Unsupervised Anomaly Detection for Safe Robot Operations

Pratik Somaiya, Marc Hanheide and Grzegorz Cielniak
Lincoln Centre for Autonomous Systems
University of Lincoln, UK
{psomaiya,mhanheide,gcielniak}@lincoln.ac.uk

https://doi.org/10.31256/Wg7Ap8J

***Abstract*—Faults in robot operations are risky, particularly when robots are operating in the same environment as humans. Early detection of such faults is necessary to prevent further escalation and endangering human life. However, due to sensor noise and unforeseen faults in robots, creating a model for fault prediction is difficult. Existing supervised data-driven approaches rely on large amounts of labelled data for detecting anomalies, which is impractical in real applications. In this paper, we present an unsupervised machine learning approach for this purpose, which requires only data corresponding to the normal operation of the robot. We demonstrate how to fuse multi-modal information from robot motion sensors and evaluate the proposed framework in multiple scenarios collected from a real mobile robot.**

***Index Terms*—anomaly detection, one-class SVM, safety**

I. Introduction

Recently robots have started replacing humans in areas where the jobs are mostly dull, repetitive or dangerous for humans. There are many examples in areas such as agriculture, tourism, logistics and transport where the robots have either fully replaced or they are accompanying humans. One of the most important concerns of operating robots in human-environment is safety. Usually, robots use sensor data to detect the presence of any kind of object or the human but the noise in the sensors or its malfunctioning can cause a disaster. Detection of these kinds of faults at the earliest is very important before causing serious damage. In the real world, it is not feasible to foresee all kinds of possible faults and therefore these can not be modelled easily. Hence, a data-driven approach to detect these kinds of anomalies is required. We propose to use an unsupervised technique which requires only data corresponding to the normal operation of the robot - namely one-class support vector machine (OCSVM). The technique was used in [1] for collision detection and collision point localisation in a humanoid which can help the remote operator to stop the robot in case of an emergency. In [2] the authors used an isolation forest-based anomaly detection method to detect the anomalous behaviour in Unmanned Aerial Vehicles (UAVs). The contribution of our paper is to use OCSVM for anomaly detection in the operation of a mobile robot based on motion data coming from the robot's motion sensors. Furthermore, we also evaluate the proposed framework in multiple real-world scenarios and present the results based on data collected from a real mobile robot.

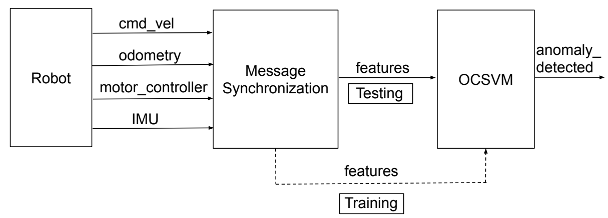

Fig. 1. An overview of the proposed data-driven anomaly detection system.

II. Methodology

A. Overview

The general system overview is presented in Fig. 1. The sensor data coming from the robot include odometry, relative motor power and speed for each wheel, linear acceleration from the IMU and issued command velocity. The frequency of received sensor messages varies and therefore in the first instance, these are synchronised. We up-sample all sensor message to 100 Hz. The synchronised messages form a feature vector of 11 values for each time instance which is an input to the one-class SVM classifier.

B. One-class SVM

OCSVM was first proposed by Schölkopf et al. [3] as an extension to the SVM. The method does not require the labelled data from two classes and can be trained using the data from one class only. OCSVM uses a kernel function $k(\mathbf{x}_i, \mathbf{x}_j)$ to map the features $\mathbf{x}$ into a high-dimensional space $\phi(\cdot)$ where it finds a hyper-plane $\mathbf{w} \cdot \phi(\mathbf{x}) - \rho = 0$ separating most of the data from the origin. This is achieved by solving the following quadratic program that maximises the distance between the hyper-plane and the origin:

$$\begin{gathered}\min_{\mathbf{w}\in F, \xi\in\mathbb{R}^\ell, \rho\in\mathbb{R}} \quad \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{v\ell}\sum_i \xi_i - \rho \\ \text{subject to } \left(\mathbf{w}\cdot\phi\left(\mathbf{x}_i\right)\right) \geq \rho - \xi_i, \quad \xi_i \geq 0\end{gathered} \tag{1}$$

where $\nu \in (0, 1]$ is an upper bound on the fraction of outliers and a lower bound on the number of training examples used as support vectors, ξ_i are slack variables and ρ is bias. By using Lagrange techniques, the decision function can be given by, $f(\mathbf{x}) = \operatorname{sgn}\left(\sum_i \alpha_i k\left(\mathbf{x}_i, \mathbf{x}\right) - \rho\right)$, where $i = 1, \ldots, \ell$ and α_i are Lagrange multipliers. An incoming datum $\mathbf{x}_n$ is determined as the anomalous if $f(\mathbf{x}_n) < 0$.

III. Experiments

A. Experimental Setup

To test the feasibility of the proposed system, we have designed a set of experiments with a real outdoor mobile robot Thorvald [4]. The robot is equipped with wheel encoders, motor controller and IMU. The data collection was performed by driving the robot manually using a joystick and issuing forward or backward command velocities. For validation and evaluation, we have also devised a set of anomalous situations by pushing the robot and recording the ground-truth using a joystick. For the normal operation, we ran the robot at fixed velocities of 0.15, 0.30 and 0.50 m/s (see Table I, dataset 1-3). Dataset 4 was created by merging sets 1-3 to represent a mix of normal behaviours. In the same way, datasets 5-10 represent examples with anomalous cases: sets 5-6, 7-8 and 9-10 correspond to speeds of 0.15, 0.30 and 0.50 m/s respectively.

TABLE I
DATA COLLECTED

number	Samples	Target Class	Anomalies
1	51091	51091	0 (0.00%)
2	19045	19045	0 (0.00%)
3	11037	11037	0 (0.00%)
4	81173	81173	0 (0.00%)
5	18156	17860	296 (1.63%)
6	11940	11681	259 (2.17%)
7	10092	9792	300 (2.97%)
8	7816	7632	184 (2.35%)
9	5508	5455	53 (0.96%)
10	5096	4978	118 (2.32%)

B. Data pre-processing

In the feature vector, some of the features have a higher order of variance than others and they might not allow the classifier to learn from other features as expected. To avoid that, we used standard scaler which scales each feature of the training data such that the mean of each feature is zero and the variance is unit. Later, these mean and variance are used to transform the test data.

C. Hyperparameter Selection

We used radial basis function (RBF) kernel for OCSVM thus the number of hyperparameters becomes two: 1) RBF kernel coefficient gamma (γ) and 2) nu (ν). The performance of OCSVM highly depends on these hyperparameters. The value of ν was set to 0.0001 based on experiments while γ was calculated with the equation, $\gamma = 1/(n * \widehat{\mathrm{Var}}(X))$, where n is the number of features, $\widehat{\mathrm{Var}}$ is the variance and X is the training data.

D. Evaluation Metrics

As the datasets are highly imbalanced, Cohen's Kappa coefficient (κ) and specificity (S) were used to evaluate the performance of the classifier. Cohen's Kappa gives the chance agreement between the observational accuracy and the expected accuracy [5] while specificity measures a classifier's ability to identify the anomalous data.

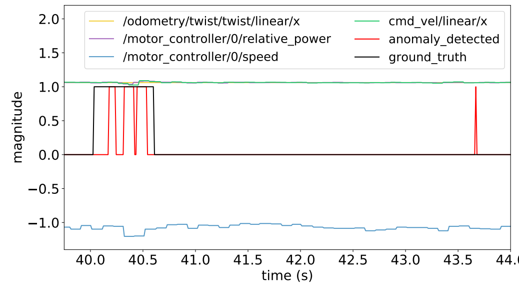

Fig. 2. Anomaly detection: true detection (left) and false detection (right)

TABLE II
OCSVM PERFORMANCE EVALUATION

		Test					
	Velocity	0.15		0.30		0.50	
		S	κ	S	κ	S	κ
Train	0.15	0.32	0.38	1.00	0.01	1.00	0.01
	0.30	0.01	0.02	0.31	0.39	1.00	0.01
	0.50	0.01	0.02	0.04	0.08	0.33	0.33
	0.15+0.30+0.50	0.13	0.22	0.07	0.12	0.38	0.3

E. Results

We combined the datasets with the same velocity from Table I and reported the results for the entire dataset in Table II. In the latter, we can see that OCSVM performed better when the training and testing data have the same magnitude of velocity. Further, when the velocity in the training was higher than the testing dataset, OCSVM missed anomalies and when it was lower, OCSVM classified most of the samples as anomalies. It is because as the velocity increases the variance of the features increase and as mentioned in III-C, γ was chosen based on the variance of the training data. For example, when the velocity in the training was lower than the testing dataset, specificity (S) was 1.00 but the value of κ was 0.01, which suggests that the performance of the classifier was poor. Finally, when the model was trained with dataset 4, it was able to predict some anomalies in the case of 0.15 and 0.50 m/s. Fig. 2 shows an example case of the anomaly detection system with the output, ground truth and some of the features.

IV. Conclusion and Future work

In this paper, we proposed OCSVM based anomaly detection method which uses multi-modal data fusion to detect the anomalous operation of a robot in human accompanied environments. We evaluated our approach in multiple real-world scenarios.

In future, we are planning to understand more about the temporal aspect of the data which might help the classifier to learn better and potentially improve the classification performance. In addition to that, we are focusing on feature selection approaches for one-class classification.

References

[1] Narukawa, Kaname, Takahide Yoshiike, Kenta Tanaka, and Mitsuhide Kuroda. ”Real-time collision detection based on one class SVM for safe movement of humanoid robot.” In 2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids), pp. 791-796. IEEE, 2017.

[2] Khan, Samir, Chun Fui Liew, Takehisa Yairi, and Richard McWilliam. ”Unsupervised anomaly detection in unmanned aerial vehicles.” Applied Soft Computing 83 (2019): 105650.

[3] Schölkopf, Bernhard, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. ”Estimating the support of a high-dimensional distribution.” Neural computation 13, no. 7 (2001): 1443-1471.

[4] Grimstad, Lars, and Pål Johan From. ”The Thorvald II agricultural robotic system.” Robotics 6, no. 4 (2017): 24.

[5] Cohen, Jacob. ”A coefficient of agreement for nominal scales.” Educational and psychological measurement 20, no. 1 (1960): 37-46.