IEEE Second International Conference on Data Stream Mining & Processing
August 21-25, 2018, Lviv, Ukraine

MEMS-based Inertial Sensor Signals and Machine Learning Methods for Classifying Robot Motion

Igor Nevlydov
Department of Computer-Integrated Technologies, Automation and Mechatronics
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
igor.nevliudov@nure.ua

Oleksandr Filipenko
Department of Computer-Integrated Technologies, Automation and Mechatronics
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
oleksandr.filipenko@nure.ua

Mariya Volkova
Department of Computer-Integrated Technologies, Automation and Mechatronics
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
mariia.volkova@nure.ua

Ganna Ponomaryova
Department of Computer-Integrated Technologies, Automation and Mechatronics
Kharkiv National University of Radio Electronics
Kharkiv, Ukraine
ganna.ponomaryova@nure.ua

***Abstract*—Robot state classification using machine-learning methods and MEMS sensors data is proposed in the paper. An experiment was performed with a three-axis MEMS gyroscope rigidly fixed to the robot body. In it we investigated the possibilities of various machine-learning methods for solving classification task.**

Keywords—robot, MEMS, classification, control

I. INTRODUCTION

MEMS sensors play a major role in the robotics and mechatronics due to their miniature size, low cost and sensitivity. The use of these sensors opens possibilities of classification features of the robot movement, balance control system [1-2].

Development of our project "PromoRobot" is caused by the need for high-quality information support, promotion (promotion) of services in places of mass presence of people, including airports, railway stations, business centers, hotels, libraries, exhibitions, educational institutions, government institutions, etc. and takes into account the public interest in robotics, new information technologies, artificial intelligence tools.

Generally, the "PromoRobot" control system corresponds to the concept of an intelligent robotic agent with a feedback control. But in some cases, the data about the robotics system is not enough to correctly work out the task. First of all, it is the task of moving on complex surfaces: up or down a slope, uneven surfaces, etc.

The solution of the task should be to create an algorithm for classifying such states of the robot that corresponded to these complexities. This will allow to take into account these features in the autonomous robot control system.

II. EASE OF USE

A. Robot Specifications

The robotic platform has a two-wheeled chassis whose elements are shown in Fig. 1. For our experiments, we use module MPU-9265. The MPU-9265 devices combine a 3-axis gyroscope, 3-axis accelerometer and 3-axis compass in the same chip together with an onboard Digital Motion Processor capable of processing the complex MotionFusion algorithms [2]. The output signals of the accelerometers (Ax, Ay, Az) and the gyros (wx, wy, wz) are converted directly by an Analog to Digital Converters inside the microcontroller ATMEL ATmega32. This microcontroller has 8 channels of 10-bit Analog to Digital Converters, a USART (Universal Asynchronous serial Receiver and Transmitter) port and a sampling rate - 200 Hz.

Fig. 1. "PromoRobot" project

B. Classification task

Formally, the problem of robot states classification can be represented as follows: let X be the set of data on the work state obtained from the MEMS gyroscope (along the Ox, Oy, Oz axes). Y is a finite set of classes (8 states in the work): calmness - state "0"; forward motion on the slope - state "1"; backwards motion from the slope - state "2"; backwards motion on the slope - state "3"; forward motion from the slope - state "4"; forward motion - state "5"; rotation counter-clockwise - state "6"; clockwise rotation - state "7".

There is an unknown target addiction – reflection $y^*:X \rightarrow Y$. The value is known only on known states of the robot on the training set $X^m=\{(x_1,y_1),\ldots,(x_m,y_m)\}$.

It is necessary to develop an algorithm $a:X\rightarrow Y$ for classifying the robot's state Y according to sensor's reading $x \in X$ in real time-domain. In our case, the set of classes is $Y=\{0,1,2,3,4,5,6,7\}$.

A number of experiments were carried out with a three-axis MEMS gyroscope rigidly fixed to the robot body. Gyroscope allows you to track the robot's precise execution of the prescribed actions, possible features of its movement. Every action is a certain state.

III. Results & Analysis

The results of measurements are shown in Fig.2-4 and visualize the sensors measure in the process of robot moving. To find the clustering algorithm which is support real-time work we consider three axis of the gyroscope.

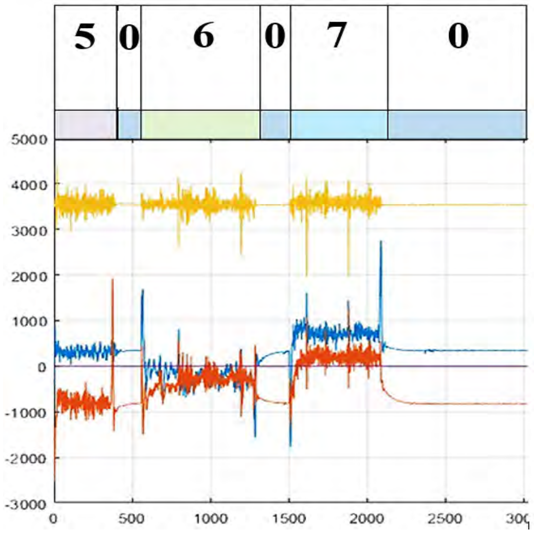

Fig. 2. " Procedure "moving forward - stop - rotation counterclockwise - stop - clockwise rotation - stop"

Figure 2 illustrates the gyroscope readings captured by the three axes. A detail visual analysis of figure 2 reveals that the relative magnitudes of the sub readings of the gyroscope could be used for event classification. For example from point 1 to point 400, robot was moving forward. And from 401 to 520 robot standing – this is indicated by the very low gyroscope values.

However it is also clear that coming up with manually defined thresholds for three sensor readings that will allow the classification of the seven events will be still a complex task. Further the raw signals captured by the sensors are noisy and will therefore have to be cleaned prior to further analysis.

The robot motion activity recognition system has to decide which of the seven events have effectively caused the measured values of the features based on real signals, which are fed from sensors. This is a general classification problem that can be dealt with by a large range of algorithms, such as logics, k-nearest Neighbor approaches, Support Vector Machines (SVMs), Artificial Neural networks [3,4], Decision Trees or Bayesian Techniques [5]. Our work focused on finding an algorithm which are realize a classification of robot motion in real-time domain.

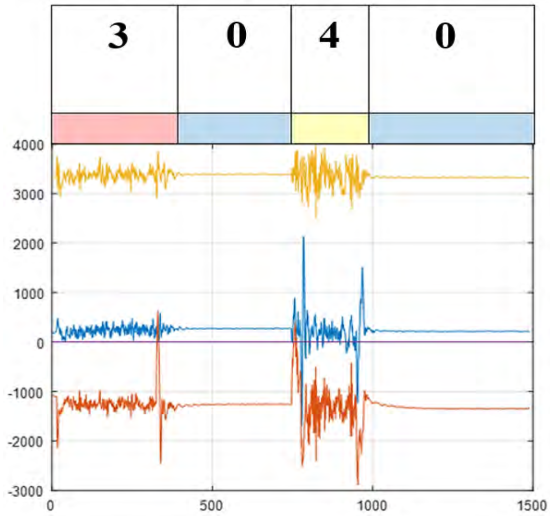

Fig. 3. Procedure "backwards motion on the slope – stop – forward motion from the slope, stop»

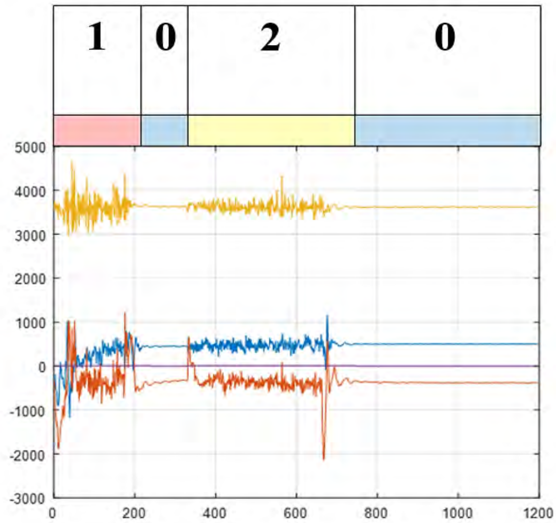

Fig. 4. Procedure "forward motion on the slope – stop – backwards motion from the slope - stop"

The thee time-domain features (three axis of the gyroscope) were used to train machine learning algorithm. The mean, standard deviation, minimum, and maximum of signal in the running window also were used as features. But using more features did not improve the quality of the classification

We examined the performance of some supervised learning algorithms and singled out most appropriate among them: Support Vector Machines (Linear SVM) [3], k-nearest neighbors algorithm (Medium KNN, Weighted KNN) [4], Boosting algorithm [6], Classification Trees (Simple Tree, Medium Tree) and Ensemble (Bagged trees) [7, 8].

Gyroscope signals from robot are sufficient to classification. Weighted KNN and Bagged trees performed slightly better than other three algorithms (the classification accuracy about 89%).

The evaluation of the quality of the trained models was carried out by such criteria as accuracy, confusion matrix, Parallel Coordinates plot ROC Curve. Results of model test can be classified according to sensitivity and specificity. Sensitivity is the ability to detect an abnormality, while specificity is the ability to distinguish an abnormality by type. Diagnostic test should also identify the frequency of false positive (FP) and false negative (FN) or true positive (TP) and true negative (TN).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{TP + FP + TN + FN}.$$

Table I presents the results of the accuracy calculation for various methods.

TABLE I. RESULTS OF THE STUDY BY VARIOUS METHODS

Metod	**Accuracy,%**
Linear SVM	66.6
Simple Tree	69.7
Medium Tree	76.7
Medium KNN	88,1
Weighted KNN	88.7
Boosted Tree	84,1
Bagged Trees	89,6

The data in an ROC analysis is used to decide which traits produce the greatest separation of two probability curves which show the likelihood of choosing wrong or right states. The standard way to interpret the data from an ROC test is to draw a ROC-curve and then measured the area under the curve. The test with the greatest area is the most accurate. The best results were shown by the Weighted KNN method.

In Fig. 5 and Fig. 6 graphs of confusion matrix are provided, which help to identify areas in which the classifier works poorly. In the first case, the lines show the current state of work, and the column shows the cjjnd classes. As can be seen from Fig. 5 the classifier works worse for determining the class "2" (backwards motion from the slope) and class "3" (backwards motion on the slope).

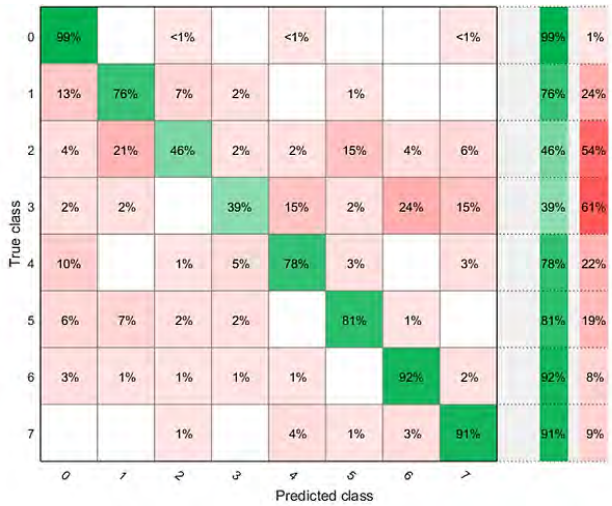

Fig. 5. Confusion matrix Weighted KNN

In fig. 3.9 in the Confusion matrix are shown false classifier actions, under the matrix green, the correct prediction is shown in each class, and the false values are shown in red. The marker on the Fig.7 shows the performance of the currently selected classifier. For our classifier false positive rate (FPR) of 0.05 indicates that the current classifier assigns 5% of the observations incorrectly to the positive class. A true positive rate of 0.99 indicates that the current classifier assigns 99% of the observations correctly to the positive class.

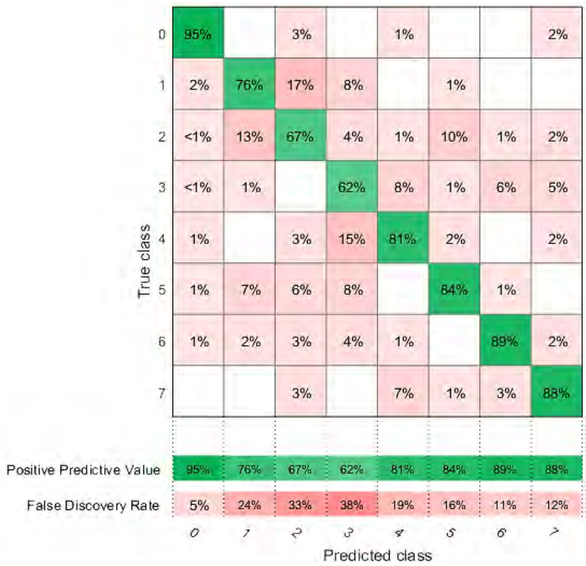

Fig. 6. Confusion matrix Weighted KNN

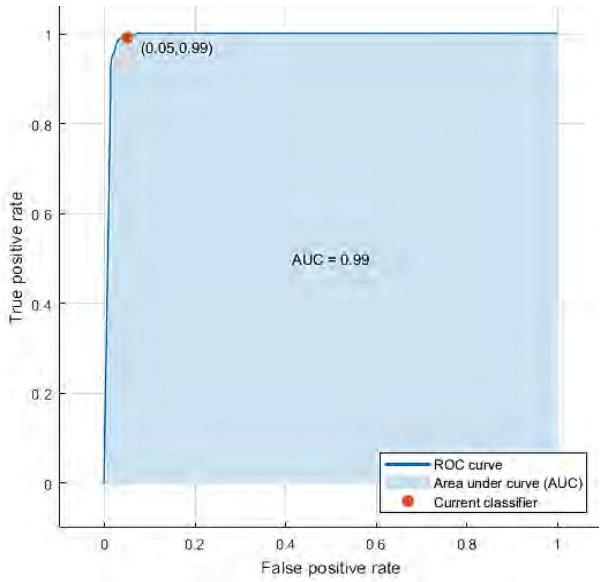

Fig. 7. ROC Curve

IV. CONCLUSIONS

The paper is devoted to solving tasks of classifying robot motion by machine learning methods. The experiments was performed based on real signals are fed from MEMS sensors on the robot board in real-time domain.

The analyzze of classificator learning results showed the possibility of using k-nearest neighbors algorithm to classify the state of a robot with 88% accuracy. An algorithm is developed based on measurements of a three-axis gyro without any pre calculations.

We are currently working on integrating a number of other in-built sensors and algorithms in the above process, allowing more detailed and complex scenarios to be identified accurately. Further development of the proposed approach can be carried out in the direction of implement the classificator in decision-making system of robot on Asus Tinker Board.

TABLE II. RESULTS OF CLASSIFIER WORK

State of robot	**Weighted KNN**		**Bagged trees**	
	Positive Predictive Value	***True Positive Rate***	***Positive Predictive Value***	***True Positive Rate***
calmness	95%	99%	96%	98%
forward motion on the slope	76%	76%	84%	84%
backwards motion from the slope	67%	46%	64%	56%
backwards motion on the slope	62%	39%	50%	34%
forward motion from the slope	81%	78%	81%	81%
forward motion	84%	81%	80%	82%
rotation counter-clockwise	89%	92%	91%	93%
clockwise rotation	88%	91%	91%	92%

REFERENCES

[1] F. Coito, A. Eleutério, S. Valtchev, and F. Coito, “Tracking a Mobile Robot Position Using Vision and Inertial Sensor,” 5th IFIP WG 5.5/SOCOLNET DoCEIS 2014, Costa de Caparica, Portugal, AICT-423, Springer, pp. 201-20, April 7-9, 2014.

[2] Inven Sense. MPU-9250 Product Specification, Revision 1.1, InvenSense Inc. [ONLINE] Available at: https://www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01- v1.1.pdf. [Accessed 26/12/2017].

[3] K. Noda, Y. Hashimoto, Y. Tanaka, and Ichiro Shimoyama, “MEMS on robot applications,” TRANSDUCERS 2009 International Solid-State Sensors, Actuators and Microsystems Conference [ONLINE] Available at: https://ieeexplore.ieee.org/document/5285608//

[4] K. Frank, J. Vera Nadales, P. Robertson, and M. Angermann, “Reliable Real-Time Recognition of Motion Related Human Activities Using MEMS Inertial Sensors,” [ONLINE] Available at: https://pdfs.semanticscholar.org/89ca/d05d53302b4b8c465f 3fc9b9eec924aff567.pdf?_ga=2.160054457.507168460.1529006792-191559443.1528161624/

[5] S. B. Kotsiantis, “Supervised machine learning: A review of classification techniques,” Informatica, vol. 31, pp. 249–268, 2007.

[6] C. Cortes, and V. Vapnik, “Support-vector network,” Machine Learning, vol. 20, issue 3, pp. 273-297, Sept. 1995. [Online]. Available: https://doi.org/10.1023/A:1022627411411.

[7] J. Zhu, S. Rosset, H. Zou, and T. Hastie, “Multiclass AdaBoost,” Technical report, Stanford Univ, 2005. Available at http://www-stat.stanford.edu/ hastie/Papers/ samme.pdf.

[8] Y. Freund, and R. E. Schapire, “Experiments with a New Boosting Algorithm,” in L.Saitta, ed., ‘Proceedings of the Thirteenth International Conference on Machine Learning (ICML’96)’, Morgan Kaufmann,1995, pp. 148–156.

[9] E. Gatnar, “Fusion of Multiple Statistical Classifiers”, in C. Preisach, H. Burkhardt, L. Schmidt-Thieme and R. Decker, eds, “Data Analysis, Machine Learning and Applications,” Studies in Classification, Data Analysis, and Knowledge Organization, Springer, Berlin/Heidelberg, 2008, pp. 19–27.