

COB-2023-0179

Quadrotor Fault Detection and Diagnosis using Multi-class Support Vector Machine

Petrus Lopes Fiore
Yohan Díaz-Méndez
Guilherme Ferreira Gomes

Institute of Mechanical Engineering, Federal University of Itajubá, Brazil
petrusfiore@unifei.edu.br, yohan.g8@unifei.edu.br, guilhermefergom@unifei.edu.br

Abstract. *Quadrotors are rotary-wing UAVs capable of vertical take-off and landing. Due to several advantages and numerous applications in all areas of knowledge, their use becomes widespread worldwide. Despite the development of new control algorithms ensuring robustness in relation to disturbances and modeling errors, flight safety in emergency situations as in case of sensor or actuator failure is still a matter of concern to drone manufacturers and is the focus of current research. Active fault tolerant control (AFTC) techniques need a fault detection and diagnosis (FDD) system in such a way that in a timely manner, the controller can be reconfigured and stability could be recovered. The main objective of this work is to use the Multi-class Support Vector Machine (SVM) classification method in order to identify the occurrence of actuator hard-over fault in a quadrotor. A numerical model is used to simulate and collect data. The Principal Component Analysis (PCA) feature extraction technique was in conjunction with SVM allowing to properly classify the type of fault injected on the system with a high accuracy. Results also demonstrated a high influence of the data collection time on the precision of the method.*

Keywords: *Quadrotor, Fault Detection and Diagnosis, Support Vector Machine, Principal Component Analysis*

1. INTRODUCTION

The possibility that drones can replace humans in potentially dangerous situations is the main factor behind investing and researching on UAVs Sadhu *et al.* (2023). The use of drones has significantly enhanced the performance of aerial operations across various industries, such as, Surveillance and Monitoring (Yang *et al.*, 2022), Mapping and Surveying (Budiharto *et al.*, 2021), Filming and Photography (Xiao *et al.*, 2022), Precision Agriculture (Meivel and Maheswari, 2021; Hafeez *et al.*, 2022) and others. In all those applications, drones are prone to external interference, they are affected by external airflow and loads, physical damage/attack, or a cyber attack, etc. resulting in route deviation which can seriously affect the meeting of mission requirements. It is well known that drones facilitate aerial inspections of infrastructure by eliminating the need for manual inspections, reducing costs and improving safety to operators. Nevertheless, in automatic delivery missions due to necessity of flying far beyond line of sight over densely populated areas and hovering inside a residential zone within touching distance of humans (Sindhvani *et al.*, 2020), makes the safety of the operations a factor that deserves important attention. According to Wang *et al.* (2020) new generations of UAVs need to be designed to accomplish their missions with not only increased efficiency but also more safety and security.

As mentioned before, during a flight, one actuator maybe break down or weather conditions may influence the dynamic as an external disturbance so designing a fault-tolerant control system (FTCS) is necessary for quadrotor UAVs. Quadrotors are more susceptible to catastrophic faults due to the lack of actuator's redundancy. FTC systems can be classified as passive fault tolerant control (PFTC) systems or active fault tolerant control (AFTC) systems. According to Hamadi (2020), in PFTC systems, the controller has a fixed structure and is designed off-line using robust control and adaptive control laws without requiring error detection which becomes PFTC methods, computationally more attractive. The main disadvantage of PFTC is its less capacity to tackle with actuator faults when compared to AFTC. On the other hand, AFTC relies on error detection from a fault detection and identification (FDI) / fault detection and diagnosis (FDD) unit scheme to perform a system recovery or control reconfiguration, so that, the stability and acceptable performance of the system can be maintained. The FDD detects and localizes the faults occurring in the system. The main advantage of this method is that since we are monitoring the system health, we can know if the system is getting close to its limits (Hamadi, 2020)

The current available methods for Unmanned Aerial Vehicle (UAV) fault diagnosis can be basically divided into three categories: analytical model-based methods, knowledge-based methods, and signal processing-based methods (Saufi *et al.*, 2019). The analytical model-based method focuses on using an accurate mathematical model and observable inputs and outputs to construct residual signals. By analyzing the residual signals, the difference between the expected behavior of the system and the actual operation mode realizes the FDI/FDD of the UAV (Zhang *et al.*, 2021). Since it is difficult to

establish accurate mathematical models of objects in some practical applications, as quadcopter flight due to modeling uncertainties, the analytical model-based method has limitations to its implementation in many cases. Knowledge-based FDI methods mainly use the knowledge of systems established by experts in this field. Due to the complexity of knowledge-based methods, the FDI/FDD accuracy highly depends on the level of expert knowledge. Meanwhile, the FDI speed decreases with the rule number. Thus, the efficiency of the knowledge-based FDI method needs to be improved (Zhang *et al.*, 2021). Finally, Signal processing-based methods extract the features of the measurement signals to achieve FDI without establishing complex mathematical models of the system and without the need of labeled data.

A somewhat less popular group of methods used for UAV fault detection are data based (data-driven, knowledge-based) methods (Puchalski and Giernacki, 2022). With the advent of the big data era, increased efficiency of sensors on board and computational capabilities of autopilot processors, some data-driven diagnosis methods have been raised. These methods rely on analyzing historical data and applying machine learning or statistical techniques to identify anomalies or patterns indicative of faults. Data-driven methods do not rely on predefined rules or expert knowledge, but instead learn patterns directly from the data. These methods can detect both known and unknown faults and have the potential to discover novel fault patterns.

The machine learning data driven technique known as Support Vector Machine (SVM) is a relatively new approach classification and regression algorithm used to solve supervised learning problems. SVM is particularly effective in classification tasks, where the goal is to separate data samples into different classes based on their characteristics. SVM is based on supervised learning, which requires to label the fault cases previously in the training data. The most forthcoming properties of SVMs are (Bronz *et al.*, 2020): (i) SVMs constructs a decision boundary that has the largest distance to the example points, called Maximum margin separator; (ii) SVMs can use so-called kernel-trick, which is to represent the linear separator in a high-dimensional space so that it can represent a non-linear separator in the original space and, (iii) SVMs are a non-parametric method, they retain training examples and potentially need to store them all. SVM can also be used and implemented to the problem of fault diagnosis in order to predict faults on-line. Its capabilities for FDD/FDI in UAVs fault detection were previously explored in Baskaya *et al.* (2017), Bronz *et al.* (2020), Bondyra *et al.* (2017) and Saied *et al.* (2017) demonstrating high accuracy in quadrotor motor and propeller faults identification.

In order to demonstrate the Fault Detection and Diagnosis (FDD) system working in a quadcopter UAV, in this paper, the data driven-based method, Support Vector Machine (SVM) is introduced. To confirm the SVM capabilities to detect and identify quadcopter faults, data-sets created from a quadrotor numerical model, subject to motor hard-over faults, are used to train a Multiclass SVM algorithm. Multiclass SVM is used for classification problems where there are more than two classes or categories to be predicted. In terms of quadrotor faults, it will be possible not only to detect whether or not there was a failure, but also to classify the cause of it. Principal Component Analysis (PCA) is used in order to reduce the dimensionless of the data-sets while preserving the most significant information facilitating and cleaning the data before SVM classifier use and to allow the results visualization. Another contribution of this work is the evaluation of the influence of data collection time on the accuracy of the SVM, time is a crucial factor in FDD systems, so that, the AFTC controller is able to act in a timely manner.

2. Quadrotor Faulty Model

In this study, first, a model of a quadrotor is simulated. This model is utilized to generate data and test the classification algorithm. The measurements of the states during and after a fault injection, are used to feed a data-set, this is done for several trajectory tracking scenarios. The nonlinear quadrotor flight dynamics for translational and rotational motion can be given as a system of first order partial differential equations, as follows (Labbadi *et al.*, 2018; Labbadi and Cherkaoui, 2019). The quadrotor scheme can be seen at Figure 1.

$$\begin{cases} \ddot{\phi} = \dot{\theta}\dot{\psi}\frac{(I_y - I_z)}{I_x} - \frac{J_r}{I_x}\Omega_r\dot{\theta} - \frac{k_{fa_x}}{I_x}\dot{\phi}^2 + \frac{1}{I_x}\tau_\phi \\ \ddot{\theta} = \dot{\phi}\dot{\psi}\frac{(I_z - I_x)}{I_y} + \frac{J_r}{I_y}\Omega_r\dot{\phi} - \frac{k_{fa_y}}{I_y}\dot{\theta}^2 + \frac{1}{I_y}\tau_\theta \\ \ddot{\psi} = \dot{\phi}\dot{\theta}\frac{(I_x - I_y)}{I_z} - \frac{k_{fa_z}}{I_z}\dot{\psi}^2 + \frac{1}{I_z}\tau_\psi \\ \ddot{x} = -\frac{k_x}{m}\dot{x} + (\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)\frac{F}{m} \\ \ddot{y} = -\frac{k_y}{m}\dot{y} + (\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi)\frac{F}{m} \\ \ddot{z} = -\frac{k_z}{m}\dot{z} - g + (\cos\phi\cos\theta)\frac{F}{m} \end{cases} \quad (1)$$

Where m is the total mass of the quadrotor, J_r is the rotor inertia, g the earth gravity acceleration and $[I_{xx}, I_{yy}, I_{zz}]$ are the quadrotor inertia moments around its body reference axis x_B, y_B, z_B respectively. $\Omega_r = \omega_1 - \omega_2 + \omega_3 - \omega_4$ and $[K_{fa_x}, K_{fa_y}, K_{fa_z}]$ and $[K_x, K_y, K_z]$ represent rotation and translation aerodynamic drag coefficients. The system states are defined as: $[\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}, x, \dot{x}, y, \dot{y}, z, \dot{z}]$ being $[\phi, \theta, \psi]$ the rotation (Euler) angles and $[x, y, z]$ the quadrotor position in relation to the earth's reference system. The control inputs are: total force F and the torques applied to the quadrotor τ_ϕ, τ_θ and τ_ψ . Due to the plus-type frame quadrotor configuration, forces and moments can be written according

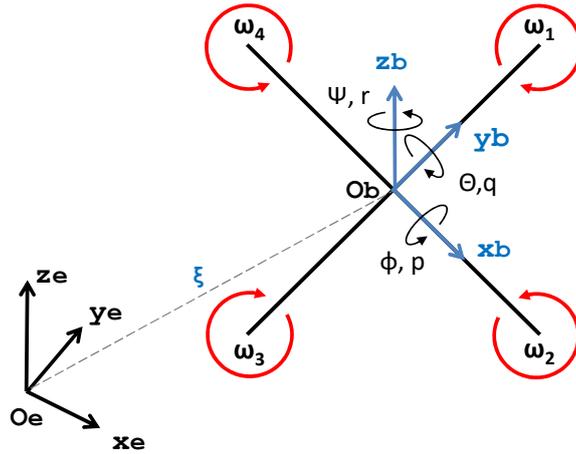


Figure 1. Quadrotor scheme (Diaz-Mendez *et al.*, 2022)

to the motor speeds as follows.

$$\begin{bmatrix} F \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -lb & 0 & lb \\ -lb & 0 & lb & 0 \\ d & -d & d & -d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \quad (2)$$

Where ω_i is the angular speed of motors $i = [1, 2, 3, 4]$, b and d are the thrust and torque constants of the rotors that depend on the propeller geometry and air density. According to their location, faults in automatic control systems can be classified as actuator, sensor or plant faults. Actuator faults or defects in a propulsion system of the quadrotor vehicles are normally catastrophic due to the lack of actuation redundancy leading to the loss of thrust generated by rotors and as a result, disturbances of thrust balance, yaw imbalance, higher power consumption and further degradation resulting in the possible crash of the vehicle (Bondyra *et al.*, 2017). In this work, a hard-over fault is injected during the quadrotor trajectory tracking execution. Hard-over refers to a fault condition where one or more of the motors or actuators of the quadrotor can get stuck or saturated to a maximum or minimum limited condition (common in motor or propeller partial damage). This fault causes a loss of control over the affected motor(s), leading to a significant imbalance in the thrust produced by the motors. The force generated by the rotors is defines as $f_i = b\omega_i^2$, then, the hard-over fault can be modeled as in Equation 3. Where t_f refers to starting fault time and γ represents the fault severity. $\gamma = 0$ (no fault) and $\gamma = 1$ (total actuator fault).

$$\begin{cases} 0 \leq f_i \leq \omega_i^{max} & t < t_f \\ 0 \leq f_i \leq (1 - \gamma)b\omega_i^{max} & t \geq t_f \end{cases} \quad (3)$$

The simulations were performed in a closed-loop system with a robust Sliding Mode Controller (SMC) acting in order to help the drone during the trajectory tracking before the fault injection and to help recovering its stability in a PFTC scheme, after the fault injection. The sliding surface s_i of the SMC controller and the corresponding control input are defined in Eq. 4

$$\begin{aligned} s_i &= \sum_{j=1}^{\rho_i-1} k_j^i e_j^i + e_{\rho_i}^i \\ u_i &= -K_i \text{sat} \left(\frac{s_i}{\mu_i} \right) \end{aligned} \quad (4)$$

Where e_j^i is the tracking error for the i -th output, ρ_i is the relative degree of the system and the positive constants $k_1^i, \dots, k_{\rho_i-1}^i$ are chosen such that the polynomial $\lambda^{\rho_i-1} + k_{\rho_i-1}^i \lambda^{\rho_i-2} + \dots + k_1^i$ is Hurwitz, which guarantees that in the sliding phase ($s_i = 0$) the tracking error and its derivatives converge to zero. K_i is the main gain of the controller (chosen as the upper bound of the control input), μ_i is the boundary layer (small region around the sliding surface where the control action switches), which reduces the chattering occurrence. Table 1 summarizes the controller configuration parameters.

Table 1. SMC Control System Parameters

SMC position	Value	SMC attitude	Value
k_1^x, K_x, μ_x	2, -5, 10	$k_1^\phi, K_\phi, \mu_\phi$	3, 12, 1
k_1^y, K_y, μ_y	2, -5, 10	$k_1^\theta, K_\theta, \mu_\theta$	3, 12, 1
k_1^z, K_z, μ_z	2, -50, 1	$k_1^\psi, K_\psi, \mu_\psi$	1, -10, 5

To the best of the author’s knowledge, the classification technique performance will be better proved if the quadrotor dynamics data is extracted in this condition (more real due to the necessity of controller on all phases of a quadrotor flight). A previous approach, was studied by Baskaya *et al.* (2017) but considering the open loop data of a fixed wing aircraft. The quadrotor physical, inertial and aerodynamic properties considered in this work are shown in Table 2.

Table 2. Quadrotor Parameters. Extracted from Labbadi and Cherkaoui (2019)

Parameters	Value	Parameter	Value
g (m/s ²)	9.81	k_y (N · s/m)	5.567×10^{-4}
m (kg)	0.486	k_z (N · s/m)	5.567×10^{-4}
I_x (kg · m ²)	3.827×10^{-3}	$k_{f_{ax}}$ (N · s/m)	5.567×10^{-4}
I_y (kg · m ²)	3.827×10^{-3}	$k_{f_{ay}}$ (N · s/m)	5.567×10^{-4}
I_z (kg · m ²)	7.656×10^{-3}	$k_{f_{az}}$ (N · s/m)	5.567×10^{-4}
I_r (kg · m ²)	2.838×10^{-5}	b (N · s ²)	2.984×10^{-3}
k_x (N · s/m)	5.567×10^{-4}	d (N · s ²)	3.232×10^{-2}

Based on the studies made by Baskaya *et al.* (2017), Bronz *et al.* (2020) and Sindhwani *et al.* (2020), linear accelerations and angular rates are important variables to be monitored and used to train classification techniques. This makes sense due to its dependency on the aerodynamic forces that are directly affected by the fault injection. The wide availability of accelerometers and gyroscopes in most real drones due to their need as a reference signal for flight controllers also justifies their use in FDD systems. Then, in this paper, the data-set used to train the classification method of the FDD system is composed by the first and second derivatives of the states measured in the quadrotor numerical model. This data is collected during the tracking of three different trajectories: hover, square (see Fig. 2) and circular (see Fig. 3) for different faults: (i) motor 1 fault, (ii) motor 2 fault, (iii) motor 3 fault, (iv) motor 4 fault, for different fault severity levels (γ), the data was collected during a ΔT time after the fault occurrence and labeled according to the motor which failed. Data points in Multi-class SVM are associated with a specific class or category label. The labeled data is essential in SVM because it provides the necessary information for the algorithm to learn the decision boundaries between different classes. The use of data from different scenarios and flight phases will help eliminate biased and partial results in order to effectively assess the performance of the classification technique.

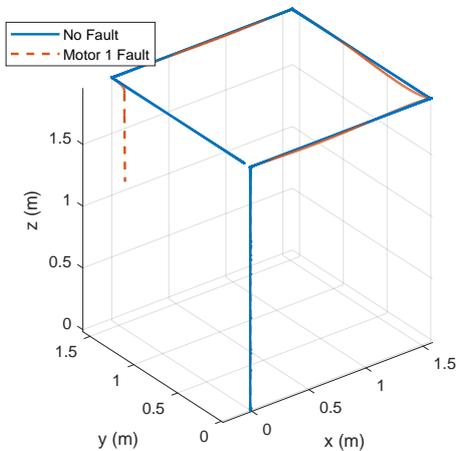


Figure 2. Quadrotor quad trajectory.
 $t_f = 10s, \gamma = 0.48, \omega_i^{max} = 800rpm$

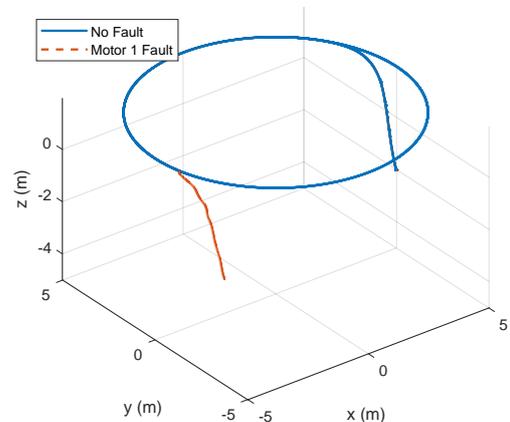


Figure 3. Quadrotor circular trajectory.
 $t_f = 20s, \gamma = 0.68, \omega_i^{max} = 1200rpm$

3. Principal Component Analysis and Multi-class Support Vector Machine for Quadrotor Fault Detection

Principal Component Analysis (PCA) is like a "feature extraction technique" that refers to the process of selecting or transforming raw data into a reduced set of representative features that capture the most relevant information for a particular task or problem. PCA is used in this work to make a dimensionality reduction that will identify the most important orthogonal components in the data-set created while preserving the most significant information. Li *et al.* (2016) recommends the PCA model to better suit the dynamic nature of UAVs and more effective fault detection. The main steps involved in PCA are as follows: (i) PCA typically begins by standardizing the input data to have zero mean and unit variance, this step ensures that each feature contributes equally to the analysis; (ii) The covariance matrix is calculated from the standardized data, it quantifies the relationships between different features and indicates their level of dependence; (iii) The next step is to find the eigenvectors and eigenvalues of the covariance matrix, the eigenvectors represent the principal components, which are the directions in the original feature space that capture the most variance in the data, the eigenvalues indicate the amount of variance explained by each principal component; (iv) The principal components are ranked based on their corresponding eigenvalues, the components with higher eigenvalues explain more variance in the data. A decision is made on how many principal components to retain; (v) The selected principal components are used as axes to create a new lower-dimensional feature space. The original data is projected onto this space to obtain the transformed data representation.

The quadrotor data-set is composed by twelve "measure" outputs: the first (rates) and second (accelerations) derivatives of the quadrotor states $[\dot{x}, \ddot{x}, \dot{y}, \ddot{y}, \dot{z}, \ddot{z}, \dot{\phi}, \ddot{\phi}, \dot{\theta}, \ddot{\theta}, \dot{\psi}, \ddot{\psi}]$. It is important to highlight that this data was extracted and stored during a ΔT time after the fault injection at the simulation. The output stored data corresponds to all possible input combinations as: (i) type of trajectory (quad, circular), fault severity (varying from 5-25%) and fault motor (from 0 to 4).

According to Baskaya *et al.* (2017), the idea behind Support Vector Machine is to find an optimal hyperplane that will linearly separate the classes. This is achieved with the introduction of maximum margin concept which is the distance in between the boundaries when they are extended until hitting the first data point. The points closest to the hyperplane (decision boundary) are called the support vectors and are the representatives of the data sets to be used for the decision process (see Figure 4).

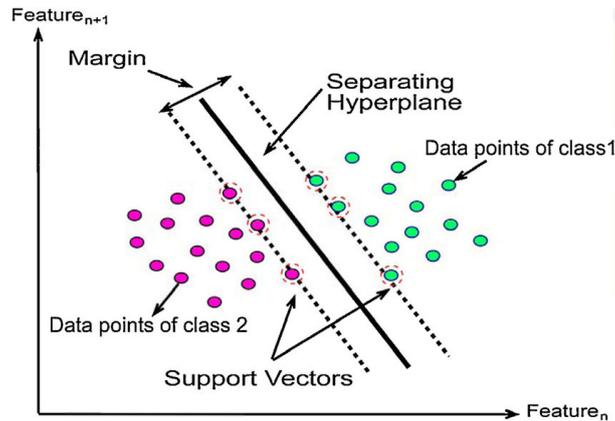


Figure 4. Linear support vector machine example (Sarp and Ozcelik, 2017)

A multi-class SVM is an extension of the binary SVM to handle classification problems with more than two classes. It is a popular machine learning algorithm used for multi-class classification tasks. The main approach to implementing a multi-class SVM is to combine multiple binary SVM classifiers. There are two common approaches: (i) In the One-vs-One (OVO) a binary SVM classifier is trained for each pair of classes in the data-set. For N classes, $N(N - 1)/2$ classifiers are trained. During classification, each classifier predicts the class between the two associated classes. The class that receives the most votes across all classifiers is assigned to the input data point. (ii) In One-vs-All (OVA) or One-vs-Rest (OVR), a binary SVM classifier is trained for each class in the data-set. For N classes, N classifiers are trained. During classification, each classifier predicts whether the input data point belongs to its associated class or not. The class associated with the classifier that produces the highest confidence score is assigned to the input data point.

In the SVM algorithm configuration, the chosen parameters are as follows: Test Size is set to 20%, determining the portion of the dataset randomly selected for evaluating model performance. Regularization Parameter (C) is set to 1, striking a moderate balance between minimizing training error and controlling weights size to avoid overfitting or underfitting. The selected Kernel is linear, assuming data separation by a straight line or hyperplane in the original feature space. Gamma is set to "auto," allowing the algorithm to calculate an appropriate value based on training data for non-linear kernels like RBF. Shrinking is enabled, reducing support vectors during training to enhance computational efficiency for large datasets while maintaining accuracy.

The corresponding PCA features map of the classified data is showed at Figure 5, this figure shows the resulted most

significant elements for a mapped feature space from twelve dimensional feature vector (6 rates and 6 accelerations) to two. It can be easily noted that Multi-class SVM method was able to classify with 99.5% of accuracy the motor which failed using the data collected during the 3 seconds after the fault occurred.

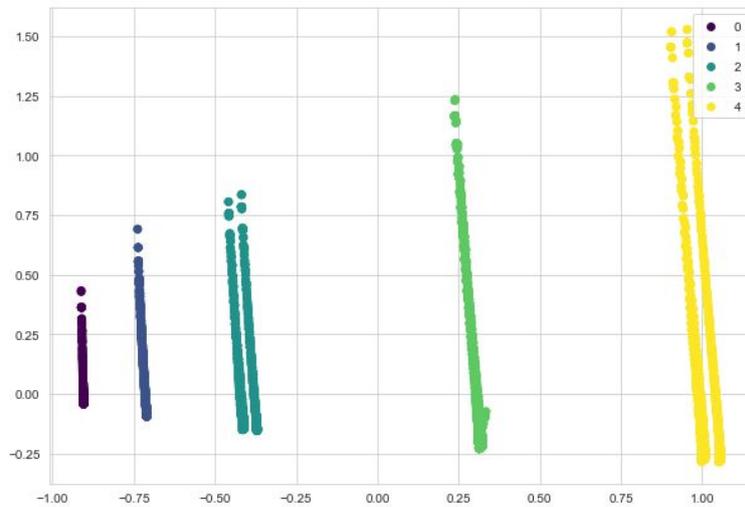


Figure 5. PCA of faulty motors data of a quadrotor in reduced dimensional feature space (for $\Delta T = 3s$)

In order to assess the influence of data collection time on the SVM classification accuracy, different size of sample points, corresponding to different times were stored and used to train the SVM model. The results are showed at Figure 6. It is possible to note that is necessary, at least, 1.5 seconds after the fault occurrence until the precision of the method reaches 90% of accuracy. Figures 7 and 8 shows, respectively, the transition of data classification between $\Delta T = 0.5s$ and $\Delta T = 2s$ revealing the impossibility of the method being able to identify engine failures in less than 0.5 seconds.

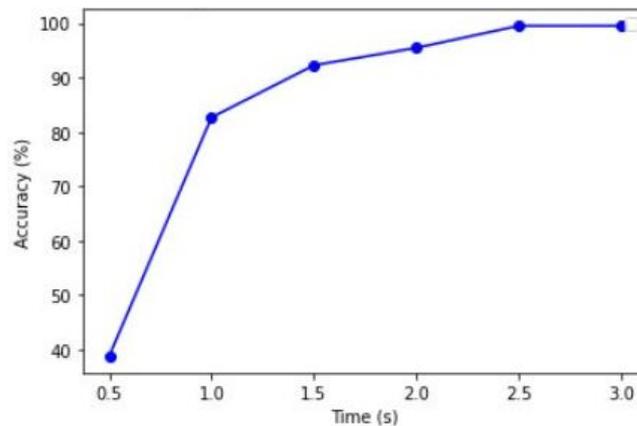


Figure 6. Effect of data collection time on SVM classification accuracy

4. Conclusions

In this work, Multi-class Support Vector Machine (SVM) method and Principle Component Analysis (PCA) were used, respectively, to classify/identify and reduce the data feature space dimension of a quadcopter simulator data-set composed by translational and linear rates and accelerations, for different combinations of fault severity in its motors. The SVM training was held offline due to the need of labeled data using as a key parameter the data collection time after the fault occurrence. The results show that for simulated measurements, SVM gives very accurate results on the classification of hard-over fault on the motors. It was verified the influence of the data collection time which showed to be crucial to the classification accuracy, this is very important for real-time FDD systems because the shorter the time used to identify the fault, the shorter the time for the controller to react allowing the AFTC to be efficient avoiding the quadcopter crash.

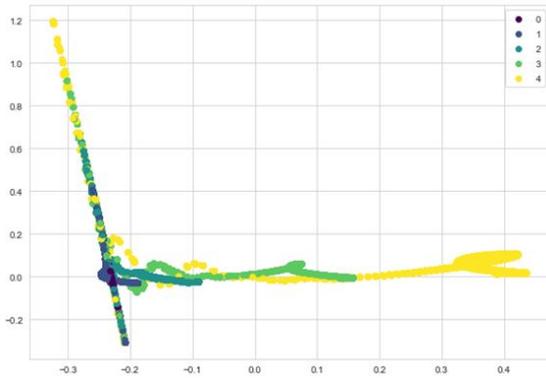


Figure 7. PCA of faulty motors data of a quadrotor in reduced dimensional feature space (for $\Delta T = 0.5s$)

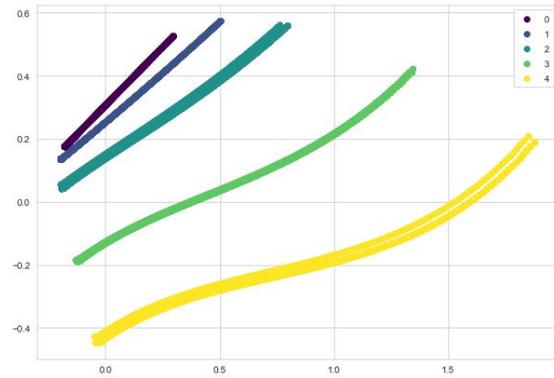


Figure 8. PCA of faulty motors data of a quadrotor in reduced dimensional feature space (for $\Delta T = 2s$)

5. ACKNOWLEDGEMENTS

This work has been supported by the State Funding Agency of Minas Gerais, Brazil (FAPEMIG), Grant Number: APQ-01656-21.

6. REFERENCES

- Baskaya, E., Bronz, M. and Delahaye, D., 2017. "Fault detection & diagnosis for small uavs via machine learning". In *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC)*. IEEE, pp. 1–6.
- Bondyra, A., Gasior, P., Gardecki, S. and Kasiński, A., 2017. "Fault diagnosis and condition monitoring of uav rotor using signal processing". In *2017 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*. IEEE, pp. 233–238.
- Bronz, M., Baskaya, E., Delahaye, D. and Puechmore, S., 2020. "Real-time fault detection on small fixed-wing uavs using machine learning". In *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*. IEEE, pp. 1–10.
- Budiharto, W., Irwansyah, E., Suroso, J.S., Chowanda, A., Ngarianto, H. and Gunawan, A.A.S., 2021. "Mapping and 3d modelling using quadrotor drone and gis software". *Journal of Big Data*, Vol. 8, pp. 1–12.
- Diaz-Mendez, Y., de Jesus, L.D., de Sousa, M.S., Cunha, S.S. and Ramos, A.B., 2022. "Conditional integrator sliding mode control of an unmanned quadrotor helicopter". *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, Vol. 236, No. 3, pp. 458–472.
- Hafeez, A., Husain, M.A., Singh, S., Chauhan, A., Khan, M.T., Kumar, N., Chauhan, A. and Soni, S., 2022. "Implementation of drone technology for farm monitoring & pesticide spraying: A review". *Information processing in Agriculture*.
- Hamadi, H., 2020. *Fault-tolerant control of a multirotor unmanned aerial vehicle under hardware and software failures*. Ph.D. thesis, Compiègne.
- Labbadi, M. and Cherkaoui, M., 2019. "Robust integral terminal sliding mode control for quadrotor uav with external disturbances". *International Journal of Aerospace Engineering*, Vol. 2019.
- Labbadi, M., Cherkaoui, M., Guisser, M. et al., 2018. "Modeling and robust integral sliding mode control for a quadrotor unmanned aerial vehicle". In *2018 6th International Renewable and Sustainable Energy Conference (IRSEC)*. IEEE, pp. 1–6.
- Li, M., Li, G. and Zhong, M., 2016. "A data driven fault detection and isolation scheme for uav flight control system". In *2016 35th Chinese Control Conference (CCC)*. IEEE, pp. 6778–6783.
- Meivel, S. and Maheswari, S., 2021. "Remote sensing analysis of agricultural drone". *Journal of the Indian Society of Remote Sensing*, Vol. 49, pp. 689–701.
- Puchalski, R. and Giernacki, W., 2022. "Uav fault detection methods, state-of-the-art". *Drones*, Vol. 6, No. 11, p. 330.
- Sadhu, V., Anjum, K. and Pompili, D., 2023. "On-board deep-learning-based unmanned aerial vehicle fault cause detection and classification via fpgas". *IEEE Transactions on Robotics*.
- Saied, M., Lussier, B., Fantoni, I., Shraim, H. and Francis, C., 2017. "Fault diagnosis and fault-tolerant control of an octorotor uav using motors speeds measurements". *IFAC-PapersOnLine*, Vol. 50, No. 1, pp. 5263–5268.
- Sarp, G. and Ozcelik, M., 2017. "Water body extraction and change detection using time series: A case study of lake burdur, turkey". *Journal of Taibah University for Science*, Vol. 11, No. 3, pp. 381–391.
- Saufi, S.R., Ahmad, Z.A.B., Leong, M.S. and Lim, M.H., 2019. "Challenges and opportunities of deep learning models for machinery fault detection and diagnosis: A review". *Ieee Access*, Vol. 7, pp. 122644–122662.
- Sindhwani, V., Sidahmed, H., Choromanski, K. and Jones, B., 2020. "Unsupervised anomaly detection for self-flying

- delivery drones”. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 186–192.
- Wang, B., Zhang, W., Zhang, L. and Zhang, Y., 2020. “Adaptive fault-tolerant control of a quadrotor helicopter based on sliding mode control and radial basis function neural network”. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, pp. 832–838.
- Xiao, M., Liang, J., Ji, L., Sun, Z. and Li, Z., 2022. “Aerial photography trajectory-tracking controller design for quadrotor uav”. *Measurement and Control*, Vol. 55, No. 7-8, pp. 738–745.
- Yang, Z., Yu, X., Dedman, S., Rosso, M., Zhu, J., Yang, J., Xia, Y., Tian, Y., Zhang, G. and Wang, J., 2022. “Uav remote sensing applications in marine monitoring: Knowledge visualization and review”. *Science of The Total Environment*, Vol. 838, p. 155939.
- Zhang, X., Zhao, Z., Wang, Z. and Wang, X., 2021. “Fault detection and identification method for quadcopter based on airframe vibration signals”. *Sensors*, Vol. 21, No. 2, p. 581.

7. RESPONSIBILITY NOTICE

The authors are solely responsible for the printed material included in this paper.