Robust classification of grasped objects in intuitive human-robot collaboration using a wearable force-myography device

Nadav D. Kahanowich and Avishai Sintov

***Abstract*—Feasible human-robot collaboration requires intuitive and fluent understanding of human motion in shared tasks. The object in hand provides the most valuable information about the intended task of a human. In this letter, we propose a simple and affordable approach where a wearable force-myography device is used to classify objects grasped by a human. The device worn on the forearm incorporates 15 force sensors that can imply about the configuration of the hand and fingers during grasping. Hence, a classifier is trained to easily identify various objects using data recorded while holding them. To augment the classifier, we propose an iterative approach in which additional signals are taken in real-time to increase certainty about the predicted object. We show that the approach provides robust classification where the device can be taken off and placed back while maintaining high accuracy. The approach also improves the performance of trained classifiers that initially produced low accuracy due to insufficient data or non-optimal hyper-parameters. Classification success rate of more than 97% is reached in a short period of time. Furthermore, we analyze the key locations of sensors on the forearm that provide the most accurate and robust classification.**

***Index Terms*—Human-Robot Collaboration, Intention Recognition.**

I. Introduction

In collaborative tasks between two humans, when one is assisted by another, some tasks can be done intuitively without verbal communication. Once one human sees the motion of his human fellow, usually his arms and manipulated objects, he recognizes an intended upcoming task and exerts supporting actions. For instance, when assembling two parts together, a human assistant can hold one part for support or can handover appropriate tools. Similarly, an upper-limb amputee would need another hand to open a bottle. In *Human-Robot Collaboration* (HRC), robotic arms should do the same to support a human in completing shared tasks [1]. Moreover, robot assistance could be carried out to prevent endangerment of humans. The major attempt has been to integrate HRC approaches in robotics to support humans with disabilities [2] or in performing tasks that require more than one participant [3].

Manuscript received: October 15, 2020; Revised: December 29, 2020; Accepted: January, 29, 2021.

This paper was recommended for publication by Editor Allison Okamura upon evaluation of the Associate Editor and Reviewers' comments.

This work was supported by the Israel Science Foundation (grant No. 1565/20).

N. Kahanowich and A. Sintov are with the School of Mechanical Engineering, Tel-Aviv University, Israel. e-mail: {kahanowich, sintov1}@{mail,tauex}.tau.ac.il.

Digital Object Identifier (DOI): .

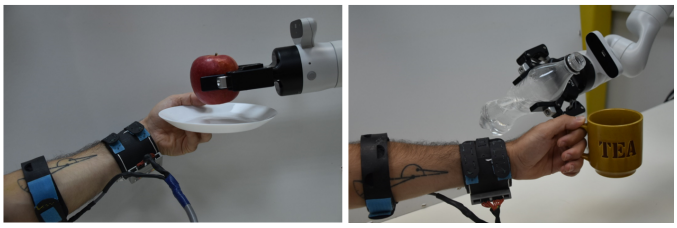


Fig. 1. Intuitive collaboration between a robotic arm and a human in (left) food serving and (right) pouring from a bottle to a mug. A wearable Force-Myography device on the human arm is used to measure musculoskeletal activities that imply on the object that is grasped by the human.

A challenging problem in HRC is to signal the robot of a desired assistive task efficiently and naturally. Some HRC solutions, however, provide non-intuitive control methods such as human gestures [4] or sensing brain activities [5]. Another common approach is Electromyography (EMG) [6] in which electrical signals from the muscles are measured through electrodes and translated to limb movements. The EMG technology, however, requires large and costly equipment, and its accuracy is compromised by sweat, electrode placement and crosstalk [7]. Early work by Amtf et al. [8] have introduced the use of body-worn force sensors to identify patterns in forearm muscle activities. Further work have shown the possibility to identify hand gestures using Force-Myography (FMG) signals [9], [10]. FMG measures perturbations of the musculoskeletal system and has been reported to be simple to acquire with a relatively high-accuracy [11]. Consequently, acquired FMG data was used in data-based classification of hand gestures [12]. However, data was recollected and a classifier was trained each time that the sensors have been placed on the arm. Hence, once the sensors have been dislocated, the previously trained classifier significantly loses its accuracy. As opposed to EMG, FMG requires low-cost sensors and a simple portable acquisition device (e.g., Arduino board), and is less sensitive to sensor positioning variations [7], [13]. Furthermore and according to the author's knowledge, there has not been any attempt to reason about objects within-hand through FMG measurements which is crucial for identification of intended tasks in HRC.

In this work, we use FMG measurements to identify objects in a human hand which, in turn, imply about the intended task (Figure 1). An object is characterised by its geometry and weight which are reflected by the musculoskeletal state of the arm grasping it. Hence, we investigate the ability of a classifier,

trained with measured FMG signals, to classify a grasped object from a given set of objects. We aim to rely solely on a low-cost FMG device directly strapped on the human forearm to provide an affordable solution. Previous work included either lower [14] or upper forearm [15] FMG bands. Yet, it is not clear what are the sensing locations required for accurate and robust predictions. Hence, we provide an analysis of the classification accuracy with regards to the placement of the sensors. In addition, we hypothesize that better coverage along the forearm will augment the model and provide better accuracy. Hence, the lightweight wearable FMG device incorporates 15 force-sensitive resistor sensors placed on the lower and upper forearms. Data collected from the FMG device is further used to train a robust classifier to identify objects in hand. We focus on observing object classification using FMG for a single participant while leaving global classifiers for future work. We show that the classifier is robust to re-positioning of the device, i.e., once the classifier has been trained over collected data, it maintains its accuracy even if the device has been taken off previously.

While we show that we can acquire a relatively accurate and robust classifier of objects in hand, we propose an *Iterative Classification* (IC) algorithm to further improve classification performance. IC is used to increase prediction certainty by sampling additional FMG data. We exploit the continuous time frame in which the user holds a certain object making more samples instantly available. The iterative method can be used with any classifier that provides a class probability distribution. Under some conditions, the iterative process will improve accuracy and robustness even for a classifier that does not provide high success rate by itself. Hence, IC can be exerted on classifiers trained with insufficient data or over a non-optimal classifier model. To conclude, the main contribution is an approach that enables accurate and robust classification of objects in a human hand using affordable, lightweight and easy to use hardware.

The proposed algorithm is able to provide fast and reliable results in real-time which is essential for practical HRC. For a task planner to decide about a future robot assistive manipulation, it must first be informed of the object in the human hand. The object provides significant information about the upcoming task even before the human arm has begun to move and enables a substantial reduction in the set of possible actions to be performed by the human. Hence, the robot can infer about future actions of the human beforehand and plan a trajectory accordingly [16]. The plan will be updated in real-time with more information about the human and object motion. Related work on the above topics are discussed next.

II. Related Work

In HRC, common control methods that have been used to efficiently signal the robot of a desired assistive task are quite limited. Many assistive robots are specially designed for a specific task in a priori known environment [17]. Other assistive arms use non-intuitive control methods such as predefined human gestures and gazes [4], [18] or brain-computer interface that sense brain activities [5]. Specially designed gloves such as in [19] measure both acceleration and flexion for motion capturing and virtual reality. However, these gloves limit the tactile sensation of the user and thus, not suitable for general use. A widely researched approach is to acquire and classify neurological activities through Electromyography (EMG) [6], [20], [21]. EMG detects electrical signals generated by muscle tissue and implies the human subject's intention. Even though this method frees the hand and allows full tactile sensation, it usually requires expensive and highly sizable equipment [7]. In addition, different artifacts and crosstalk may decrease the quality of the signal [22].

A different technique to recognize intention is through Force-Myography (FMG) where force sensors capture radially directed force distributions through expansion and contraction of the musculoskeletal system [23]. Prior works [8], [9] have shown that exterior sensing of muscle surface perturbations incorporates important information about task activity. A work study in [10], [11] has advanced the idea by identifying finger motions from muscle perturbation via force sensors on the forearm. While their methodologies did not allow having a wearable system for real-time feedback, these studies have established the feasibility of using FMG for monitoring upper-extremities gestures. More recently a wearable device has been proposed [15] that is composed of a linear set of eight force sensors. A classifier was trained to identify in real-time hand postures. Similarly, a wearable FMG feedback system was used to detect four basic hand motions in rehabilitation analysis [12].

Motion prediction based on extracted features of human motion can be divided into two categories, model-based and model-free. Model-based approaches establish an analytical function for mapping between measured features to the kinematic or dynamic behavior [24]. Model parameters are studied and refined through experiments but are challenging to evaluate accurately. On the other hand, the model-free approach is a black-box mapping acquired through machine learning [25]. A large amount of data is used to train an Artificial Neural-Network to map sensed features to the current gesture or posture of the hand. As such, the works in [26], [27] trained a neural network to map EMG signals to joint angles of the upper-limb.

III. Method

A. Wearable FMG device

As previously described, prior work included either upper or lower forearm bands. To improve accuracy and to analyze the dominant measurement locations, we have combined both and fabricated a wearable device with wider coverage along the forearm). The device is composed of 15 low-cost Force-Sensitive Resistors (FSR), model FSR-402 by Interlink Electronics. FSR sensors are made of polymer films that vary their electrical resistance upon changing pressure on their surface. The device consists of three main components: (a) upper forearm band with six FSR sensors, (b) lower forearm band with nine sensors organized in two rows and (c) a data acquisition system based on an Arduino Mega 2560 board. The FSR sensors were positioned in equal spacing along the bands

as seen in Figure 2. We note that the device includes two bands for prototyping considerations but can easily be fabricated as one unit. The bands were fabricated by 3D printing with an elastic polymer (Thermoplastic elastomer). They include a flexible bulge for each sensor to ensure proper attachment to the skin while maintaining flexibility during arm motion. Each FSR sensor is connected to an analog pin of the Arduino through a voltage divider of 4.7kΩ resistor. Such acquisition configuration provides real-time data stream of all the given sensors in a frequency of up to 300Hz. The described system is composed of low-cost and light-weight hardware which is appealing and suited for easy arm movements.

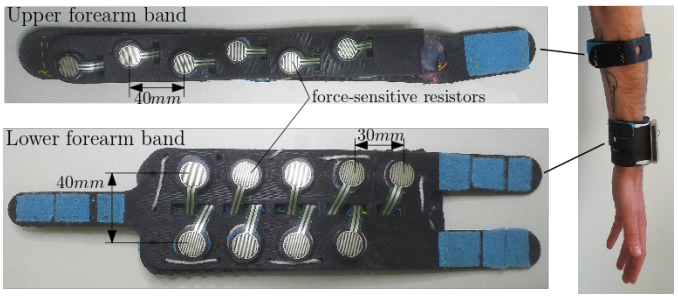


Fig. 2. Prototype of the FMG wearable device made of two parts for the upper and lower forearm. Each part includes a set of force-sensitive resistors (FSR) designed to sense perturbations of the musculoskeletal system.

B. Neural-Network classifier

We aim to identify an object grasped by the human solely by measuring FMG signals measured by the device. Given a set of m objects $\{\mathcal{O}_1, \dots \mathcal{O}_m\}$, we require to identify an object from the set. That is, we require real-time classification based on pattern recognition of the input signals. This is achieved through supervised learning with the use of a feed-forward Neural-Network (NN) trained over labeled signals.

Let $\phi \in \mathbb{R}^n$ be the observable state of the musculoskeletal system measured by the FMG system with n FSR sensors. For each object $\mathcal{O}_i$, training data is collected by holding it as seen in Figure 3. We record grasps of objects as intended during tasks, e.g., grasping scissors by their ring handles. To increase data variance, the data was recorded in various arm postures, i.e., while arbitrary moving the shoulder, elbow and wrist. Ultimately, the resulting training data is a set of M labeled FMG signals $\Phi = \{(\phi_1, l_1), \dots, (\phi_M, l_M)\}$ where label l_i corresponds to object $\mathcal{O}_{l_i}$. However and as described previously, we aim to acquire a model robust to replacing of the FMG device. Recording data after a one time positioning of the device will not be robust and a trained model is most likely to fail after taking-off and re-positioning.

Classification failure while using the above formulation will happen for two reasons: inability to re-position the device at the same location and inability to tighten it with the same forces each time. For the former challenge, we collect data in N episodes where, in each episode, the device is taken-off and re-positioned. To cope with the different tightening forces at each episode, we consider episode values relative to the initial forces after strapping in. Thus, at the beginning of episode j, the user is required to perform a simple calibration process in which the muscles are relaxed prior to picking up the object. The FMG baseline signal $\phi_o^{(j)}$ during the relaxation is subtracted from the episode measurements to decrease variance, i.e., a signal is now given by $\tilde{\phi}_i^{(j)} = \phi_i^{(j)} - \phi_o^{(j)}$. Next, since the data is significantly noisy, we apply a simple Mean Filter of width w to each sensor measurement. A Median filter exerted similar results. However, while Mean filters suppress noise, unique features of the data may be lost. Hence, a multitude of measurement vectors of different classes may be described with the same mean, leading to low classification accuracy. Therefore, we include the standard deviation of each sensor measurement which retrieves some previously lost distinctive features of the class. Hence, a processed signal of episode j is now $\mathbf{x}_i^{(j)} \in \mathbb{R}^{2n}$ including both mean and standard deviation of n signals along a window of length w.

The labeled dataset is now composed of N episodes $\Phi = \Phi_1 \cup \Phi_2 \cup \ldots \cup \Phi_N$ where $\Phi_j = \{(\mathbf{x}_1^{(j)}, l_1), \dots, (\mathbf{x}_M^{(j)}, l_M)\}$. Dataset Φ can now be used to train a NN mapping to classify a set of objects, i.e., train map $h : \mathbb{R}^{2n} \rightarrow [0,1]^m$ such that a class probability distribution $\mathbf{p} = \{p_1, \dots p_m\}$ (where $\sum_m p_i = 1$) using $h(\mathbf{x})$ is computed by $\mathbf{p} = h(\mathbf{x})$. Next, we present an iterative classification method in which models, even with low certainty, can be improved by sequentially drawing more samples.

C. Iterative Classification

A trained classifier may yield object identification with low certainty due to noisy measurements, insufficient measurement data or non-optimal model. Nevertheless, we may exploit the continuous time frame in which the user holds a certain object. While common classification tasks rely only on one sample for prediction, in our case, we may rapidly acquire additional samples while being certain that they originate from the same class. Consider FMG signals arriving sequentially $\{\mathbf{x}_1, \mathbf{x}_2, \dots\}$ in real-time while holding an unknown object. It is required to estimate conditional probability for class $\mathcal{O}_i$ given k sequential samples, i.e., $P_k(\mathcal{O}_i | \mathbf{x}_1, \dots \mathbf{x}_k)$. We propose to use an iterative process where a score is given in each iteration based on prediction certainty. Unlike sequential Bayesian update [28], the proposed approach can be used with any type of classifier that provides a probability distribution.

The *Iterative Classification* (IC) process is described in Algorithm 1. We track the scores of the classes based on the predictions for each sample provided by any chosen classifier. We maintain a vector $\mathbf{s} = (s_1, ..., s_m)$ of cumulative scores for the classes. In each iteration, a signal $\mathbf{x}$ is sampled, followed by acquiring a class probability distribution $\mathbf{p} = h(\mathbf{x})$. Generally, function h can be any trained classifier that outputs a probability distribution of the prediction. The probability output is considered as the certainty of the classifier to its prediction and the score to the class prediction. Hence, the highest probability p_i in $\mathbf{p}$ is the iteration score for class i and is added to s_i. This process is repeated until the normalized cumulative score $\hat{s}_{max}$ for some class reaches above a lower bound $\lambda \in [0, 1]$. It is also possible to *fully accumulate* (FA) all class scores by updating $\mathbf{s}$ with all iteration probabilities, i.e.,

replace lines 5-6 in Algorithm 1 with $\mathbf{s} \leftarrow \mathbf{s} + \mathbf{p}$. However, this will result in an excessive number of iterations for ill-trained classifiers while requiring a carefully tuned λ. For a sufficiently trained classifier, FA would provide only marginal accuracy improvement as will be seen in the experimental section.

Algorithm 1: `iterative_classification`(λ)

```
1  Initiate elements of s = {s_1, ..., s_m} to 0;
2  repeat
3      x ← sample();
4      p ← h(x);
5      i ← arg max(p);
6      s_i ← s_i + p_i;
7      o ← arg max(s);
8      if first iteration then
9          ŝ_max ← s_o;
10     else
11         ŝ_max ← s_o / (Σ_i s_i);
12 until ŝ_max > λ;
13 return o ;          /* return class index */
```

Let $P(l = j|\mathcal{O}_i)$ be the probability for classifier h to assign label j to a grasped object $\mathcal{O}_i$ such that

$$\sum_{j=1}^{m} P(l = j|\mathcal{O}_i) = 1. \tag{1}$$

A sufficiently trained classifier must satisfy

$$P(l = i|\mathcal{O}_i) > P(l = j|\mathcal{O}_i) \tag{2}$$

for any $i, j \in \{1, ..., m\}$ and $j \neq i$. Naturally, higher values of $P(l = i|\mathcal{O}_i)$ for all $i \in \{1, \ldots, m\}$ mean a more accurate classifier. In many cases, prediction probability of incorrect predictions tend to be lower than the prediction probability for correct examples [29]. Hence, given $p_{max} = \max(\mathbf{p})$, the expected value for p_{max} when successfully classifying object $\mathcal{O}_i$ would be larger than an erroneous prediction. That is, statement

$$\mathbb{E}(p_{max}|l = i, \mathcal{O}_i) > \mathbb{E}(p_{max}|l = j, \mathcal{O}_i) \tag{3}$$

holds for any $j \neq i$. According to Algorithm 1, vector $\mathbf{s}$ accumulates scores for class predictions with the increase of iterations. In addition, a score is given to s_j only if label $l = j$ is assigned to the query object in a particular iteration. Hence, the expected normalized value $\hat{s}_j$ of component s_j given object $\mathcal{O}_i$ is

$$\mathbb{E}(\hat{s}_j|\mathcal{O}_i) = \mathbb{E}(p_{max}|l = j, \mathcal{O}_i)P(l = j|\mathcal{O}_i) \tag{4}$$

for any $j \in \{1, \ldots, m\}$ and where $\hat{s}_j = s_j / (\sum_i s_i)$. One may view $\hat{s}_j$ as the probability approximation for grasping $\mathcal{O}_j$ after some number of iterations, i.e., $\hat{s}_j \approx P_k(\mathcal{O}_j|\mathbf{x}_1, \ldots \mathbf{x}_k)$. From (2)-(4), it must be that

$$\mathbb{E}(\hat{s}_i|\mathcal{O}_i) > \mathbb{E}(\hat{s}_j|\mathcal{O}_i), \quad \forall j \neq i. \tag{5}$$

While we acknowledge that class probability distributions outputted from a *softmax* layer may not always reflect the true certainty of its prediction [30], preliminary results show that condition (3) holds in our case. Nevertheless, even with a more strict assumption, where p_{max} for any prediction (erroneous or not) has a uniform distribution $p_{max} \sim U(\frac{1}{m}, 1)$ such that

$$\mathbb{E}(p_{max}|l = i, \mathcal{O}_i) = \mathbb{E}(p_{max}|l = j, \mathcal{O}_i) = \frac{m + 2}{2m}, \tag{6}$$

statement (5) remains valid due to (2).

The above statements imply that as long as a classifier satisfies (2), the expected cumulative score $\mathbb{E}(\hat{s}_i|\mathcal{O}_i)$ will increase and converge to an higher value than $\mathbb{E}(\hat{s}_j|\mathcal{O}_i)$ $(j \neq i)$, with the increase of classification iterations, i.e., $\hat{s}_i$ is more likely to be equal to $\hat{s}_{max}$. Hence, the certainty about the prediction will grow with the addition of more samples. In turn, this will result in continuous improvement of the classifiers success rate. Let $m_p \in \{0, 1, \ldots, m\}$ be the number of grasped objects that satisfy (2) for a given classifier. If a classifier is sufficiently trained such that $m_p = m$, condition (2) will be satisfied for all objects. Hence, the total success rate would converge to 100% with the increase of iterations. However, when $m_p < m$, condition (2) is satisfied only for the m_p objects. Consequently, the classification success rate will increase only for these objects while declining for the remaining $m - m_p$ ones. This means that the success rate upper limit for a certain classifier is

$$\xi = \frac{m_p}{m} \times 100\%. \tag{7}$$

The convergence rate depends on the quality of the classifier, that is, on the accuracy $P(l = i|\mathcal{O}_i)$ and prediction certainty $\mathbb{E}(p_{max}|l = i, \mathcal{O}_i)$, for all i.

The proper amount of iterations to reach some level of accuracy or certainty is not known beforehand. Hence, when required to acquire a classification in a short period of time, we cannot set the termination criterion to some arbitrary number of iterations. Therefore, Algorithm 1 sets a termination criterion when reaching some certainty above a threshold λ. Alternatively, classification can be done without a termination criterion for a long time horizon with continuous certainty improvement. The collaborative robot, however, will need to identify task completion by other means. These will be shown and analyzed through experiments in the next section.

IV. Experiments and Analysis

In this section, we test and analyse the proposed FMG device and classification method over a set of objects. We have picked five everyday objects shown in Figure 3 including a bottle, a mug, a screwdriver, scissors and a plate. A training set is acquired by recording $N = 40$ episodes for each object with a single participant. The participant grasped the objects in a task-based pose (e.g., by the handle of the screwdriver) similar to the taxonomy described in [31]. In addition, the FMG device is taken off and re-positioned between the episodes. While doing so, the object is also put down and picked up again in a task-based grasp with some variations. It is important to note that the data is recorded while the arm is moving through various configurations to increase variance and include different grasping postures. For each episode and

object, a batch of M =10,000 data points is recorded and labeled. All data is pre-processed as described in Section III-B with $w = 100$, resulting in dimension 30 (combining mean and standard deviation of an $n = 15$ dimensional measurement). 10% of the data was dedicated for testing and hyper-parameters optimization, and not included in the training. Similarly, we recorded and pre-processed five episodes for each object to form a validation set yielding 4,950 samples for each object and 24,750 in total. We use the validation data to test classification success rate in a standard fashion over all tests. The validation data was completely decoupled from the training and testing phases. While we perform validation tests off-line, data is taken sequentially as recorded as if done in real-time.

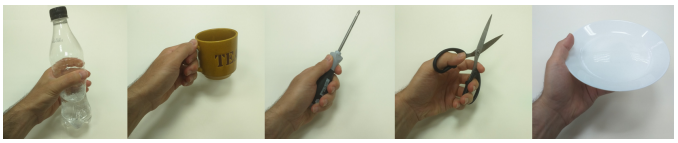


Fig. 3. Five objects used in the experiments and their typical grasps. From left to right: bottle, mug, screwdriver, scissors and a plate.

A. Model evaluation

Using the training data, we have optimized a feed-forward NN classifier. The resulting NN has two hidden layers of 398 neurons each and a Rectified Linear Unit (ReLU) activation. A dropout of 50% and an L2 regularizer (factor 10^{-5}) were included to control apparent overfitting. Additionally, the ADAM optimizer was used with a sparse categorical cross-entropy loss function. The network was trained with the back-propagation algorithm. Furthermore, we have conducted a comparison to other common classifiers, including: Nearest-Neighbors, Naive Bayes, Support Vector Machines (SVM) with a linear kernel, Random Forests, Decision Trees, Adaptive Boosting (AdaBoost) and Linear Discriminant Analysis (LDA) [32].

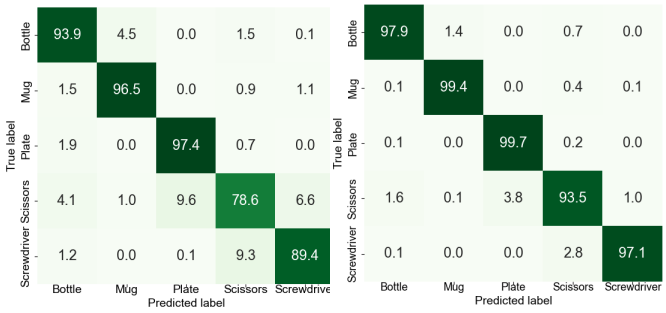
Table I reports the classification success rate for these classifiers over the validation data. We note that these are the rates for one-time calls without IC, i.e., classification according to one sample as input. The table also shows the importance of the pre-processing step (no Mean filter and standard deviation inclusion). The pre-processing step slightly improves in most cases with a significant increase for the NN classifier. All tests, however, include signal substraction by the initial episode relaxation measurements to compensate varying tightening forces. Without filtering the training data, the most accurate NN with input dimension of 15 reach success rate of 63.01%. When applying the Mean filter, the success rate significantly increases to 88%; and 91.17% when also including standard deviation. The results indicate that both mean and standard deviation values embed valuable data, where the mean values are of greater importance for the NN performance. Overall, it is clear that the NN classifier outperforms and, therefore, further used in our experiments.

Figure 4a presents the confusion matrix for the NN classifier, denoted as classifier O, with a total success rate of 91.17%. The most erroneous classification is for the scissors in which some grasp variations can be confused with a plate

TABLE I
SUCCESS RATE COMPARISON FOR DIFFERENT CLASSIFIERS

	Classification success rate	
Classifier	w/o pre-process	w/ pre-process
Nearest Neighbors	75.19%	83.68%
Naive Bayes	64.41%	62.11%
Linear SVM	75.52%	80.86%
Random Forests	70.90%	63.29%
Neural-Network	63.01%	91.17%
Decision Trees	60.95%	63.59%
AdaBoost	72.38%	73.39%
LDA	73.68%	78.99%

or a screwdriver. This motivates the observation of multiple signals through time to increase certainty and success rate.



(a) (b)

Fig. 4. (a) Confusion matrix for the Neural-Network classifier (O) with a total success rate of 91.17%. (b) Confusion matrix using IC with classifier O for $\lambda = 0.99$. The total classification success rate is 97.5% with 1.18 average number of iterations.

In the next analysis, we wish to observe the robustness property with regards to the number of recorded episodes. Recall that each episode contains 10,000 recorded signals and the FMG device is removed and re-positioned before each episode. In order to observe the success rate with regards to the number of episodes N, the NN classifier was repeatedly trained over 10 trials for each given $N = 1, ..., 40$ while sampling different episodes from the dataset in each trial. Results can be seen in Figure 5. The classification success rate improves with the increase of episodes, reaching approximately 90% success rate with more than 30 episodes. Also, the standard deviation over the 10 trials decreases as the number of episodes rise. This behaviour confirms the main issue dealing with data originating from body-muscle related source; the placement and replacement after every episode creates considerable variations to the data. As expected, additional episodes in the training data results in a more robust classifier to variations in device placements and tightening forces.

B. Sensor placement analysis and Feature Importance

As discussed in Section I, previous work have positioned FSR sensors either on the lower or upper forearm. We now investigate the contribution of each sensor location to the success rate of the classification. Hence, we compare between various configurations of FMG measurements: an upper forearm band (UF), a lower forearm band with one row of sensors (LF-1), a lower forearm band with two rows of sensors (LF-2) and using all sensors in the FMG device. For each configuration,

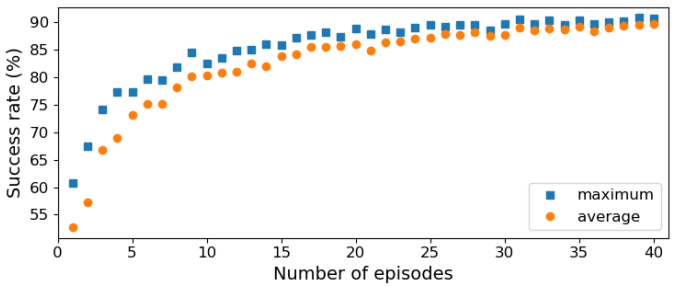


Fig. 5. Classification success rate acquired with regards to the number of episodes recorded. The results show maximum and average values for 10 training attempts while sampling different episodes in each.

we optimized the NN hyper-parameters. Classification success rate is summarized in Table II. We see that sensors on the lower forearm contribute the most valuable information while sensors on the upper forearm by themselves are not sufficient.

TABLE II
CLASSIFICATION SUCCESS RATE WITH REGARDS TO DIFFERENT PLACEMENTS OF FMG SENSORS

	UF	LF-1	LF-2	All (O)
Num. of sensors	6	5	9	15
Success rate	57.20%	75.13%	82.80%	91.17%

To achieve better understanding of the FSR sensors effect on the model's performance, we observe the impact of each sensor on predictions. Permutation feature importance is a common method to quantify the contribution of each feature in an NN [33]. This is done by measuring the increase in the prediction error after permuting the values of each single feature separately, which breaks the relationship between the feature and the true label. We use classifier O and randomly permute the validation data, a single feature each time. The importance score of feature i is the error e_i relative to the non-permuted model. In other words, the score is defined as the decline in accuracy resulting from the permutation of a sensor's values. The score is computed according to $e_i = \frac{q-q_i}{q} \times 100\%$, where q is the success rate of the non-permuted model and q_i is the success rate when feature i is permuted. The results of feature importance evaluation over 30 repetitions are shown in Table III and illustrated in Figure 6 along with sensor locations. The relative accuracies indicate a relatively strong dependence on the lower forearm sensors and correlate to the results in Table II. The above feature importance correlates with the layout of the forearm muscles. The most significant sensors, 8 and 14, lay on top of the flexor carpi radialis and flexor digitorum superficialis, which have an important role in operating the wrist and fingers. While some sensors are more important than others, the results show that all sensors along the forearm contribute to an accurate and robust classification.

C. Iterative classification analysis

The results of Section IV-A show the ability to train a classifier with relatively high success rate. However, it may be possible that a model is not accurate enough due to insufficient data or non-optimal NN hyper-parameters. Hence, we now

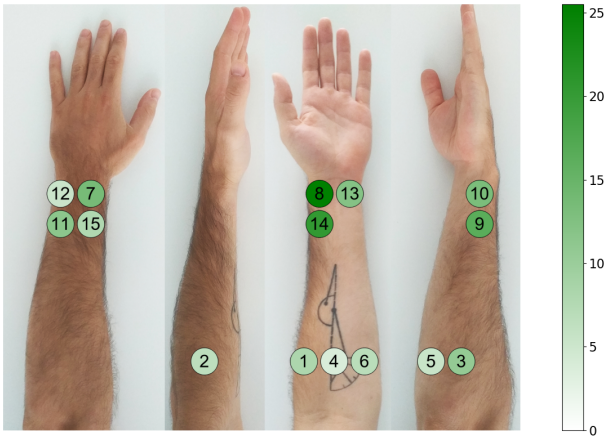


Fig. 6. Illustration of the sensor locations and importance score computed with the permutation feature importance method.

TABLE III
IMPORTANCE SCORE FOR THE SENSOR ON THE FMG DEVICE

Sensor index	Importance score	Sensor index	Importance score	Sensor index	Importance score
1	8.23%	6	6.90%	11	11.47%
2	6.58%	7	13.62%	12	5.33%
3	10.67%	8	25.50%	13	12.21%
4	3.78%	9	16.50%	14	20.46%
5	5.52%	10	13.16%	15	7.89%

analyse the proposed IC algorithm described in Section III-C (Algorithm 1) aimed to raise the certainty of the output given by a classifier. We first apply the algorithm to classifier O. When exerting IC with $\lambda = 0.99$, the total success rate increases to 97.5% with 1.18 average number of iterations. Figure 4b shows the confusion matrix of the iterated classifier. Adding more samples increases certainty and, therefore, success rate. Specifically, significant improvement is seen for the scissors and screwdriver when comparing to Figure 4a. The above results show that IC along with a sufficiently good classifier can exhibit accurate and robust classifications even after the re-positioning of the FMG device.

We now explore the use of the iterative algorithm for trained classifiers that initially achieved lower success rate. Four classifiers were chosen and their confusion matrices are seen in Figure 7. Classifiers A and B were trained with only 4 and 16 episodes, and reached success rate of 76.88% and 84.77%, respectively (confusion matrices are seen in Figures 7a and 7b). Another two classifiers, C and D (Figures 7c and 7d), that use the entire training set (40 episodes) but with non-optimal hyper-parameters, reach success rates of 50.10% and 60.12%, respectively. We note that the confusion matrices are an approximation of $P(l = j|\mathcal{O}_i)$ for $i, j \in \{1, 2, 3, 4, 5\}$ where the diagonal elements approximate $P(l = i|\mathcal{O}_i)$. Hence, classifiers C and D do not satisfy (2).

Table IV presents the success rate improvement when applying IC with $\lambda = 0.99$ over the validation data. The results for classifiers A and B clearly show that a classifier can be trained with less episodes (i.e., smaller sized training set) and, achieve high and robust accuracy with a low number of iterations. The success rates for classifiers C and D show limited improvement while the number of iterations is relatively high. This is the result of the low initial accuracy of the classifiers and

TABLE IV
RESULTS OF IC WITH DIFFERENT NN CLASSIFIERS ($\lambda = 0.99$)

Classifier	*A*	*B*	*C*	*D*	*O*
Initial success rate (%)	76.88	84.77	50.10	60.12	91.17
Success rate w/ IC (%)	87.15	92.39	58.71	69.13	97.5
Difference (%)	10.27	7.62	8.61	9.01	6.33
Avg. iterations	1.31	1.17	2.06	2.04	1.18

inability to satisfy condition (2). With FA, the success rate is 98.1% with 1.16 average number of iterations for classifier *O*, providing only marginal improvement. On the other hand, FA for classifiers *A*-*D* failed to converge with a high number of iterations for many trials making the approach infeasible.

We next analyze the performance of the algorithm over a long horizon when removing the λ termination criterion. Figure 8 shows the success rate behavior with respect to the number of iterations. Classifiers *O*, *A* and *B* (represented by Figures 4a, 7a and 7b) show satisfaction of condition (2). Consequently, Figure 8 shows continuous success rate increase toward converging to a common upper limit of $\xi = 100\%$ as expected from (7). Classifier *O* converged to 100% relatively fast while the other two require more iterations due to their lower success rate for some objects. Nevertheless, the algorithm contributes significant improvement compared to the initial results.

As can be seen in Figures 7c and 7d, classifiers *C* and *D* share a common property where they both have exactly one object that does not satisfy condition (2), i.e., $m_p = 4$. Therefore and according to (7), they are both expected to exhibit convergence to an upper limit of $\xi = 80\%$ success rate. Figure 8 indeed shows moderate increase in success rate for the two classifiers toward 80%. Additional trials show that they reach a success rate of approximately 76% after 1,000 iterations slowly approaching their upper bound.

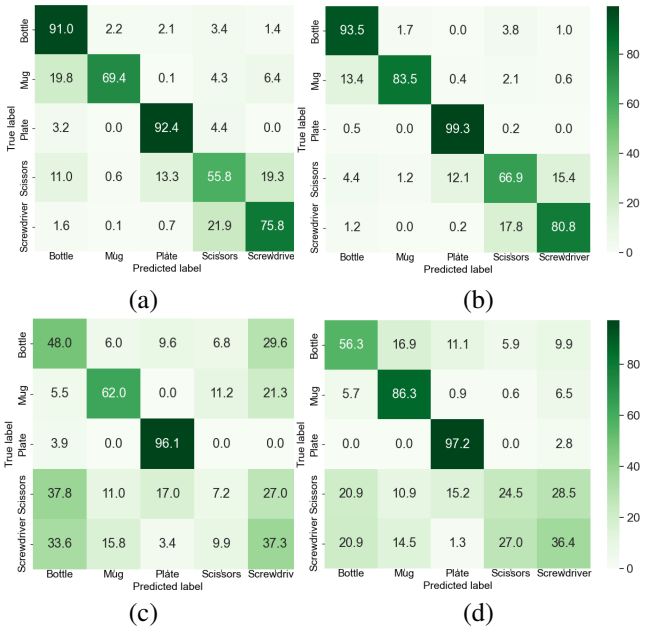


Fig. 7. Confusion matrices for lower success rate classifiers. Classifiers (a) *A*, (b) *B*, (c) *C* and (d) *D* have a total success rate of 76.88%, 84.77%, 50.10% and 60.12%, respectively.

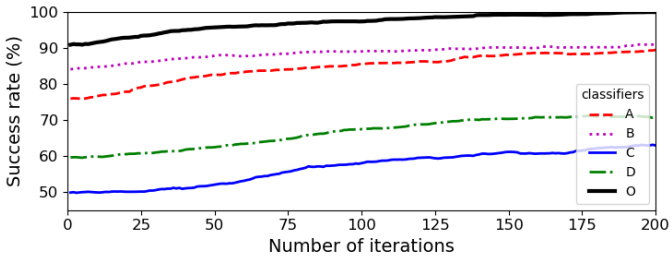


Fig. 8. The classification success rate with regards to the number of iterations for classifiers *O*, *A*, *B*, *C* and *D*.

Finally, we report the computational time required to classify an object within hand. We have experimented on an Intel-Core i7-9700 Ubuntu machine with 16GB of RAM. The average computation time over 5,000 trials when $\lambda = 0.99$ is 24 milliseconds. Alternatively, if the λ termination criterion is not used and the algorithm keeps improving the prediction certainty, iteration frequency reaches 53.5Hz. These results show that accurate classification can be done very fast and in real-time.

D. Sensor failure

When a sensor in the FMG device fails, a well trained classifier can lose its accuracy. We now test the use of IC with an already trained classifier when various sensors fail. We virtually fail a sensor by setting its values in the validation set to zero. We note that failure can occur with other non-zero values or increased noise, leading to different success rates. Table V shows the results of several different sensors that fail individually with and without using IC. Indices of the sensors can be viewed in Figure 6. We note that these results do not match the general failure importance values (Table III) as they only reflect a special case of setting the failed value to zero. Failure of some sensors lead to significant loss of accuracy. The loss magnitude depends on the impact of the sensor on the classification success rate for specific objects and on the respective feature importance. IC, in such case, can provide valuable improvement while not enough in some cases.

TABLE V
RESULTS OF IC WITH DIFFERENT FAULTY SENSORS ($\lambda = 0.99$)

Sensor index	3	4	8	9	14
Initial success rate (%)	58.17	86.23	78.37	78.11	62.77
Success rate w/ IC (%)	63.22	93.80	89.72	88.03	70.64
Difference (%)	5.05	7.57	11.35	9.92	7.87
Avg. iterations	1.2	1.17	1.22	1.22	1.25

V. CONCLUSIONS

We have presented the problem of classifying an object in hand using simple FMG measurements accurately and robustly. We have shown the use of 15 FSR sensors on a low-cost FMG device worn on the forearm. Measurements from the sensors are recorded during the handling of various objects while removing and replacing the device. The data is used to train a NN classifier that exhibited high classification success rate. We further proposed an iterative classification algorithm to augment the classification. The IC has shown to significantly improve certainty of predictions by sampling additional signals. Furthermore, success rate is improved even

for less accurate classifiers. Hence, a single user can train a robust and relatively accurate classifier quickly. In addition, we have performed an analysis to understand the key locations for FSR sensors. Our proposed method has shown a robot's ability to reason about the object in hand without verbal communication or visual feedback. The FMG device provides fast and robust classification of grasped objects that imply about the intended task of the human. The information about the object will be part of the decision making for a task planner to take assistive actions.

Future work could focus on increasing the resolution of the FSR sensors and reasoning about the pose and weight of an object in hand. In addition, more sensors can be added along the arm to acquire additional information about its pose and to predict future trajectory. We note that some objects with similar geometry and a task-based grasp (e.g., a screwdriver and a spatula) cannot be distinguished by solely observing finger poses. In such case, future work may consider the observation of FMG measurements during the task over time or add context information. A global classifier or model transfer to a new user are interesting topics to further be explored.

REFERENCES

[1] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib, "Progress and prospects of the human-robot collaboration," *Autonomous Robots*, 10 2017.

[2] T. L. Chen, M. Ciocarlie, S. Cousins, P. M. Grice, K. Hawkins, K. Hsiao, C. C. Kemp, C. King, D. A. Lazewatsky, A. E. Leeper, H. Nguyen, A. Paepcke, C. Pantofaru, W. D. Smart, and L. Takayama, "Robots for humanity: using assistive robotics to empower people with disabilities," *IEEE Rob. & Aut. Mag.*, vol. 20, no. 1, pp. 30–39, 2013.

[3] M. A. Goodrich and A. C. Schultz, "Human-robot interaction: A survey," *Foundations and Trends in Human-Computer Interaction*, vol. 1, no. 3, pp. 203–275, 2007.

[4] B. Gleeson, K. MacLean, A. Haddadi, E. Croft, and J. Alcazar, "Gestures for industry: Intuitive human-robot communication from human observation," in *Proceedings of the ACM/IEEE International Conference on Human-robot Interaction*, 2013, pp. 349–356.

[5] L. F. Nicolas-Alonso and J. Gomez-Gil, "Brain computer interfaces, a review," *Sensors*, vol. 12, no. 2, pp. 1211–1279, 2012.

[6] Z. Khokhar, Z. Xiao, and C. Menon, "Surface EMG pattern recognition for real-time control of a wrist exoskeleton," *Biomedical engineering online*, vol. 9, p. 41, Aug. 2010.

[7] E. Fujiwara, Y. T. Wu, C. K. Suzuki, D. T. G. de Andrade, A. R. Neto, and E. Rohmer, "Optical fiber force myography sensor for applications in prosthetic hand control," in *Proceedings of the IEEE International Workshop on Advanced Motion Control (AMC)*, 2018, pp. 342–347.

[8] O. Amft, H. Junker, P. Lukowicz, G. Troster, and C. Schuster, "Sensing muscle activities with body-worn sensors," in *Int Work Wearable Implant Body Sens Networks*, 05 2006, pp. 138–141.

[9] G. Ogris, M. Kreil, and P. Lukowicz, "Using FSR based muscule activity monitoring to recognize manipulative arm gestures," in *IEEE Int Symp Wearable Comput*, 2007, pp. 45 – 48.

[10] N. Li, D. Yang, L. Jiang, H. Liu, and H. Cai, "Combined use of FSR sensor array and SVM classifier for finger motion recognition based on pressure distribution map," *Journal of Bionic Engineering*, vol. 9, no. 1, pp. 39–47, 2012.

[11] X. Li, Q. Zhuo, X. Zhang, O. W. Samuel, Z. Xia, X. Zhang, P. Fang, and G. Li, "FMG-based body motion registration using piezoelectret sensors," in *Int. Conf. of the IEEE Eng. in Medicine and Biology Society*, 2016, pp. 4626–4629.

[13] X. Jiang, L.-K. Merhi, Z. G. Xiao, and C. Menon, "Exploration of force myography and surface electromyography in hand gesture classification," *Medical Eng. & Physics*, vol. 41, pp. 63 – 73, 2017.

[12] H. K. Yap, A. Mao, J. C. H. Goh, and C. Yeow, "Design of a wearable FMG sensing system for user intent detection during hand rehabilitation with a soft robotic glove," in *IEEE Int. Conf. on Biomedical Rob. and Biomech.*, 2016, pp. 781–786.

[14] P. B. Shull, S. Jiang, Y. Zhu, and X. Zhu, "Hand gesture recognition and finger angle estimation via wrist-worn modified barometric pressure sensing," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 4, pp. 724–732, 2019.

[15] Z. Xiao and C. Menon, "Towards the development of a wearable feedback system for monitoring the activities of the upper-extremities," *Journal of neuroengineering and rehabilitation*, vol. 11, p. 2, 01 2014.

[16] D. Paulius, Y. Huang, R. Milton, W. D. Buchanan, J. Sam, and Y. Sun, "Functional object-oriented network for manipulation learning," in *IEEE/RSJ Int. Conf. on Intel. Robots and Sys.*, 2016, pp. 2655–2662.

[17] T. Mukai, S. Hirano, M. Yoshida, H. Nakashima, S. Guo, and Y. Hayakawa, "Tactile-based motion adjustment for the nursing-care assistant robot riba," in *Proceedings of the IEEE International Conference on Robotics and Automation*, May 2011, pp. 5435–5441.

[18] J. Newn, B. Tag, R. Singh, E. Velloso, and F. Vetere, "AI-mediated gaze-based intention recognition for smart eyewear: Opportunities and challenges," in *Proceedings of the ACM International Symposium on Wearable Computers*, New York, NY, USA, 2019, pp. 637–642.

[19] "Cyberglove," http://www.cyberglovesystems.com/.

[20] L. Bi, A. G. Feleke, and C. Guan, "A review on EMG-based motor intention prediction of continuous human upper limb motion for human-robot collaboration," *Biomedical Signal Processing and Control*, vol. 51, pp. 113 – 127, 2019.

[21] M. Tavakolan, Z. Xiao, and C. Menon, "A preliminary investigation assessing the viability of classifying hand postures in seniors," *Biomedical engineering online*, vol. 10, p. 79, Sep 2011.

[22] P. Artemiadis, "EMG-based robot control interfaces: Past, present and future," *Advances in Robotics and Automation*, vol. 01, 2012.

[23] E. Cho, R. Chen, L.-K. Merhi, Z. Xiao, B. Pousett, and C. Menon, "Force myography to control robotic upper extremity prostheses: A feasibility study," *Frontiers in Bioeng. and Biotech.*, vol. 4, 2016.

[24] B. Borbely and P. Szolgay, "Real-time inverse kinematics for the upper limb: A model-based algorithm using segment orientations," *BioMedical Engineering OnLine*, vol. 16, 12 2017.

[25] C. Loconsole, S. Dettori, A. Frisoli, C. A. Avizzano, and M. Bergamasco, "An EMG-based approach for on-line predicted torque control in robotic-assisted rehabilitation," in *IEEE Haptics Symposium (HAPTICS)*, Feb 2014, pp. 181–186.

[26] Y. M. Aung and A. Al-Jumaily, "Estimation of upper limb joint angle using surface EMG signal," *International Journal of Advanced Robotic Systems*, vol. 10, no. 10, p. 369, 2013.

[27] Q. Zhang, R. Liu, W. Chen, and C. Xiong, "Simultaneous and continuous estimation of shoulder and elbow kinematics from surface EMG signals," *Frontiers in Neuroscience*, vol. 11, p. 280, 2017.

[28] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. USA: Prentice-Hall, Inc., 1993.

[29] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in *International Conference on Learning Representations*, 2017.

[30] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proceedings of the International Conference on Machine Learning*, vol. 70, 2017, p. 1321–1330.

[31] M. R. Cutkosky, "On grasp choice, grasp models, and the design of hands for manufacturing tasks," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 3, pp. 269–279, 1989.

[32] A. Singh, N. Thakur, and A. Sharma, "A review of supervised machine learning algorithms," in *International Conference on Computing for Sustainable Global Development*, 2016, pp. 1310–1315.

[33] J. Yang, K. Shen, C. Ong, and X. Li, "Feature selection for mlp neural network: The use of random permutation of probabilistic outputs," *IEEE Trans. on Neural Net.*, vol. 20, no. 12, pp. 1911–1922, 2009.