# Dynamic Time Warping based features selection method for selecting foot gesture cobot operation mode.

**Gilde Vanel Tchane Djogdom [1,2], Martin J.-D. Otis [1], Ramy Meziane [1,2]**

[1] Laboratory of Automation and Robotic interaction (LAR.i), Department of Applied Sciences, Université du Québec à Chicoutimi (UQAC), 555 Boulevard de l'Université, Chicoutimi, QC G7H 2B1; e-mail: martin_otis@uqac.ca

[2] ITMI (Technological institute of industrial maintenance), Sept-iles College, 175 Rue de la Vérendrye, Sept-Îles, QC G4R 5B7Canada

* Correspondence: gilde-vanel.tchane-djogdom1@uqac.ca

Received: date; Accepted: date; Published: date

**Abstract: Problem:** The emerging needs of human beings are pushing manufacturing companies from mass production to mass customization. The occurrence of these new challenges leads to a change of scenario where the robot no longer works isolated from human to a scenario in which the robot collaborates with the human in the same workspace (collaborative robotics). **Aims**: Wearable sensors using inertial measurement unit (IMU) are widely used to capture human upper body gestures in which the set of gesture being recognize is very large. However, foot gesture approach is starting to gain some places in applications where human's hands are occupied when interacting with robots. **Method:** This study presents an insole-based foot gesture recognition method for cobot operation mode selection. The insole is composed of an IMU and four force sensors. The classification algorithm uses a support vector machine (SVM) classifier based on features extracted by means of Dynamic Time Warping (DTW) applied to only one reference gesture signal. Five human participants are used for the dataset. As a case study, the system was interfaced in real-time (real time classification algorithm) using a Simulink 2020a scheme with Universal Robots UR5 (5 kg payload). **Results**: The worst-case recognition accuracy is around 88%. **Conclusion:** The algorithm is able to adequately discriminate between 10-foot gestures by means of a wearable insole sensor incorporated into the insole. Moreover, this study shows that, the control gesture can accurately being recognize from other current activities such as walking, turning, climbing the stairs and similar.

**Keywords:** Human-Robot Collaboration; Instrumented Insole; Foot Gesture Recognition; Support Vector Machine; Dynamic Time Warping.

## 1. Introduction

The advent of collaborative robotics has led to the development of new applications such as third-hand robotics where robots work as an extension of the human limb as a support and assistant [1,2]. These new applications require the development of new intuitive, user-friendly and ergonomic communication interfaces between the robot and the human [3]. In doing so, portable and intuitive communication devices have emerged and enable various robot control modes in the industry. Recent examples deal with the recognition of human hand gestures acquired by means of inertial measurement units for robots mode change and control applications in the manufacturing environment [4]. The advantage of using inertial measurement units lies in their mobility and small size. It does not restrict human movements and appears to be more robust to environmental disturbances and constraints such as noise, brightness etc. [4, 5]. Studies dealing with the recognition of human gestures based on inertial measurement sensors are of various types and make it possible to detect both gestures of the upper parts of the human [3, 6, 7] and very recently those of the lower parts [8-11] based on foot gestures. However, aside the nature of the input command gestures, there is a concern for the processing of time series data derived from the different gestures, particularly, on the topic of real time segmentation and classification. It is commonly assumed in the literature that the best classification result for time series data in term of accuracy is achieved using Dynamic Time Warping (DTW) combined with 1-NN (nearest neighbour) [12, 13]. In such process, the input signal is compared with the different signals from the database or key signals of each class considered. This approach explores the concept of similarity in the sense that the class with the closest distance is the one that best matches the signal under evaluation. However, for systems with low processing capacities and for real time implementation objectives, this structure turns out to be costly in terms of computation time.

This article aims to address applications such as the third-arm robotic where lower body gestures are desired for hand free interaction with the robot. Moreover, a particular emphasis is placed on the DTW-based classification mechanisms used as a tool for determining the signal features based on a single reference gesture rather than considering either all of them [13] or each representatives gestures for different classes of the dataset [12].

The project suggests controlling robotic actions through 10 simples and compounds foot gestures for controlling possible modalities of high dimensionality cobot with a low dimensionality wearable device such as a smart insole. The contributions to this article are as follows:

- Recognition of 10 simples and compounds foot gestures foot by means of a sensor placed inside an instrumented insole.
- The use of DTW as a tool for determining the temporal characteristics of gestures **based on a single reference gesture signal**. The aim is to compute rather than the similarity between classes, the dispersion base on a single reference gesture.
- Discrimination between control gestures and those of everyday life applications such as walking, turning, going up and down stairs without the need of a locking gesture.

The major contribution to this article is to show that the DTW approach based on a single reference foot gesture can been used as features for an SVM classifier and adequately discriminate between command and no command gestures such as walking, turning, going upstairs, going downstair. The proposed method is simple and extensible and can be potentially further improved by combining with other features related method such as mean, standard deviation etc. which perform well in time series classification.

The rest of this work is organised a follow: Section 2 of this article reviews the related works to contextualize the contribution of this research work. Section 3 presents the material used and the paper's primary contributions: which is the use of DTW approach based on one reference gesture for the selection of cobot operating mode. Section 4 presents the experimentation and the results obtained. Section 5 presents an overview of the limit of the study and section 6 presents the conclusion and future works.

## 2. Related Works

Firstly, the related work on foot gesture recognition as command center is covered in section 2.1 and then a brief review of the most different existing methods for foot gesture recognition based wearable sensors is analyzed in section 2.2. In these related works, the previous studies on foot gestures-based pressure sensor matrices and features selection method such as DTW are particularly covered with other classification algorithm such as SVM (Support Vector Machine) classifier.

### *2.1. Foot gesture as command center*

Control based on foot gestures is a fairly recent research topic which tends to impose itself in applications mainly for people suffering from limb deficit in the context of the control of prostheses [14]. This control approach is done depending on whether you are standing or sitting. According to a study carried out in [15], which demonstrates that, for healthy people interacting with a mobile phone, for example, there are configurations according to which the command based on foot gestures would be more beneficial than that based on hand gestures with a satisfaction rate of nearly 70%. From this observation, it follows that, for an application such as the third robotic hand where one is often led to operate the robot in a standing position, the command based on foot gestures appears to be the ideal solution even thought the feet also fulfill the main function of supporting the limbs of the human when the latter is in a standing position [8, 16]. Various works going in this field have made it possible to set up these strategies both for control of mobile phone [15, 17], creation of music from foot gestures recognition [18] or performing of navigational tasks in interactive 3D environments [11]. Other applications have focused on the field of surgical assistance [19]. One of the first applications of this technology in the context of robotic control is inherited from Sasaki et al., 2017 [16], which proposes an interactive system for controlling the position of two robotic arms by the movement of the user's foot and the grip of each arm is controlled by the toes. Recently a UR5 robotic system control approach is explored in [8] without referencing any real-time application of the proposed control strategy. Independently of the field of application, two technologies of portable sensors are the most recurrent, namely the systems based on sEMG (surface Electromyography) and those based on inertial measurement unit (IMU). Moreover, independently of the type of sensor being used the need of segmentation and classification for gestures recognition arises [20].

### *2.2. Time series based classification approaches*

Time series classification is usually based on either features-based method, model-based method or distance-based method.

Independently of the method being used, the necessity of accurate signal segmentation arises. The purpose is to determine at which time the command gesture is set to start and when it is set to finish. Usually, the segmentation approaches use a window length calibrated on the gesture duration and the starting point might either been a sliding window or a given threshold position as defined by [18]. Once the segmentation is done, time series classification is required. For time series recognition approaches in general, **distance-based approaches** using DTW like 1-NN DTW appear to be a state of art in term of accuracy. However, such algorithm has a computational issue and for simple online application, it requires high computational capacities. Therefore, **features-based time series classification** has been considered in the latter and it is commonly used in the field of gesture input modalities for cobot or mobiles phones control. **Table 1** presents an overview of the different recognition methods used.

**Table 1 :** Overview of the differents classification method uses for upper and lower body recognition of input signal.

| Article | Upper body | Lower body | Method | Comment |
|---|---|---|---|---|
| [3] | ☒ | ☐ | ANN (Artificial Neural Network) | Hand gestures (8 statics gestures and 4 dynamics gestures) |
| [6] | ☒ | ☐ | CNN + NN (Convolutional Neural Network & Neural Network) | Hand gesture (10 statics gestures) |
| [8] | ☐ | ☒ | 2D-CNN | 5 foot gestures |
| [9] | ☐ | ☒ | 2D-CNN | 1 foot gesture |
| [11] | ☐ | ☒ | 2D-CNN | 4 foot gestures |
| [18] | ☐ | ☒ | SVM | 5 foot gestures |
| [29] | ☐ | ☒ | 2D-CNN | 8 foot gestures |
| [30] | ☐ | ☒ | LDA (Linear Discriminant Analysis) | 6 foot gestures |
| [31] | ☐ | ☒ | LR (logistic regression technique) | 1 foot gesture |

Features based time series classification involves automatic time series or hand-crafted times series features selection. The state-of-the-art result in feature-based time series classification lies in CNN (Convolutional Neural Network). Recently, Aswad et al., 2021 [8] achieve nearly a 99% classification accuracy recognition from timeseries classification based on 2D-CNN. However, for the same reason stated above concerning the computational burden required, 2D-CNN was not considered for the application being proposed. Moreover, in this paper, the dimension of gestures has to be the same (windows length) to transfer the selected features of the data inside each pixel of an image and this segmentation is done manually. Another method with state of art result is the 1D-CNN used for the classification of time series with consideration of some temporal dependencies between sensor signal being analysed. However, it is required to define a specific structure according to the frame of signal being analysed [21]. Others approaches uses statistical features in time and/or frequency domain to compute for features and then classify through simple SVM classifier [18]. Those approaches are characterised with low computational burden but cannot account for temporal distortion in the time series signal. Therefore, DTW which can manage signal dilatation, tends to be of great interest if it is used as features extraction method. This line of thought was firstly introduced by Kate, 2016 [13]. In his study, the author uses DTW as features extractor and compute DTW distances between every set of the training samples and then uses the distance acquired in combination with SAX method to train an SVM classifier. However, the method proposed is computationally dependent of the training size. Another approach based on DTW as features extraction method uses a centroid data to represent each class for which the DTW will then be computed and used for training purposes of an SVM or a clustering approach [22]. More recently, one approach combine 1D-CNN with local DTW features extraction method from each class centroid for recognition processing [23].

However, From the author's point of view, no work has considered only one reference signal or gesture using DTW features extraction method to discriminate between time series signal classes. Thus, in this work, three hypotheses are formulated as follows:

1. It's possible to discriminate between a set of 10 command gestures and non-command gestures by means of a single time series reference gesture with high accuracy,
2. The classification algorithm is mainly based on the nature of the reference gesture being used and
3. It's possible to compute features selection based on DTW by means of a static reference gesture (the standing position).

## 3. Methodology

First, the insole hardware and software used for the foot gesture command is presented in section 3.1, then the data processing and pipeline approaches used are presented in 3.2. The gestures dictionary used the for cobot control is defined in 3.3. The data processing and preprocessing adopted are presented in 3.4. Section 3.5 presents the concept of dynamic time warping for time series signal and section 3.6 presents a proof of concept on the advantages of using such approach in the case of foot gesture recognition. Finally, section 3.7 presents a comparison of the different classifiers in order to choose the most suitable one for the application.

### *3.1. Insole hardware and software architecture*

The insole device presented in **figure 1** is located at the foot arch position. The detailed design was previously presented in [25]. It contains a 9-axis motion processing unit MPU9250 [26], which measures the foot's acceleration, velocity, and orientation through a set of 3-axis accelerometer, 3-axis gyroscope, and 3-axis magnetometer combined with a digital motion processor (DMP). Moreover, four force-sensitive resistors (FSR), two in the forefoot position and two in the heel position were also integrated to measure the pressure applied on the insole. The analog signals acquired from the pressure sensors were converted by an analog-to-digital converter (ADC) with a 12-bit resolution acquired with an ESP32 WiFi module which is also used to send data to the Linux server using MQTT protocol.


**Figure 1** : Insole's device sensors

The overview of the proposed foot gesture recognition system is illustrated in **figure 2.**

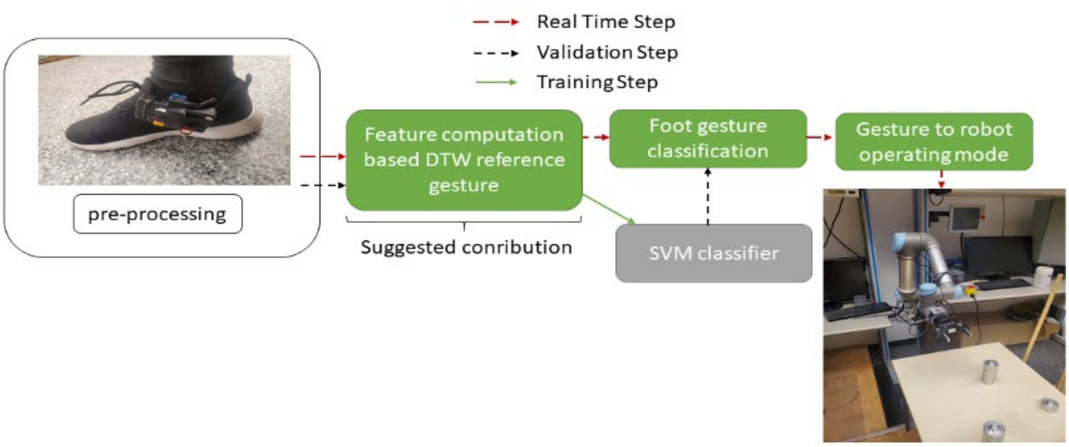



**Figure 2** : Suggested pipeline for the training, validation, and real-time execution

The signal processing steps used in this article is the same as the one depicted in Aswad et al., 2021 [8]. As the system computes foot gesture command detection, it requires data information from the human's foot. The aim of the recognition is to control UR5 (Universal Robots, 5kg payload) robot through foot gesture. The instrumented insole acquires, processes, and uses MQTT protocol to transmits wirelessly the data to the computer running a ROS server. Then a communication channel is set between the ROS server and MATLAB-Simulink 2020a for online data acquisition and recognition. The sampling frequency used in the data processing and transmission is 500 Hz [24].

*3.2. Experimental protocol with human participants*

The experimental protocol is conducted with five (5) participants which consists of four (4) distinct phases as shown in **figure 3.**

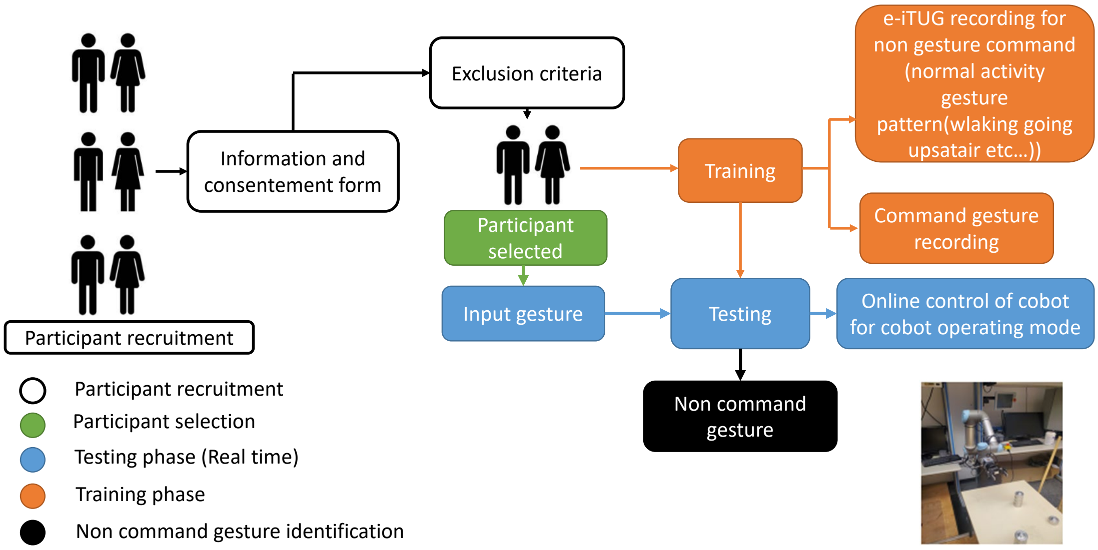



**Figure 3** : Experimental protocol

For each participant taken individually, the first phases consist of protocol agreement and exclusion criteria evaluation. The exclusion criteria are: the participant should be able to stand without a supportive device, they must have both physical motor and intellectual impairment and female participant must not ne pregnant. 5 male participants with an average age of 27,5 were recruited among our lab’s colleagues. This study is approved by the University of Quebec at Chicoutimi (UQAC) Ethics Committee (Research Ethics Board) under number 2022-837. All participants signed an informed consent form.

The second phase involves as training and data acquisition. Here, each participant is asked to do three main set of actions. The first one is the recording of the command gestures. This recording is based on the use of a fixed window size of 2 seconds. This window is considered sufficient to be able to perceive all the dynamics of one command gesture. Moreover, in opposition to the moving window techniques widely used in the case of human activity recognition [14], a windowing system based on an input conditions is used. Indeed, it’s assumed that all control gestures begin with a stable equilibrium position without which there would be no possibility of sending command to the robot by means of foot gesture without enhancing the risk of falling or poor posture. This entry condition is subordinated to a standing position of the user. It is materialized by two joint conditions, the activation of all the FSR's sensors and the reset of y-axis acceleration to offset values i.e. 0 for some participants and 1 for others. When the triggering condition is activated, the participant is asked to perform the 10 predefined gestures which are presented to him by means of a video. In order to control the sequences of recording or not of the data, the operators have the latitude to leave or not the standing position by slightly bending the foot so as to break the condition on the activation of the FSR sensors. For each recording gesture, the participant is required to perform each gesture 10 times according to its different rhythms (fast, slow, medium).

The last activity in this phase is the recording of normal human behavior in everyday life. In doing so, the participant is asked to execute the extended instrumented time up and go test (e-iTUG) about 3 or 5 times. This test is implemented by repeating a set of movements in a cyclic way without the need to concern about the activated or not activated state of the system. The participant is invited to do the following set of movement as described in figure 4.

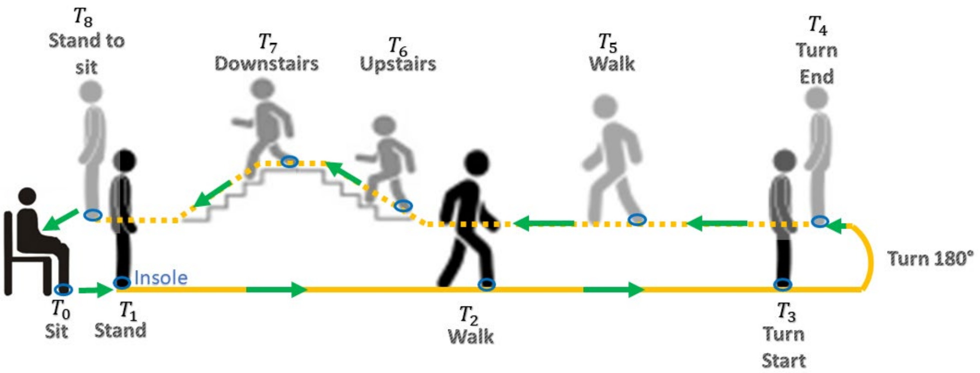



**Figure 4 :** e-iTUG for normal activity recording

Each participant is asked to do the following in one cycle activity: get up from a seated position, walk in a straight line, turn 180 degrees, walk in a straight line, go towards the stairs, go upstairs, go downstairs and sit. The recording process follows the same segmentation approaches base on the triggering condition used in the recording phase of the command gesture. Moreover, the participant is asked to stay in standing position for almost 5 seconds in order to get the reference signal gesture as it is assumed to be the best choice in this case. All the data recorded during this e-iTUG are categorized as non-gesture commands (class 11) and presented in [28]. The last phase is real time implementation of the proposed foot gesture recognition process. In this phase, the data from foot gesture are acquired through the same segmentation process as the one used for the training (triggering condition and moving window of fixed size).

The proposed real time implementation can be summarized in **figure 5**. The data recording is conducted by a fixed window of 2 seconds when the triggering condition is satisfied. This triggering condition is related to the FSR's sensors and y-axis values of acceleration as it is assumed that when standing, all the FSR's sensors might be activate and the y-axis acceleration might be constant or equal to the offset values depending on human's way of standing. The algorithm then proceeds to compute DTW features based on the reference gesture which is then used in a classic SVM classifier for performing the SVM based DTW classification for gesture recognition and submit an operating mode to the cobot. In this experimentation, the human is required to assemble in accordance with the cobot partner, part of a motor. Therefore, a set of cobot operating mode can be choose solely by the recognition of human's foot gesture input. The cobot is then required to selects an appropriate algorithm from the available operating modes such as trajectory tracking, collision avoidance, etc.

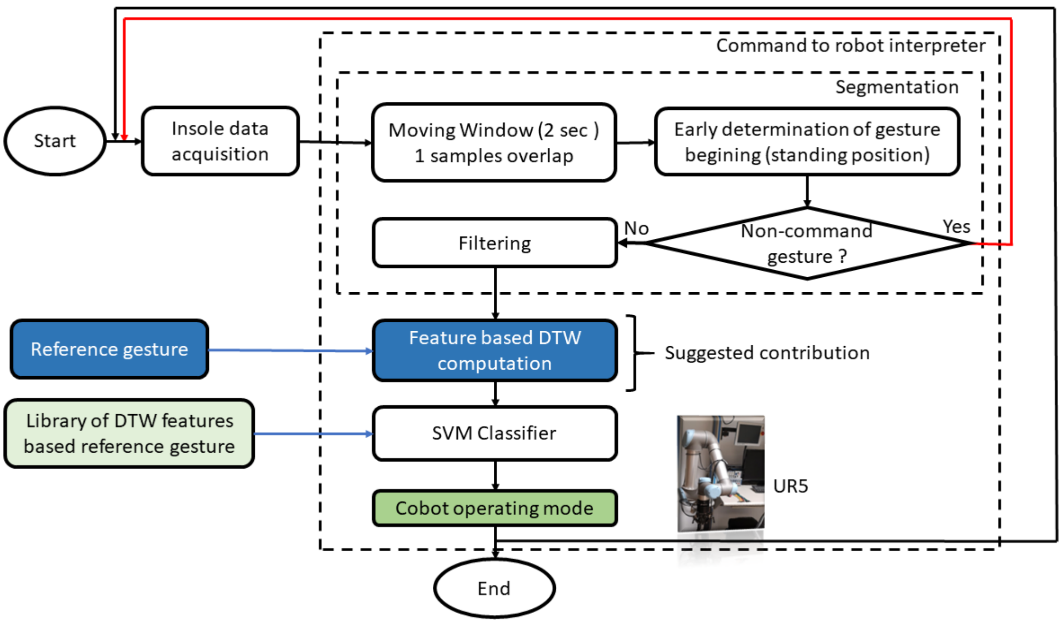



**Figure 5 :** Real-time execution algorithm from data acquisition to the execution of a cobot command for operating mode selection

*3.3. Foot-based command: Gesture Dictionaries*

When referring to **Table 1**, one can observe that foot gesture input modalities often have a limited number of possibilities (8). In this research work, one aim is to extend the gesture input modalities to 10 for the control of complex system operation. Thus, a dictionary of 10 command gesture has been formulated. It is composed of an extension of the five simple foot gestures derived from Aswad et al.,2021 [8] and compound gestures as define in **Tables 2** and **3**. The suggested algorithm should be able to differentiate these 10 gestures from those executed in the e-iTUG, which represents daily activities (not associated to a command for the cobot).

**Table 2:** Representation of the five proposed gestures denoted from G1 to G5 as defined in Aswad et al.,2021 [8]

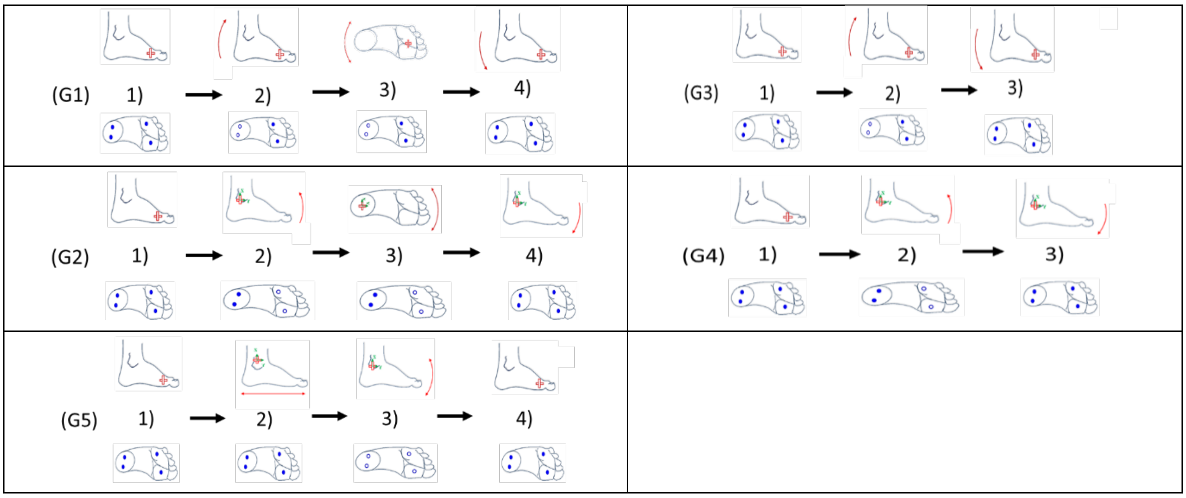
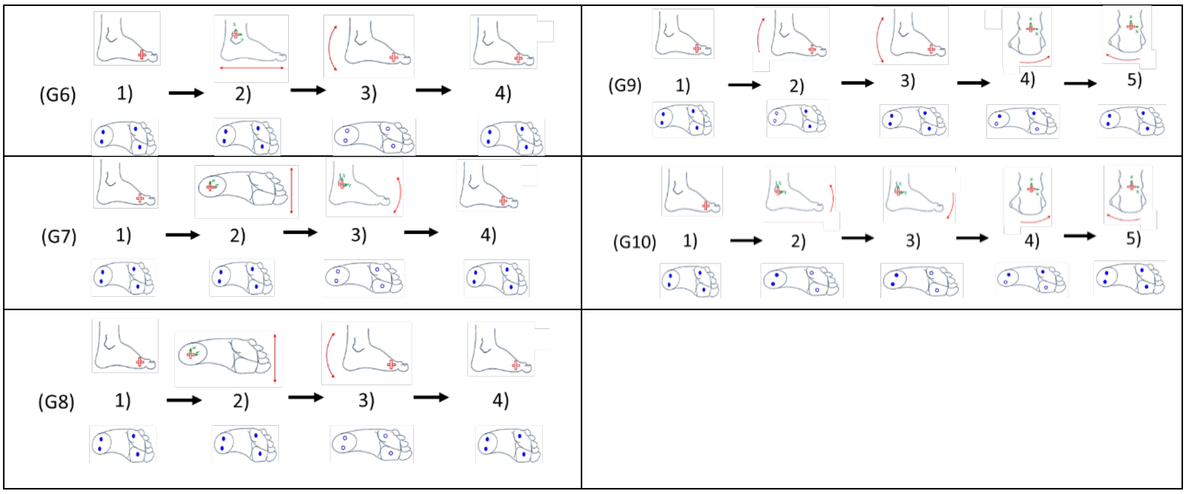



**Table 3 :** Representation of the five new proposed gestures denoted from G6 to G10.




Once identified, the foot gestures are then mapped with a set of cobot operating mode. In this study, a set of different cobot states which can help the assembly process has been defined. The different Cobot's modes uses in this article can be activated at any time when the mapping gesture is performed. Those mode are defined as follows:

- **Free drive mode**: with this mode, the robot can be held by hand and taken to a given target location for learning.
- **Autonomous mode**: the robot performs a given motion by taking a piece from a position A to the assembly path.
- **Learning new assembly process and part locations:** The parts location can be modified and indicated through the robot using the free drive mode, then learning new task is defined as the ability of the robot to learn the given parts locations.
- **Force control mode**: It is defined as humans having physical interaction with the robot (force control).
- **Others general movements are also defined like:** Precise trajectory control, fast trajectory control, moving robot to home position, stopping robot, turning left or right the robot configuration.

The following commands with mapping gestures are presented in **Table 4.**

**Table 4 :** Foot mapping gesture

| Foot gesture | Cobot operating mode |
|---|---|
| **G1** | Free drive mode |
| **G2** | Fast trajectory control |
| **G3** | Precise trajectory control (Slow) |
| **G4** | Autonomous action in shared activity |
| **G5** | Stopping the robot |
| **G6** | Learning new tasks for assembling process |
| **G7** | Physical collaboration / force control mode |
| **G8** | Moving robot to home position |
| **G9** | Turning left (robot) |
| **G10** | Turning right (robot) |

The proposed foot-based dictionary mapped with cobot operating mode must be decoded in order to accurately scope the user's intention when interacting with the cobot. The next section proposed the overall process for data acquisition and features selection.

### *3.4. Data Acquisition, segmentation and filtering*

The gestures presented in **Tables 2** and **3** are acquired by an instrumented insole worn in the left foot. In this study, the gestures of 5 participants (healthy adults) were recorded. The measurement time of each gesture was set at two (2) seconds. For numerical simulation, signals from the 3-axis accelerometer, 3-axis gyroscope, and the 4 FSRs are exploited. The details from the insole's signals are provided in Table 5. They are them used as entry for the DTW features based SVM classification.

**Table 5 :** Insole's device signals

| Signal's name | Description | Signal's Origin |
|---|---|---|
| ***AcX, AcY, AcZ*** | Acceleration in the 03 axis (X, Y, Z) | 3-axis accelerometer |
| ***VaX, VaY, VaZ*** | Angular velocity in the 03 axis (X, Y, Z) | 3 axis gyroscopes |
| ***P*** | Euler's angle: P (Pitch) | DMP (Digital Motion Processor) |
| ***R*** | Euler's angle: R (Roll) | |
| ***Y*** | Euler's angle: Y (Yaw) | |
| ***F1, F2, F3, F4*** | FSR sensors displayed at the forefoot (right and left) and the heel (right and left) | FSR sensors |

The gestures are assumed to start from a standing position and end in the same position. In fact, this is what happens in reality, so the data are recorded using this principle. As for real time implementation, the same approach is used as depicted in **Figure 5.** Thus, the authors formulate the hypothesis that, it's possible to compute features selection based DTW by means of a static reference gesture (the standing position). When the foot gesture signal data are given as input, the set of signals according to the defined window of two (2) seconds is proceeded to signal filtering block which is based on a low pass fourth order FIR (Finite Impulsion Response) Butterworth filter with a cut off frequency of 75Hz. The cut off frequency is designed based on the obw() MATLAB function which help identify the portion of signal in the frequency domain belonging to the human being. Then, the filter design MATLAB function (FilterDesigner) is used to design the filter.

*3.5. Dynamic time Warping: distance feature*

DTW is a distance tool used to measure the dissimilarity between two times series sequences after aligning them. It allows similar shapes to match even if they are out of phase allowing elastic (warping) shifting of the time series [13]. Given two-time series Q and R, DTW distance is computed by first finding the best alignment between them. To align the two time series, an n-by-m D matrix is constructed whose (i, j) element is given by $D_{i,j} = \left(q_i - r_j\right)^2$; it which represents the cost to align the point $q_i$ of time series Q with the point $r_j$ of time series R. An alignment between the two time series is represented by a warping path, $W = w_1, w_2, \ldots, w_k$, in the matrix which has to be contiguous, monotonic, start from the bottom-left corner and end at the top-right corner of the matrix. The best alignment is then given by a warping path through the matrix that minimizes the total cost of aligning its points, and the corresponding minimum total cost is named the DTW distance. Hence, as defined in [12], $DTW(Q, R) = W_{NN}$ with $W_{ij} = D_{ij} + \min(w_{i-1,j}, w_{i-1,j-1}, w_{i,j-1})$. The minimum cost alignment is computed using a dynamic programming algorithm. DTW also has a multivariate version commonly used for multi class series classification but it is well overtaken by 1-NN DTW univariate time series classifier [20]. As one might consider, 1-NN DTW appears to be time consuming due to the need of computing DTW between a time series T and each time series present in each class [13] or more recently in each centroid (a centroid represents a central time series which can well represent its class) [22]. Moreover, for a set of n classes, $n^2$ DTW distance computation is requires, which is time consuming. The proposed approach uses the human standing posture as reference gesture signals and then, the dataset is composed of a basic DTW distances computed for every one of our 13 signals channels with the reference gesture signals. An analysis of the impact of the reference signal choice is shown in section 4.3 below. Therefore, the accuracy achieved is purely dependent on the accurate choice of the reference signal. In the case of foot gesture recognition as implemented in this study, it appears that the standing posture is an excellent choice for classification purpose.

*3.6. Dynamic Time Warping as features selection method based one reference gesture signals : proof of concept*

In order to evaluate the capacity of the proposed gestures to be able to determine whether or not a characteristic allows good features identification of gestures as suggested in [27], the ANOVA statistical analysis is used. It's calculated from the null hypothesis which implies that the distribution of all the calculated characteristics distribution is similar. The null hypothesis considers that if the probability (p-value) is less than 0.05, the characteristic is set to be significantly different. The ANOVA's results is computed with MATLAB 2020a for the dataset presented in [28]. It is composed of the 5 participants foot gestures. Each participant has a set of 11 gesture group (10 for command gesture and 1 for non-command gesture). For analysis purpose, a part of the dataset, comprising of a set of 10 samples per gestures (110 samples for each participant), is used. The features that are discriminated are the channels univariate DTW distance for each element of the dataset with the reference gesture. The results of the statistical one-way ANOVA evaluation for each of the five participants are given in **Table 6**.

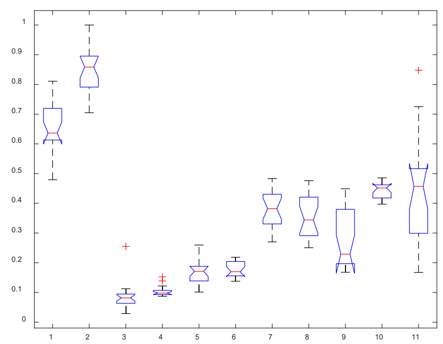
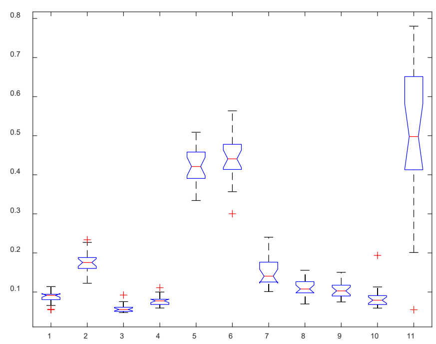
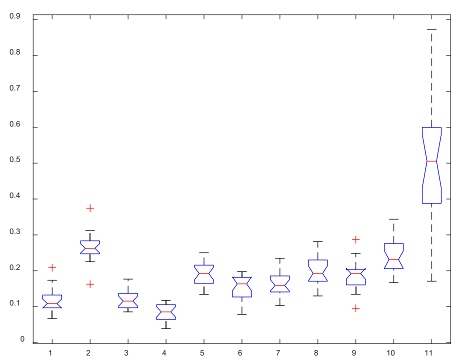
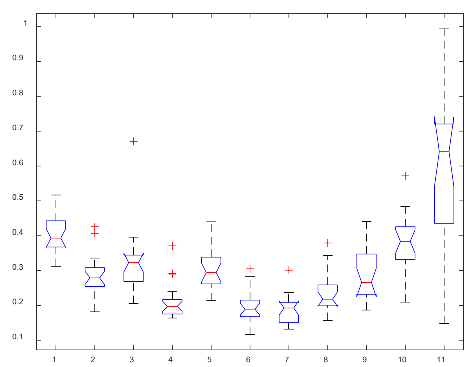
**Table 6 :** Probability (p-values) derived from one way ANOVA

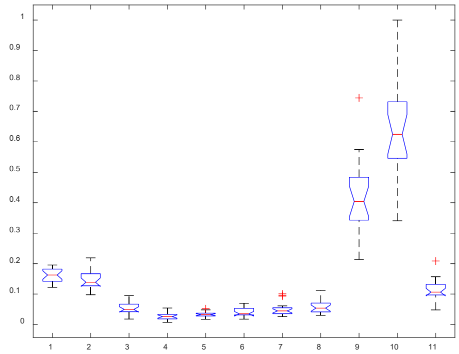
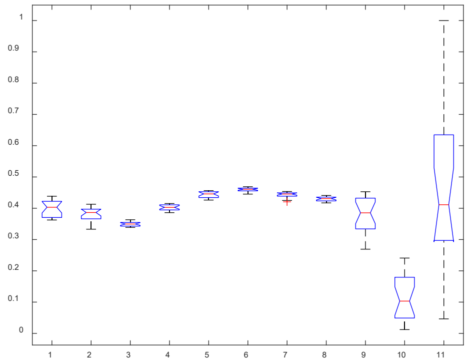
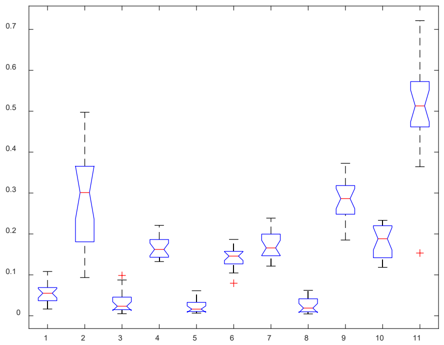
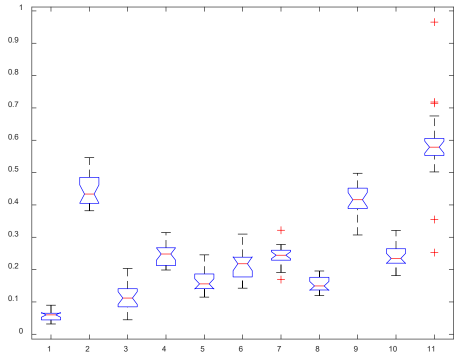
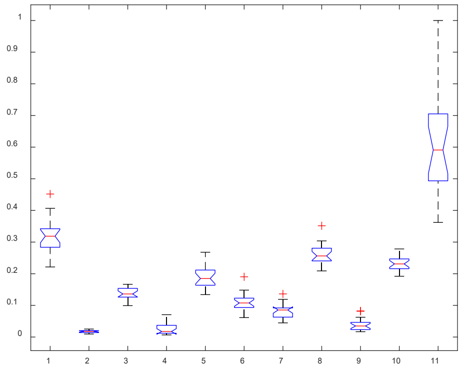
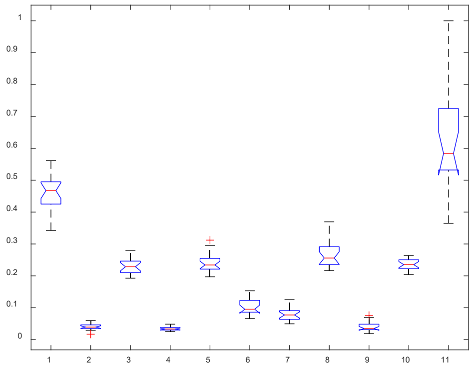
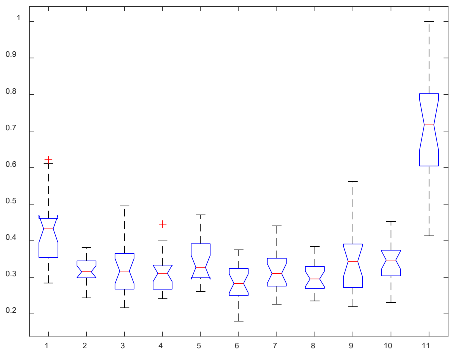
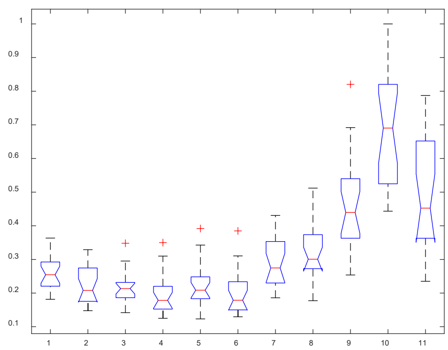
| Sensor channel | Probability (p-values) | | | | |
|---|---|---|---|---|---|
| | Participant 1 | Participant 2 | Participant 3 | Participant 4 | Participant 5 |
| ***AcX*** | 7.86e-97 | 7.86e-27 | 2e-33 | 1.87e-17 | 7.95e-14 |

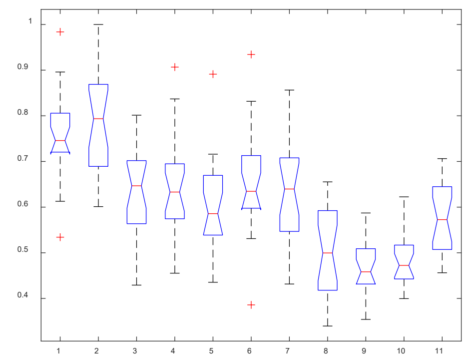
| | | | | | |
|---|---|---|---|---|---|
| ***AcY*** | 9.53e-87 | 1.37e-23 | 7.09e-16 | 2.49e-18 | 2.39e-10 |
| ***AcZ*** | 9.88e-61 | 2.54e-13 | 6.94e-12 | 9.78e-9 | 9e-4 |
| ***R*** | 2.49e-38 | 1.15e-6 | 1e-4 | 1.22e-24 | 4.67e-30 |
| ***P*** | 1.81e-102 | 1.92e-27 | 1.12e-28 | 1.37e-15 | 2.19e-6 |
| ***Y*** | 3.41e-29 | 3.3e-3 | 6.11e-13 | 1.46e-11 | 2.14e-11 |
| ***F1*** | 7.89e-82 | 3.27e-12 | 1.01e-16 | 6.52e-15 | 1.27e-26 |
| ***F2*** | 2.11e-94 | 3.62e-19 | 6.37e-27 | 6.38e-7 | 2.85e-25 |
| ***F3*** | 2.15e-99 | 1.2e-12 | 2.67e-32 | 5.24e-15 | 4.66e-54 |
| ***F4*** | 1.24e-103 | 3.41e-14 | 1.26e-31 | 1.56e-8 | 5.76e-51 |
| ***VaX*** | 3.55e-49 | 1e-4 | 1,79e-7 | 0.5749 | 1.5e-2 |
| ***VaY*** | 1.13e-49 | 9.48e-6 | 2.22e-11 | 4e-4 | 1e-3 |
| ***VaZ*** | 8.16e-27 | 4.597e-6 | 8.03e-12 | 0.7382 | 3e-4 |

The probabilities (p-values) are significantly less than 0.05 apart from ***VaX*** and ***VaZ*** for participant 4**.** This means that, except for this participant, the proposed features might be of great interest for classification purposes. In order to deal with the disparities observed between each participant, it is decided not to remove the above features for participant 4 because they are considered as part of his singularity. **Table 7** below presents participant 1 ANOVA's data. This participant is one author of this paper. The ANOVA representation allows to visually evaluate the ability of the DTW features to discriminate between the 11 set of classes ranging from 1 to 11.

**Table 7 :** ANOVA's results distribution

| ***AcX*** | ***AcY*** |
|---|---|
|  |  |
| ***AcZ*** | ***R*** |
|  |  |
| ***P*** | ***Y*** |

| | |
|---|---|
|  |  |
| ***F1*** | ***F2*** |
|  |  |
| ***F3*** | ***F4*** |
|  |  |
| **VaX** | **VaY** |
|  |  |
| **VaZ** | **X-Axis signification** |




- **1 refer to command gesture (G1)**
- **2 refer to command gesture (G2)**
- **3 refer to command gesture (G3)**
- **4 refer to command gesture (G4)**
- **5 refer to command gesture (G5)**
- **6 refer to command gesture (G6)**
- **7 refer to command gesture (G7)**
- **8 refer to command gesture (G8)**
- **9 refer to command gesture (G9)**
- **10 refer to command gesture (G10)**
- **11 refer to non command gesture (G11)**

A Tukey-Kramer post hoc test was conducted in order to confirm the ability of the proposed DTW feature to adequately discriminate between the 11 different classes. Also, based on the information presented in **Table 6,** one can end up concluding that it is visually possible to adequately discriminate between all different classes. The proposed DTW features are then proceed through a classifier for recognition purposes.

### *3.7. Classifiers comparison and performance validation*

Once the features are extracted, the selection of the best classifier is attempted. For the selection method of the best suitable classifier, MATLAB 2020a classifier application without any optimisation is used. The aim was to find the best classifier in terms of prediction and speed for real time implementation purposes. In the classifier learner apps of MATLAB 2020a, all the classifier proposed are trained for each participant. However, only the ones with the best results according to a given set of metrics for every participant are retrieved for comparison purposes. They are: Fine Tree, linear discriminant, Naive Bayes (Gaussian), linear and quadratic SVM (one vs one), Fine KNN, Cosine KNN, weighted KNN, Ensemble subspace discriminant, and Ensemble subspace KNN. The dataset used is based on that presented in [28] and it is divided for each participant as a ratio of 70% for training and 30% for the testing phase. This dataset consists of 5 participant's gestures recorded. For each participant, a dataset of 10 samples per gesture is obtained for the command gesture and 12 samples for the non-command gesture acquired by implementing three (3) e-iTUG (walking, sitting, standing, turning, going upstairs, going down stair). However, for participant #1, which is one of the authors of this research work, a more much information of 20 samples per command gesture and five (5) e-iTUG test were recorded. The comparison metric used for this classification are as follows:

- ***The accuracy***: it's referred to as the level of good classification. It is a number between 0 and 100 and it is defined by the number of good predictions on the overall number of input samples.
- ***False Positive*** (FP): in this specific application, because of the issue of discriminating with high priority, command from non-command gesture, FP refers to cases in which the model knows that it is a non-command gesture, but the classifier predicts it as a command gesture. This is very important in the recognition process because of the need to keep the level of inappropriate activation of cobot operating mode very low when a non-gesture command is in process.

- ***False Negative (*FN): which infers the reverse scenario. E.g.: it is a command gesture but the classifier define it as non-command gesture (this refers to the sensibility of the system to react to user's input command gesture).
- ***Misclassification level (MC):*** it refers to level of confusion between different command gesture. It's important for such application as cobot behaviour must be predictive; when given an input gesture the cobot behaviour output needs to be known in advance.
- ***Prediction speed:*** it refers to how much observation is made in a given time. Its gives information about the classifier speed and for the application purpose, its indicate whether or not the classifier is suitable for real time. This information is a result obtained from the MATLAB 2020a classifier application.

Moreover, as inspired by [18], the above list of classifier has been augmented with a SVM (support machine) based Gaussian-RBF (Radial based Function) kernel classifier with the principles of one versus all, this means that, for each class i considered, it is always a binary operation that is implemented. The problem is reframed as belonging to the class i or not. So, the other classes are then labeled as non class i. Tables 8, 9, 10,11 and 12 presents the results for each participant.

**Table 8** : Classifier comparison results for participant #1

| Classifier | Accuracy % | FP % | FN % | MC % | Prediction speed (observation/sec) |
|---|---|---|---|---|---|
| Fine Tree | 92.8 | 3.61 | 0 | 3.61 | 33000 |
| Linear discriminant | 96.4 | 3.61 | 0 | 0 | 12000 |
| Naive Bayes (Gaussian) | 98.8 | 0 | 1.2 | 0 | 9600 |
| SVM linear one vs one | 96.4 | 2.41 | 0 | 1.2 | 2300 |
| SVM quadratic one vs one | 98.8 | 0 | 0 | 1.2 | 1500 |
| SVM RBF (Gaussian) One vs all | 98.8 | 0 | 0 | 1.2 | 5900 |
| KNN fine | 97.6 | 1.2 | 0 | 1.2 | 15000 |
| Cosine KNN | 96.4 | 1.2 | 0 | 2.4 | 9600 |
| Weighted KNN | 98.8 | 1.2 | 0 | 0 | 10000 |
| Ensemble subspace discriminant | 97.6 | 2.4 | 0 | 0 | 1200 |
| Ensemble subspace KNN | 98.8 | 0 | 0 | 1.2 | 1400 |

For participant #1 the best overall accuracy is achieved by Naïve Bayes, SVM (quadratic and gaussian), weighted KNN and Ensemble subspace KNN. However, weighted KNN was excluded for to recognizing non command gesture as command gesture. For this participant the best result is achieved using Naïve Bayes because of a low-rate misclassification of command gesture. Indeed, cobot operating mode requires the system to be predictable; thus, a low misclassification rate between command gesture is highly important. Although the presence of possible confusion between a command gesture recognised as a non command one, the rate is low and just refers to the capacity of the system to be sensitive to command input gesture. Moreover, the second-best classifier with the highest computation time is the SVM based Gaussian-RBF kernel function.

**Table 9 :** Classifier comparison results for participant #2

| Classifier | Accuracy % | FP % | FN % | MC % | Prediction speed (observation/sec) |
|---|---|---|---|---|---|
| Fine Tree | 84.8 | 0 | 3.03 | 12.12 | 6400 |
| Linear discriminant | 90.9 | 3.03 | 0 | 6.06 | 5300 |
| Naive Bayes (Gaussian) | N/A | N/A | N/A | N/A | N/A |
| SVM linear one vs one | 78.8 | 6.06 | 0 | 15.15 | 290 |
| SVM quadratic one vs one | 90.9 | 0 | 3.03 | 6.06 | 270 |
| SVM RBF (Gaussian) One vs all | 90.9 | 0 | 3.03 | 6.06 | 3700 |
| KNN fine | 90.9 | 0 | 3.03 | 6.06 | 1700 |
| Cosine KNN | 78.8 | 3.03 | 3.03 | 0 | 310 |
| Weighted KNN | 90.9 | 0 | 3.03 | 6.06 | 3100 |
| Ensemble subspace discriminant | 90.9 | 3.03 | 0 | 6.06 | 470 |
| Ensemble subspace KNN | 90.9 | 0 | 3.03 | 6.06 | 400 |

For participant #2 the best accuracy is achieved with linear discriminant, quadratic and gaussian SVM, fine KNN, weighted KNN, ensemble subspace discriminant and ensemble subspace KNN. Ensemble subspace discriminant and linear discriminant are rejected due to their ability to confuse non command gesture with command one which in fact is very bad compared to what is proposed by others. Moreover, considering the computation speed required, the Gaussian-RBF kernel SVM appear to be the best classifier. Naïve Bayes which was the best for participant one could not even compute, so it was rejected. In doing so it appears that even for participant one, SVM based Gaussian-RBF kernel is the best classifier.

**Table 10 :** Classifier comparison results for participant #3

| Classifier | Accuracy % | FP % | FN % | MC % | Prediction speed (observation/sec) |
|---|---|---|---|---|---|
| Fine Tree | 76.5 | 2.94 | 8.82 | 11.76 | 300 |
| Linear discriminant | 94.1 | 2.94 | 0 | 2.94 | 530 |
| Naive Bayes (Gaussian) | 94.1 | 0 | 2.94 | 2.94 | 950 |
| SVM linear one vs one | 88.2 | 0 | 0 | 11.8 | 150 |
| SVM quadratic one vs one | 85.3 | 2.94 | 0 | 11.8 | 290 |
| SVM RBF (Gaussian) One vs all | 97.1 | 0 | 0 | 2.94 | 2000 |
| KNN fine | 94.1 | 0 | 0 | 5.9 | 1100 |
| Cosine KNN | 97.1 | 2.94 | 0 | 0 | 3100 |
| Weighted KNN | 94.1 | 2.94 | 0 | 2.94 | 5500 |
| Ensemble subspace discriminant | 94.1 | 2.94 | 0 | 2.94 | 520 |
| Ensemble subspace KNN | 94.1 | 0 | 0 | 5.88 | 520 |

For Participant #3, the best accuracy result is achieved by SVM based Gaussian-RBF kernel and cosine KNN. However, due to the ability of Cosine KNN to confuse non gesture command with command one, the best classifier is achieved using SVM based Gaussian-RBF kernel.

**Table 11 :** Classifier comparison results for participant #4

| Classifier | Accuracy % | FP % | FN % | MC % | Prediction speed (observation/sec) |
|---|---|---|---|---|---|
| Fine Tree | 77.8 | 2.78 | 0 | 19.44 | 590 |
| Linear discriminant | 88.9 | 2.78 | 0 | 8.33 | 470 |
| Naive Bayes (Gaussian) | 72.2 | 5.56 | 2.78 | 19.44 | 720 |
| SVM linear one vs one | 86.1 | 2.78 | 2.78 | 8.33 | 210 |
| SVM quadratic one vs one | 88.9 | 0 | 0 | 11.1 | 210 |
| SVM RBF (Gaussian) One vs all | 88.9 | 0 | 0 | 11.1 | 850 |
| KNN fine | 86.1 | 2.78 | 0 | 11.1 | 630 |
| Cosine KNN | 61.1 | 5.56 | 0 | 33.33 | 2900 |
| Weighted KNN | 83.3 | 0 | 0 | 16.7 | 5500 |
| Ensemble subspace discriminant | 88.9 | 5.56 | 0 | 5.56 | 380 |
| Ensemble subspace KNN | 88.9 | 0 | 0 | 11.1 | 310 |

For participant #4 the best accuracy result with the highest prediction speed is achieved with SVM based Gaussian-RBF kernel classifier.

**Table 12 :** Classifier comparison results for participant #5

| Classifier | Accuracy % | FP % | FN % | MC % | Prediction speed (observation/sec) |
|---|---|---|---|---|---|
| Fine Tree | 77.8 | 2.78 | 0 | 19.44 | 14000 |
| Linear discriminant | 72.2 | 2.78 | 2.78 | 22.22 | 7100 |
| Naive Bayes (Gaussian) | 80.6 | 0 | 5.56 | 13.89 | 640 |
| SVM linear one vs one | 77.8 | 2.78 | 2.78 | 16.67 | 800 |
| SVM quadratic one vs one | 80.6 | 0 | 0 | 19.44 | 710 |
| SVM RBF (Gaussian) One vs all | 86.1 | 0 | 11.1 | 13.89 | 3100 |
| KNN fine | 88.9 | 0 | 0 | 11.1 | 4600 |
| Cosine KNN | 77.8 | 0 | 5.56 | 16.67 | 4600 |
| Weighted KNN | 86.1 | 0 | 2.78 | 11.11 | 2600 |
| Ensemble subspace discriminant | 80.6 | 2.78 | 0 | 16.67 | 550 |
| Ensemble subspace KNN | 94.4 | 0 | 0 | 5.56 | 470 |

For participant #5 the best classifier in term of accuracy is achieved using ensemble subspace KNN. For all the participants, it appears that SVM based Gaussian-RBF kernel is the best in term of accuracy, computation time (real time application) and false positive rate of non command gesture.

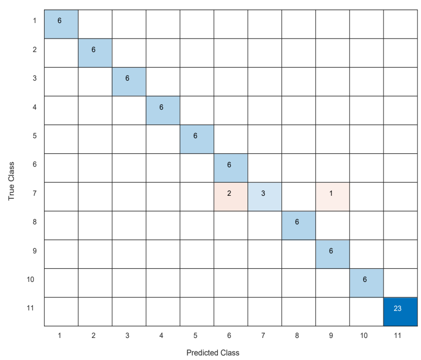
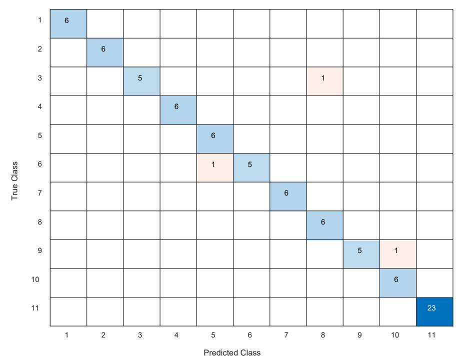
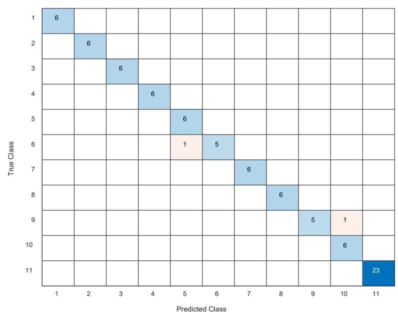
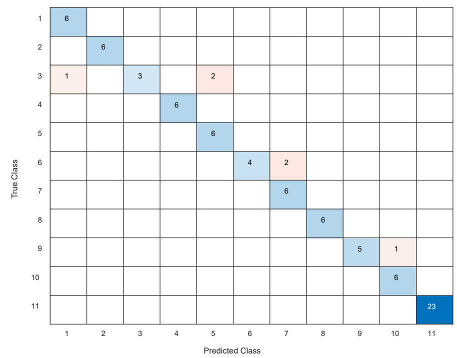
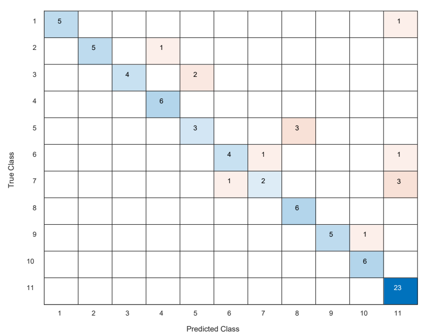
## 4. Experimentation and results

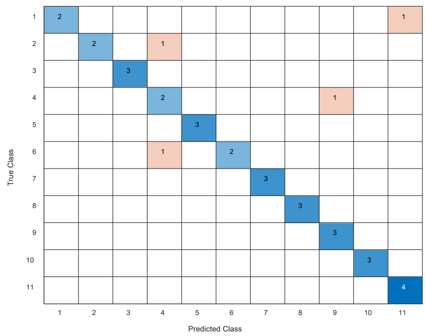
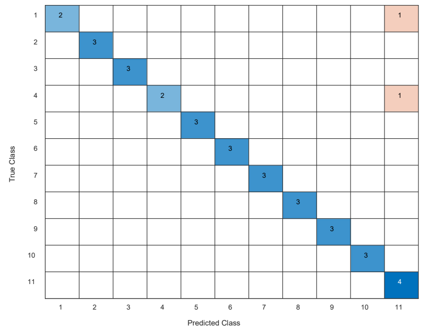
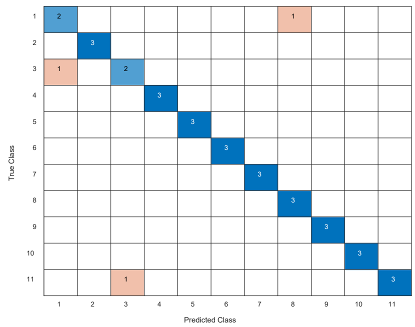
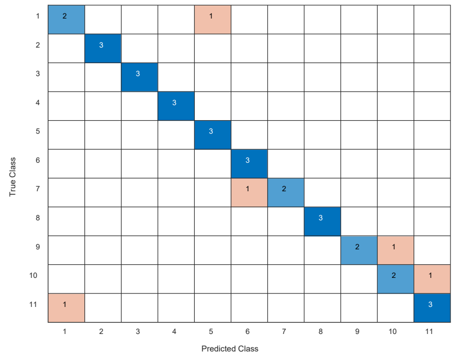
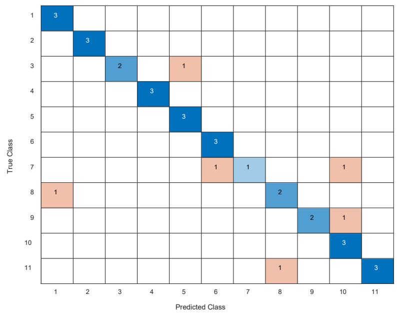
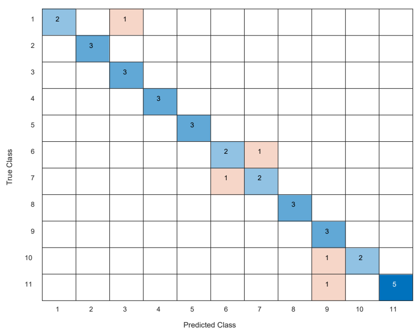
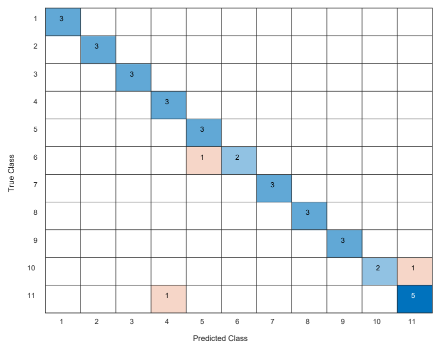
The experimentation is set in two main phases: 1) Training and testing for the first phases (section 4.1) and 2) real-time application for the second phase (section 4.2). Furthermore, the evaluation of the impact of changing the reference gesture on the recognition performances is presented in section 4.3.

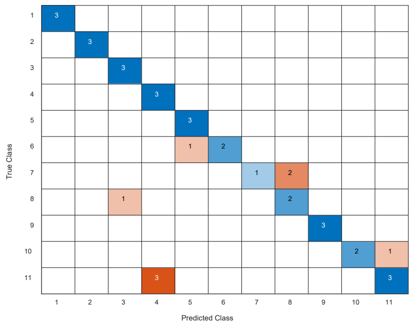
### *4.1. Training and testing*

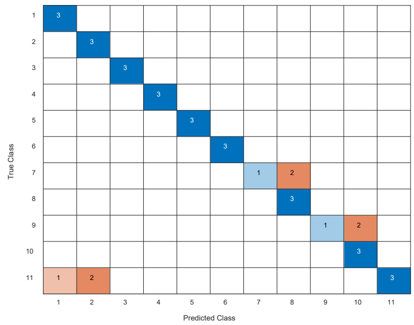
Based on the best classifier identified, the gaussian-RBF kernel SVM, the aim of this first phase is to demonstrate how well the proposed features approaches outperform temporal conventional ones such as mean, standard deviation, kurtosis, skewness, etc. In doing so, a comparison protocol is attempted by using the same training set in terms of number and index for each temporal characteristic and each participant. The same thing was done for the testing phase. The dataset used in this step is the same one used in section 3.5 above. **Table 13** presents the results of the different temporal features considered for foot gesture recognition; 70% of the data are used as training set with a 5-fold validation and 30% for testing set. The classes are labelled from 1 to 11 namely G1 to G10 for command gesture as defined in the dictionary in **section 3.3** and G11 for non command gesture as defined in the e-iTUG.

**Table 13** : Classification results

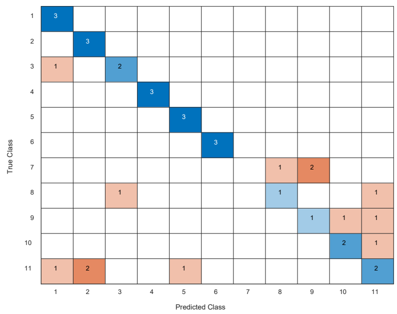
| | **Mean** | **Proposed DTW approach** |
|---|---|---|
| Participant 1 |  |  |
| | **Standard deviation** | **Skewness** |
| |  |  |
| | **Kurtosis** | |
| |  | |
| | | |
| | **Mean** | **Proposed DTW approach** |

<table>
<tr><td rowspan="5">Participant 2</td><td>
</td><td>
</td></tr>
<tr><td>Standard deviation</td><td>Skewness</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>Kurtosis</td><td></td></tr>
<tr><td>
</td><td></td></tr>
<tr><td colspan="3"></td></tr>
<tr><td rowspan="3">Participant 3</td><td>Mean</td><td>Proposed DTW approach</td></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td>Standard deviation</td><td>Skewness</td></tr>
</table>



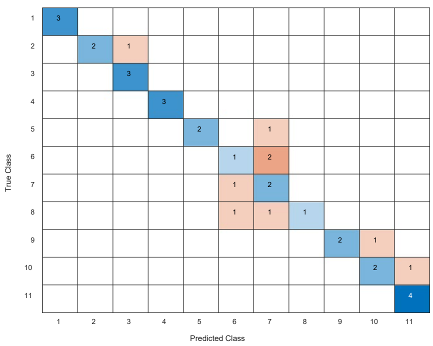



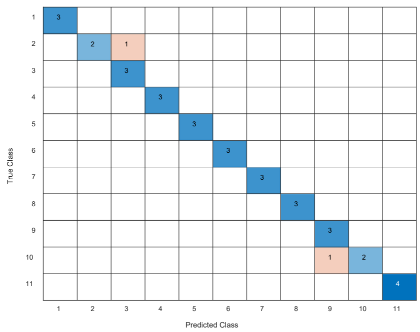

**Kurtosis**



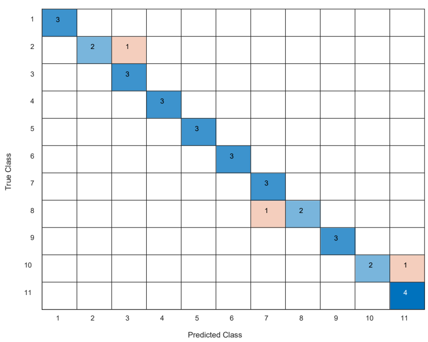

Participant 4

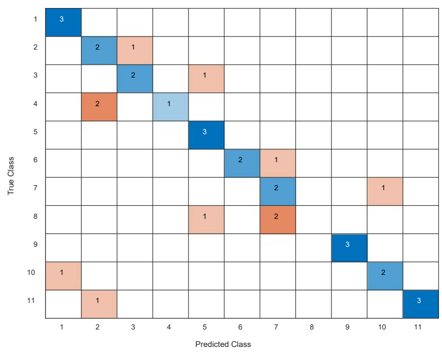
| **Mean** | **Proposed DTW approach** |
|---|---|



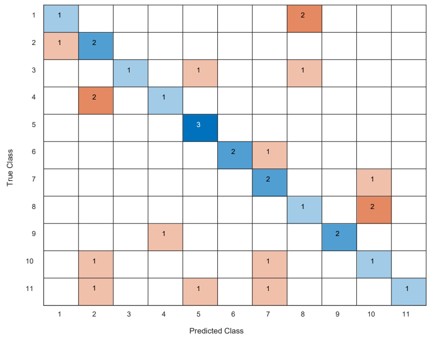
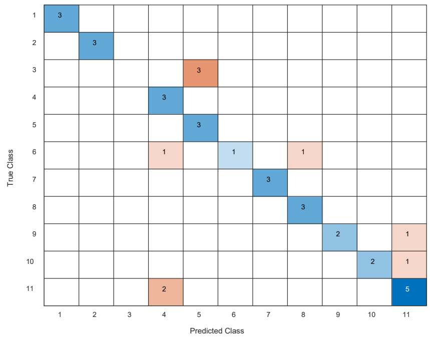
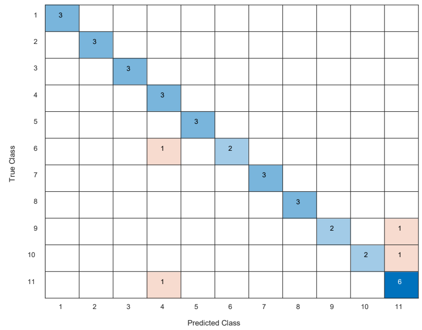
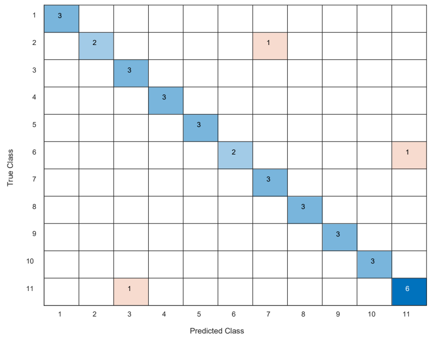
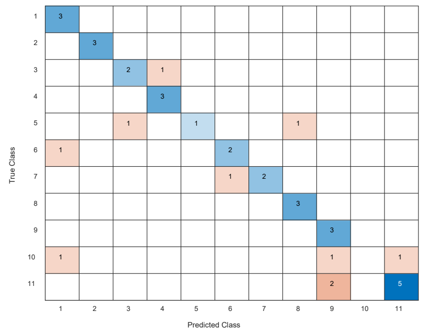
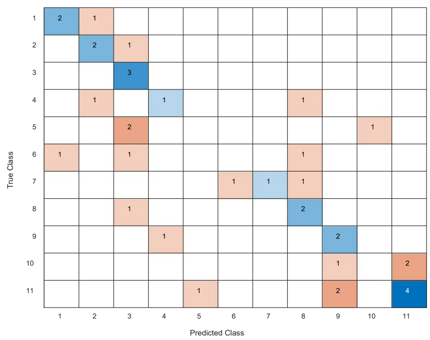




| **Standard deviation** | **Skewness** |
|---|---|




**Kurtosis**

| | | |
|---|---|---|
| |  | |
| | | |
| Participant 5 | **Mean** | **Proposed DTW approach** |
| |  |  |
| | **Standard deviation** | **Skewness** |
| |  |  |
| | **Kurtosis** | |
| |  | |
| | | |

Form the results above, different metrics were estimated like the ones presented in section 3.7. Table 14 presents the different metrics for each participant.

**Table 14 :** Comparison metric of differents set of features used for SVM classifier for each participant

| Participant 1 | | | | |
|---|---|---|---|---|
| % | Accuracy | FP | FN | MC |
| Proposed DTW feature | 96.39 | 0 | 0 | 3.61 |
| Mean | 96.39 | 0 | 0 | 3.61 |
| Standard deviation | 97.59 | 0 | 0 | 2.41 |
| Kurtosis | 92.77 | 0 | 0 | 7.33 |
| Skewness | 83.13 | 0 | 6.02 | 10.84 |

| Participant 2 | | | | |
|---|---|---|---|---|
| % | Accuracy | FP | FN | MC |
| Proposed DTW feature | 94.12 | 0 | 5.88 | 0 |
| Mean | 88.24 | 0 | 2.94 | 8.82 |
| Standard deviation | 91.18 | 2.94 | 0 | 5.88 |
| Kurtosis | 85.29 | 2.94 | 29.4 | 8.82 |
| Skewness | 82.35 | 2.94 | 0 | 14.71 |

| Participant 3 | | | | |
|---|---|---|---|---|
| % | Accuracy | FP | FN | MC |
| Proposed DTW feature | 91.67 | 2.78 | 2.78 | 2.78 |
| Mean | 83.33 | 2.78 | 0 | 11.11 |
| Standard deviation | 75 | 8.33 | 2.78 | 11.11 |
| Kurtosis | 77.78 | 8.33 | 0 | 11.11 |
| Skewness | 61.11 | 8.33 | 8.33 | 16.67 |

| Participant 4 | | | | |
|---|---|---|---|---|
| % | Accuracy | FP | FN | MC |
| Proposed DTW feature | 94.12 | 0 | 0 | 5.88 |
| Mean | 73.53 | 0 | 2.94 | 23.53 |
| Standard deviation | 91.18 | 0 | 2.94 | 5.88 |
| Kurtosis | 67.65 | 2.94 | 0 | 29.41 |
| Skewness | 50 | 8.82 | 0 | 41.18 |

| Participant 5 | | | | |
|---|---|---|---|---|
| % | Accuracy | FP | FN | MC |
| Proposed DTW feature | 89.19 | 2.7 | 5.41 | 2.7 |
| Mean | 75.68 | 5.41 | 5.41 | 13.51 |
| Standard deviation | 91.89 | 2.7 | 2.7 | 2.7 |
| Kurtosis | 72.97 | 5.41 | 2.7 | 18.92 |
| Skewness | 45.95 | 8.11 | 5.41 | 40.54 |

For participants #1 and #5, the best classification is achieved by means of standard deviation approach. Moreover, for the same participant, the proposed DTW approach appears to end up with a high level of accuracy even though it is not considered the best in terms of accuracy, false negative and misclassification rate. However, for participants #2, #3 and #4, the best classification rate is achieved using the proposed DTW approach. Furthermore, for participant #2, the standard deviation based approach, aside of presenting a lower accuracy level, presents a rate of false positive which is different from zero. This means that for such participant, the use of standard deviation approach can end up in a case when the user is implementing non command gestures such as walking, turning etc. and the system recognizes it as an input command for the cobot. This in fact is very bad compared to the result achieved using the proposed DTW approach. Another point of interest is observed in participant #3; it appears that the classification rate is very low with the use of standard deviation and has a high rate of false positive detection of non-command gesture.

In conclusion, from one participant to another, it appears that even if there are some cases where the use of standard deviation approach alone slightly outperforms the proposed DTW, there are cases where the classification result is very bad compared to the proposed DTW approach. Thus, they require for each input participant, to implement feature selection phase to rightly choose of the best temporal feature to use for implementation purposes. However, the proposed DTW is more robust to individual specificity. It can accurately classify foot gesture for different participant better than classical approaches as mean, kurtosis, skewness, standard deviation by only comparing results of the signal corresponding to the standing position of each participant at any time.

### *4.2. Real-time evaluation as the application*

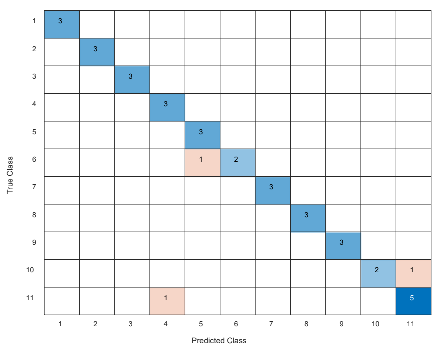
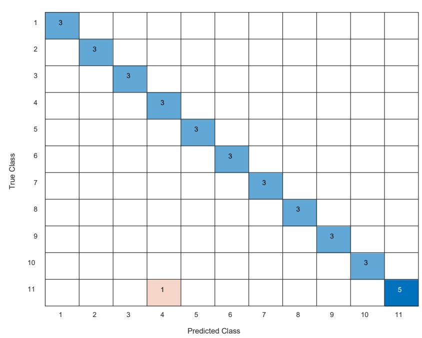
Online cobot operating mode control is evaluated using the proposed DTW-SVM approach based on the model trained in section 4.1 for each participant. The recognition rate for all five participants, in real time was at a range of 66% of accuracy with a set of FP (false positive) at 8% (mainly non command gesture (G11) confused as G4), false negative (FN) at 10% and misclassification between command gesture (MC) of 16%. The biggest confusion was observed between (G9 and G10) and (G5 and G6).

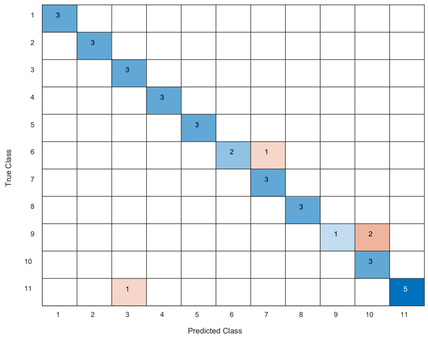
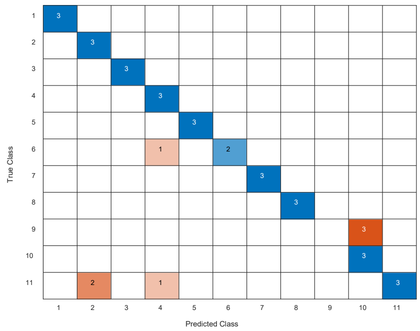
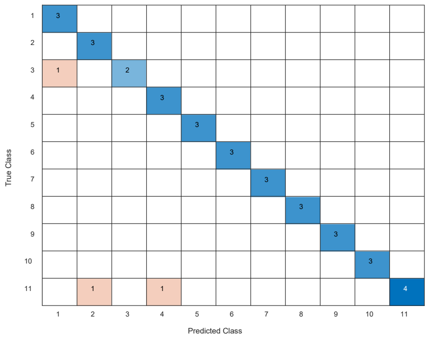
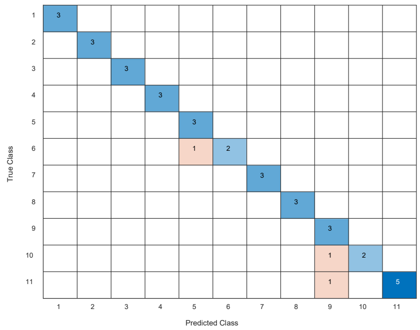
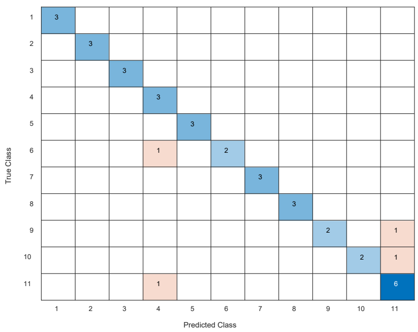
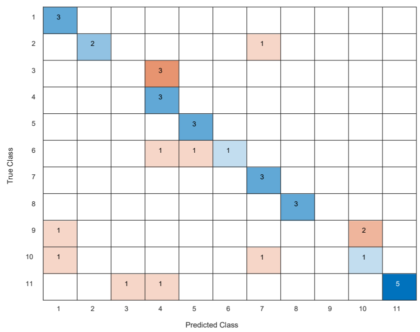
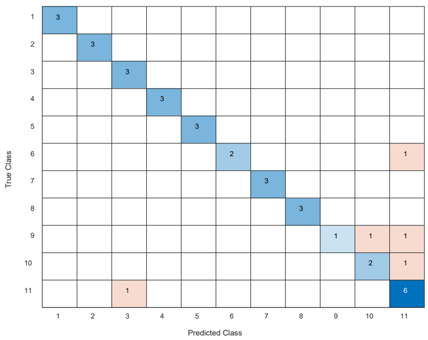
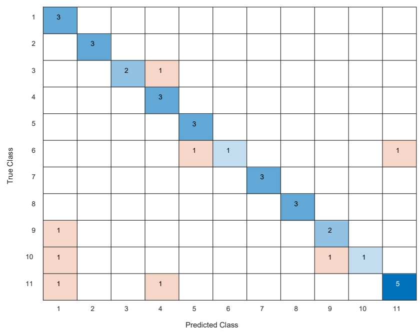
Moreover, because real-time application mainly relies on the capacity of the system to detect the command gestures in time, an evaluation was conducted with the different participant to estimate the computation time. It appears that the computation time of the proposed DTW approach based gaussian SVM classifier is greatly adapted for such a non-real time platform as our MS window computer. The computation time achieved for one classification was about 3.7418e-4 sec obtained using tic and toc MatLAB function used in a MatLAB Script box included in Simulink. It is a very conservative measure based on MatLAB implementation and execution. The Simulink is executed with the real time workshop and the frequency transmission rate from the insole to Simulink is 500Hz.

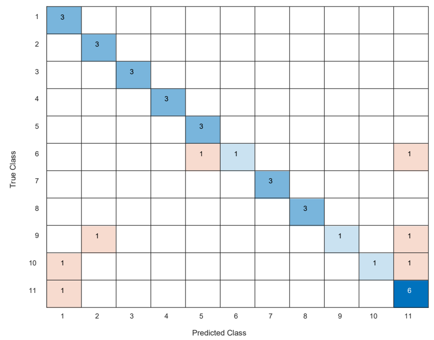
### *4.3. Impact of reference gesture changing on the classification rate.*

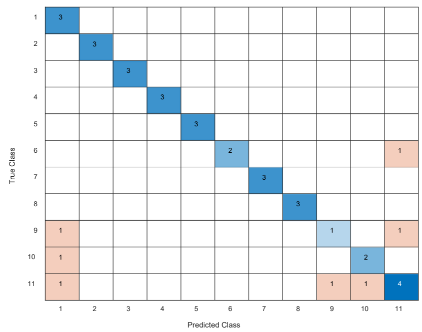
The aim of this research work is to present the usefulness if using standard DTW computation based on one reference gesture for cobot operating mode. Till now, the focused was put on the use of the standing gesture as the reference gesture because of the assumption that every command or non command gesture at some point pass through the standing position before been executed. This section presents foot gesture recognition result when changing randomly the reference gesture being use. To presents this approach, it has been decided to conduct for two (2) participants (#3 and #5) a set of five (5) changing of reference gesture. In doing so, the same dataset and comparison approach together with the same metrics explored in section 4.1 were used. Table 15 display the results of each participant and a reference taken randomly from five different classes.

**Table 15 :** Confusion matrices results of reference gesture change

| | **Reference at standing position** | **Reference at class G1** |
|---|---|---|
| Participant 3 |   |   |

<table>
<tr><td rowspan="4"></td><th>Reference at class G2</th><th>Reference at class G6</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><th>Reference at class G8</th><th>Reference at class G9</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><td colspan="3"></td></tr>
<tr><td rowspan="5">Participant 5</td><th>Reference at standing position</th><th>Reference at class G1</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><th>Reference at class G2</th><th>Reference at class G6</th></tr>
<tr><td>
</td><td>
</td></tr>
<tr><th>Reference at class G8</th><th>Reference at class G9</th></tr>
</table>




The results comparison metric is presented in table 16.

**Table 16 :** Classification metrics for reference changing signal

| Participant 3 | | | | | Participant 5 | | | | |
|---|---|---|---|---|---|---|---|---|---|
| % | Accuracy | FP | FN | MC | % | Accuracy | FP | FN | MC |
| **Reference at standing position** | 91.67 | 2.78 | 2.78 | 2.78 | **Reference at standing position** | 89.19 | 2.7 | 5.41 | 2.7 |
| **Reference at class G1** | 97.22 | 2.78 | 0 | 0 | **Reference at class G1** | 64.86 | 5.41 | 0 | 29.73 |
| **Reference at class G2** | 88.89 | 2.78 | 0 | 8.33 | **Reference at class G2** | 86.49 | 2.7 | 8.11 | 2.7 |
| **Reference at class G6** | 80.56 | 8.33 | 0 | 11.11 | **Reference at class G6** | 78.38 | 5.41 | 2.7 | 13.51 |
| **Reference at class G8** | 91.67 | 5.56 | 0 | 2.78 | **Reference at class G8** | 81.08 | 2.7 | 8.11 | 8.11 |
| **Reference at class G9** | 91.67 | 2.78 | 0 | 5.56 | **Reference at class G9** | 81.08 | 8.11 | 5.41 | 5.41 |

By taking a random reference for participant #5, it appeared that the change in reference signal led to a change in the classification result. Moreover, for this participant, it seemed that, the use of the standing posture as the reference signal gives the best result. However, for participant #3, the best results is achieved using a random reference signal taken in class G1. Taking the standing position as reference gesture is not the best but is all the least able to accurately classify between different gestures. The change in reference gesture can lead to a decrease of performance as seen with participant 3 or in an increase of performance as seen with participant #5.

Based on these results, it appears that it is possible to find better reference gesture for a given participant. But one can imply that when the standing position is used as the reference one the recognition rate is very good without the need to actively search for the best one.

## 5. Discussion

The aim of this study was to analyse whether or not the proposed DTW feature approach based on a single reference gesture (standing pose) can be useful for online foot gesture cobot control. There are with four (4) main conclusions regarding the performance of the proposed approach as feature input for a classical SVM classifier:

1) The proposed DTW approach can well discriminate the ten (10) command gesture between them as well as non command gesture with the lowest accuracy rate of 88% obtained in the training/ testing phase. Moreover, even if in real time implementation, the overall accuracy dropped to 66% due to either confusion between command gesture (G9 and G10) or (G5 and G6) and confusion between the non command gesture.
2) The proposed DTW approach used alone can outperform common temporal feature based approach and can be easily implemented through different participants with high accuracy.
3) When looking at the classification results of the proposed DTW approach, aside for participants #3 and #5, the level of false positive is very low. Thus, one can imply that it's possible to discriminate between command and non command gesture without the need of a locking gesture even if in real time evaluation, confusion between command gesture G4 and non command gesture G11 exist. The only requirement is a secure process in order to avoid unwanted activation of G4.
4) The classification rate of the proposed approach is highly dependent on the nature of the reference gesture being used as shown in section 4.3. One assurance given at the end of this work is to say that, by using the standing posture as a reference gesture for online cobot control based foot recognition system, the accuracy is highly to be very high and at some point, be the highest. Even though all the other possibility of using another reference gesture for the approach hasn't been tried, as far as this article author knowledge, the best result considering all the five participants is achieved by using the standing pose of each other as the reference gesture.

## 6. Limit of the study

Limitations in this study can been seen on three main aspects. Firstly, the proposed classification scheme uses only five participants since the approach is dependent on participants. Therefore, the necessity to compute training for any new users appears and the number of participants is likely enough to demonstrate this situation. Secondly, the study has been conducted in a supervised environment where noise arises from environmental consideration like vibrations has been taken out, thus requires enhance disturbance robustness for all industrial applications. Thirdly, the proposed approach is not tested in a real industrial case study, where high accuracy and responsiveness is needed to achieve a safe human robot interaction.

## 7. Conclusions and future works

This paper presents a foot gesture recognition scheme for cobot control based on DTW features input for an SVM classifier. Foot gestures are collected from an insole device and then DTW computation with the reference signal is done and later transmitted to SVM classifier for activity (command) recognition. Then, an interface with a UR5 robot is implemented in order to operate robot change control-based foot gesture recognition.

There are three hypotheses suggested in section 2. The goal is to demonstrate the possibility of using only one reference signal (standing position in our case) as DTW based feature extraction methods. The study shows the ability of the proposed scheme to recognize command foot gestures (10) and to actively discriminate between non-command gestures and others (hypothesis 1 is confirmed). Based on the results, the classification algorithm is mainly dependant of the nature of the reference gesture being use (hypothesis 2 is confirmed) and a static reference gesture can be used (hypothesis 3 is confirmed). .

Future research aims at the real time deployment of the proposed solution in a real industrial case scenario and for the perspective of generalisation purposes so that a more refined method can been used for two or three users without the need to conduct training phase. Moreover, the automatically detection of the best reference gesture (signal) to be used for a given dataset without prior knowledge of the purpose application is still in exploration.

**Author Contributions:** Conceptualization, G.V.T.D. and M.O.; methodology, G.V.T.D, and M.O.; software, G.V.T.D; validation, G.V.T.D.; formal analysis, G.V.T.D.; investigation, G.V.T.D.; resources, M.O.; data curation, G.V.T.D.; writing—original draft preparation, G.V.T.D.; writing—review and editing, G.V.T.D., M.O.; visualization, G.V.T.D. and M.O.; supervision, M.O, R.M..; project administration, M.O.; funding acquisition, M.O. and R.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work received financial support from the Fonds de recherche du Québec—Nature et technologies (FRQNT), under grant number 2020-CO-275043 (Ramy Meziane) and NSERC Discovery grant number RGPIN-2018-06329 (Martin Otis). This project uses the infrastructure obtained by the Ministère de l'Économie et de l'Innovation (MEI) du Quebec, John R. Evans Leaders Fund of the Canadian Foundation for Innovation (CFI) and the Infrastructure Operating Fund (FEI) under the project number 35395.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References


1. Krüger, J., T.K. Lien, and A. Verl, *Cooperation of human and machines in assembly lines.* CIRP annals, 2009. **58**(2): p. 628-646.
2. Lopes, M., et al., *Semi-Autonomous 3rd-Hand Robot.* Robot. Future Manuf. Scenar, 2015. **3**.
3. Safeea, M., P. Neto, and R. Bearee, *On-line collision avoidance for collaborative robot manipulators by adjusting off-line generated paths: An industrial use case.* Robotics and Autonomous Systems, 2019. **119**: p. 278-288.
4. Neto, P., et al., *Gesture-based human-robot interaction for human assistance in manufacturing.* The International Journal of Advanced Manufacturing Technology, 2019. **101**(1): p. 119-135.
5. Ende, T., et al. *A human-centered approach to robot gesture based communication within collaborative working processes*. in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2011. IEEE.
6. Jiang, W., et al., *Wearable on-device deep learning system for hand gesture recognition based on FPGA accelerator.* Mathematical Biosciences and Engineering, 2021. **18**(1): p. 132-153.
7. Juang, J.-G., Y.-J. Tsai, and Y.-W. Fan, *Visual recognition and its application to robot arm control.* Applied Sciences, 2015. **5**(4): p. 851-880.
8. Aswad, F.E., et al., *Image Generation for 2D-CNN Using Time-Series Signal Features from Foot Gesture Applied to Select Cobot Operating Mode.* Sensors, 2021. **21**(17): p. 5743.
9. Crossan, A., S. Brewster, and A. Ng, *Foot tapping for mobile interaction.* Proceedings of HCI 2010 24, 2010: p. 418-422.
10. Hua, R. and Y. Wang, *A customized convolutional neural network model integrated with acceleration-based smart insole toward personalized foot gesture recognition.* IEEE Sensors Letters, 2020. **4**(4): p. 1-4.
11. Valkov, D., et al. *Traveling in 3d virtual environments with foot gestures and a multi-touch enabled wim*. in *Proceedings of virtual reality international conference (VRIC 2010)*. 2010.
12. Gudmundsson, S., T.P. Runarsson, and S. Sigurdsson. *Support vector machines and dynamic time warping for time series*. in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. 2008. IEEE.
13. Kate, R.J., *Using dynamic time warping distances as features for improved time series classification.* Data Mining and Knowledge Discovery, 2016. **30**(2): p. 283-312.
14. Li, W., P. Shi, and H. Yu, *Gesture recognition using surface electromyography and deep learning for prostheses hand: state-of-the-art, challenges, and future.* Frontiers in neuroscience, 2021. **15**: p. 621885.
15. Fan, M., et al. *An empirical study of foot gestures for hands-occupied mobile interaction*. in *Proceedings of the 2017 ACM International Symposium on Wearable Computers*. 2017.
16. Sasaki, T., et al., *MetaLimbs: multiple arms interaction metamorphism*, in *ACM SIGGRAPH 2017 Emerging Technologies*. 2017. p. 1-2.
17. Kim, T., et al., *Usability of foot-based interaction techniques for mobile solutions*, in *Mobile Solutions and Their Usefulness in Everyday Life*. 2019, Springer. p. 309-329.
18. Maragliulo, S., et al., *Foot gesture recognition through dual channel wearable EMG System.* IEEE Sensors Journal, 2019. **19**(22): p. 10187-10197.

19. Huang, Y., et al., *Design and evaluation of a foot-controlled robotic system for endoscopic surgery.* IEEE Robotics and Automation Letters, 2021. **6**(2): p. 2469-2476.

20. Asghar, A., et al., *Review on electromyography based intention for upper limb control using pattern recognition for human-machine interaction.* Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine, 2022. **236**(5): p. 628-645.

21. Kiranyaz, S., et al. *1-D convolutional neural networks for signal processing applications*. in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* 2019. IEEE.

22. Ismail Fawaz, H., et al., *Deep learning for time series classification: a review.* Data mining and knowledge discovery, 2019. **33**(4): p. 917-963.

23. Iwana, B.K. and S. Uchida, *Time series classification using local distance-based features in multi-modal fusion networks.* Pattern Recognition, 2020. **97**: p. 107024.

24. *Datasheet ESP32*. 11/03/2021]; Available from: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf.

25. Barkallah, E., et al., *Wearable devices for classification of inadequate posture at work using neural networks.* Sensors, 2017. **17**(9): p. 2003.

26. *Datasheet Mpu9250*. 02/03/2017]; Available from: https://www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf.

27. Wu, C., et al., *sEMG measurement position and feature optimization strategy for gesture recognition based on ANOVA and neural networks.* IEEE Access, 2020. **8**: p. 56290-56299.

28. TCHANE DJOGDOM, Gilde Vanel; Meziane, Ramy; Otis, Martin, 2022, "*Insole sensor data for foot gestures*", https://doi.org/10.5683/SP3/C4UQCW, Borealis, V1.

29. Lin, C., et al., Foot Gesture Recognition with Flexible High-Density Device Based on Convolutional Neural Network. In : 2021 6th IEEE International Conference on Advanced Robotics and Mechatronics (ICARM). IEEE, 2021. p. 306-311.

30. Lyons, K.R. et Joshi, S. S., Upper limb prosthesis control for high-level amputees via myoelectric recognition of leg gestures. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2018, vol. 26, no 5, p. 1056-1066.

31. Chawuthai, R., et Sakdanuphab, R., The analysis of a microwave sensor signal for detecting a kick gesture. In : 2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST). IEEE, 2018. p. 1-4.