Data-driven Anomaly Detection for Quadcopter UAV Indoor Flight Platform

Gengyu Li, Chun Fui Liew, Naoya Takeishi, and Takehisa Yairi

Department of Advanced Interdisciplinary Studies, The University of Tokyo, Tokyo, Japan
$\{$*li-gengyu, liew, ntake, yairi*$\}$ *@g.ecc.u-tokyo.ac.jp*

ABSTRACT

Ensuring the safe operation of unmanned aerial vehicles (UAVs) requires timely and accurate detection of anomalies that may indicate system faults or external disturbances. In this study, we experimentally investigate a data-driven approach for unsupervised anomaly detection in UAVs, leveraging a newly developed multimodal dataset that includes synchronized telemetry, sensor measurements, motion capture data, and pilot inputs. Our method learns representations of normal UAV behavior from healthy flight records and is applied to fault-injection scenarios to identify potential anomalies. Preliminary results on experimental data suggest that the approach can capture subtle deviations from expected behavior across multiple data modalities, including flight dynamics and environmental feedback. This work lays the foundation for data-driven UAV health monitoring through unsupervised learning. It complements our publicly released dataset and analysis tools and aims to facilitate broader research on autonomous anomaly detection, early fault diagnostics, and the development of resilient UAV systems in safety-critical applications.

1. INTRODUCTION

Unmanned aerial vehicles (UAVs) have become increasingly prevalent across a wide range of civilian, industrial, and defense applications, including inspection, surveillance, mapping, and delivery (Liew & Yairi, 2020). Recent advancements in small-scale UAVs have also expanded their potential into companion or social interaction roles, where their unique ability to fly offers unprecedented possibilities for human–drone interaction. However, these advantages come with challenges such as safety concerns, operational noise, and so on, making them susceptible to a wide variety of faults and anomalies.

Prognostics and Health Management (PHM) provides a systematic framework to address these challenges by integrating sensing, data analysis, and predictive modeling to enable early fault detection, diagnosis, and remaining useful life estimation (Zonta et al., 2020). In the context of UAVs, PHM aims to ensure safe, reliable and efficient operation by detecting anomalies that may indicate system faults or environmental disturbances before they escalate into mission-critical failures. Unsupervised anomaly detection by isolation forest has been carried out using the aero-propulsion system simulation dataset as a benchmark (Khan et al., 2019). For device-level anomaly detection of UAVs, a reinforcement learning-based anomaly detection system that monitors motor temperature has been conducted (Lu et al., 2017). Also, with the rapid development of deep learning, deep neural network-based approach for anomaly detection and recovery has already been studied (Yang et al., 2023) However, despite the growing interest in UAV health monitoring, publicly available datasets suitable for PHM research remain scarce, and the limited availability of such datasets — particularly in the UAV domain, regardless of whether the type is rotary-wing or fixed-wing — has become a major bottleneck for advancing PHM research. Current existing UAV datasets focus on flight control, navigation, or mapping tasks, and few provide synchronized, multimodal sensor data collected under controlled fault or anomaly conditions. This lack of comprehensive and annotated datasets limits the development, benchmarking, and reproducibility of data-driven PHM algorithms for UAVs.

To address this gap, this work uses a newly released multimodal UAV dataset with synchronized telemetry, sensor measurements, motion capture data, and pilot control inputs, recorded under nominal and fault-injection conditions. In addition, we present a standardized pipeline for unsupervised anomaly detection, enabling researchers to train models on healthy flight data and evaluate their performance in detecting injected faults. In this work, we release a new multimodal UAV dataset with synchronized telemetry, sensor, and pilot input streams, explicitly designed to support unsupervised anomaly detection research. We employ several standard data-driven anomaly detection methods based on reconstruction errors in the workflow as shown in Figure 1. Rather

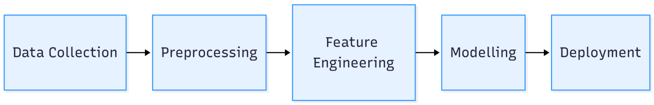


Figure 1. A typical data-driven PHM workflow.

than pursuing algorithmic novelty, our goal is to investigate how these standard models behave under controlled fault injection scenarios and to offer solid, reproducible baselines for future research. It should be noted that our objective is not to provide definitive performance evaluations (e.g., in terms of AUC) of these baseline methods, as a fixed train–test split is not defined for the dataset. This is primarily because the currently released initial version of the dataset is not yet fully mature, making it premature to establish a standard benchmark. Instead, the emphasis is on offering transparent reference implementations that can be adapted and evaluated under user-defined experimental setups. The remainder of this paper details the detection methods (Section 2), dataset description and experimental details (Section 3), results and discussion (Section 4), and conclusion (Section 5).

2. Problem Statement and Methodologies

Let $\mathbf{X} \in \mathbb{R}^{N\times D}$ denote the input data matrix, where N is the number of time steps, and D is the number of observed variables. We formulate the problem as an unsupervised anomaly detection task, where a model is trained exclusively on normal data and evaluated on test samples.

Among various approaches for unsupervised anomaly detection, reconstruction error-based methods are particularly well-suited to this task. These methods are typically trained on normal samples to learn a compact latent representation that captures the underlying structure of healthy behavior. Test samples that cannot be faithfully reconstructed are regarded anomalous and can be further analyzed for fault isolation. We then briefly describe methods used in this work.

Principle Component Analysis PCA is one of the most widely used dimensionality reduction technique and is a natural baseline for anomaly detection because of its simplicity and maturity (Takeishi & Yairi, 2014). In practice, PCA is commonly implemented using Singular Value Decomposition (SVD) on the centered data matrix $\mathbf{X}$ to improve computational efficiency, especially in high-dimensional cases as $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, where $\mathbf{U} \in \mathbb{R}^{N\times N}$ and $\mathbf{V} \in \mathbb{R}^{D\times D}$ are orthogonal matrices, and $\mathbf{\Sigma} \in \mathbb{R}^{N\times D}$ is a diagonal matrix containing the singular values along its main diagonal. By selecting latent dimensions $K < D$, we can project $\mathbf{X}$ into a K-dimensional latent subspace and then reconstruct it back to the original feature space. The reconstructed matrix by PCA using the truncated right singular vector matrix $\mathbf{V}_K$ is given by $\hat{\mathbf{X}} = \mathbf{X}\mathbf{V}_K\mathbf{V}_K^T$.

Kernel Principle Component Analysis To relax the linear assumption of PCA, Kernel PCA introduces nonlinearity by constructing a kernel matrix $\mathbf{M} \in \mathbb{R}^{N\times N}$ through kernel function $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle$, and then applies standard PCA in a high-dimensional feature space (Schölkopf et al., 1998). The reconstruction in kPCA is usually defined in the feature space between the input's mapped representation and its projection onto the top-$\boldsymbol{K}$ principal components. This allows capturing nonlinear anomalies beyond the capacity of standard PCA.

Autoencoder AE is an unsupervised neural network trained to learn reconstructions close to the observation inputs. It is composed of an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$. The encoder takes the input $\mathbf{X}$ and projects them into a latent space, while the decoder reproduces the compressed representation $\mathcal{E}(\mathbf{X})$ back to the observation inputs. The nonlinearity of AE is introduced by nonlinear activation functions such as ReLU and this nonlinear reconstruction capability makes AE suitable for anomaly detection tasks (Sakurada & Yairi, 2014). AE is trained to minimize the reconstruction error on normal data samples and the reconstructed result for AE is $\mathcal{D}(\mathcal{E}(\mathbf{X}))$.

Variational Autoencoder VAE is a generative model combines the ideas of AE and probabilistic modeling. Instead of directly encoding input $\mathbf{X}$, it learns a probabilistic encoder, typically modeled as a Gaussian. During inference, the latent variable is sampled from the distribution and decoded to reconstruct the input. Different from AE, VAE is trained by maximizing the evidence lower bound (ELBO), which balances reconstruction fidelity and latent regularization. This probabilistic nature and inherent ability to model data distributions make VAE a popular choice for anomaly detection (An & Cho, 2015).

When reconstructing test data using a reconstruction-based anomaly detection method, the greater the difference between the reconstruction result and the input value, the more likely it is that the data is anomalous, where the Frobenius norm of the difference between the reconstructed and original data matrices $\|X - \hat{X}\|_F$ is used as the reconstruction error.

3. Dataset description

We conducted three sets of experiments under three different types of abnormal situation. Each dataset contains one set of data from completely normal experiments and one set from abnormal experiments, with each set consisting of three (or two) flights, as shown in Table 1. The test sets of the first and second datasets consist of flight data collected under unbalanced propeller conditions. The propeller in the test data of the second dataset is more unstable than that in the test data of the first dataset. In the third dataset, approximately 3 cm part was cut from the end of a propeller to simulate a crack situation. Each flight in the dataset lasts approximately one to two minute, during which the UAV undergoes takeoff, cruising, and landing phases. All training data were collected under fully healthy conditions, while for the test data, an ab-

Table 1. Overview of the quadcopter UAV indoor flight dataset. Training sets contain only normal flights, while test sets include different states.

Dataset	Train		Test	
	Flights	State	Flights	State
1	3	Normal	3	Unbalanced propeller
2	3	Normal	3	Unbalanced propeller
3	3	Normal	2	Propeller crack

normal condition was introduced prior to takeoff. As a result, the test data are expected to exhibit responses distinct from the training data over the full duration of pilot operation.

As for the data composition, the dataset is organized on a per-experiment basis, with each flight stored in separate files. The primary files include PX4 log files, INAV log files, and motion capture (MoCap) data files. The PX4 log files contain the onboard flight controller data, such as sensor readings, control inputs, and state estimates, recorded by the PX4 autopilot firmware. The INAV log files provide navigation and control data from the INAV flight control system, with the most relevant information are vibration and temperature measurements. The MoCap files record the high-precision position and orientation of the UAV, obtained from an external motion capture system, serving as ground truth for trajectory and state estimation. In the task of anomaly detection, the above-mentioned primary files are considered to provide sufficient information for model training.

4. EXPERIMENT

4.1. Data Preprocessing

As mentioned, we have mainly three recorded files per flight. Since the different files and even different sensor measurements recorded within a file, may have different sampling frequencies and timestamps, the first step we need to take is to synchronize all the files. The code for synchronization is provided together in the dataset. After synchronization is complete, all sensor recordings will have the same timestamps. However, the discrepancy in initial timestamps led to a significant number of missing values in both high-frequency and low-frequency data streams. So we applied a fill operation to impute all missing values, ensuring temporal alignment and data completeness across all channels. Finally, the sampling frequency of all sensors are synchronized to 8 kHz. Figure 2 shows the roll, pitch, yaw and throttle in the synchronized PX4 and four motor outputs signals in INAV logged file of the first flight trial in Dataset 1. It can be seen that due to the fill operation, the signals exhibit a horizontal plateau at both the beginning and the end. Figure 3 illustrates the three-dimensional trajectory of the UAV captured by the motion capture (MoCap) system during a single experimental trial. The plotted path shows the spatial positions of the UAV from takeoff to landing, with the start and end points marked for reference. During the cruising phase, the UAV performed repeated back-and-forth translations and rotations within the flight area, as reflected in the oscillatory patterns of the trajectory. Considering both computational efficiency and the types of anomalies of interest in the dataset, we downsampled the data to a final sampling rate of 8 Hz.

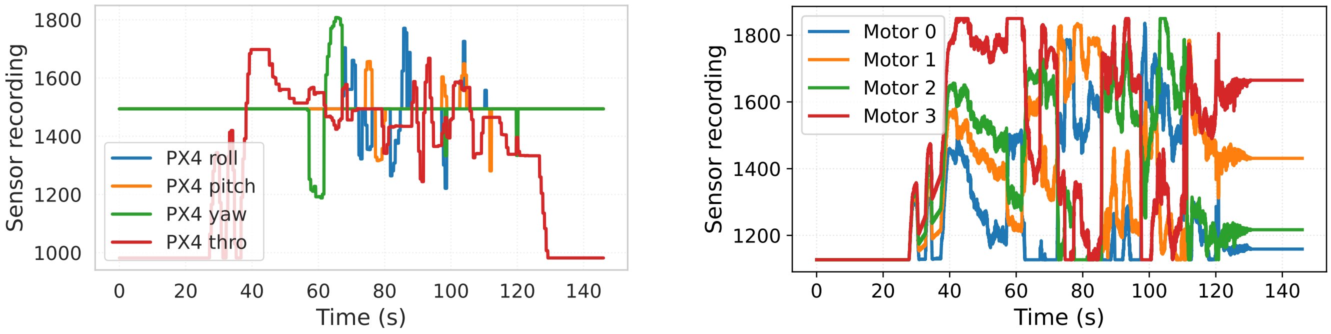
4.2. Feature Selection

Feature selection is a crucial step in almost all tasks of PHM research. Effective feature selection aims to retain informative variables for the task while removing redundant ones, thereby improving both computational efficiency and prognostic accuracy.

The feature selection process in this study consisted of three stages. In the first stage, we performed a manual inspection of the experimental data to verify the validity of each feature and to determine whether each sub-file contained variables relevant to anomaly detection. This reduced the total number of features to 900. In the second stage, we applied a filter based on the count of valid values for each feature. This step aimed to remove data streams from each sub-file that did not contain sufficient valid information. By setting the threshold to at least three valid values, the feature dimensionality was significantly reduced to 286. In the final stage, we addressed cases in which certain features required merging due to being stored across multiple columns. After completing this merging process, the final dimensionality of the dataset was reduced to 205. We have provided the results and process code for standard feature selection. Based on the settings, the final number of features used for training may be different.

4.3. Implement Details

Since the drone was already in standby mode at the start of the experiment and during the technical portion, this data was deleted based on the *landed* parameter in the *takeoff_status* of the PX4 subfile as shown in Figure 4. Figure 5 shows the kernel density estimations (KDE) of four randomly selected example features from the training and test datasets. KDE is employed here to visualize and compare the probability distributions of features across the two datasets, providing an intuitive way to detect potential distribution shifts. In the context of anomaly detection, such shifts are of particular importance, as they often indicate that the test data originate from a different operating condition or system state than the training data. When models are trained solely on normal data, a distribution mismatch in the test phase typically leads to increased prediction errors. In our provided datasets, the plots reveal a clear difference between the training and test feature distributions, suggesting that most anomaly detection methods would be able to readily identify anomalies in the test flights. After feature selection, all data are standardized using the mean and standard deviation of the training set by Z-score.



(a) PX4 plot

(b) INAV Motor plot

Figure 2. Example plots of PX4 and INAV signals from the first flight trial in Dataset 1.

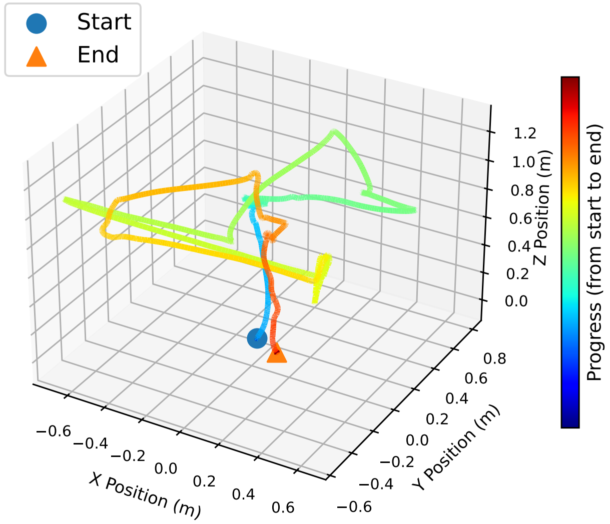


Figure 3. 3D Trajectory of MoCap Data from the first flight trial in Dataset 1.

Before model training, it is required to ensure that the latent space dimensionality is consistent across different methods. So we first apply PCA to the dataset. Figure 6 illustrates the results of the PCA analysis performed on the training data. As can be seen from the figure, the first 48 principal components explain 0.8 of the variance in the training data, while the cumulative variance explained by the first 100 principal components is close to 0.95. In the subsequent experimental section of this paper, the dimension of the latent space will be uniformly set to 60 for comparison purposes. Figure 6b shows the projection of the first training flight and the first test flight onto the first two principal components. Apart from the beginning and end of the experiment, the anomalous flight exhibits a large discrepancy from the training flight during most of the flight. Also note that a part of non-flying periods have already been removed, so the data shown here mainly represent valid flight segments.

4.4. Results and Discussion

In the final experiments, we employ PCA, kPCA, AE, and VAE for reconstruction. As described in the previous subsection, the dimension of the latent space was set to 60 to facilitate a fair comparison. For kPCA, we select the Gaussian kernel to capture nonlinear structures in the data. For AE and VAE, we adopt multi-layer perceptron (MLP) for both the encoder $(500, 200)$ and decoder $(200, 500)$. In the case of the AE, the encoder ultimately compresses the input data into a 60-dimensional latent vector. In contrast, the VAE employs the encoder to map the input into two 60-dimensional vectors representing the mean and standard deviation of a Gaussian distribution.

Figure 7 illustrates the temporal evolution of the 60-dimensional principal components or latent representations obtained by the four methods after being trained on the training dataset. The latent space of the VAE does not correspond to fixed numerical values and it learns a probability distribution for each sample. To facilitate a fair comparison with the other methods, the results for the VAE are plotted using the mean of the learned distribution, denoted by the mathematical symbol μ. It can be observed that the principal components extracted by PCA exhibit a clear and pronounced trend over flight/time. The components extracted by kPCA also shows certain trends, but are weaker and appear more irregular compared to PCA. The trend-related principal components extracted in PCA and kPCA are considered to be caused by variables such as battery usage and temperature rises measured by temperature sensors. Since the training dataset is constructed by concatenating the recordings of three separate flights, distinct boundaries between flights can be clearly identified in certain principle components. This indicates that PCA retains global variation patterns that reflect differences between flight sessions. In contrast, the latent spaces learned by AE and VAE still allow the flight segments to be distinguished from the values, but the overall trends are comparatively weaker.

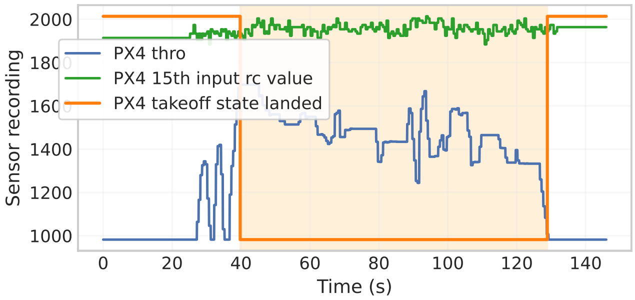


Figure 4. Illustration of the final sampled area for sensor sequences.

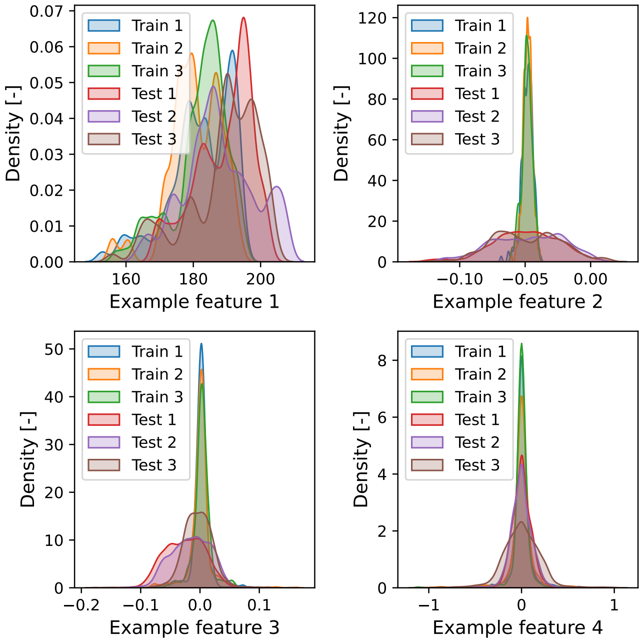
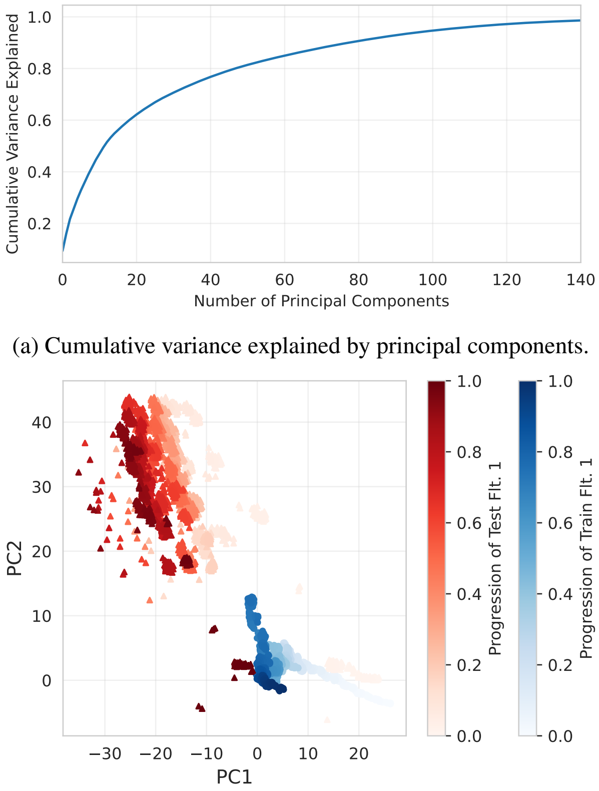


Figure 5. Kernel density estimations of four randomly selected example features from three training flights (Train 1–3) and three anomalous flights (Test 1–3) in Dataset 2.



(a) Cumulative variance explained by principal components.

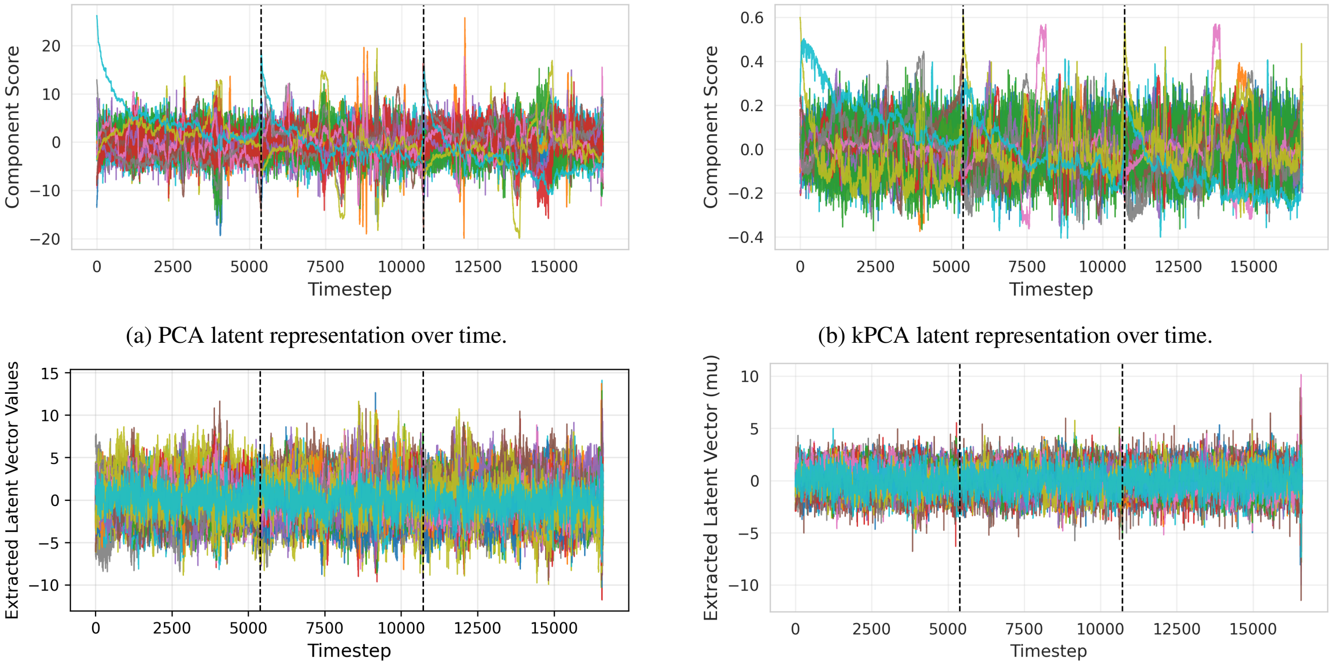
(b) PCA projection of Train Flight 1 and Test Flight 1.

Figure 6. PCA results of the Dataset 2.

In our dataset, since the anomalous flights exhibit abnormal conditions from the very beginning of the experiments, the reconstruction errors are generally pronounced. Therefore, in the comparison, instead of evaluating the overall reconstruction error, we focus on the reconstruction performance of several key parameters in the original observation space for a more targeted comparison. In our experiments, both hover thrust and motor output are expected to be affected under the simulated anomalies, making them relevant features for anomaly detection and analysis. Therefore, we specifically compare the reconstruction performance on these parameters.

Figure 8 shows the reconstruction error of hover thrust and four motor outputs. For PCA, the reconstruction errors remain relatively small across all five parameters. Noticeable increases in error occur mainly in the hover thrust during the first two test flights, indicating that PCA is sensitive to large deviations in global thrust demand but otherwise maintains stable reconstruction performance. For AE, a pronounced reconstruction error appears in the hover thrust during the third anomalous test flight. However, across the four motor outputs, AE consistently produces larger reconstruction errors, suggesting that it is more sensitive to localized deviations in individual actuators and thus potentially more effective at detecting distributed anomalies in the propulsion system. VAE shows overall reconstruction patterns similar to PCA, with relatively low error magnitudes and limited variation across flights, indicating comparable robustness to PCA but potentially less sensitivity to small actuator-level anomalies than AE. It can also be observed that during the first two test flights, all three methods respond clearly to anomalies, whereas in the third flight AE still shows a discernible response, suggesting a sustained ability to detect anomalies in that segment.

These differences may be explained by the physical impact of the simulated anomalies and the used methods. Unbalanced propeller (and propeller crack) primarily alter the aerodynamic efficiency of the rotor system, increasing the thrust required to maintain altitude. This manifests as a noticeable change in hover thrust, which is a global parameter aggre-



(a) PCA latent representation over time.

(b) kPCA latent representation over time.

(c) AE latent representation over time.

(d) VAE latent representation over time.

Figure 7. Temporal variations of principal components (PCA, kPCA) and latent space representations (AE, VAE) for Dataset 2. The results are obtained by concatenating the three flight sequences in chronological order along the timestep axis. Black dashed vertical lines indicate the boundaries between consecutive flight data.

gated over all motors. Since PCA emphasizes global variance, it is particularly sensitive to such coordinated changes across the propulsion system. In contrast, these anomalies also introduce asymmetric load distribution among the motors, causing certain motors to work harder while others operate at reduced loads to maintain overall stability. Such localized deviations may not strongly influence global parameters like hover thrust but are directly visible in the individual motor outputs. AE, with its nonlinear mapping and ability to capture fine-scale relationships, is more responsive to these channel-specific deviations, producing larger reconstruction errors for motor outputs. VAE, while also a nonlinear model, imposes a probabilistic constraint on the latent space, which can smooth out local variations. This regularization helps preserve general reconstruction stability but may reduce sensitivity to subtle anomalies in motor-level compared to AE.

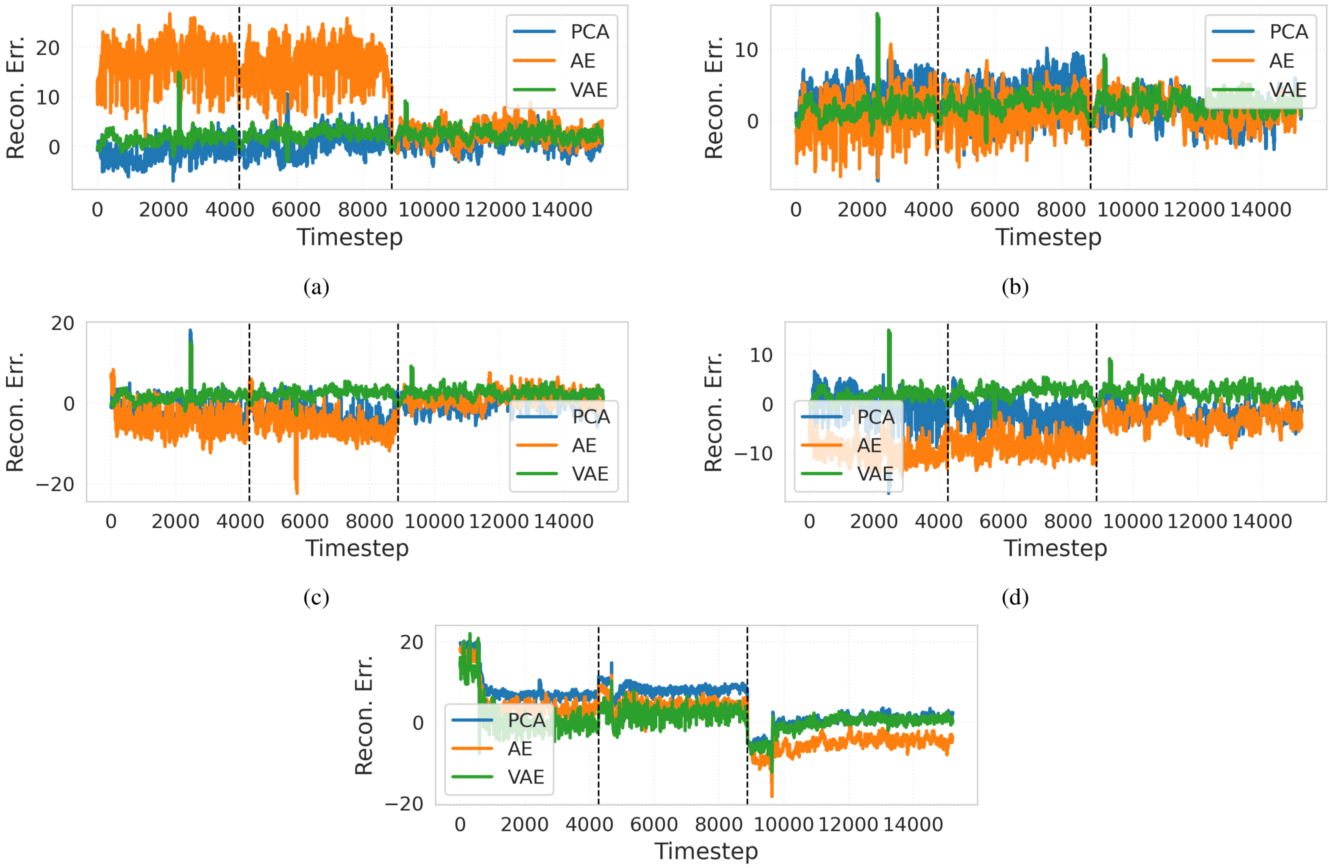
5. CONCLUSION

In this study, we introduced a multimodal UAV dataset specifically designed for unsupervised anomaly detection under controlled fault-injection scenarios, including unbalanced propeller and propeller crack conditions. We tried several reconstruction-based methods on synchronized telemetry, motion capture, and control data, with a unified latent dimensionality for fair comparison. Instead of evaluating the overall reconstruction error across all features, we focused on reconstructing key parameters that are directly affected by the simulated anomalies. Our analysis demonstrated that different methods exhibit distinct sensitivities in our datasets. These findings highlight the importance of selecting models based on the nature of the anomalies of interest. The released dataset, preprocessing pipeline, and baseline results provide a reproducible foundation for further research in UAV prognostics, fault isolation, and the development of more robust anomaly detection algorithms.

One limitation of the current study is that the training and test flights were conducted sequentially, which introduces distribution shifts in parameters affected by cumulative or time-varying effects such as battery voltage and temperature. These shifts can lead to inflated reconstruction errors unrelated to the targeted anomalies. In future dataset releases, we plan to provide updated versions with more balanced and interpretable data distributions, enabling fairer evaluation and more reliable diagnostic performance. Nevertheless, the currently released version of the dataset can still support user-defined train–test splits and facilitate various model development experiments.

ACKNOWLEDGMENT

This work was supported by JSPS International Joint Research Program JPJSJRP20221501.



(a)

(b)

(c)

(d)

(e)

Figure 8. Reconstruction errors of (a)(b)(c)(d) four motors outputs and (e) hover thrust and in Dataset 2. The three flight sequences are concatenated along the timestep axis, with black dashed vertical lines indicating the boundaries between consecutive flights.

REFERENCES

An, J., & Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. *Special lecture on IE*, *2*(1), 1–18.

Khan, S., Liew, C. F., Yairi, T., & McWilliam, R. (2019). Unsupervised anomaly detection in unmanned aerial vehicles. *Applied Soft Computing*, *83*, 105650.

Liew, C. F., & Yairi, T. (2020). Companion unmanned aerial vehicles: A survey. *arXiv preprint arXiv:2001.04637*.

Lu, H., Li, Y., Mu, S., Wang, D., Kim, H., & Serikawa, S. (2017). Motor anomaly detection for unmanned aerial vehicles using reinforcement learning. *IEEE internet of things journal*, *5*(4), 2315–2322.

Sakurada, M., & Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the mlsda 2014 2nd workshop on machine learning for sensory data analysis* (pp. 4–11).

Schölkopf, B., Smola, A., & Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, *10*(5), 1299–1319.

Takeishi, N., & Yairi, T. (2014). Anomaly detection from multivariate time-series with sparse representation. In *2014 ieee international conference on systems, man, and cybernetics (smc)* (pp. 2651–2656).

Yang, L., Li, S., Li, C., Zhu, C., Zhang, A., & Liang, G. (2023). Data-driven unsupervised anomaly detection and recovery of unmanned aerial vehicle flight data based on spatiotemporal correlation. *Science China Technological Sciences*, *66*(5), 1304–1316.

Zonta, T., Da Costa, C. A., da Rosa Righi, R., de Lima, M. J., Da Trindade, E. S., & Li, G. P. (2020). Predictive maintenance in the industry 4.0: A systematic literature review. *Computers & industrial engineering*, *150*, 106889.