM-AE) neural network data-driven method for unsupervised anomaly detection and recovery of UAV flight data. First, UAV flight data are preprocessed by combining the Savitzky-Golay filter data processing technique to mitigate the effect of noise in the original historical flight data on the model. Correlation-based feature subset selection is subsequently performed to reduce the reliance on expert knowledge. Then, the extracted features are used as the input of the designed LSTM-AE model to achieve the anomaly detection and recovery of UAV flight data in an unsupervised manner. Finally, the method's effectiveness is validated on real UAV flight data.

unmanned aerial vehicle (UAV), anomaly detection, spatiotemporal correlation based on long short-term memory and autoencoder (STC-LSTM-AE), Savitzky-Golay, feature selection

Citation: Yang L, Li S B, Li C J, et al. Data-driven unsupervised anomaly detection and recovery of unmanned aerial vehicle flight data based on spatiotemporal correlation. *Sci China Tech Sci*, 2023, 66: 1304–1316, <https://doi.org/10.1007/s11431-022-2312-8>

1 Introduction

In recent years, unmanned aerial vehicle (UAV) technology has shown strong application prospects in many fields, such as agriculture, healthcare, and traffic [1–3]. However, the safety and reliability of UAVs have received unprecedented attention due to the lack of real-time observations and rapid decision-making by pilots and the limitations of UAV size

and weight [4]. Flight data are critical indicators to evaluate the performance of UAV flight status and comprehensively reflect the functional status of key UAV components and the operator's operational information [5]. With the continuous accumulation of historical UAV flight data, processing and analyzing abnormal data information has become the primary means to evaluate the flight quality of UAVs, investigate the causes of UAV accidents, and improve the efficiency of UAVs [6]. Therefore, the study of UAV flight data anomaly detection and recovery methods is an effective

*Corresponding author (email: lishaobo@gzu.edu.cn)

way to improve the reliability and safety of UAVs.

Anomaly detection aims to identify items, events, or observations that do not fit the expected pattern or dataset [7,8]. It is widely used in critical areas, such as urban road traffic [9,10], the industrial Internet of Things [11,12], and finance [13]. According to the definition of time-series anomalies [8], flight data can be divided into sequence and point anomalies, also called flight-level and instantaneous anomalies [14,15], respectively. Flight-level anomalies are sequences or sub-sequences of in-flight data aggregated into unusual shapes [16]. Instantaneous anomalies are outliers in in-flight data that deviate from the expected values [17]. Since there are often contextual dependencies (temporal or spatial) in time-series anomalies between adjacent data points, instantaneous anomalies are also called contextual anomalies [18].

UAV flight data anomaly detection methods are usually classified as knowledge-based, model-based, and data-driven approaches [19]. Qi et al. [20] proposed an adaptive threshold neural network (NN) scheme for rotor UAV sensor fault detection, effectively detecting flight data anomalies, such as acceleration, position, and angular velocity. Bu et al. [21] combined particle filter estimated state residuals with a fuzzy inference system decision system for the detection of anomalies in UAV airborne navigation sensors data information. Abbaspour et al. [22] used extended Kalman filtering to update the weight parameters of an NN to detect anomalies associated with rudders, elevators, aileron deflection, and other UAV-related parameters. López-Estrada et al. [23] designed a linear parameter variation system that accurately represents a nonlinear model that can detect and isolate anomaly parameters, such as UAV position, velocity, and angular velocity. Alos et al. [24] used principal component analysis (PCA), together with Pearson correlation coefficient (PCC), Y-intercept, and slope of linear regression, to detect abnormal data conditions after regressing the UAV variables, such as longitude, latitude, and altitude. However, by summarizing the experience and knowledge of UAV-specific domain experts to detect flight data anomalies, the knowledge-based approach [20,21] heavily relies on prior knowledge, resulting in its poor anomaly detection capability. Moreover, model-based approaches require establishing accurate physical models of UASs, which limits the performance of their methods to some extent [22,23]. Compared with knowledge-based and model-based approaches, the data-driven methods do not require modeling the complex physical characteristics of UASs and extensive expert knowledge, which can maximize flight data information to enable anomaly detection [24]. Similarly, in other fields, including health status prediction of lithium-ion batteries and bearing fault diagnosis, data-driven methods, such as convolutional neural network (CNN) and long short-term memory (LSTM), have also shown their clear ad-

vantages [25,26].

Data-driven approaches can be further classified into temporal and spatiotemporal correlation methods [19]. The temporal correlation methods refer to detecting anomalies in flight data from the correlation of the time dimension. For example, Bronz et al. [27] used a support vector machine (SVM) to classify anomalous and normal flight states. The method's anomaly detection accuracy on parameters, such as gyroscope and accelerometer measurements, was examined by reducing the dimensionality of the feature space. Duan et al. [28] proposed a kernel principal component analysis (KPCA)-based anomaly detection method. This method can quickly detect anomalies in sensing data through a dimensionality reduction of high-dimensional UAV sensor data. Since many UAV flight parameters are correlated in the temporal and spatial dimensions, the temporal correlation approaches ignore the correlation of UAV flight parameters in the spatial dimension, which limits the application of their methods [27,28]. In addition, supervised methods are often impractical due to the difficulty and cost of anomalous labeling data [27]. By contrast, unsupervised methods do not require prior knowledge about actual anomalies to obtain complex mapping relationships between data and are more suitable for the requirements of practical UAV flight data anomaly detection tasks. The spatiotemporal correlation methods consider the correlation of UAV flight parameters in the temporal and spatial dimensions. For example, Wang et al. [29] proposed an LSTM recurrent NN (RNN) method to detect northward velocity and aerodynamic lift speed point anomalies. Wang et al. [30] proposed a multivariate regression based on the LSTM with residual filtering (LSTM-RF) data-driven method. They used an LSTM model to capture the spatiotemporal correlation among UAV flight parameters. Although these methods can effectively detect abnormal flight data, they still rely on expert knowledge to select parameters [29,30]. The PCC can be used to extract correlation parameter features, which is limited when the feature space's variance is significant [24]. Although it is possible to directly select all parameters of UAV flight data for analysis, the presence of many redundant parameters has no positive impact on the model [31]. Furthermore, due to the unpredictable random noise in the flight data, many unobvious anomalies may not be detected.

Motivated by the above challenges, this paper proposes a spatiotemporal correlation based on LSTM and autoencoder (STC-LSTM-AE) NN data-driven approach. Before training and testing the model, the data were preprocessed by the Savitzky-Golay filter to mitigate the effect of noise. It reduces the sensitivity of the model to noise, thus allowing more obscure anomalies to be easily detected. Then, correlation-based feature subset selection was performed to overcome the limitations of the PCC and reduce the dependence on expert knowledge in parameter selection. The

correlated features were inputted into the unsupervised LSTM-AE model to realize the mapping between different features. Anomaly detection was achieved by comparing the threshold value determined by model training with the loss function values tested by the model. In particular, the proposed method not only detects anomalies but also recovers the anomalies present in the detected parameters. It has important practical significance for the safe flight of UAVs. Finally, the effectiveness of the STC-LSTM-AE method in anomaly detection and recovery was verified using real UAV flight data.

2 Preliminaries

2.1 Maximum information coefficient

The maximum information coefficient (MIC) is used to measure the degree of correlation between variables [32]. It can effectively perform feature selection without considering the mean or variance significance. Given a pair of ordered samples of two variables $S=(X_{os}, Y_{os})$, the x and y axes of the two-dimensional coordinate system are divided into a grid of $x \times y$. $S|_G$ represents the probability distribution and is calculated from the frequencies of the samples $S=(X_{os}, Y_{os})$ in the grid distribution. $I(S|_G)$ is the size of the mutual information value defined on $S|_G$. Thus, the maximum mutual information can be defined as

$$I_m(S, x, y) = \max I(S|_G). \quad (1)$$

Based on eq. (1), the feature matrix can be defined as

$$M(S)_{x,y} = \frac{I_m(S, x, y)}{\log \min\{x, y\}}, \quad (2)$$

where $\log \min\{x, y\}$ is the element of the normalized feature matrix. Combining eqs. (1) and (2), the MIC is defined as

$$\text{MIC} = \max_{xy \leq B(n)} \{M(S)_{x,y}\}, \quad (3)$$

where $B(n)$ is the maximum number of cells in the grid G and $|\text{MIC}| \in [0, 1]$. When the MIC value is higher than a given threshold $|\zeta|$, a correlation is considered to exist between the variables X_{os} and Y_{os} .

2.2 LSTM

LSTM is a variant of RNNs. It overcomes the gradient explosion and gradient disappearance problems of RNNs in deep network structures [33]. The LSTM network model structure consists of an input layer, some hidden layers, and an output layer. The hidden layer usually consists of a forget gate, an input gate, an output gate, and a memory cell (as shown in Figure 1).

The forget gate is associated with the process of discarding

or ignoring some information. It accepts the output from the previous unit to determine which data should be discarded and sends retained information to the next gate. The input gate controls how much input information that comes in and out. It determines which data is saved to the internal state. Like the input gate principle, the output gate aims to control the model's output information. It also saves the current data to the next hidden layer cell. The memory cell determines and sends the data to the next level. The calculations associated with each gate and memory cell are shown as follows [34]:

$$f_t = \sigma(U_f h_{t-1} + W_f x_t + b_f), \quad (4)$$

$$i_t = \sigma(U_i h_{t-1} + W_i x_t + b_i), \quad (5)$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot a_t, \quad (6)$$

$$o_t = W_o h_{t-1} + U_o x_t + b_o, \quad (7)$$

$$h_t = o_t \tanh(C_t), \quad (8)$$

$$a_t = \tanh(W_a h_{t-1} + U_a x_t + b_a). \quad (9)$$

In eqs. (4)–(9), U_f and W_f , U_i and W_i , U_o and W_o , and U_a and W_a are the weights of the forget gate, input gate, output gate, and memory cell, respectively. σ is the sigmoid activation function. b_f , b_i , b_o , and b_a are the bias of the forget gate, input gate, output gate, and memory cell, respectively. h_{t-1} is the hidden state value of the last layer. x_t is the input information at moment t . C_t is the external state at moment t .

2.3 AE

An AE is an unsupervised deep-learning NN that contains two parts: an encoder and a decoder [35]. Its model structure is shown in Figure 2. The core of the AE is representation learning of the input data, such as data dimensionality reduction and data compression.

Assume that the input sample is $\chi = R^\ell$. The weight matrix between the encoder layer and the input layer is $W_e \in R^{\ell \times q}$ ($q < \ell$). The biases of the encoder and decoder layers are $b_e \in R^\ell$ and $b_d \in R^q$, respectively. The activation functions of the encoder and decoder are σ_e and σ_d , respectively. The AE first completes the encoding of the input χ using a linear mapping and a nonlinear activation function to intermediate variables $H \in R^q$, i.e.,

$$H = \sigma_e(W_e \chi + b_e). \quad (10)$$

The decoder decodes the intermediate variables to obtain the output $\chi' \in R^q$, i.e.,

$$\chi' = \sigma_d(W_d^T H + b_d), \quad (11)$$

where $W^T \in R^{q \times \ell}$ is the transpose of the weight matrix W ,

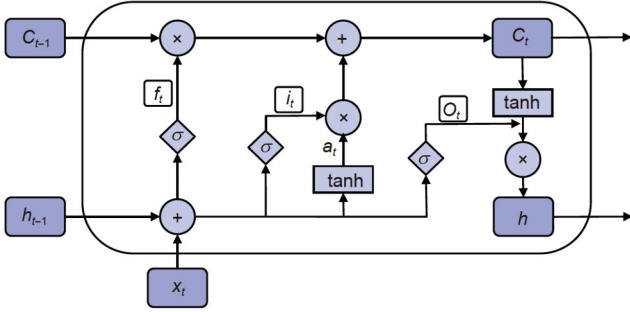


Figure 1 (Color online) Structure of the LSTM model.

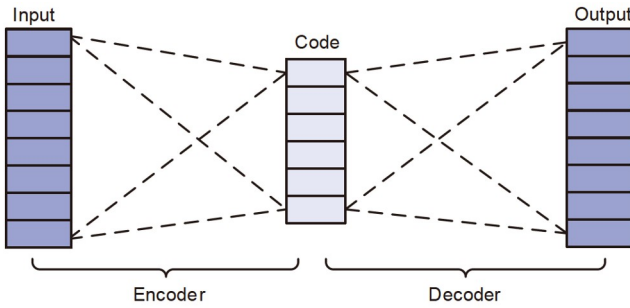


Figure 2 (Color online) Structure of the AE model.

and χ' is the reconstructed data of χ . The training objective of the AE model aims at minimizing the reconstruction error, i.e.,

$$L_{\min}(\chi - \chi') = \operatorname{argmin}_{W, b} L(W, b), \quad (12)$$

where $L(W, b)$ is the loss function of AE, $W = [W_e, W_d]^T$, and $b = [b_e, b_d]^T$.

3 Methodology framework

Figure 3 shows the schematic diagram of the STC-LSTM-AE framework for the unsupervised anomaly detection and recovery of UAV flight data with spatiotemporal correlation. Specifically, the STC-LSTM-AE consists of the following steps: First, data preprocessing is performed on the original n -dimensional UAV flight data features, including data normalization and Savitzky-Golay filtering. Second, the correlation subset features are extracted using the MIC method. Then, the features with correlation are divided into training and test sets. The training set trains and adjusts the STC-LSTM-AE model to obtain a threshold to determine whether the data are abnormal. The test set is used to test and evaluate the model's effectiveness. Finally, anomaly detection is achieved by comparing the model loss function with the threshold value, and recovery is achieved by replacing the anomalous data with reconstructed values.

3.1 Data preparation

The experimental data used in this study are publicly available UAV flight data collected by the University of Minnesota [36]. Given the characteristics of different Thor UAV datasets, the anomalous Thor Flight 69 dataset was used to verify the effectiveness of the proposed method. The dataset is the flight data recorded during the 69th flight of a Thor-type UAV from the University of Minnesota UAV Laboratory and contains 75 features. In this study, part of the flight data from takeoff to landing was used, and there were 19000 sampling points for each feature with a sampling frequency of 50 Hz. The anomalous data points in the abnormal features were marked as 905, i.e., [0:905]. The anomaly rate was 0.0476.

3.2 Data preprocessing

UAV flight data parameters have different units of measurement. The original flight parameters were normalized in the range [0,1] using the min-max normalization method [37] shown in eq. (13) to eliminate the effect of different magnitudes on the model:

$$X = \frac{X - X_{\min}}{X_{\max} - X_{\min}}, \quad (13)$$

where X is the original data and X_{\min} and X_{\max} are the minimum and maximum values of X , respectively. In addition, to reduce the effect of noise on the model, the normalized data were smoothed and denoised using the Savitzky-Golay filter [38]. The principle of the Savitzky-Golay filter is described as follows:

$$y_j^* = \sum_{v=-m}^m \frac{c_v y_{j+v}}{N}, \quad (14)$$

where y is the original signal, y^* is the signal after noise reduction, c_v is the noise reduction coefficient of the v th time, m is the half-width of the smoothing window, N is the number of data points and is equal to the smoothing window size ($2m+1$), and j is the feature index and represents the j th sample in the data sample set. The essence of the Savitzky-Golay filter is to fit the original data using a polynomial for consecutive data points in the data to be processed, based on the least squares method, and selecting the appropriate fitting order. The fitted curve values of the data points were then used as the filtered data values, and the other data to be processed were processed accordingly. The polynomial can be expressed as follows:

$$f_k(v) = p_0 + p_1 v + \dots + p_k v^k = \sum_{\tau=0}^k p_{\tau} v^{\tau}, \quad v \leq 2m, \quad (15)$$

where p_{τ} and k represent the coefficients and degree of the polynomial, respectively, and k and N are the key parameters of the Savitzky-Golay filter. Through extensive exploratory experiments, in this study, the values of k and N were set to 3

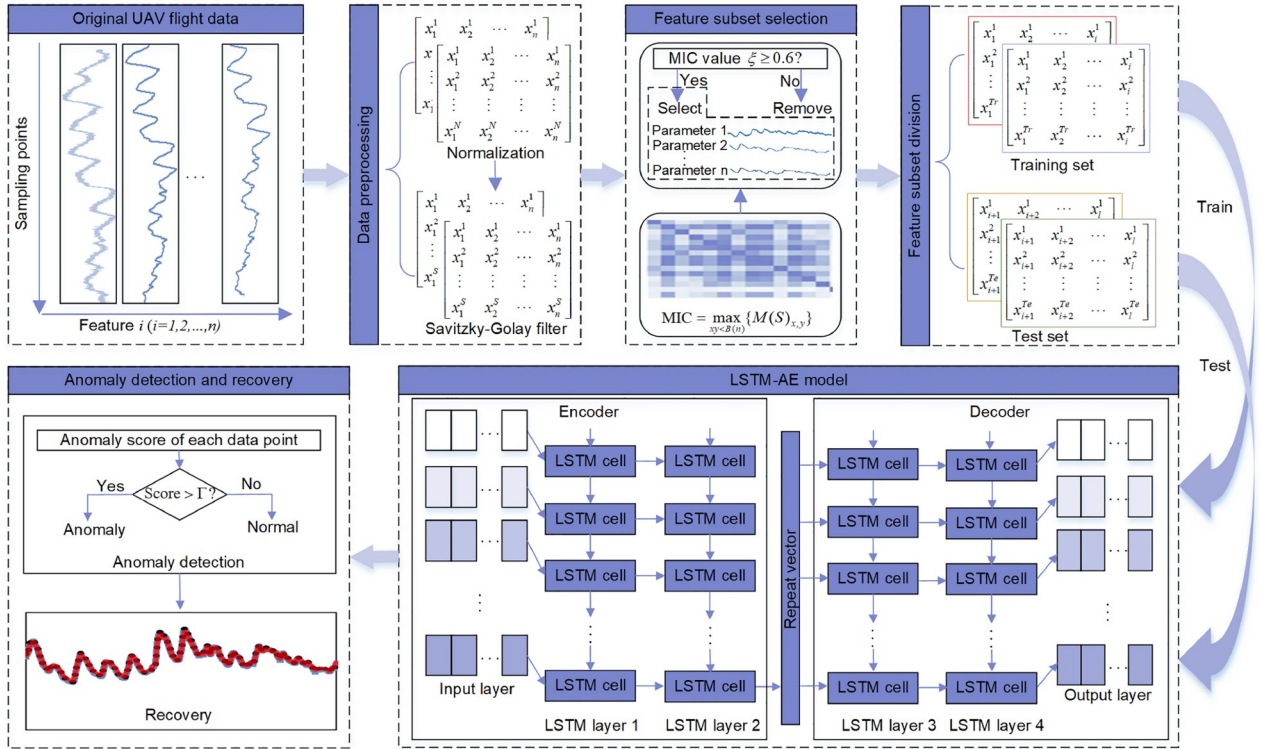


Figure 3 (Color online) Proposed framework of STC-LSTM-AE.

and 115, respectively.

An example is presented in Figure 4, where Figure 4(a) shows the original data of the barometric altitude above ground (h), which is processed by normalization and the Savitzky-Golay filter, as shown in Figure 4(b). The Savitzky-Golay filter can minimize the loss of useful information, thus improving the smoothness of the data and reducing the interference of noise.

3.3 Feature subset selection

The UAS structure is complex and has many flight parameters. Therefore, a feature subset selection technique is needed to avoid unnecessary computational overhead and undesirable effects on model training. It also plays a vital role in reducing the difficulty of troubleshooting and maintenance [19]. PCC is commonly used to measure the correlation between features, but it is limited when the variance obtained from the feature space has relatively high importance. Therefore, the MIC, described in Section 2.1, is used for the correlation feature subset extraction to overcome this problem. It allows the selection of features without considering the significance of the mean or variance. The MIC approach reduces the reliance on expert knowledge and translates UAV domain knowledge into general data analysis problems, which is helpful for further research related to UAV flight data.

Among the 75 features, 55 features of the Thor Flight 69 dataset were used for the feature selection analysis. To avoid the bad effects of anomalous data on the correlation analysis, the UAV flight data with 55 features, including only normal sampling points, were normalized and filtered using eqs. (13) and (14), respectively, to measure the features' MIC values. The visualization of the MIC values is shown in Figure 5. Since the proposed framework focuses on the anomaly detection and recovery of flight state performance data, a subset of flight altitude data (i.e., GPS altitude, NAV altitude, and h) with high correlation was selected as the reference features. In this work, the MIC value was determined as $|\xi| = 0.6$ to automatically exclude unrelated features. It reduced the number of 56 features to 24. Table 1 describes the reference and selected features in detail.

3.4 Detail of the feature subset division

After selecting the subset of correlation features according to the MIC method described in Section 3.3, the subset was then partitioned to contain the training, validation, and test sets. The training set contains only normal data, whereas the test set contains abnormal data. Considering the time-series nature of the Thor Flight 69 flight dataset and data characteristics, the segment [0:2000] (approximately 10% of the subset) of the selected subset was used as the test set, and the segment [2000:4000] (approximately 10% of the subset) of

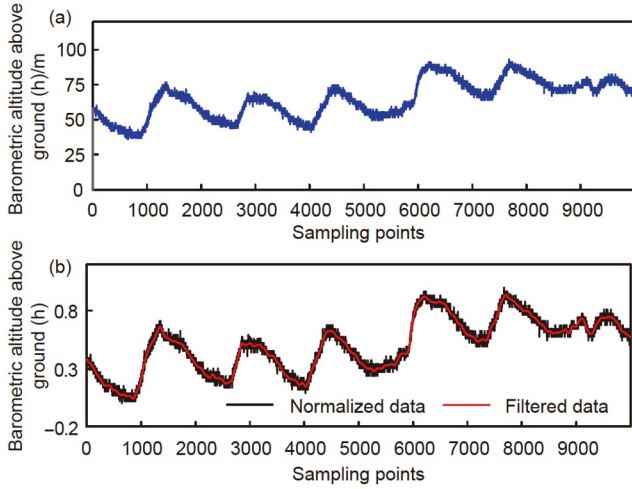


Figure 4 (Color online) (a) Original data; (b) normalized and filtered data.

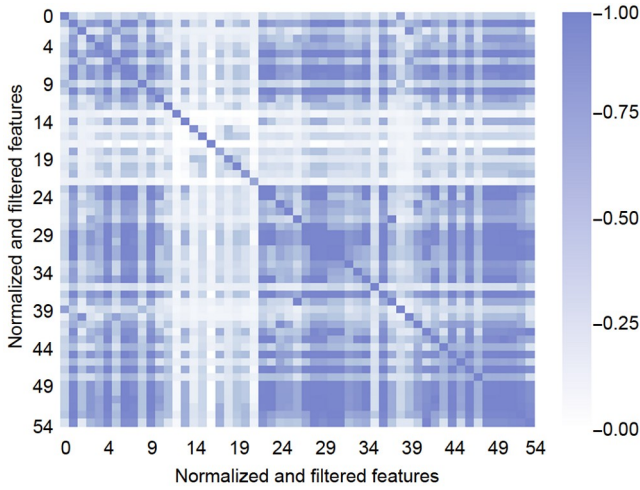


Figure 5 (Color online) Visualization of MIC values.

the selected subset was used as the validation set. The segment [4000:19000] (approximately 80% of the subset) of the selected subset was used as the training set.

3.5 Anomaly detection and recovery based on the STC-LSTM-AE

The core of the STC-LSTM-AE is based on the LSTM-AE to build an unsupervised anomaly detection and recovery model by extracting a subset of features with a spatio-temporal correlation. The LSTM-AE is an NN that fuses LSTM networks with an AE, which combines the feature extraction capability of the AE with the temporal data processing capability of the LSTM to preserve the temporal data information in the extracted features. Figure 6 shows the LSTM-AE model structure diagram proposed in this paper. The model has one input layer, one output layer, four LSTM layers, and one repeat vector. The input and output shapes of

Table 1 Description of the reference and selected features with the MIC

Field name	Description	Units	Class
alt	GPS altitude (WGS84)	m	1
navalt	NAV altitude (WGS84)	m	1
h	Barometric altitude above ground (AGL)	m	1
aoa_cmd	—	—	0
ax	X-axis acceleration	m/s ²	0
ax_bias	Accelerometer bias, X-axis	rad/s	0
ay_bias	Accelerometer bias, Y-axis	rad/s	0
ay_bias_true	—	—	0
az_bias	Accelerometer bias, Z-axis	rad/s	0
h_filt	—	—	0
ias	Indicated airspeed	m/s	0
lat	GPS latitude	deg	0
lon	GPS longitude	deg	0
navlat	NAV latitude	deg	0
navlon	NAV longitude	deg	0
navvn	NAV north velocity	m/s	0
p_bias	Gyro bias, X-axis (roll)	rad/s	0
Ps	Differential pressure	kPa	0
psi	Euler roll angle	rad	0
q_bias	Gyro bias, Y-axis (pitch)	rad/s	0
r_bias	Gyro bias, Z-axis (yaw)	rad/s	0
vd	GPS down velocity	m/s	0
ve	GPS east velocity	m/s	0
vn	GPS north velocity	m/s	0

a) “1” represents the reference feature, and “0” represents the selected feature.

each LSTM layer and repeat vector are shown in Table 2. LSTM layer 2 can also be called the bottleneck layer, i.e., the minimum dimension after the input data are encoded.

(1) Model construction with spatiotemporal correlation

As shown in Figure 6 and Table 2, in the encoding phase, the correlation feature subset of dimension (l, M) is assumed to be used as the input of LSTM layer 1, which is defined as $F_n^{(l)} = (F_n^t, F_n^{t+1}, \dots, F_n^{t+l-1}) \in R^{l \times M}$. l is the sliding time window (with time_step 1). M is the number of features contained in the subset of correlation features determined by the MIC method. F_n^t is the data input to a single LSTM cell at the current moment t . h_{t-1}^E is the hidden representation encoded by the LSTM cell at the moment $t-1$. The hidden representation h_t^E at the moment t was used as one of the inputs for the next LSTM cell. The encoding process of LSTM layer 1 ends at the l -th LSTM cell. The encoding process of LSTM layer 2 is similar to the encoding process of LSTM layer 1. The output data dimension was expanded by a repeat vector after encoding by LSTM layer 2 to derive the output $V_n^{(l)} = (V_n^t, V_n^{t+1}, \dots, V_n^{t+l-1})$ as the input of LSTM layer 3.

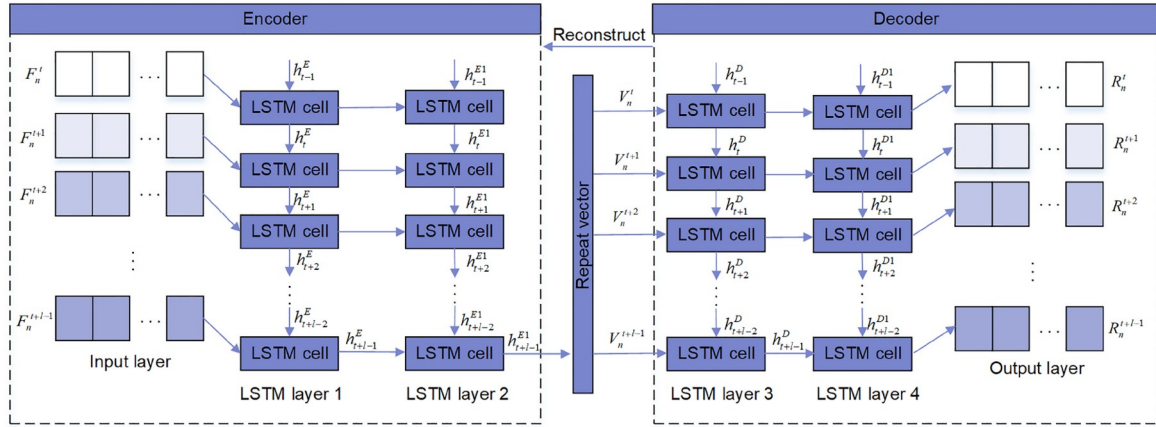


Figure 6 (Color online) Proposed LSTM-AE structure.

The decoding process of LSTM layers 3 and 4 is similar to the encoding process. When the decoding of LSTM layer 4 is completed, the reconstructed data $R_n^{(t)} = (R_n^t, R_n^{t+1}, \dots, R_n^{t+l-1})$ of the input data $F_n^{(t)}$ were obtained.

(2) Model training with the spatiotemporal correlation

LSTM-AE model training is based on the backpropagation algorithm. The algorithm updates the network weights according to the network structure's weight gradient until the model accuracy requirement is satisfied. In this study, the mean squared error (MSE) shown in eq. (16) was used as the loss function of the LSTM-AE model. The goal of the model training is to minimize the loss function. The proposed model's activation function is ReLU, and the Adam optimizer was used to update the network weights.

$$\text{loss} = \frac{1}{l} \sum_{t=1}^{t+l-1} R_n^{(t)} - F_n^{(t)2}. \quad (16)$$

Since a large number of LSTM parameters may cause model training overfitting, this study adopted the discard method, i.e., some cells are randomly dropped in the training phase to avoid affecting forward propagation. By contrast, the output of all LSTM cells was used in the testing phase to affect the model results and weight coefficients. Once the training was completed, a reconstruction error can be obtained as the threshold.

(3) Anomaly detection

In the model testing phase, the reconstruction loss function was calculated in the same way as described in eq. (16). It is

assumed that the threshold value obtained from the model training is Γ . If the reconstruction loss value of the model testing is greater than the threshold Γ , the data point is marked as abnormal. Otherwise, it is normal. Thus, anomaly detection is achieved by the following definition:

$$\text{Anomaly} = \begin{cases} \text{true,} & \text{if } \text{loss} > \Gamma, \\ \text{false,} & \text{else.} \end{cases} \quad (17)$$

The anomaly score can be obtained as follows:

$$\text{Score} = \text{loss}. \quad (18)$$

(4) Recovery

In the encoding phase, assume that the input data at the moment $t+j$ ($1 < j < l-1$) are F_n^{t+j} and the hidden state value at the moment $t+j-1$ is h_{t+j-1}^E of LSTM layer 1. The hidden state values of LSTM layers 1 and 2 at the moment $t+j$ can be calculated according to eqs. (19)–(21).

$$H_E^{t+j} = \sigma_E(W_E F_n^{t+j} + B_E), \quad (19)$$

$$h_{t+j}^E = Z_E(W_E [h_{t+j-1}^E, H_E^{t+j}] + B_E'), \quad (20)$$

$$h_{t+j}^{E1} = Z_{E1}(W_{E1} [h_{t+j-1}^{E1}, h_{t+j}^E] + B_{E1}'). \quad (21)$$

In the decoding phase, assuming that the input data at the moment $t+j$ are V_n^{t+j} and the hidden state value at the moment $t+j-1$ is h_{t+j-1}^D of LSTM layer 3. The hidden state values of LSTM layers 3 and 4 at the moment $t+j$ can be calculated according to eqs. (22)–(24).

$$H_D^{t+j} = \sigma_D(W_D V_n^{t+j} + B_D), \quad (22)$$

$$h_{t+j}^D = Z_D(W_D [h_{t+j-1}^D, H_D^{t+j}] + B_D'), \quad (23)$$

$$h_{t+j}^{D1} = Z_{D1}(W_{D1} [h_{t+j-1}^{D1}, h_{t+j}^D] + B_{D1}'). \quad (24)$$

Then, the reconstructed data at the moment $t+j$ can be calculated by eq. (25):

Table 2 Input and output sizes of each LSTM layer and repeat vector

Layer	Input shape	Output shape
LSTM layer 1	(None, time_step, M)	(None, time_step, 16)
LSTM layer 2	(None, time_step, 16)	(None, 8)
Repeat vector	(None, 8)	(None, time_step, 8)
LSTM layer 3	(None, time_step, 8)	(None, time_step, 16)
LSTM layer 4	(None, time_step, 16)	(None, time_step, M)

$$R_n^{t+j} = \sigma_{D1}(W_{D1} h_{t+j}^{D1} + B_{D1}). \quad (25)$$

In eqs. (19)–(25), $H \in [H_E^{t+j}, H_D^{t+j}]$ is an encoded or decoded variable as the input to the LSTM cells at the moment $t+j$. $\sigma \in [\sigma_E, \sigma_D, \sigma_{D1}]$, $W \in [W_E, W_D, W_{D1}]$, and $B \in [B_E, B_D, B_{D1}]$ are the activation function, weight coefficients, and bias of the LSTM layer network structure, respectively. $Z \in [Z_E, Z_{E1}, Z_D, Z_{D1}]$ denotes the structure of the LSTM cell unit. h_{t+j-1}^{E1} is the hidden state at the moment $t+j-1$ of LSTM layer 2. $W' \in [W'_E, W'_{E1}, W'_D, W'_{D1}]$ and $B' \in [B'_E, B'_{E1}, B'_D, B'_{D1}]$ are the weight coefficients and bias of the LSTM cells, respectively. E or $E1$ denotes encoding, and D or $D1$ denotes decoding.

4 Experimental results and analysis

4.1 Experimental setup

The experiments in this study were realized with the system setup shown in Table 3. In addition, the hyperparameters used in the model training phase were as follows: learning rate = 0.0003, batch size = 32, epochs = 30, and dropout = 0.2.

4.2 Anomaly detection performance metrics and results

In the anomaly detection task, the focus is on the model's ability to detect anomalies. In this study, the true positive rate (TPR), false positive rate (FPR), and accuracy (ACC) were used as the performance evaluation criteria for the experimental model [30]. The calculation formulas for the TPR, FPR, and ACC are given as follows:

$$\text{TPR} = \frac{TP}{TP+FN} \times 100\%, \quad (26)$$

$$\text{FPR} = \frac{FP}{TN+FP} \times 100\%, \quad (27)$$

$$\text{ACC} = \frac{TP+TN}{FP+TP+TN+FN} \times 100\%, \quad (28)$$

where FP denotes the false positive, FN denotes the false negative, TP denotes the true positive, and TN denotes the true negative. TPR indicates the ratio of predicted actual normal samples to the total number of all normal samples. FPR indicates the number of predicted normal but actual

abnormal samples as a percentage of the total number of anomalous samples. ACC indicates a composite indicator of the model's performance. Higher values of ACC and TPR and smaller values of FPR represent the better performance of the model.

Figure 7 shows the original data for the subset of correlation features h , $navalt$, and alt in the Thor Flight 69 dataset at the segment of the sampling points [0:19000]. Figure 8(a) shows the features h , $navalt$, and alt after normalization and Savitzky-Golay filtering. In the anomalous segment [0:905], the feature alt and $navalt$ show significant drift anomalies. This type of anomaly is called a contextual anomaly; i.e., the abnormal features in the subset show a different trend than the other normal ones. Figure 8(b) shows the anomaly score and threshold α after the min-max normalization method, where the blue square bar represents the false positive and the pink square bar represents the false negative. Although there are a small number of missed and false detections during the model training and testing phases, the STC-LSTM-AE is still effective in detecting the vast majority of data points (with 880 anomalies) in the anomaly segment.

To further verify the method's effectiveness in this paper, the AE, SVM [27], KPCA [28], K-nearest neighbor (KNN) [5], LSTM [29], LSTM-RF [30], and WSTC-LSTM-AE (without the Savitzky-Golay filter processing) were compared with the STC-LSTM-AE. The AE is an anomaly detection method based on data reconstruction, as described in Section 2.3. The SVM is a two-class classification model whose basic principle is a linear classifier that finds separated hyperplanes in a feature space with maximized intervals. The KPCA is a nonlinear data processing method. Its core idea is to project data from the original space into a low-dimensional feature space through a nonlinear mapping and perform PCA-based data processing in the low-dimensional

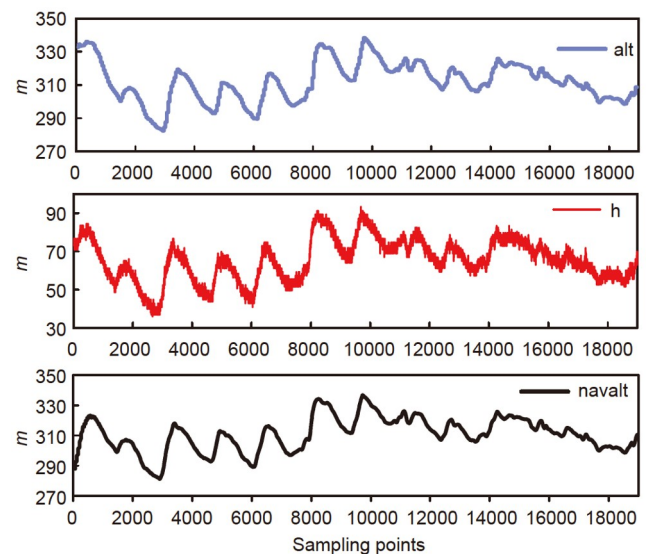


Figure 7 (Color online) Original data of features h , $navalt$, and alt .

Table 3 Implementation environment

Name	Description
Processor	AMD Ryzen 5 3600 6-Core, 3.6 GHz
Operating system	Windows 10 (64-bit)
RAM	16 GB
Packages used	TensorFlow 2.2.0, sklearn 0.21.3

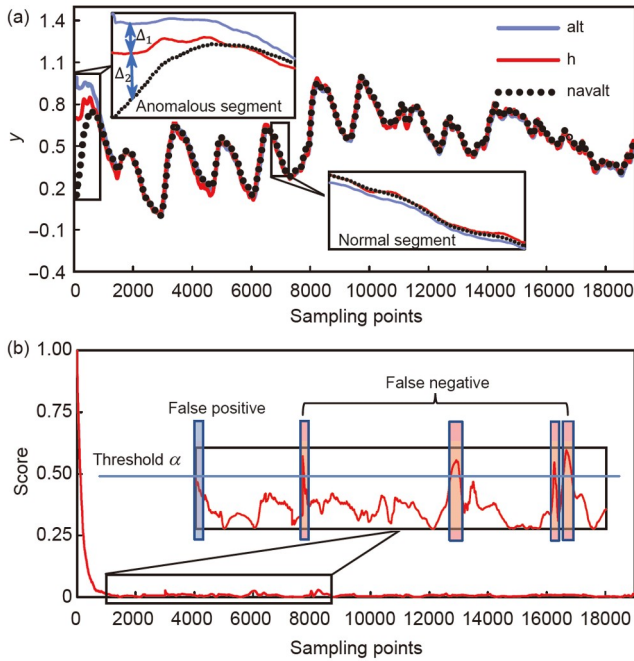


Figure 8 (Color online) Anomaly detection results of the STC-LSTM-AE. (a) Normalized and filtered features alt, h, and navalt; (b) anomaly score.

feature space. The KNN is a temporal correlation anomaly detection method. It is performed by calculating the distance between the predicted data points and the nearest K sample points and discriminating the abnormal samples according to the threshold value. The LSTM and LSTM-RF, as described in Sections 2.2 and 1, are spatiotemporal correlation anomaly detection methods. The approaches determine if the data are abnormal by comparing the difference values between the predicted and actual data.

Figure 9 shows the test results of the anomaly detection methods mentioned above. The TPR values of the STC-LSTM-AE and KPCA are all 1, indicating that all normal data can be detected by these methods. Although the KNN can detect all the outliers, i.e., FPR=0, the method detects most normal data as outliers, which is not allowed in practical applications. The TPR and ACC values of the KNN are 23.20% and 57.95%, respectively. The main reason is that the KNN is susceptible to anomalous data among k proxies

due to its poor fault tolerance for training data. The TPR, FPR, and ACC values of the AE are 97.17%, 4.97%, and 96.20%, respectively, which indicates that it has a good detection performance. The SVM does not perform well in all the performance metrics, and its TPR, FPR, and ACC values are 88.13%, 23.63%, and 82.80%, respectively. The TPR, FPR, and ACC values of the LSTM are 90.87%, 41.44%, and 76.25%. The relatively poor results of the LSTM are attributed to the random noise and dramatically changing points, which make it unable to effectively detect anomalies. Accordingly, this problem was overcome and enhanced by the proposed method using the AE and Savitzky-Golay filter. Similarly, by filtering the residual values between the predicted and actual values, the anomaly detection performance of the LSTM-RF is significantly improved. TPR, FPR, and ACC values are 93.06%, 3.20%, and 94.75%, respectively. The TPR, FPR, and ACC values of the WSTC-LSTM-AE are 93.33%, 4.42%, and 94.10%, respectively, indicating that the anomaly detection performance of the method is reduced without the Savitzky-Golay filter processing.

Table 4 shows the performance metric difference values of the STC-LSTM-AE compared with the AE, KPCA, SVM, KNN, LSTM, LSTM-RF, and WSTC-LSTM-AE. “+” denotes an increase, and “−” represents a decrease. Each performance metric value of the STC-LSTM-AE is shown in parentheses. The TPR value of the STC-LSTM-AE is increased by 2.83%, 11.87%, 76.80%, 9.13%, 6.94%, and 6.67% compared with the AE, SVM, KNN, LSTM, LSTM-RF, and WSTC-LSTM-AE, respectively. Although the FPR value of the STC-LSTM-AE is 2.76%, which is higher than that of the KNN, it is 2.21%, 4.97%, 20.86%, 38.68%, 0.44%, and 1.66% lower compared with that of the AE, KPCA, SVM, LSTM, LSTM-RF, and WSTC-LSTM-AE, respectively. The ACC value of the STC-LSTM-AE is 98.75%, which increases by 2.55%, 2.85%, 15.95%, 40.80%, 22.50%, 4.00%, and 4.65%, respectively, compared with the AE, KPCA, SVM, KNN, LSTM, LSTM-RF, and WSTC-LSTM-AE. The results show that the STC-LSTM-AE can effectively mine the dependency relationship between spatiotemporal correlation features and detect more obscure anomalies.

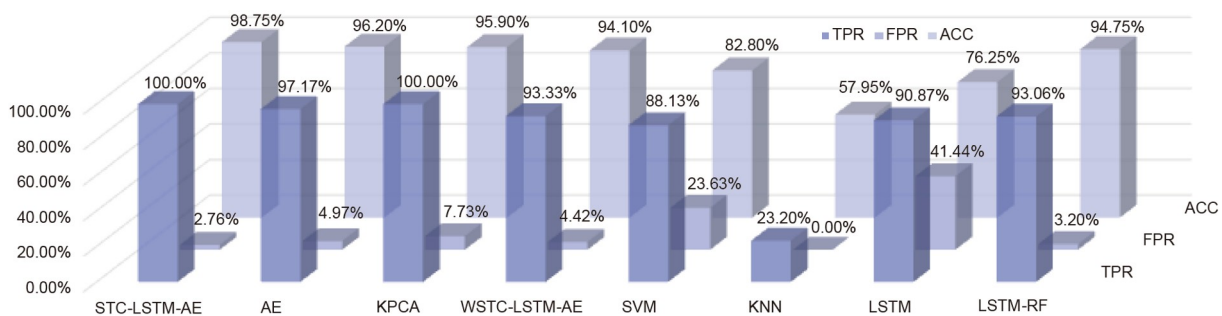


Figure 9 (Color online) Anomaly detection results of the STC-LSTM-AE, AE, KPCA, WSTC-LSTM-AE, SVM, KNN, LSTM, and LSTM-RF.

Table 4 Performance metric difference values of the AE, KPCA, SVM, KNN, LSTM, LSTM-RF, and WSTC-LSTM-AE compared with the STC-LSTM-AEa)

Model	Δ_{FPR} (STC-LSTM-AE)	Δ_{TPR} (STC-LSTM-AE)	Δ_{ACC} (STC-LSTM-AE)
AE	-2.83% (100%)	+2.21% (2.76%)	-2.55% (98.75%)
KPCA	0 (100%)	+4.97% (2.76%)	-2.85% (98.75%)
SVM	-11.87% (100%)	+20.86% (2.76%)	-15.95% (98.75%)
KNN	-76.80% (100%)	-2.76% (2.76%)	-40.80% (98.75%)
LSTM	-9.13% (100%)	+38.68% (2.76%)	-22.50% (98.75%)
LSTM-RF	-6.94% (100%)	+0.44% (2.76%)	-4.00% (98.75%)
WSTC-LSTM-AE	-6.67% (100%)	+1.66% (2.76%)	-4.65% (98.75%)

a) The bold values outside of parentheses in Table 4 represent other methods that performed better than or the same as the proposed method. The bold values in parentheses indicate the best performance of the proposed method in some indexes.

4.3 Recovery performance metrics and results

The purpose of UAV flight data anomaly detection is to detect and minimize the impact of anomalies on time. In addition, it is necessary to recover the abnormal values and convert them into normal data to help operators better master the UAV flight status performance and take relevant emergency measures, which is vital to ensure the safety of UAV flights. This study used the MSE and mean absolute error (MAE) [39] to reflect the recovery accuracy of the actual data. They are calculated as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2, \quad (29)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{y}_i - y_i|, \quad (30)$$

where y_i is the original data and \hat{y}_i is the recovered value at time i . n is the length of selected features.

Usually, smaller MSE and MAE values indicate better recovery performance of the model. However, the experimental object of this study is UAV flight data contextual anomalies with spatiotemporal correlation. For the features alt and navalt, larger MSE and MAE values for the anomalous segment and smaller MSE and MAE values for the normal segment indicate better model performance. For the normal feature h , smaller MSE and MAE values indicate better model performance. To further illustrate the reason, in Figure 8(a), the distances Δ_1 and Δ_2 between the anomalous segment features alt and h and between navalt and h are significantly larger than those of the normal segment. This finding indicates that in the anomaly segment, the differences between the recovered feature alt and navalt and the normal feature h need to be as small as possible. That is, the MSE and MAE values need to be as large as possible, implying a better model effect. Meanwhile, in the normal segment, the differences among the recovered features alt, navalt, and h need to be as small as possible with themselves; i.e., the MSE and MAE values need to be as small as possible, implying a better model effect.

For the above comparison methods, the KPCA and SVM focus on anomaly detection. Although the LSTM and KNN can achieve data recovery, their model performances are relatively poor, and there are many missed or false detections. Therefore, the STC-LSTM-AE, LSTM-RF, and AE were selected to test the recovery effect. Figure 10 shows the recovery results of the STC-LSTM-AE, LSTM-RF, and AE. The STC-LSTM-AE, LSTM-RF, and AE can efficiently recover original data on the training set. However, on the test set, the distances between the recovered alt and navalt of the STC-LSTM-AE and the original alt and navalt data are the largest compared with those of the LSTM-RF and AE, as shown in Figure 10(a) and (b). For normal h , compared with the LSTM-RF and AE, the STC-LSTM-AE minimizes the distance between the recovered h and original h data, as shown in the left panel of Figure 10(c).

Table 5 and Figure 11 show the MSE and MAE values of the STC-LSTM-AE, LSTM-RF, and AE and the increase or decrease in the MSE and MAE values of the STC-LSTM-AE compared with those of the LSTM-RF and AE. The recovery performance metrics MSE and MAE values of the STC-LSTM-AE on the training set were 0.000052 and 0.005547, 0.000031 and 0.004502, and 0.000083 and 0.006912 and were reduced by 86.73% and 64.86%, 93.16% and 74.01%, and 76.55% and 55.79%, respectively, compared with those of the AE. They were also decreased by 91.65% and 70.19%, 94.00% and 75.37%, and 77.63% and 57.06%, respectively, compared with those of LSTM-RF. On the test set, the recovery performance metrics MSE and MAE values of the STC-LSTM-AE for the abnormal features alt and navalt were 0.004775 and 0.050933, and 0.007262 and 0.045475, which were increased by 92.85% and 48.86%, and 21.26% and 29.77%, respectively, compared with those of the AE. They were also increased by 54.38% and 26.83%, and 18.04% and 21.48%, respectively, compared with those of the LSTM-RF. By contrast, for the normal feature h , the recovery performance metrics MSE and MAE of the STC-LSTM-AE were 0.000276 and 0.013704, which were 68.58% and 44.30%, and 84.54% and 43.81% lower than

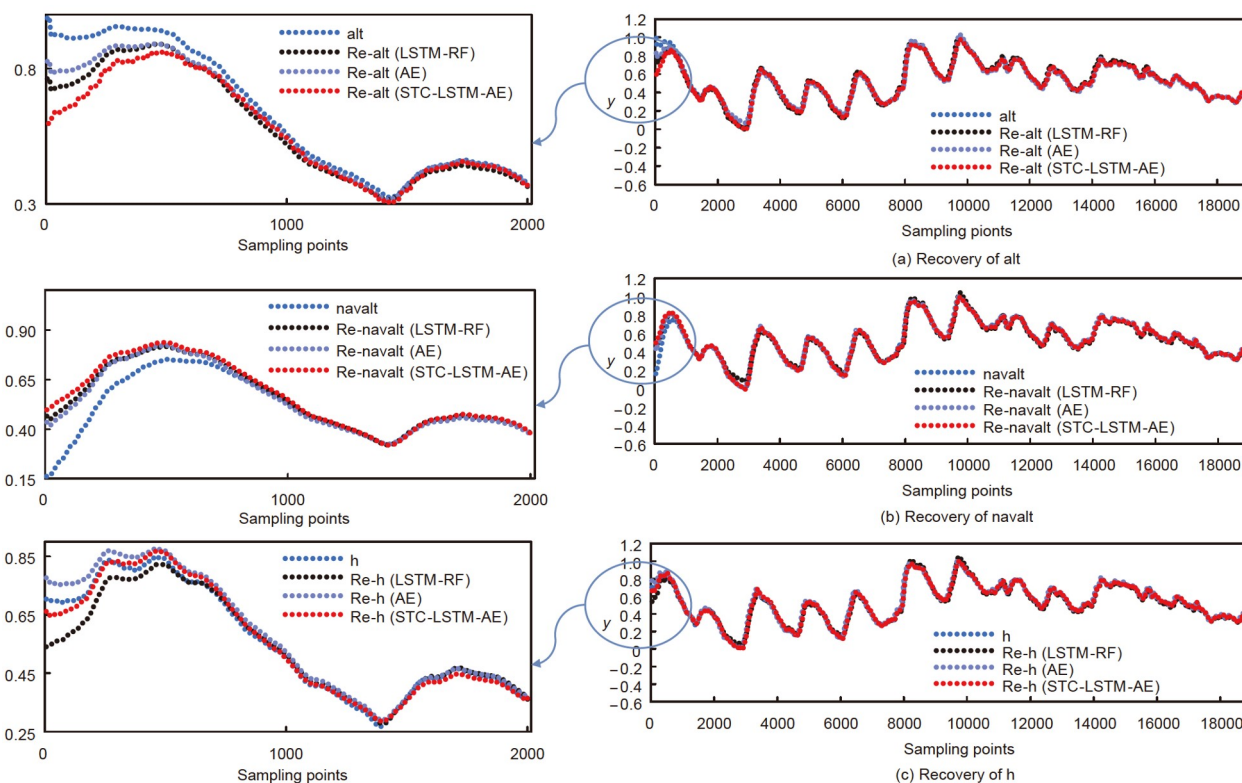


Figure 10 (Color online) Recovery results of the STC-LSTM-AE, LSTM-RF, and AE.

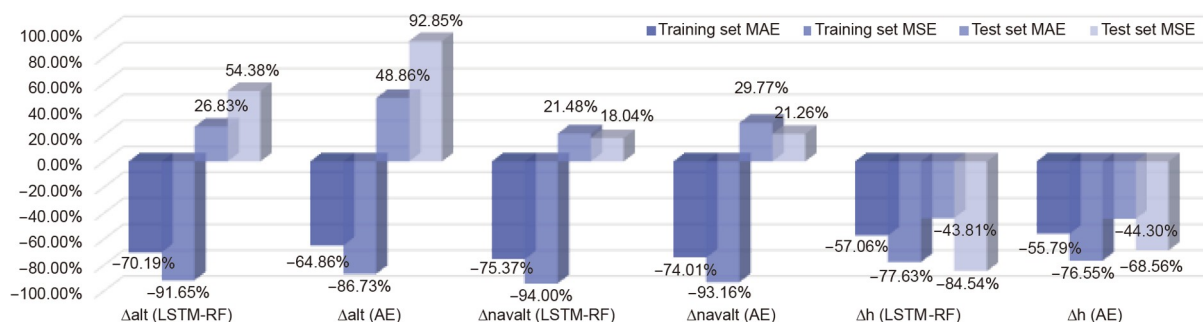


Figure 11 (Color online) MSE and MAE values of the STC-LSTM-AE increase or decrease compared with those of the AE and LSTM-RF.

Table 5 MSE and MAE values of the STC-LSTM-AE, LSTM-RF, and AE

Model	Feature	Training set		Test set	
		MAE	MSE	MAE	MSE
STC-LSTM-AE	alt	0.005547	0.000052	0.050933	0.004775
	navalt	0.004502	0.000031	0.045475	0.007262
	h	0.006912	0.000083	0.013704	0.000276
LSTM-RF	alt	0.018605	0.000623	0.040158	0.003093
	navalt	0.018277	0.000517	0.037435	0.006152
	h	0.016097	0.000371	0.024387	0.001785
AE	alt	0.015785	0.000392	0.034215	0.002476
	navalt	0.017322	0.000453	0.035043	0.005989
	h	0.015635	0.000354	0.024605	0.000878

those of the AE and LSTM-RF, respectively. The results show that the STC-LSTM-AE can capture the spatio-temporal correlation among the features and the slight variation trend between anomalous features to achieve highly accurate anomaly recovery.

5 Conclusions

This study designed an unsupervised anomaly detection and recovery framework STC-LSTM-AE for UAV flight data. First, the data were preprocessed using the Savitzky-Golay filter to mitigate the noise problem in the flight data. Second, the MIC method was used to achieve automatic flight feature extraction as the model input, effectively reducing parameter selection dependence on expert knowledge. Finally, the unsupervised LSTM-AE model was used to achieve anomaly detection and recovery of UAV flight data.

This study conducted experiments using the anomalous Thor Flight 69 real public dataset. The anomaly detection test results for STC-LSTM-AE were TPR=1, FPR=2.76%, and ACC=98.75%. Compared with the AE, SVM, KNN, LSTM, LSTM-RF, and WSTC-LSTM-AE, the TPR of the STC-LSTM-AE was increased by 2.83%, 11.87%, 76.80%, 9.13%, 6.94%, and 6.67%, the FPR of the STC-LSTM-AE was reduced by 2.21%, 4.97%, 20.86%, 38.68%, 0.44%, and 1.66%, and the ACC of the STC-LSTM-AE was 2.55%, 2.85%, 15.95%, 40.80%, 22.50%, 4.00%, and 4.65% higher, respectively. In terms of recovery, the performance metrics of the STC-LSTM-AE outperformed those of the AE and LSTM-RF on the training and test sets. The experimental results showed that the STC-LSTM-AE could effectively mine the interdependence between the data change trends and detect more obscure anomalies to achieve high-precision anomaly detection and recovery.

In the future, on the one hand, we will further optimize the model performance, such as reducing the FPR and deploying it to the platform for real-time anomaly detection. On the other hand, since features with correlation directly reflect the performance of specific parts of UAVs, we will also consider the research on failure mode matching based on abnormal correlation features.

This work was supported by the National Key Research and Development Program of China (Grant No. 2020YFB1713300), the Guizhou Provincial Colleges and Universities Talent Training Base Project (Grant No. [2020] 009), the Guizhou Province Science and Technology Plan Project (Grant Nos. [2015]4011, [2017]5788), the Guizhou Provincial Department of Education Youth Science and Technology Talent Growth Project (Grant No. [2022]142), the Scientific Research Project for Introducing Talents from Guizhou University (Grant No. (2021)74), and the Guizhou Province Higher Education Integrated Research Platform Project (Grant No. [2020] 005).

- Ullah S, Kim K I, Kim K H, et al. UAV-enabled healthcare architecture: Issues and challenges. *Future Gen Comput Syst*, 2019, 97: 425–432
- Aggarwal S, Kumar N, Alhussein M, et al. Blockchain-based UAV path planning for healthcare 4.0: Current challenges and the way ahead. *IEEE Network*, 2021, 35: 20–29
- Butilă E V, Boboc R G. Urban traffic monitoring and analysis using unmanned aerial vehicles (UAVs): A systematic literature review. *Remote Sens*, 2022, 14: 620
- Li C, Li S, Zhang A, et al. A Siamese hybrid neural network framework for few-shot fault diagnosis of fixed-wing unmanned aerial vehicles. *J Comput Des Eng*, 2022, 9: 1511–1524
- Liu Y, Ding W. A KNNs based anomaly detection method applied for UAV flight data stream. In: 2015 Prognostics and System Health Management Conference (PHM). Beijing, 2015. 1–8
- Zhao W, Li L, Alam S, et al. An incremental clustering method for anomaly detection in flight data. *Trans Res Part C-Emerging Tech*, 2021, 132: 103406
- Hawkins D M. Identification of Outliers. London: Chapman and Hall, 1980
- Keogh E, Lin J, Lee S H, et al. Finding the most unusual time series subsequence: Algorithms and applications. *Knowl Inf Syst*, 2007, 11: 1–27
- Djenouri Y, Belhadi A, Lin J C W, et al. A survey on urban traffic anomalies detection algorithms. *IEEE Access*, 2019, 7: 12192–12205
- Zhao C, Chang X, Xie T, et al. Unsupervised anomaly detection based method of risk evaluation for road traffic accident. *Appl Intell*, 2023, 53: 369–384
- Summerville D H, Zach K M, Chen Y. Ultra-lightweight deep packet anomaly detection for Internet of Things devices. In: 2015 IEEE 34th International Performance Computing and Communications Conference (IPCCC). Nanjing, 2015. 1–8
- Wang X, Garg S, Lin H, et al. Toward accurate anomaly detection in industrial internet of things using hierarchical federated learning. *IEEE Internet Things J*, 2021, 9: 7110–7119
- Anandakrishnan A, Kumar S, Statnikov A, et al. Anomaly detection in finance: Editors' introduction. In: KDD 2017 Workshop on Anomaly Detection in Finance. Halifax, 2018. 1–7
- Li L, Hansman R J, Palacios R, et al. Anomaly detection via a Gaussian Mixture Model for flight operation and safety monitoring. *Trans Res Part C-Emerging Tech*, 2016, 64: 45–57
- Schumann J, Rozier K Y, Reinbacher T, et al. Towards real-time, on-board, hardware-supported sensor and software health management for unmanned aerial systems. *Intl J Prog Health Man*, 2015, 6: 1–27
- Gupta M, Gao J, Aggarwal C C, et al. Outlier detection for temporal data: A survey. *IEEE Trans Knowl Data Eng*, 2013, 26: 2250–2267
- Puranik T G, Mavris D N. Identifying instantaneous anomalies in general aviation operations. In: 17th AIAA Aviation Technology, Integration, and Operations Conference. Denver, 2017. 3779–3794
- Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey. *ACM Comput Surv*, 2009, 41: 1–58
- Zhong J, Zhang Y, Wang J, et al. Unmanned aerial vehicle flight data anomaly detection and recovery prediction based on spatio-temporal correlation. *IEEE Trans Rel*, 2021, 71: 457–468
- Qi J, Zhao X, Jiang Z, et al. An adaptive threshold neural-network scheme for rotorcraft UAV sensor failure diagnosis. In: International Symposium on Neural Networks. Berlin, 2007. 589–596
- Bu J, Sun R, Bai H, et al. Integrated method for the UAV navigation sensor anomaly detection. *IET Radar Sonar Navigat*, 2017, 11: 847–853
- Abbaspour A, Aboutalebi P, Yen K K, et al. Neural adaptive observer-based sensor and actuator fault detection in nonlinear systems: Application in UAV. *ISA Trans*, 2017, 67: 317–329
- López-Estrada F R, Ponsart J C, Theilliard D, et al. LPV model-based tracking control and robust sensor fault diagnosis for a quadrotor UAV. *J Intell Robot Syst*, 2016, 84: 163–177
- Alos A M, Dahrouj Z, Dakkak M. A novel technique to assess UAV

- behavior using PCA-based anomaly detection algorithm. *Int J Mech Eng Robot Res*, 2020, 9: 721–726
- 25 Ma G, Xu S, Jiang B, et al. Real-time personalized health status prediction of lithium-ion batteries using deep transfer learning. *Energy Environ Sci*, 2022, 15: 4083–4094
 - 26 Yuan Y, Ma G, Cheng C, et al. A general end-to-end diagnosis framework for manufacturing systems. *Natl Sci Rev*, 2020, 7: 418–429
 - 27 Bronz M, Baskaya E, Delahaye D, et al. Real-time fault detection on small fixed-wing UAVs using machine learning. In: 2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC). San Antonio, 2020. 1–10
 - 28 Duan Y, Xu Y Q, Zhao Y P, et al. Unmanned aerial vehicle sensor data anomaly detection using kernel principle component analysis. In: 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI). Yangzhou, 2017. 241–246
 - 29 Wang B, Wang Z, Liu L, et al. Data-driven anomaly detection for UAV sensor data based on deep learning prediction model. In: 2019 Prognostics and System Health Management Conference (PHM-Paris). Paris, 2019. 286–290
 - 30 Wang B, Liu D, Peng Y, et al. Multivariate regression-based fault detection and recovery of UAV flight data. *IEEE Trans Instrum Meas*, 2019, 69: 3527–3537
 - 31 Lee H, Li G, Rai A, et al. Real-time anomaly detection framework using a support vector regression for the safety monitoring of commercial aircraft. *Adv Eng Inf*, 2020, 44: 101071
 - 32 Sun G, Li J, Dai J, et al. Feature selection for IoT based on maximal information coefficient. *Future Gener Comput Syst*, 2018, 89: 606–616
 - 33 Xia X Z, Cheng L. Adaptive Takagi-Sugeno fuzzy model and model predictive control of pneumatic artificial muscles. *Sci China Tech Sci*, 2021, 64: 2272–2280
 - 34 Van Houdt G, Mosquera C, Nápoles G. A review on the long short-term memory model. *Artif Intell Rev*, 2020, 53: 5929–5955
 - 35 Su L, Zhang S Y, Ji Y, et al. A novel approach for flip chip inspection based on improved SDELM and vibration signals. *Sci China Tech Sci*, 2022, 65: 1087–1097
 - 36 Taylor B. Thor Flight 69. Retrieved from the University of Minnesota Digital Conservancy, 2012. <https://hdl.handle.net/11299/174347>
 - 37 Han J, Kamber M. *Data Mining: Concepts and Techniques*. 2nd ed. Waltham: Morgan Kaufmann, 2006
 - 38 Liu Y, Dang B, Li Y, et al. Applications of Savitzky-Golay filter for seismic random noise reduction. *Acta Geophys*, 2016, 64: 101–124
 - 39 Sun Q, Jiang B, Zhu H, et al. Hard thresholding regression. *Scand J Statist*, 2019, 46: 314–328