Stability Prediction Of Quadruped Robot Movement Using Classification Methods And Principal Component Analysis

Mohammad DIVANDARI [1], *Delaram GHABI*[1], *Abdol Aziz KALTEH* [1]

[1]Department of Electrical Engineering, Aliabad Katoul Branch, Islamic Azad University, Aliabad Katoul, Iran

m.divandari@aliabadiau.ac.ir,delaram.ghabi@yahoo.com, kalte@aliabadiau.ac.ir

DOI: 10.15598/aeee.v21i4.5215

Article history: Received May 14, 2023; Revised Sep 17, 2023; Accepted Oct 20, 2023; Published Dec 31, 2023.

Abstract. *This paper introduces a novel technique for predicting the stability of quadruped robot locomotion using a central pattern generator (CPG). The proposed method utilizes classification methods and principal component analysis (PCA) to predict stability. The objective of this study is to anticipate the stability or instability of robot movement by modifying controlling parameters, referred to as features. The simulations of robot locomotion are conducted in MATLAB/SIMULINK®, generating a dataset of 82 observations with different parameters. Machine learning (ML) techniques are then applied, using classification methods and PCA, to determine the stability condition. Six classification methods, including K-nearest neighbors (KNN), support vector classifier (SVC), Gaussian Naïve Bayes (GaussianNB), logistic regression (LR), decision tree (DT), and random forest (RF) are implemented using Scikit-learn, an open-source ML library in Python. The performance of these classifiers is evaluated using four metrics: precision, recall, accuracy, and F1-score. The results indicate that KNN and SVC exhibit higher metric values compared to the other classifiers, making them more effective for stability prediction.*

Keywords

Quadruped robot, stability, prediction, classification methods, principal component analysis (PCA).

1. Introduction.

Legged robot has made much attention of researchers worldwide because of exploring in the complex areas, space, rescue operation, accomplishing a task without human intervention, industrial use, etc. Among all legged robot, quadruped robots are superior due to the benefits of load capacity and balanced structure, exploring in all terrains and locomotive stability [1, 2]. In order to achieve real-time speed and stable patterns of natural quadrupedal movement like a cat, dog, horse, the developed control system, and dynamical gait generation are required [3].

Central pattern generator (CPG), a model-free control method, has become a noteworthy technique in dynamic locomotion control of multi-legged robots. It is a bionic technique in quadruped robot control. Operating as a biological nervous system, it can be constructed with a group of coupled neurons that generate rhythmic locomotion by coordinating the connection of oscillators. CPG can also modify gait transition by simple control signals. It adjusts output by high level nerve signals and external feedback. [4]. CPG output signals were first achieved by Matsuoka utilizing oscillator model adjustment [5]. This oscillator can only generate a positive signal, which is often inappropriate for engineering control purposes. On the basis of Matsuoka's CPG oscillator, Kimura et al. utilized excitatory and inhibitory model neurons to simulate the impulses of nervous system of animals and to control the gait of a quadruped robot called Tekken [6]. Moreover, the functions of the oscillators successfully were improved by proposing further reflexes to the controller and applied great walking tests of Tekken on irregular terrains.

The Wilson-Cowan neural oscillator is also a well-known CPG model to control legged robots. It describes the dynamics of interactions between population of simple excitatory and inhibitory model neurons. The Wilson-Cowan neural oscillator controller was presented by Li et al. to control the quadruped robot locomotion [7]. The rhythmic movement of one leg from the others in a quadruped robot is controlled with four Wilson-Cowan neural oscillators. It is easy to control the amplitude and period of the CPG model for generating diverse gaits, but the stability control and the real time of the oscillation system regulation require more modification.

In different methods of stability control, Liu and Chen suggested zero moment point (ZMP) model to control the locomotive stability of quadruped robot, which achieves some improvements but low flexibility and efficiency of planning and switching the gait [8]. Junmin Li et al. proposed the improved Wilson-Cowan nervous oscillator model and presented new controller algorithm based on central pattern generator-zero moment point (CPG-ZMP) to control the locomotion of a quadruped robot. The stability of robot's locomotion was improved significantly but it needs a greater improvement [9].

On the basis of intelligent control, the control methods have received considerable attentions from the majority of researchers in recent years. Some artificial intelligent algorithms are also utilized in the locomotion control methods, such as genetic algorithms, fuzzy control, deep learning and neural networks. Besides, there are some advantages to using classic and intelligent control together in different researches. First, Naghmeh Mirrashid et al. [10] presented a new super-twisting algorithm (STA) controller based on teaching-learning-based optimization (TLBO) for an upper limb rehabilitation robot. The suggested controller guaranteed finite-time convergence, reduced chattering, stability and accurate performance. Second, Hadi Hasanpour et al. [11] studied the adaptive back-stepping-based control of a two-joint arm robot and dynamic model of the robot using the Euler -Lagrange method. The results of tracking error were brought to zero and the stability of the model was guaranteed.

In [12], Ololade O. Obadina et al. presented the improved grey-box model and fuzzy logic to control real time of trajectory tracking and position of a four degree-of-freedom leader-follower robot (LFR). This approach successfully achieved a high-performing model of the LFR manipulator system and a better performance of trajectory tracking. Moreover, in [2], Qinglei Ji et al. applied a deep reinforcement technique in order to accomplish the efficient gait of a four-legged robot for different walking however some aspects require to be improved in case of traing process and additional sensors. Also, M. Naya-Varela et al. [13] utilized morphological development in neural network in order to control the locomotion of quadruped robot. An artificial neural network (ANN) was utilized to execute experimental researches to recognize the weaknesses and strengths of an ANN in navigation systems for legged robots in [14]. Four experiments showed that changing different parameters including accelerating the training process and optimization can increase the value of ANN. A new gait planning technique was proposed for a 2n-legged robot with passive-spines movement in [15]. The finite state machine (FSM) theory is introduced into the undulatory gait planning method. The accuracy of the gait planning method was proved and the results identified that the lateral locomotion of the legs can improve the weaknesses of gaits. Danilo S. Jodas [16] presented the developed control system for the navigation of an independent mobile robot. Track images were utilized to control the navigation using pre-processing them and then extracted features were submitted to a support vector machine (SVM) and an ANN to find the most appropriate track. The results determined that SVM had better performance and less execution time in the training step. In addition, in [17], Tong Li and et al. presented robot grasping system and grasp stability prediction based on flexible tactile sensor array using KNN, SVC, LR and ensemble learning methods. The model achieved accuracy 98% in grasp stability prediction.

In [18], Jiawei Chen et al. designed an adaptive omnidirectional walk for the quadruped robot applying the extended ZMP on the center of inertia to obtain the stable criterion in the rough terrain. Results illustrate that the relevant robot using the adaptive omnidirectional walk can walk on the rough terrain. Yanbin Zhang et al. [19] prepared a neural control framework based on the CPG-RBF-hyper network. By modulating the frequency of the CPG, basic motor patterns for different speeds can be generated. The radial basis function (RBF) network is a premotor network that alters the shape of the CPG patterns. The motor patterns under several walking speeds are first learned utilizing probability-based black-box optimization (PIBB) by an incremental learning method. By combining PIBB and supervised learning, the CPG-RBF-hyper network can enable a quadruped robot to perform stable and robust walking at different speeds. Takahiro Fukui et al. [20] illustrated the effectiveness of vestibular feedback to CPG employed for the movement of quadruped robots. Vestibular feedback causes autonomous gait transitions at different speeds between different types of gates.

The locomotion control of a quadruped robot requires powerful system. Machine learning techniques are the great key to anticipate whether a corresponding locomotion is stable or not.

All studies done by researchers for improvement and stability of robot locomotion in the past, were actually based on robot behavior. While, prediction of robot stability using adjusted control parameters, is highly valuable. The state-of-the-art of this article is prediction-based classification of quadruped robot stability by adjusting control parameters. In fact, main

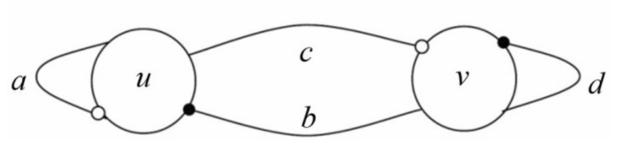

Fig. 1: Configurations of the Wilson-Cowan neural oscillators. [9], • Inhibitory connection and o excitatory connection.

contribution of this study is to guarantee the stability during locomotion before switching the gait such as walk, trot, pace and gallop. In addition, the new approach makes the robot capable of predicting stability when parameters are adjusted in real time. To achieve this, the stable and unstable locomotion of a quadruped robot is studied during trotting. In this study, 82 tests using different parameters are executed on quadruped robot that 30% of them are test data. In addition, six classifiers: K-nearest neighbors (KNN), support vector classifier (SVC), Gaussian Naïve Bayes (GaussianNB), logistic regression (LR), decision tree (DT), random forest (RF) and a dimensional reduction model based on PCA are employed to help in prediction of stability or instability.

This paper is organized as follows: Section 2. refers to the improved Wilson-Cowan nervous oscillator. Section 3. explains the proposed techniques in details. Section 4. introduces our data preprocessing. Section 5. applies pipeline method, Section 6. briefly discusses precision, recall, accuracy and f1-score plots. Section 7. discusses simulation and classification results. Finally, conclusion is addressed in Section 8. .

2. Transition of Quadruped Robot

This paper represents two typical gaits, including walk and trot that were generated by improved Wilson-Cowan nervous oscillator [9].

2.1. Improved Oscillators model

Fig. 1 shows the Wilson and Cowan's oscillator model consisting of excitatory neuron u and inhibitory neuron v with synaptic connections. Where d is the inhibitory connection gain of neuron and a is the excitatory connection gain of neuron, c is the excitatory connection gain of u to v and b is the inhibitory connection gain of v to u. The improved Wilson-Cowan nervous oscillator

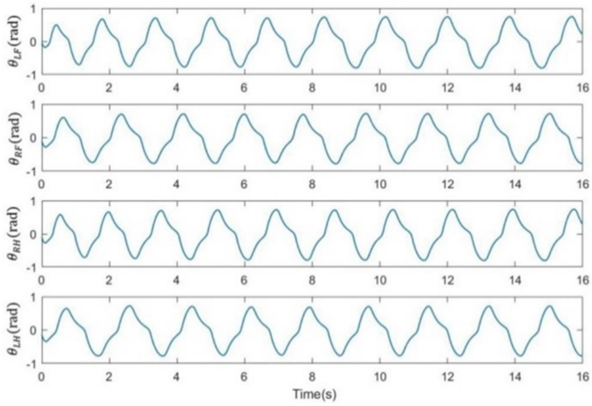

Fig. 2: The CPG control topology structure of quadruped robot (a) Walk network connection topology structure. (b) Trot network connection topology structure [9].

model was calculated as follows:

$$T_{u_i}\frac{du_i}{dt}+u_i=f_\mu\left(au_i-bv_i+\sum_{j=1}^{n}w_{ij}u_j+\sum_{k=1}^{m}g_k s_{ik}+S_{u_i}\right), \quad (1)$$

$$T_{v_i}\frac{du_i}{dt}+v_i=f_\mu\left(cu_i-dv_i+\sum_{j=1}^{n}w_{ij}v_j+\sum_{k=1}^{m}g_k s_{ik}+S_{v_i}\right), \quad (2)$$

Where $f_\mu(x) = \tanh(\mu x), i, j = 1,2,3,4, k = 1,2,3,\ldots,m$.

Also, i and j are the numbers of central neural oscillators, $W \in R4\times4$ is a matrix constructed of w_{ij} and w_{ij} is the connection weight between oscillators. External feedback of CPG control model was proposed, where g_k is coefficient of s_{ik}, s_{ik} is the reflection information and u and v are the adverse vectors of reflection coefficients. The parameters S_{u_i} and S_{v_i} are the external signals that are usually DC inputs, T_{u_i} is the rise-time constant of step input, T_{v_i} is the fatigue time constant, $f_\mu(x)$ is the coupling function, and μ is the gain of $f_\mu(x)$. The output of the corresponding CPG model was normalized as follows:

$$y^i_{out} = p(u_i - v_i), \quad (3)$$

in order to adjust the outputs of u and v, p was proposed as amplitude limiting coefficient; y_{out} controls the locomotion of corresponding leg as an output of linear synthesis.

2.2. Gait planning

Walk and trot are discussed and their gait switch is shown in this study. Walk is putting up and down each leg in turn and the phase of each leg movement

Tab. 1: Parameters of the CPG differential equations [9].

Parameters	Value
T_{u_i}, T_{v_i}	0.2
a	5.6
d	-2.4
b	5.6
c	2.4
S_{u_i}, S_{v_i}	0.02
μ	1
p	0.5
m	1
g_k	0.1

lasts a quarter cycle. Trot means that the legs move diagonally and they put up and down at the same time. The phase of two diagonal legs lasts half of cycle.

1) Weight matrixes of walk and trot

The CPG model with different weight matrixes can generate rhythmic signals by different gaits including walk and trot. The connection weight matrixes W_{walk} and W_{trot} are as follows [9]:

$$W_{walk}\begin{bmatrix} +0.0 & -0.1 & -0.1 & -0.1 \\ -0.1 & +0.0 & -0.1 & -0.1 \\ -0.1 & -0.1 & +0.0 & -0.1 \\ -0.1 & -0.1 & -0.1 & +0.0 \end{bmatrix} \tag{4}$$

$$W_{trot}\begin{bmatrix} +0.0 & -0.1 & +0.1 & -0.1 \\ -0.1 & +0.0 & -0.1 & +0.1 \\ +0.1 & -0.1 & +0.0 & -0.1 \\ -0.1 & +0.1 & -0.1 & +0.0 \end{bmatrix} \tag{5}$$

The CPG control topology structure of quadruped robot with connection weight matrixes W_{walk} and W_{trot} is shown in Fig. 2. As shown in this Figure, the movement of each leg will be controlled with other legs by three bidirectional connections.

2) Gait transition

With diverse gait matrixes, the CPG model can generate various gaits. Gaits transition can be realized by changing the connection weight matrix. The parameters in (2) and (3) are shown in Table 1.

The two gaits of quadruped robot generated by corresponding CPG are shown in Fig. 3. It is shown that the circuit can change the rhythmic patterns promptly. The x-axis is second and the unit of y-axis is radian. Gait transition begins at t = 8s.

3. Classification methods

The data classification is a supervised learning technique to predict the correct category for a given input using machine learning. A classifier organizes data into categories that make it easy to retrieve, locate and

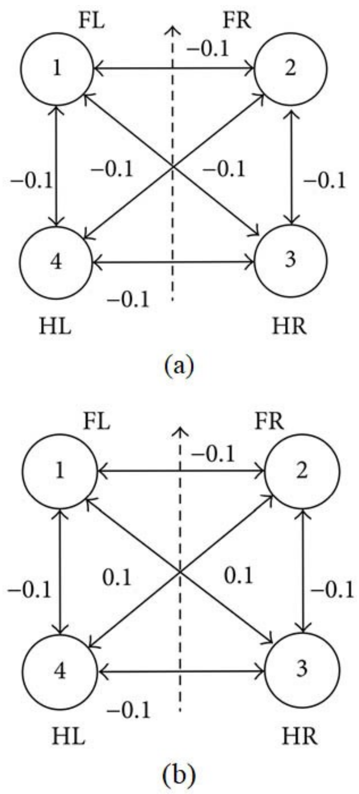

Fig. 3: Gait transition from walk to trot. (a) Left front leg. (b) Right front leg. (c) Right hind leg. (d) Left hind leg.

store for future use. In other words, the model of classification is completely trained to the train data, and then it is evaluated on test data and finally the results are measured using several metrics to know how well the corresponding classifier performed on the input data and whether it is appropriate for future prediction on new datasets or not (Appendix).

3.1. Classifiers

As mentioned before in this study, five well-known types of classifiers based on supervised learning techniques are described as follows:

1) DT

DT is a decision support tool that utilizes tree-like model of Boolean decisions. In this technique, each path starts from the root and indicates a sequence of data splitting until a Boolean result is reached at the leaf node [21]. The purpose is to anticipate the value of a target variable by learning decision derived from the data features.

2) RF

RF constructs decision trees on randomly selected samples. It anticipates from each tree and takes the average of all the predictions. Important features in a database are also selected by RF [22].

3) KNN

KNN classifier is one of the simplest learning algorithms, working off the assumption that similar points can be found near one another. KNN stores datasets at the training phase and when it receives new data, it classifies them into a class which is almost similar to the new data [23].

4) SVC

SVC maps data points to a high-dimensional space and then recognizes the optimal hyperplane that divides the data into two classes. SVM is an integrated model in order to import SVC. According to recent researches, using SVM provides proper solution for control purposes [24].

5) LR

LR is another well-understood technique. In this model a linear combination of inputs is transformed to non-linear outputs. In fact, it utilizes a sigmoid function to model a binary output variable. LR is a robust method for classification prediction [25, 26].

6) GaussianNB

Naive Bayes is considered as a popular probabilistic classifier. It relies on the Bayes theorem. NB classifier has stable efficiency [27]. A GaussianNB is a special type of NB algorithm. It also assumes that all the features have a Gaussian distribution [28].

3.2. Evaluation metrics

Several measures are utilized to evaluate how efficient the proposed classifier performs. The statistics are described as follows [29], (Appendix)

1) Accuracy

Accuracy illustrates the fraction of correct predictions of the corresponding classifier.

$$\mathrm{Accuracy} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}. \tag{6}$$

2) Recall

Recall determines the number of positive cases that the classifier correctly predicted. It also refers to the sensitivity.

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{7}$$

3) Precision

Precision is also called positive predictive value. It indicates that how many of positive findings are the real positive.

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}. \tag{8}$$

4) F1-score

F1-score incorporates recall and precision and measures the model accuracy.

$$\mathrm{F1} = 2 * \frac{\mathrm{precision} * \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}. \tag{9}$$

The corresponding dataset requires to be checked and modified before using classification methods.

4. Data preprocessing

This stage is used to process the dataset including analysing the type of features, null values, and duplicated data before classification. For achieving better results from the applied model in machine learning project the format of the data has to be in a proper manner. In order to prepare the dataset for classification, dropping unimportant columns or modifying the dataset is considered [28]. Pandas is the most widely-used library for data analysis [30]. In this study, the column of samples that is unnecessary for classification was dropped and the total number of null values was zero.

5. Pipeline method

The main key of constructing a pipeline is improved readability. It can execute a series of operations with only one call. Pipeline method performs multiple transformers in a list (name, transform). The output of each transformation is the input of the next. Eventually, pipeline would be fitted to the train data [31]. In this work, the operations applied in pipeline are a dimension reduction based on PCA and six classification methods respectively (Appendix).

5.1. PCA method

In this study, PCA is considered as the first transformation of pipeline technique. PCA is a well-known unsupervised learning technique utilized for reducing the dimensionality of a dataset by minimizing information loss. This algorithm seeks for directions of maximum variance regardless of labels [32]. The PCA method implementation is applied in six steps [33] (Appendix).

6. Classification results

In this study, trot dataset is classified using six classification methods with different PCAs. There are 82 observations including 62 train samples and 20 test samples. Each sample is achieved by CPG model running in Matlab/Simulink with five modified parameters or features. Classification algorithms perform based on the stability of quadruped robot locomotion. Therefore, the labels of unstable and stable outputs are considered as 0 and 1 respectively. The train data has thirty-one 0 and thirty-one 1. On the other hand, test data has ten outputs of each one. Each sample has five features: a, b, c, d and g1. A range of values for each feature is defined that in these ranges the trot stability of quadruped robot is guaranteed. Extremum points (minimum and maximum) of corresponding features are set in Table 2.

Tab. 2: The extremum values of features in trot dataset.

Features	Minimum	Maximum
g1	0.05	0.15
a	5.5	5.6
b	5.5	5.6
c	2.1	2.5
d	-2.1	-2.5

6.1. Accuracy

As can be observed in bar graphs of Table 3, the accuracy value of KNN and SVC is maintained 95% with all dimensions of PCA. The accuracy value of GaussianNB with PCA = 1 is 100%. It falls to 85% with PCA=2 and this value remains constant with other PCAs. The accuracy value of LR with all PCAs is 70% except with PCA = 2 that shows a drop by 10%. The accuracy of DT and RF with PCA=1 is 100% but with other PCAs it can change significantly when DT and RF methods run again. The range of changes with different numbers of PCA is between 60% to 100%. For instance, Table 3, illustrates that by one run, the accuracies of DT and RF classifiers with all numbers of PCA are 100% but reveals a drop by 10% for RF with PCA=3 and a fall by 20% for DT with PCA=4.

6.2. Recall

As shown in Table 3, there are three classification methods that maintain their recall value with all numbers of PCA. First, KNN and SVC with the value of 100% and secondly, GaussianNB with 90%. Recall value of LR is 80% with PCA = 1 but it decreases to 60% by increasing the number of PCA. Recall score of DT and RF with all numbers of PCA except PCA=1 can alter from 60% to 100% by restarting RF and DT methods. Recall score of RF and DT with PCA= 1 is 100%. In the given report of recall scores, the values of RF and DT are decreased to 60% with PCA=2. The recall values with PCA=3 rise to 100% and 70% for DT and RF respectively. PCA=4 decreases the value of DT to 60% and increases the value of RF to 100%. PCA=5 alters the values of DT and RF to 100% and 60% respectively.

6.3. Precision

According to the Table 3, the precision score of DT and RF with all numbers of PCA is 100%. This value is constant with PCA=1 but it can fluctuate (from 60% to 100%) with other numbers of PCA by restarting DT and RF methods. According to the given values, the precision scores of DT and RF is 100% with all numbers of PCA. The precision value of KNN and SVC is 90.99% with all PCAs. Precision value of LR is 66.66% in PCA=1 but it slightly decreases to 60% in PCA=2 and it rises to 75% in other PCAs. Precision value of GaussianNB is 100% with PCA = 1 and PCA=2. There is a drop in this score to 81.81% with the rest of PCAs.

6.4. F1-score

As observed in Table 3, the f1-score of KNN and SVC is 95.23% with all PCAs. F1-score of LR with PCA=1 is 72.72%. This value decreases to 60% in PCA=2 and it minimally increases to 66.66% with PCA=3 and it is maintained with other PCAs. F1-score of GaussianNB is 100% with PCA=1. It drops to 83% in PCA=2 then it slightly increases to 85.71% in PCA=3 and it maintains this value in higher PCAs. The f1-score of DT and RF is 100% with PCA=1 but several fluctuations can be seen by restarting the two classifiers with different PCAs.

According to the given figure, there is a fall to 75% in the f1-score of DT with PCA=2 and it rises to 100% with PCA=5. The f1-score of RF fluctuates from 100% to 75% with PCA=2 and increases to 95% with PCA=3, then it turns back to 100% with PCA=4.

7. Discussion

The trot dataset is classified using six classification methods in an effort to find which classifiers work efficiently in prediction of movement stability of the quadruped robot.

The parameters of different classifiers are set, including n_estimators=5 for RFC, n_neighbors =3 for KNN, kernel="rbf" for SVC and solver="liblinear" for LR. PCA utilized as a feature reduction is considered from 1 to 5. Scikit-learn, an open-source library in Python, is utilized to import classifiers, PCA and pipeline technique [34]. Confusion matrixes of six classifiers with PCA=1, are set in table 3. Each classifier is made a

Tab. 3: Results of classifiers for different PCAs.

PCA	Classifier	Accuracy score (%)	Recall score (%)	Precision score (%)	F1-score (%)
1	**DTC**	100	100	100	100
	RFC	100	100	100	100
	KNN	95	100	90.9	95.23
	SVC	95	100	90.9	95.23
	LR	70	80	66.66	72.72
	GaussianNB	100	90	100	100
2	DTC	100	60	100	75
	RFC	100	60	100	75
	KNN	95	100	90.9	95.23
	SVC	95	100	90.9	95.23
	LR	60	60	60	60
	GaussianNB	85	90	100	83
3	DTC	100	100	100	75
	RFC	90	70	100	95
	KNN	95	100	90.9	95.23
	SVC	95	100	90.9	95.23
	LR	70	60	75	66.66
	GaussianNB	85	90	81.81	85.71
4	DTC	80	60	100	75
	RFC	100	100	100	100
	KNN	95	100	90.9	95.23
	SVC	95	100	90.9	95.23
	LR	70	60	75	66.66
	GaussianNB	85	90	81.81	85.71
5	DTC	100	100	100	100
	RFC	100	60	100	100
	KNN	95	100	90.9	95.23
	SVC	95	100	90.9	95.23
	LR	70	60	75	66.66
	GaussianNB	85	90	81.81	85.71

total of 20 predictions. Confusion matrix presents the number of correct and incorrect predictions and the type of errors made by classifiers [35]. Confusion matrixes determine that the number of errors made by DT, RF and GaussianNB are zero. KNN and SVC have one incorrect prediction. Confusion matrix of LR shows that out of 20 predictions, 14 are true predictions, and 6 are incorrect predictions.

8. Conclusion

In this study, a novel stability prediction method for quadruped robot locomotion based on CPG has been employed. The proposed technique utilizes ML approaches, employing classification methods and PCA, to predict the stability condition of robot locomotion. Six classification methods, including DT, RF, KNN, SVC, LR, and GaussianNB, have been applied and evaluated using four metrics: accuracy, recall, precision, and F1-score, in order to classify the stability or instability of robot locomotion. To reduce the dimensionality of features, the PCA method has been employed with various values. Unlike previous studies that focused on analyzing the stability of legged robot locomotion during specific gaits [6, 7, 8, 9], the primary objective of this study is to predict stability prior to initiating a gait with different parameters in robots. Based on the results presented in Table 3, it can be observed that the LR classifier exhibited the weakest performance in predicting stability. On the other hand, the DT and RF classifiers demonstrated the best performance when PCA was set to 1. However, for other PCA values, the KNN and SVC classifiers displayed higher metric values, indicating their effectiveness in predicting the stability of robot locomotion. In this technique the stability of a quadruped robot is anticipated before applying the relevant locomotion and step switching with specific range of adjusting parameters. In addition, the best predictors are represented to guarantee the stable and unstable move. It is proposed in future research to implement this technique to real legged robots especially to quadrupeds the superior robots because anticipating the stability is vital in both the gait changing and adjusting the parameters in practice because in different terrains, different types of robots should be utilized to maintain the relevant stability.

Author Contributions

M.D. developed the theoretical formalism of quadruped robot, performed the analytic calculations. D.G. simulated the quadruped robot model. Due to stability prediction of the robot, both D.G. and M. D. performed machine learning techniques on the quadruped robot model. A. A. K. and M. D. authors contributed to the final version of the manuscript. M.D. supervised the project.

References

[1] Priyaranjan B., Prases K. Mohanty. Synthesizing the optimal gait of a quadruped robot with soft actuators using deep reinforcement learning. *Ain Shams Engineering Journal*. 2021, vol.12 DOI: 10.1016/j.asej.2020.11.005.

[2] Qinglei J., Shuo F., Kaige T., Seshagopalan Thorapalli M., Karin Lagrelius, David D., Georgios A., Xi Vincent W., Lihui W., Lei F. Synthesizing the optimal gait of a quadruped robot with soft actuators using deep reinforcement learning. *Robotics and Computer–Integrated Manufacturing*. 2022, vol. 78, DOI: 10.1016/j.rcim.2022.102382.

[3] Raibert M. Legged robots. *Commun ACM*. 1986, vol. 29, no. 6, pp. 499-514.

[4] Jiawei C., Kun X., Xilun D. Adaptive gait planning for quadruped robot based on center of inertia over rough terrain. Biomimetic Intelligence and Robotics. 2022, vol. 2, DOI: 1016/j.birob.2021.100031.

[5] Matsuoka K. Mechanisms of frequency and pattern control in the neural rhythm generators. *Biological Cybernetics*. 1987, vol. 56, no. 5, pp. 345-353.

[6] Kimura H., Fukuoka Y., Cohen A.H. Adaptive dynamic walking of a quadruped robot on natural ground base on biological concepts. *Int. J. Robot. Res.* 2008, vol. 26 no. 5, pp. 475-490, DOI: 10.1177/2F0278364907078089.

[7] Li B., Li Y., and Rong X. Gait generation and transitions of quadruped robot based on Wilson-Cowan weakly neural networks. *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, 2010, DOI: 10.1109/ROBIO.2010.5723296.

[8] Liu F. and Chen X. P. Gait evolving method of quadruped robot using zero-moment point trajectory planning. *Robot*. 2010.

[9] Li J., Wang J., Yang S. X., Zhou K., and Tang H. Gait Planning and Stability Control of a Quadruped Robot. *Research Article*. 2016, DOI: 10.1155/2016/9853070.

[10] Mirrashid N., Alibeiki E., Rakhtala S. M. New Supper-twisting Sliding Mode Control of an Upper Limb Rehabilitation Robot Based on the TLBO Algorithm. *Journal of Applied Dynamic Systems and Control*. 2022, Vol.5, No.2, pp. 44-53.

[11] Hasanpour H., Alibeiki E., Ghadami S. M. Flexible Robot Arm Control Using Adaptive Control Structure. *Journal of Applied Dynamic System and Control*. 2022, Vol.5, No.1, pp. 5-10.

[12] Obadina O. O., Thaha M. A., Mohamed Z. Greybox modelling and fuzzy logic control of a Leader-Fllower robot manipulator system: A hybrid Grey Wolf-Whale Optimisation approach. *ISA Transactions*. 2022, DOI: 10.1016/j.isatra.2022.02.023.

[13] Naya-Varela M., Faina A., Mallo A., Duro R.J. A study of growth based morphological development in neural network controlled walkers. *Neurocomputing*, 2022, DOI: 10.1016/j.neucom.2021.09.082.

[14] Verbitsky N. S, Chepin E. V., Gridnev A. A. Experimental studies of a convolutional neural network for application in the navigation system of a mobile robot. *Procedia Computer Science*. 2018.

[15] Tang Y., Zhang G., Ge D., Ren C., Ma S. Undulatory gait planning method of multi-legged robot with passive-spine. *Biomimetic Intelligence and Robotics*. 2022, DOI: 0.1016/j.birob.2022.113271.

[16] Jodas D. S., Marranghello N., Pereira A. S., Guido R. C. Comparing support vector machines and artificial neural networks in the recognition of steering angle for driving of mobile robots through paths in plantations. *Procedia Computer Science*. 2013.

[17] Li T., Sun X., Shu X., Wang C., Wang Y., Chen G., and Xue N. Robot Grasping System and Grasp Stability Prediction Based on Flexible Tactile Sensor Array. *Machines*. 2021, DOI: 10.3390/machines9060119.

[18] Chen J., Xu K., Ding X. Adaptive gait planning for quadruped robot based on center of inertia over rough terrain. *Biomimetic Intelligence and Robotics*. 2022,DOI: 10.1016/j.birob.2021.100031.

[19] Zhang Y., Thor M., Dilokthanakul N., Dai Z., Manoonpong P. Hybrid learning mechanisms under a neural control network for various walking speed generation of a quadruped robot. *Neural Networks*. 2023, DOI: 10.1016/j.neunet.2023.08.030.

[20] Fukui T., Fujisawa H., Otaka K., Fukuoka Y. Autonomous gait transition and galloping over unperceived obstacles of a quadruped robot with CPG modulated by vestibular feedback. *Robotics and Autonomous Systems*. 2018, DOI: 10.1016/j.robot.2018.10.002.

[21] Jijo B. T., Abdulazeez A. M. Classification Based on Decision Tree Algorithm for Machine Learning. *Journal of Applied Science and Technology Trends*. 2021, Vol. 02, No. 01, pp. 20 - 28.

[22] Han S., Kim H., Lee Y. S. Double random forest. *Machine Learning*. 2020, DOI: 10.1007/s10994-020-05889-1.

[23] Barfi M., Karami H., Faridi F., Sohrabi Z., Hosseini M. Improving robotic hand control via adaptive Fuzzy-PI controller using classification of EMG signals. *Heliyon*. 2022, DOI: 10.1016/j.heliyon.2022.e11931.

[24] Noboa E., Racz M., Szucs L., Galambos P., Marton G., and Eigner G. Development of an EMG based SVM supported control solution for the PlatypOUs education mobile robot using MindRove headset. *IFAC PapersOnLine*. 2021, pp. 54-15, DOI: https://creativecommons.org/licenses/by-nc-nd/4.0/.

[25] Stojanov D., Lazarova E., Veljkova E., Rubartelli P., and Giacomini M. Predicting the outcome of heart failure against chronic-ischemic heart disease in elderly population - Machine learning approach base don logostic regression, case to Villa Scassi hospital Genoa, Italy. *Journal of King Saud University - Science*. 2023, vol 35 DOI: 10.1016/j.jksus.2023.102573.

[26] Bartosik A., Whittingham H. The Era of Artificial Intelligence, Machine Learning, and Data Science in the Pharmaceutical Industry - Chapter 7 - Evaluating safety and toxicity. 2021, DOI: 10.1016/B978-0-12-820045-2.00008-8.

[27] Ye Z., Song P., Zheng D., Zhang X., Wu J. A Naive Bayes model on lung adenocarcinoma projection based on tumor microenvironment and weighted gene co-expression network análisis. *Infectious Disease Modelling*. 2022, DOI: 10.1016/j.idm.2022.07.009.

[28] Singh S. K., Taylor R. W., Pradhan B., Shirzadi A. Predicting sustainable arsenic mitigation using machine learning tecgniques. *Ecotoxicology and Environment Safety*, 2022, DOI: 10.1016/j.ecoenv.2022.113271.

[29] Markoulidakis I., Rallis I., Georgoulas I., Kopsiaftis G., Doulamis A., and Doulamis N. Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem. *technologies*. 2021, DOI: 10.3390/technologies9040081.

[30] McKinney W. Python for data analysis: Data wrangling with Pandas, NumPy, and IPython. *O'Reilly Media, Inc,*. 2017.

[31] Bibal A., Clarinval A., Dumas B., Frenay B. IXVC: An interactive pipeline for explaining visual clusters in dimensionality reduction visualizations with decisión tres. *Array* 2021, DOI: 10.1016/j.array.2021.100080.

[32] Jolliffe L. T., Cadima J. Principal component analysis: a review and recent developments. *The Royal Society - PHILOSOPHICAL TRANACTIONS A*. 2015, DOI: 10.1098/rsta.2015.0202.

[33] Hussain A. N., Abboud S. A., Jumaa B. A., Abdullah M. N. Impact of feature reduction techniques on classification accuracy of machine learning techniques in leg rehabilitation. *Measurement: Sensors*, 2023, vol. 25, DOI: 10.1016/j.measen.2022.100544.

[34] Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M. et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 2011, vol. 12.

[35] Valero-Carreras D., Alcaraz J., Landete M. Comparing two SVM models through different metrics based on the confusion matrix. *Computers and Operations Research*. 2023, vol. 152, DOI: h10.1016/j.cor.2022.106131.

About Authors

Delaram GHABI was born in Gonbad kavus, Iran. She received her B.Sc. from Islamic Azad University in 2021. Her research interests include robotics and machine learning.

Mohammad DIVANDARI (corresponding author) was born in Sabzevar, Iran. He received the B.Sc. in electronic engineering and M.Sc. in control engineering degrees in 1998 and 2003, respectively, both from Islamic Azad University, Iran. He received a PhD degree in electrical engineering and control systems at Babol Noshirvani University of technology 2019. His research interests include special electrical machines and drives, mechatronics, control solutions and machine learning. He is assistant professor of control systems engineering at Islamic Azad University, Iran.

Abdol Aziz KALTEH was born in Bandarturkman, Iran. He received the B.Sc. in electronic engineering from Ferdowsi Universty of Mashhad and M.Sc. in communication engineering from Sahand University of Technology, Iran. He received a PhD degree in communication engineering at Islamic Azad University Science and Research Branch, Tehran, Iran 2011. His research interests included microwave, antenna and propagation, pattern recognition, robotics, communication systems, signal processing and machine learning. He is assistant professor of electrical and electronic engineering at Islamic Azad University, Iran.

Appendix A

```
ACC_DTC = []
ACC_RFC = []
ACC_KNN = []
ACC_SVC = []
ACC_LR = []
ACC_GaussianNB = []
PCA_dim = []
for i in range(1, 6):
.  step1 = [('dimension reduction',
PCA(n_components= I)), ('classifier',
DTC())]
.  step2 = [('dimension reduction',
PCA(n_components= I)), ('classifier',
RFC(n_estimators= 3))]
.  step3 = [('dimension reduction',
PCA(n_components= i)), ('classifier',
KNN(n_neighbors= 3))]
.  step4 = [('dimension reduction',
PCA(n_components= i)), ('classifier',
SVC(kernel= 'rbf'))]
.  step5 = [('dimension reduction',
PCA(n_components= i)), ('classifier',
LR(solver= 'liblinear'))]
.  step6 = [('dimension reduction',
PCA(n_components= i)), ('classifier',
GaussianNB())]
.  clf1 = Pipeline(steps= step1)
.  clf2 = Pipeline(steps= step2)
.  clf3 = Pipeline(steps= step3)
.  clf4 = Pipeline(steps= step4)
.  clf5 = Pipeline(steps= step5)
.  clf6 = Pipeline(steps= step6)
.  clf1.fit(X_train, y_train)
.  clf2.fit(X_train, y_train)
.  clf3.fit(X_train, y_train)
.  clf4.fit(X_train, y_train)
.  clf5.fit(X_train, y_train)
.  clf6.fit(X_train, y_train)
.  y_pred1 = clf1.predict(X_test)
.  y_pred2 = clf2.predict(X_test)
.  y_pred3 = clf3.predict(X_test)
.  y_pred4 = clf4.predict(X_test)
.  y_pred5 = clf5.predict(X_test)
.  y_pred6 = clf6.predict(X_test)
.  ACC_DTC.append(accuracy_score(y_test,
y_pred1)*100)
.  ACC_RFC.append(accuracy_score(y_test,
y_pred2)*100)
.  ACC_KNN.append(accuracy_score(y_test,
y_pred3)*100)
.  ACC_SVC.append(accuracy_score(y_test,
y_pred4)*100)
.  ACC_LR.append(accuracy_score(y_test,
y_pred5)*100)
.  ACC_GaussianNB.append(accuracy_score(
y_test, y_pred6)*100)
PCA_dim.append(i)
```