Engaging with Robotic Swarms: Commands from Expressive Motion

DAVID ST-ONGE, MISTLab, Department of Computer Engineering and Software Engineering, École Polytechnique de Montréal, Canada
ULYSSE CÔTÉ-ALLARD, Department of Electrical Engineering, Laval University, Canada
KYRRE GLETTE, RITMO, Department of Informatics, University of Oslo, Norway
BENOIT GOSSELIN, Department of Electrical Engineering, Laval University, Canada
GIOVANNI BELTRAME, MISTLab, Department of Computer Engineering and Software Engineering, École Polytechnique de Montréal, Canada

In recent years, researchers have explored human body posture and motion to control robots in more natural ways. These interfaces require the ability to track the body movements of the user in 3D. Deploying motion capture systems for tracking tends to be costly, intrusive, and requires a clear line of sight, making them ill-adapted for applications that need fast deployment. In this paper, we use consumer-grade armbands, capturing orientation information and muscle activity, to interact with a robotic system through a state machine controlled by a body motion classifier. To compensate for the low quality of the information of these sensors, and to allow a wider range of dynamic control, our approach relies on machine learning. We train our classifier directly on the user to recognize (within minutes) which physiological state his or her body motion expresses. We demonstrate that on top of guaranteeing faster field deployment, our algorithm performs better than all comparable algorithms, and we detail its configuration and the most significant features extracted. As the use of large groups of robots is growing, we postulate that their interaction with humans can be eased by our approach. We identified the key factors to stimulate engagement using our system on 27 participants, each creating their own set of expressive motions to control a swarm of desk robots. The resulting unique dataset is available online together with the classifier and the robot control scripts.

CCS Concepts: • **Computer systems organization → Robotic control**; **External interfaces for robotics**; *Sensors and actuators*;

Additional Key Words and Phrases: machine learning,natural user interface,body motion,gesture recognition,human-swarm interaction

ACM Reference Format:
David St-Onge, Ulysse Côté-Allard, Kyrre Glette, Benoit Gosselin, and Giovanni Beltrame. 2019. Engaging with Robotic Swarms: Commands from Expressive Motion. *ACM Trans. Hum.-Robot Interact.* 1, 1, Article 1 (January 2019), 26 pages. https://doi.org/10.1145/3323213

Authors' addresses: David St-Onge, MISTLab, Department of Computer Engineering and Software Engineering, École Polytechnique de Montréal, 2900 Blvd Edouard-Montpetit, Montreal, QC, H3T 1J4, Canada, david.st-onge@polymtl.ca; Ulysse Côté-Allard, Department of Electrical Engineering, Laval University, 1095 Medecine Av. Quebec, QC, H3T 1J4, Canada, ulysse.cote-allard@ulaval.ca; Kyrre Glette, RITMO, Department of Informatics, University of Oslo, Oslo, Norway, kyrrehg@ifi.uio.no; Benoit Gosselin, Department of Electrical Engineering, Laval University, 1095 Medecine Av. Quebec, QC, H3T 1J4, Canada, benoit.gosselin@ulaval.ca; Giovanni Beltrame, MISTLab, Department of Computer Engineering and Software Engineering, École Polytechnique de Montréal, 2900 Blvd Edouard-Montpetit, Montreal, QC, H3T 1J4, Canada, giovanni.beltrame@polymtl.ca.

1 INTRODUCTION

The presence of robots in daily activities (cleaners, personal assistants, autonomous vehicles) as well as in critical scenarios (emergency response, planetary exploration), is continuously being extended with new tasks, new integration tools, and new hardware. Each of these domains constitutes an opportunity to develop a more natural and intuitive relationship between robots and humans, working on the robots' capacity to detect social attitudes and adopt expressive stances. While social robotics has mostly focused on interactions with single humanoids and zoomorphic robots, new forms of robots are entering the scene. Abstract robot swarms are one of them, composed of large numbers of robots that can evolve in formation and adapt easily to multiple environments. An expensive and complex robot can be outperformed in many scenarios by a group of smaller and simpler ones: e.g., covering a large area (for search and rescue or sensor network deployment) [Mottola et al. 2014], or when redundancy is mandatory (e.g. firefighting or long-term deployments) [Penders et al. 2011]. While several decentralized algorithms were introduced in the past decades, few were implemented on hardware platforms and only a handful integrate user commands. Based on the work of our research group on a programming language specific to robotic swarms [Pinciroli and Beltrame 2016], we implement a set of common control algorithms (pursuit, aggregation, formation, etc.) as states for the user to select.

Unfortunately, the command and control of decentralized behaviors are challenging, mostly because group intelligence is a difficult and abstract concept to grasp [Kolling et al. 2016]. The nascent field of Human-Swarm Interaction (HSI) [Podevijn et al. 2016] addresses the challenge with two questions: how does a user perceive the artificial collective state and intention? How can a user influence the collective intelligence? This work aims at giving one possible answer to the latter. The interaction strategy described in this paper, and validated with a user study, paves the way to an abstract natural interaction approach with swarms, centered on the user.

In Human-Robot Interaction (HRI), engagement generally refers to the process of creating and maintaining a connection [Levillain et al. 2017]. A human-robot connection can be achieved with voice communication, user centered motion (i.e. follow me), or cues given by a graphical user interface. However, these require a level of concentration (that is, cognitive load) that can distract the user from his or her other tasks. In this study, we are interested in the use of *expressive motion*. Expressive motion, as put by Simmons and Knight, refers to "movements that are not directly related to achieving a task, but that help convey an agent's attitude towards its task or its environment" [Simmons and Knight 2017]. As shown in their work, expressive motion can be used to interact with humans. Indeed, psychological states such as emotions, trigger physiological states that can be detected and classified based on facial expressions, physiological response (i.e. galvanic skin response or heart rate), and whole-body motion. In this work, we observe a user's expressive motion, and use it to make a swarm react, stimulating a connection with the user. We use the term "moods" to refer to the physiological states of the user associated with an observed and classified expressive motion. Since our work does not aim to be a universal mood classifier, but rather an intuitive interaction modality, we assume the moods to be labeled by the users.

Our work targets the use of swarms for critical scenarios: from emergency response to artistic performance, with a prior focus on the latter. In these context, an intelligent robotic system is one that can infer the physiological state of its operator and adapt its behavior accordingly. On top of such benefits derived from the observation of the user expressive motions, we postulate that this approach is more suitable than classic fixed gestures to command a robotic system in these dynamic and mobile scenarios. In the rest of the paper, we use *users* or *performers* to refer to both contexts without distinction.

To capture a user's movement in an unobtrusive manner, a recording device must be small, lightweight, wireless, and robust. Since muscular activity signals (electromyograms - EMG) alone are difficult to classify, the ideal device should include extra sensing modalities. An inertial measurement unit (IMU) is particularly well suited for such requirement.

We recorded two datasets with the Myo, a low-cost, low-sampling rate (200Hz), 8-channel, consumer-grade, dry electrode EMG armband which integrates a nine degree-of-freedom IMU. Our approach is to design a lightweight, portable, live learning algorithm that is able to identify a user's mood based on his or her expressive motion. The user designs the expressive motions and labels them with moods to control a robotic system. We used the *design* dataset, based on a single participant, to determine the best classifier, its parameters and select the number of sensors, their placement as well as which features to extract. We then tested and validated the design on the *evaluation* dataset.

The main contributions of this work are technical: a set of the most significant features extracted from IMU and EMG to classify body motion expressive states, a detailed working classifier configuration, a set of platform-agnostic behaviors scripts for robotic swarms; and scientific: the identification of the key factors to stimulate engagement in expressive motions based HSI. This work also made it possible to share a unique dataset publicly for future research.

After presenting the inspiration of this work in Section 2, we describe the design methodology for the learning algorithm, as well as the features generated from the Myo outputs in Section 3. We then compare the selected classifier with other algorithms and discuss its limitations in Section 4. In order to close the loop with our robotic swarm, we present our software and hardware infrastructure and detail the decentralized control scripts in Section 5. Using the learning system together with our tabletop robots equipped with scripted expressive motions, we then assess the accuracy of the classifier as well as the level of engagement felt by the participants and its main contributing factors (Section 6). Finally, we show that our classifier outperforms the classic neural network configurations for the specific task used in this paper (Section 6.4) and we end the paper with a discussion on the results and the following steps (Section 7).

2 RELATED WORK

2.1 Natural Interfaces

The interpretation of human behavior is an active field feeding industrial collaborative robotic as well as personal assistants. Numerous modalities were explored: visual movement detection, speech recognition, haptic devices (force sensing), smart watches, etc. [Song et al. 2007]. Closing the loop with robotic systems is the fascinating challenge of NUI [Aggarwal and Cai 1999; Sanna et al. 2013]. NUI can decrease the cognitive load associated with the control of robotic systems, but often constrain the translation of the users' intent [Petersen et al. 2010]. Indeed, extracting pre-processed actions from data features is not very flexible and prone to user fatigue [Cosentino et al. 2014]. Therefore, we present a new take on the use of human body motion to seamlessly control robots with the projected emotions of their body.

Since gestures can represent a direct expression of mental concepts, it is one of the most popular types of interaction in HRI literature [Fernandez et al. 2016]. In gesture recognition, the focus is on accurately detecting a number of simple gestures within a small, predetermined workspace. For example, the Leap Motion Controller™ can track and recognize hand and finger gestures [Weichert et al. 2013]. The well-known Kinect™ systems can perform hand and arm gestures [Herrera-Acuña et al. 2015] with some limited finger movement tracking. The Kinect was successfully used to command motion primitives on quadrotors [Sanna et al. 2013]. For wider workspaces and movements, a motion-capture system such as the Vicon™ is better suited. It is however expensive, and often cumbersome, to set up cameras for

short term interventions. Some research groups have tackled the problem of whole body recognition in larger spaces through the use of stereo-camera rigs [Ahn et al. 2009], or monocular camera systems for low resolution gestures at long range [DoHyung et al. 2013]. All of these systems require an unobstructed line of sight, which can be hard to obtain, notably in outdoor scenarios.

The use of EMGs can give information on the wearer's motions and gestures from a more intimate, proprioceptive standpoint, instead of an absolute external recognition. Furthermore, EMGs remove the requirement of a clear line of sight. However, these signals are difficult to interpret. EMGs are well known in medical applications for diagnostics, and also for interfacing with prosthetic devices and wheelchairs. For the latter, neural and Bayesian networks were successfully applied to classify hand gestures [Bu et al. 2009; Liu et al. 2016]. [Allard et al. 2016; Artemiadis and Kyriakopoulos 2011] proposed a neural network to control a robot arm using EMGs. However, most of these algorithms require large datasets and powerful processing units for their training. In many critical applications, the classifier must be trained within minutes on a standard computer to allow for rapid deployment on any user. For instance, on a rescue mission, a robotic system may have been calibrated on a first responder not available for that call, and the success of the rescue should not have to rely on a new calibration or training in the field (often with weak internet connection preventing the use of cloud-based services). Faster algorithms such as Random Forest were also explored, but only in the context of precise grasping tasks [Liarokapi et al. 2013], or for conscious fixed gestures [Scheme and Englehart 2011].

The device used in this work, the Myo armband, was already subject to a handful of works [Allard et al. 2016; Sathiyanarayanan 2016], for instance to operate a sound software [Nymoen et al. 2015]. To the best of our knowledge, no currently available system meets the requirements for a live adaptive mapping of complex and dynamic body motions to the control of an abstract robotic system. We introduce a solution that provides fast live training using a lightweight interface, making the interpretation of live expressive motion easy and flexible.

2.2 Human-Swarm Interaction

As described in the work of [Pendleton and Goodrich 2013]:

> For thousands of years, humans have used a variety of methods to direct and influence [these] decentralized systems (e.g. cattle ranching and sheep herding). The field of swarm robotics seeks to understand the principles underlying these systems and encode them into robots to create large robust robot teams. The goal of human-swarm interaction (HSI) is to develop methods for directing and influencing these large decentralized systems.

Inspired by biological swarms, the work of [Pendleton and Goodrich 2013] used attracting and repulsing potential forces to control the robots and influenced the group behavior through *leaders*, *predators* and *stakeholders*. Nevertheless, increasing the group size, even for a coherent group entity, has effect on the cognitive load and on the psycho-physiological state of the operator [Podevijn et al. 2016]. The cognitive load required to control a swarm through stakeholders do not scale well for most control styles and was only applied to potential-based strategy (e.g. flocking, formation, foraging). Smaller reaction time generally enhance engagement, but with swarms the emergent state from such latency was shown to help in some situations [Walker and Nunnally 2012]. Attractors (potential field sink) and repulsors (potential field source) based interactions, also referred to as leaders and predators, are sometimes more intuitive, for instance by using a mission planer to set goals or with a virtual leader (e.g. a user's phone or laptop perceived in the swarm as a member). The Naval Postgraduate School Advanced Robotic Systems Engineering Laboratory (ARSENL) developed a swarm of quadcopters loaded with multiple behavior binary code files before launch,

so the user can activate and deactivate them while in flight. This approach falls into the same type of control as our work: behavior selection. Other types of swarm interaction, such as proximal interaction, parameters settings and environmental control will not be covered here since they are more challenging to the user [Kolling et al. 2016].

Robotic swarm interfaces can also benefit from unique concepts such as a haptic geometric device to control a patrol formation [McDonald et al. 2017] and emergent states from user input timing (neglect benevolence) [Walker and Nunnally 2012]. Swarm intelligence is still an uncharted territory in terms of novel interaction paradigms. Our work leverages the most robust and scalable approach, state control (where states are expressive motions, referred to as moods), but rely on the faculty of humans to make sense of group motions. Indeed, the work of [Brown et al. 2015] showed that operators are able to distinguish and interpret collective states in robotic swarms. Consequently, we postulate that relating the user's expressive motions to the swarm collective states, designed as expressive motions, will increase the perceived connection.

In terms of existing platforms for swarm user interfaces, a notable contribution was the release of the tabletop robots named Zooids [Goc et al. 2016]. These robots led to a first study of robots group motion perception (emotional response) [Dietz et al. 2017] and they were used to examine the perception of abstract robotic displays [Kim and Follmer 2017], an interesting approach for ubiquitous robots. The interactions in these studies are limited to physical manipulation of individual robots by the operator and passive observation of their behaviors. Our user study is a first attempt at establishing a bond between these Zooids robots and an operator.

2.3 Expressive motion

Human body language is complex and meaningful. In robotic applications, the focus is most often on simple gesture recognition [Xu et al. 2008]. However, whole-body motion can bring significantly more information, such as intention and mood. Dance is an active and conscious activity playing on the potential of human body motion. Dancers and choreographers are trained to identify the connections between body motion and its mapping to the audience's emotions. Without this expertise and a large emotional encyclopedia, this ability is challenging to formalize. Thus, the designation of body movements and their emotional association should primarily be driven by the dancers and choreographers. With such knowledge, it is possible to design a system reacting to *perceived* emotions by categorizing the body motion [Venture et al. 2016].

Central to this work, the concept of a mood attributed to an expressive motion is taken from choreography. Choreographic theory identifies four key elements in dance: design, dynamics, rhythm, and form [Blom and Chaplin 1982]. Dances can be analyzed and designed in terms attributed to these domains. A method that characterizes the dance movements is a dance notation, to which Rudolf Laban made the most significant contribution by developing one base on four domains: body, effort, state and shape [de Souza 2016]. As pointed out by [Simmons and Knight 2017], a handful of works have used dance notation to characterize people's emotions and behaviors. [Simmons and Knight 2017] used the Laban Efforts notation to mimic users' motion with small fixed robots and showed positive effect on children's behaviors.

In our study, moods are physiological states, observed from clusters of body motions, based on the perceived physiological state, such as emotions, that they each create, or were designed to inspire. The moods in our study were designed by dancers or choreographers. We did not restrict the classification to fit any notation model, but our hypothesis is that a classifier can be trained to recognize expressive motion structure. For instance, Bayesian binning was shown to perform well on the recognition of Taekwon-Do movement primitives with a Vicon system, for only a specific set of kick primitives [Endres et al. 2016].

The use of wearable devices for natural interaction has also been demonstrated in the dance community: the German group Palindrome previously used EMG for visual amplification of a dancer's movements [Salter 2010]; Bill Vorn from Concordia University in Canada created the *Grace State Machine*, a parallel platform mechanism stacked in columns moving according to the biometrics of a dancer, including EMG sensors [Vorn 2016].

3 CLASSIFIER DESIGN

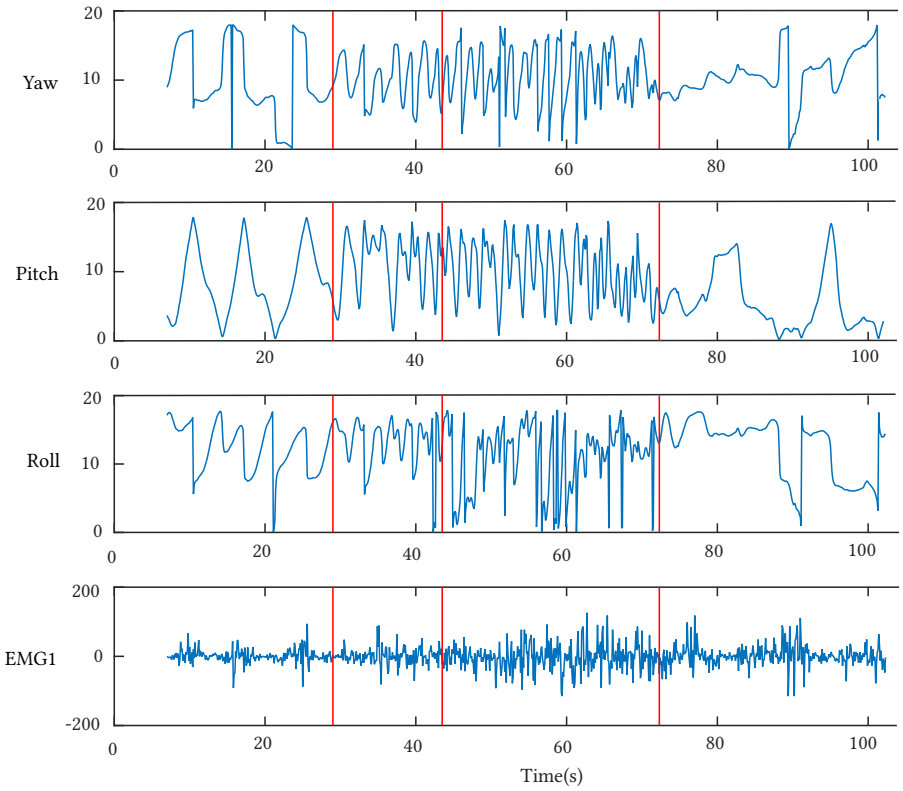

Fig. 1. An example of signals for the EMG and the IMU of the Myo armband, for four different dance movements. Vertical red lines indicate the transition between two movements. The goal of our system is to identify these four movements automatically.

The core contribution of this work is to use machine learning to distinguish between the users' moods from signals captured by the Myo armband, aiming for a system robust to uncertainties and noise. Let us recall here that in this work moods are the user defined labels associated to expressive whole-body motions. Fig. 1 shows a number of such Myo signals for four different expressive motions. As opposed to gesture recognition, expressive motions are dynamic. The fluctuations of the sensors signals over time are quite meaningful in this context, and features must be selected accordingly. As a design criterion, we restrict the classification delay to a maximum of 1 second, in order to let the system perceive a change over time in the motion rhythm and dynamics .

The rest of this section discusses the creation of the design dataset, the principal features used, and the methodology used to extract the most meaningful features. The use of *windows* in the following text refers to a portion of a determined time length in the dataset extracted to be classified.

3.1 Dataset building

To develop a new learning solution, the most basic requirement is a reliable dataset. Unfortunately, when it comes to whole-body motion, as opposed to image processing for instance, datasets are rare, and generally use external motion capture systems. The user study presented in this paper aims at correcting this situation for the next researchers willing to work in this area with a public release of the evaluation dataset [St-Onge et al. 2018a]. Initially, a professional dancer and choreographer assisted our team in building a first dataset, referred to as the design dataset.

Fig. 2. Dancer Claudia Tremblay during one of the performance for the design dataset at Laval University.

Our collaborator (see Fig. 2) wore two Myo armbands: one on the forearm and one on the calf. This configuration was selected to ensure a single Bluetooth receiver (on a laptop or a smart device) can capture data with the highest rate on all armbands. An armband on the upper body and one on the lower body allow a broader range of movements than using only half the body. This was qualitatively observed when testing with our first collaborator. Both IMU orientations were extracted at 50 Hz and the forearm's eight EMG channels at 200 Hz (the armband's maximal frequency). The performer was instructed to develop three different choreographic moods. Fig. 2 illustrates different body postures related to the same mood. The moods were differentiated by the emotion that they each subjectively represented for the performer. The training sets were recorded with the performer repeating each sequence for 20 seconds. This method of collecting training data offers the advantage of obtaining ground truth labels without any input from the dancer/choreographer. From this lexicon, the dancer then created three performances of about three minutes each. One of these served as a validation set, and the other two to test the performance of the classifier. A fourth performance was created to measure the impact of variable body orientation on the classifier's accuracy, using the same lexicon (as discussed in Section 4.3). Even if, as stated in Section 2.3, our objective is not to extract the movements' characteristics to build a notation model, the most relevant signals' features detailed in this section for a single user are shown to work on many users (as demonstrated in Section 6).

3.2 Overlapping Time-window Classification

To increase the training set size (as data is expensive and time-consuming to collect), overlapping windows are employed as a form of data augmentation. New windows were generated by sliding the window by 100 *ms* (corresponding to five

IMU samples and 20 EMG samples). For a time window of duration L, this means that $L - 1$ IMU samples were shared between consecutive windows. On top of providing more samples, this approach minimizes the latency of classification during a live performance, without the need to generate synthetic examples.

3.3 Initial Feature Extraction

The training algorithms require a set of meaningful features composed from the sensors outputs. A state-estimation filter built in the IMU already provides the orientation of the device on three axes. However, the raw EMG data is very noisy and difficult to directly associate with a specific movement. In both cases, the selected features need to consider the evolution of the motion (dynamics) over time. Based on the general similarity between accelerometer data from tactile sensing and the EMG signals of the Myo, some features were borrowed from work pertaining to surface identification [Giguere and Dudek 2011], complemented by features used for gesture recognition [Phinyomark et al. 2010, 2009]. In the end, we extracted the following list of features from temporal windows of one second from the raw signals:

- Minimum and maximum values,
- Mean, variance, skewness, kurtosis and the fifth moment,
- Integrated Values (IV - sum of the absolute values of the signal),
- Mean Absolute Value (MAV - average of the sum of absolute value of the signal),
- MAV1 (similar to MAV but using a weighted average with more weight given on central values),
- MAV2 (as MAV1, with a different weight distribution),
- Root Mean Square (RMS - The square root of the mean square of the signal),
- Zero Crossing (ZC - Number of time the signal cross the value zero),
- Waveform Length (WL - The sum of the difference between two consecutive data point over the signal),
- Slope Sign Change (SSC - The number of change between positive and negative slope of the signal),
- Willison Amplitude (WAMP - number of overreached difference in the signal amplitude), and
- Median Frequency.

All these features are regrouped in feature ensembles generated from the sensors' data to form as many examples for the training algorithms. With the eight channels of the EMGs and the three channels of the IMU, the dataset has 11 channels per armband and 17 features for each channel, thus a feature ensemble of 187 features per device. The algorithms are trained on one-second windows, called examples, each made of ten adjacent, non-overlapping sub-windows. Employing multiple sub-windows within a window allows for a measure of the features' variation through time within one example. Note that as the IMU and EMG sensors are operating at different sampling rate, the number of data points per sub-window for each modality was set so that they covered the same amount of time.

3.4 Automatic Feature Selection

Feature selection was performed automatically, in an iterative manner. We exploited the fact that Random Forest (RF) classifiers naturally select the most informative features for classification purposes. At each iteration, the RF classifier was trained ten times with 500 estimators, then the average contribution of each feature was computed. During this procedure, the maximum number of features per tree equals to $log(N)$, where N is the number of features of the current iteration. The number of estimators and the number of features per tree were chosen based on the random search described in Section 4. Any feature that was not present among the top 20% was rejected, as the RF did not consider

them as part of the most relevant group. Features were treated individually for each signal, as the most informative features might vary from one sensor to another. After removing every feature not present in the top 20%, the process iterates. This procedure is repeated until *i)* the accuracy on the validation set starts to decrease for two consecutive iterations or *ii)* the top 20% features has not changed between iterations. At this point, the best feature set is the one that provided the highest accuracy in validation. The final features selected for the *RF Classifier* were:

- for the IMU orientation: Maximum, Mean, Variance, IV, MAV and RMS;
- for the EMGs: Variance, IV, RMS and MAV.

Other experiments with RF on EMGs suggest to preserve all features and distribute weights among them [Xu et al. 2012]. Minimizing the feature space is more suitable considering our real-time requirement, as doing otherwise would incur in undesirable processing delays. Furthermore, we use only two of the eight available EMG channels (selected by cross-validation on the validation set). This was done following the automatic selection process: many of the EMG channels were redundant in our use case.

4 CLASSIFIER SELECTION AND LIMITATIONS

Several classifiers were tested to select the one achieving the best accuracy given a 1 *Hz* classification constraint. The four classifiers were AdaBoost, SVM-RBF, SVM-Linear and Random Forest (RF). They are all well known, state-of-the-art learning algorithms. This section discusses the selection of hyper-parameters for the various classifiers and the observed limitations of the proposed solution.

4.1 Classifier and hyper-parameter selection

For each of the four previously mentioned classifiers, hyper-parameters were selected using a 20 iterations random search with a 3-fold cross-validation on the training set (totaling 60 training per classifier). The reported metric is the average success rate over these three runs. The hyper-parameter list is as follows:

- RF: the number of estimators and the number of features used. The number of estimators available to the random search were 10, 30, 50, 70, 100, 200, 500 and 1000. The maximum number of features was either equal to $\sqrt{N}$, $log(N)_{base2}$ or the total number of features N;
- SVM-RBF (also designed as SVM-B): the soft margin tolerance hyper-parameter C and the parameters γ (the parameter related to Radial Basis Function RBF kernel). They were chosen from 10^{-5} to 10^{5} on a logarithm scale, with 20 values equally distributed along the range of each hyper-parameters;
- SVM-Linear: C (same as above) was chosen from 10^{-5} to 10^{5} on a logarithm scale with 20 values.
- AdaBoost: the number of estimators and the learning rate. The number of estimators was the same as RF. The learning rate was between 10^{-2} and 1 on a logarithm scale, with 20 values equally distributed on the scale.

All algorithms were taken from `scikit-learn` implementation [Pedregosa et al. 2011].

The results presented in Fig. 3 show the four classifiers trained with the features selected in section 3.4 with the test datasets created in Section 3.1. In these experiments, RF was clearly superior. This is expected, as RFs can cope with a large proportion of noisy features in very high-dimensional space. Indeed, RFs can handle thousands of features with a relatively small dataset. It also has the capacity to perform well on datasets that are non-linear and where features have a high-level of interaction between them [Qi 2012]. RF could also be trained, within a minute by a standard laptop. This is a key characteristic in a live training context, as the users need to get feedback rapidly about the reliability of the detection.

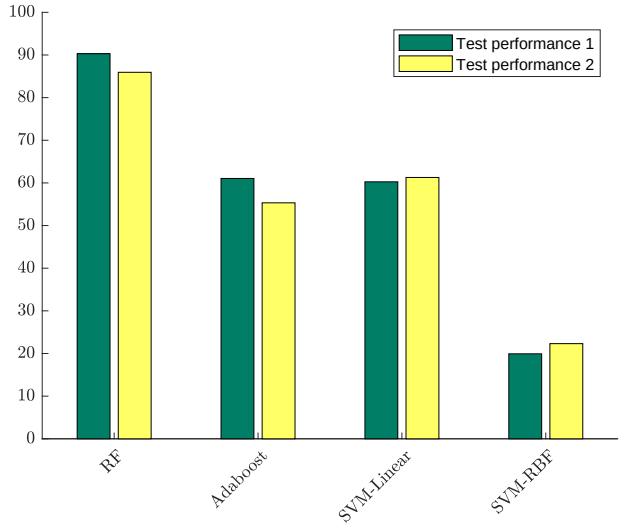

Fig. 3. Validation result comparing RF with AdaBoost, SVM-L and SVM-B.

4.2 Impact of IMU vs. EMG data for single and dual armbands

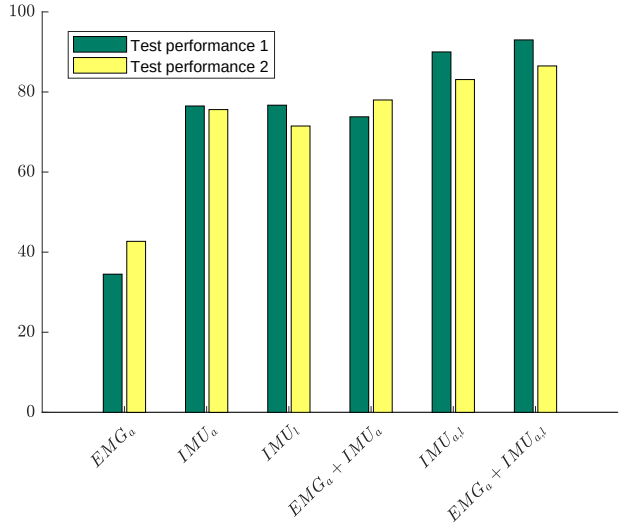

Fig. 4. Comparison of RF classification performances on the two test sets of design dataset using IMU and EMG on the arm (a) and the leg (l).

We compared the informative value of the EMG from the arm (EMG_a) against the IMU of the leg (IMU_l) and arm (IMU_a) using the accuracy of RF on the three classes. The classification results for all possible combinations are shown

in Fig. 4. The IMU, as a single source of information, performs better than the EMGs, which is not surprising given the complexity and the noise of the latter. Taken separately, the IMUs on the forearm and calf give similar performances but together they increase the performance of the classifier by more than 10%. Valuable information can still be gained from the EMG, however. Alone, the forearm Myo is better without EMGs signals; but when combined with the calf and forearm IMU, the two EMGs channels increased the accuracy up to 3%.

4.3 Absolute orientation over-fit

The dataset created for the design of our classifying solution uses the filter output of the armband: full device orientation. While these inputs are likely to stabilize the features extracted, they are coupled with absolute orientation of the body. In many applications, the execution of the users unrelated tasks will require changes of the body orientation without any influence on his psychological state.

A fourth performance of the design dataset was dedicated to test this issue. The performer was asked to reuse the previously designed moods, but this time with specific instructions to change her whole body orientation (north-east-south-west) as much as possible during each training. The new training dataset is referred to as *orientation training dataset*. The dancer also created a new performance of three minutes, again while continuously changing her body orientation. Apart from this, all the other parameters in this test (e.g. number of training/test examples, hyper-parameters, features) are the same as the ones used earlier. The accuracy on this new performance using the precedent training dataset is 75% for the three moods. However, when using the *orientation training dataset*, the accuracy grows to 84%. To overcome RF over-fitting the absolute orientation of the performer using the YAW data, only the variance of the YAW (for both IMU) is employed as it was the only feature that was agnostic to the absolute orientation of the dancer.

4.4 Random Forest performances on the design dataset

In the end, based on the most significant features selected, and integrating all the strategies discussed above, the two test sets of the design dataset led to the performance in Fig. 4. The final settings for this classification are:

- Random Forest algorithm with the previously detailed hyper-parameters (number of estimators and number of features)
- One second time windows to detect a class, with ten sub-windows.
- Overlaps of $0.1s$ between successive windows.
- Measurements of the orientation of the right forearm and right calf, with two EMG channels on the forearm.

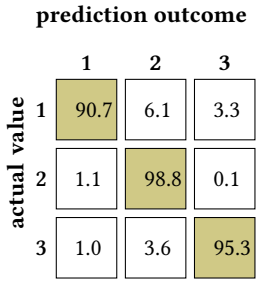

Fig. 5. Confusion matrix (in %) for RF trained on three *moods* of the test set of the design dataset.

It resulted in a performance of 94% for the three moods (average on both performances for 100 runs each) with most of the errors arising around the transitions. Fig. 5 shows the distribution of the classification errors.

5 ROBOTIC SWARM EXPRESSIVE MOTION

The work detailed in the previous sections is one-sided: our classifier observe and learn the moods of a user. To close the loop with a robotic system, we selected an abstract hardware implementation of non-intuitive control: a small tabletop robotic swarm. Its decentralized control is less common and intuitive than a one-on-one classic tele-operation configuration and required a dedicated software architecture. From these motion control premises, we built a user-friendly interface to explore and design expressive motion for the swarm. While the control modules are leveraged from previously published works and the exploration of a robotic swarm expressiveness with our interface is still undergoing experiments for further publications, their introduction here is mandatory to the understanding of the user study results and their scope.

5.1 Software ecosystem

The development of a common behavior for a swarm can be very challenging, especially considering that swarms are in essence decentralized systems, the behavior of which is based only on local interactions. To accelerate the implementation of swarm behaviors, our group designed a programming language specifically for swarm deployment: Buzz [Pinciroli and Beltrame 2016]. Buzz provides special constructs to address three essential concepts: a) shared memory (virtual stigmergy), b) swarm aggregation and c) neighbor operations. In a glimpse, Buzz includes a virtual machine processing the scripts without the need to compile them, in a platform agnostic environment. Its virtual machine must run on every unit of the swarm and with the exact same script, but units may have various capacities (heterogeneous swarm), leveraging swarm aggregation, or specialized sub-swarms. Example scripts are available online[1], as well as the Buzz virtual machine source code[2] and the exact behaviors described in this section[3], on Github.

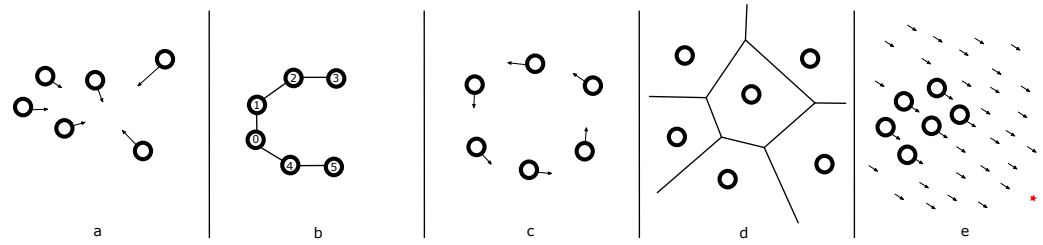

Fig. 6. Common swarm behaviors implemented: a- aggregation, b- graph formation, c- pursuit, d- random deployment, e- group movement with potential fields.

Since our implementation targets a small sized swarm, we leveraged from Buzz primitives the concepts of virtual stigmergy and neighbor operations. We use the former to agree on swarm-wide variables, such as the next state of the swarm in a given state machine. The latter serves most swarm intelligence algorithms: local interactions with neighbors, including receiving their relative position, are the core requirements of the algorithms implemented. As shown in Fig. 6, a set of common swarm behaviors were implemented in Buzz scripts: aggregation, formations from

[1] http://the.swarming.buzz/ICRA2017/cheat-sheet/
[2] https://github.com/MISTLab/Buzz
[3] https://github.com/MISTLab/ROSBuzz/blob/master/buzz_scripts/include/act/states.bzz

graphs, pursuit, random deployment and group movement through potential fields. With n robots in the swarm, b_{ji}, the bearing between robot i and j and d_{ij}, the distance between these two robots, we define:

(1) Aggregation as a simple behavior sending all robots to the swarm centroid, with each robot linear and rotational velocity, similar to the work of [Gauci et al. 2014], given by, respectively:

$$v_i = \frac{\sum_j^n d_{ij}}{n}, \text{ and } \omega_i = \frac{\sum_j^n b_{ij}}{n} \tag{1}$$

(2) Graph formation as an assignment problem attributing a graph node to each robot, similar to the work of [Pinciroli et al. 2016]. Our implementation takes a leader in the group to gather the bids of each robot for assignments, based only on their distance to the graph nodes, and allocate the position to each robot.

(3) Pursuit as a behavior to patrol around a point of interest, with each robot linear and rotational velocity, similar as the work of [Kubo et al. 2014], given by, respectively:

$$v_i = f b_{ij}, \text{ and } \omega_i = \frac{v_i}{r} - k\cos(\alpha), \tag{2}$$

with r the distance to point of interest, α, the bearing toward this point, and f,k, parameters of the pursuit behavior.

(4) Random deployment by creating a random tessellation of a user determined area and sending the robots to each cell centroid. Our cells are generated from Voronoi method, using Fortune sweep line algorithm, a method similar to some extent as the work of [Cort et al. 2004].

(5) Potential field group movement by computing a velocity vector from the swarm centroid and a point of interest:

$$v_i = r - \frac{\sum_j^n d_{ij}}{n}, \text{ and } \omega_i = \alpha - \frac{\sum_j^n b_{ij}}{n}, \tag{3}$$

with r the distance to point of interest, α, the bearing toward this point.

In a Buzz script, a function called *moveto* takes the velocity vector (direction and magnitude) computed from the behavior above, as argument and process it through a C *closure*[4]. This closure deals with low-level control using available hardware on-board controller. In the end, while the exact path of each robot is not determined, the group motion parameters and goal locations are scripted.

5.2 Hardware implementation

The selection of the robotic platform was mainly driven by: 1- size and portability to conduct the study in various studios, 2- abstract shapes (non-anthropomorphic nor zoomorphic) to focus on group movement expressiveness. The Zooids are small tabletop cylindrical robots of 2.6 cm diameter, localized from structured light emitted by a ceiling projector [Goc et al. 2016]. While our behavioral scripts can be ported on any hardware platform (as explained above), we selected the Zooids for the minimal setup time and ease of transportation. A robotic swarm sharing the same space as the user may most likely be perceived differently in interactions such as the one we designed, but we are looking for command center scenarios for remote control of swarms and thus we find it more suitable to have a separated space for the user and the robots (as discussed in Section 6.3).

[4]Buzz closures are C functions registered in BVM, available for use within Buzz scripts. They provide the BVM with access to external input and output, such as ways to interact with the hardware.

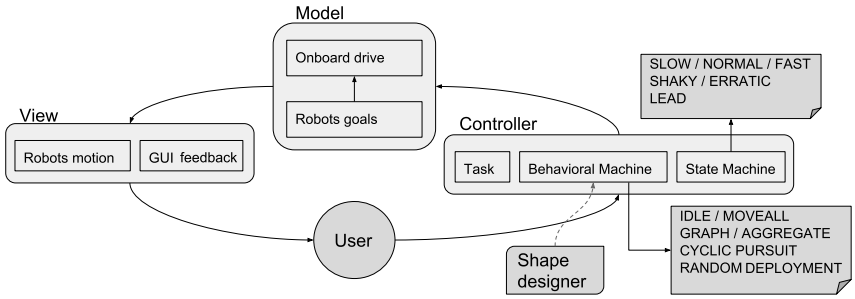

Fig. 7. Architecture of the swarm motion design software

The Buzz *moveto* closure implemented on the Zooids calls their on-board position control function, which has its source code available online[5]. In order to be able to explore the expressiveness of the robot's motion from quality of their movements, we manipulate some low-level variables of the on-board controller:

(1) the **maximum velocity** allows to change the general velocity of the Zooids movement without rendering the system unstable (such as playing with the controllers gains can do).
(2) adding artificial **positioning noise** can either introduce small variations on the linear motions or large movement artifacts.
(3) adding artificial **delay to the movement** commands allow us to manipulate the synchronicity of the movements, such as creating the impression of a leader in the swarm.

5.3 Motion design

To be as flexible as possible, we designed a software infrastructure for non-expert operators to design robotic swarm motions, based on the previous behaviors and parameters. The architecture we propose is derived from a classic model-view-controller (MVC) approach, integrating the swarm motion descriptors (tasks, behavioral, and state machine), the robots control, the system feedback, and the user multimodal inputs. We detail below the main components in the overall view of Fig. 7. The physical implementation requires the robots to comply with two constraints: 1) a communication path exists to all robots in the swarm, including the control station, and 2) a global positioning system is available. The second hypothesis is verified for most outdoor swarms using GPS as well as many indoor robots, either using motion-capture systems, structured light, or beacons.

In order to design various motions and manipulate their movement qualities, the first element of the design interface is the **behavioral machine**. The behavior of a swarm is the set of rules dictating their cohesion and their interaction with one another. Fig. 7 shows the behaviors discussed above together with *IDLE*, a state forcing all robots to stop where they are and wait until the next change of state.

Notwithstanding the behavioral machine, the robots can move following different dynamics, available in the **state machine**. However, several parameters can be controlled, such as the velocity range, the drive controller gains, and the accuracy of the target goal for each robot. To simplify the exploration of the parameter space and ensure stability we defined a set of available states:

- NORMAL: the default state, using the baseline motion parameters

[5] https://github.com/ShapeLab/SwarmUI/blob/master/Software/Microcontroller/Zooid_v2/src/position_control.c

- SLOW: uses a set of parameters that slows down all robot movement
- FAST: uses a set of parameters that accelerates all robot movement
- SHAKY: generates random individual goals so each robot oscillates near their target
- ERRATIC: generates random erroneous goals once in a while for a limited number of robots
- LEAD: move to a goal or to a shape in two steps, first the swarm leader, than all others.

6 HYBRID PERFORMANCE WITH ROBOTS

A number of challenges arise when designing an experiment for this work. Our aim is to demonstrate: 1- the performance of the moods classifier from expressive motions, 2- the potential of expressive motions to stimulate a connection with an abstract robotic system: a swarm. To begin with, we intentionally targeted a specific background in participants to give us sensible insights for a study aiming at evaluating the use of expressive motion: dancers and choreographers. They are experts of body motion, let it be human or artificial. We believe the conclusions obtained from their answers can better help us define the motion parameters for a broader spectrum of users.

The tasks performed by the robots under the users' command shall preferably be artistic to fit the target participants, but also because we are still at the inception of understanding and manipulating the expressivity of a robotic swarm [Levillain et al. 2018]. Thus in order to explore the expressivity of the swarm, we tasked choreographers not taking part in the study, with the design of expressive motions for a small tabletop robotic swarm. The concept of a hybrid performance is that the swarm ultimately determines its own movements from the performer's, allowing the emergence of an improvised and hybridized "pas de deux".

6.1 Methods

We recruited 27 participants with good knowledge and experience of dance. From the 27 participants, 4 are men, 22 women and 1 selected 'other'; two thirds are dance students (19), while the others are freelancers (8). All participants had training on choreographic work on top of their interpretation work and the freelancers were all experienced choreographers as well as performers. The participants did not receive any kind of financial compensation for the study: they participated out of curiosity for natural interaction with robotic systems. We made blocks of user study sessions at various schools of dance in Montreal, Canada, setting up the robotic platform in their studio to simplify the participants schedule. The study protocol was approved by the Polytechnique de Montréal's ethical committee. Participants signed an informed consent form to partake in the study.

The classifier detailed in the previous sections was used to control a swarm of tabletop robots in a hybrid dance performance. Technically, these performances consist of live mapping of moods expressed by the users through their body motion to the corresponding mood of the robotic swarm. This mapping stimulates the perception of a human-swarm connection. To conduct multiple sequential studies, we alternated between two sets of six Zooids robots.

Before conducting the study on our 27 participants, we tasked a small group of choreographers with the design of six expressive motions using our swarm of six Zooids: fear, happiness, sadness, surprise, disgust, anger. Six emotions that are known to be the easiest to name (self-recognize) when we experience them ourselves [Ekman 1992]. Using the motion design interface discussed in Section 5, the focus group came to agree on the six expressive motions. We then conducted a small (six participants) qualitative user study to confirm the perceived moods (in this case, emotions) for each expressive motion designed and we made small adjustments according to the participants' feedback. The resulting six Zooids' moods were meant to be mapped on the users' self-defined moods for their improvised hybrid performance.

Fig. 8. Six Zooids performing with a dancer.

For each participant, the armband was systematically tightened to its maximum and slid up the user's forearm/calf until the circumference of the armband matched that of the forearm/calf. This was done in an effort to reduce bias from the researchers, and to emulate the wide variety of armband positions that users might adopt without prior knowledge of optimal electrode placement.

Our protocol uses three parts: 1) the participant passively observes the Zooids, 2) the participant creates expressive dance motions, and 3) the participant executes an improvised performance to which the robots react live. At the beginning, the researcher conducting the experiment explains how swarm intelligence is used to stimulate motion on the robots, without controlling each individual, but rather through a common objective and a handful of cohesion parameters. In part 1, the participants are asked to observe and comment scripted motion sequences of the Zooids (these results are part of a separate study [Levillain et al. 2018]), thus decreasing the novelty effect on their following assessments. In part 2, they have ten minutes to create from three to six expressive dance sequences selecting any of the suggested themes: fear, happiness, sadness, surprise, disgust, anger. This constraint allowed us to prescript expressive motions on the robots for each theme (see Section 5). When they find their expressive motions to be ready, they perform each one repeatedly in order to create three training sets for each (of variable lengths between 10s and 120s). We specifically instruct them to design their expressive motions without constraints, not considering, for instance, the armbands location. In part 3, following a fast (less than 2 minutes) training of the classifier on their expressive motions, the participant executes an improvised performance of three to six minutes structured with his or her lexicon

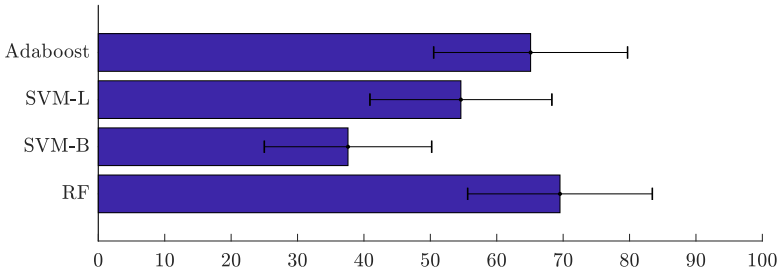

Fig. 9. Average classification performance of the RF, SVM-L, SVM-B and Adaboost classifiers on the 27 participants.

of expressive motions. During this performance, the Zooids are reacting live to the dancer motion and the resulting hybrid performance is recorded on video. At the end, the participant completes a survey to assess the level of connection perceived and the main contributing factors by observing the performance video. Even if the recordings were only meant to give them an outside point of view on the interaction, most were so enthusiastic of the results that they asked to keep the file. A subset of the 27 performance videos is available online [St-Onge et al. 2018b].

6.2 Classification performance

The resulting dataset of performances from the 27 participants of our study was labeled using the video recordings of the training together with the performance. To decrease the subjectivity impact in labeling, three researchers labeled each video and the resulting labels were compared. The labeling of the videos were made using *second* units and the overall agreement between the researchers, using Cohen's Kappa, is 96%. Most of the disagreements arose in mood transitions, as it is not clear, even for humans, when precisely the following mood starts to be prominent. This ground truth grants the possibility to assess the overall performance of the algorithm, as illustrated in Fig 9. The figure shows that the results of RF still outperform the comparable classifiers discussed previously. To be fair in the comparison, each algorithm used a random search to find its optimal hyper-parameters (as described in Section 4). The performances of the classifier may not seem convincing at first, but one must recall that the classes are expressive motion sequences, i.e. dance qualities, and as such really variable through time. In order to challenge the system, many participants merged expressive motions together and played with their execution of the moods.

It is also important to note that gestures were not created/selected/designed by the researchers. As such, they were not designed to be easily detected by our system. Rather, they were developed by the dancer, with aesthetic criteria in mind, to create a performance. The participants were not allowed to prepare before these short sessions, which, together with the variety of styles, grant us confidence on the flexibility of the system to various types of expressive motions.

Other parameters influence the results such as the number of moods and the experience of the performer. For instance, in Fig. 10, the confusion matrix to the left shows amazing predictions for a professional dancer developing three moods only, while the one to the right show more scattered results with five moods executed by a first-year student in dance. Nevertheless, the confusion matrix of the five mood performance demonstrates good results for most classes, the fifth, in this case 'anger', was often confused with the second, 'happiness', most likely because both were energetic and 'happiness' had more samples in the training set than 'anger'.

From the questionnaire results, overall, the participants were not very satisfied ($\tilde{x} = 45\%, \sigma = 17.8\%$) with their hybrid performance. Under the circumstances, lacking time to practice and explore the relation with the robots, it

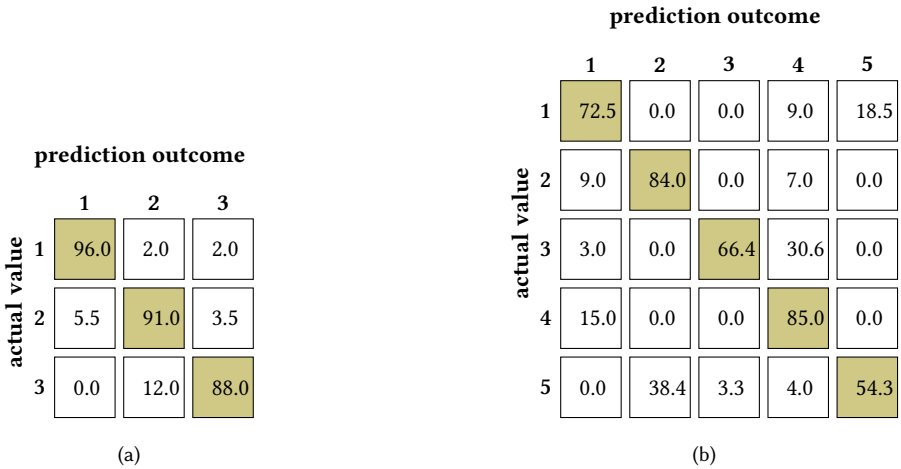

Fig. 10. Confusion matrix (in %) for RF trained on (a) three moods for the best performance and (b) five moods for an average performance of the user study dataset.

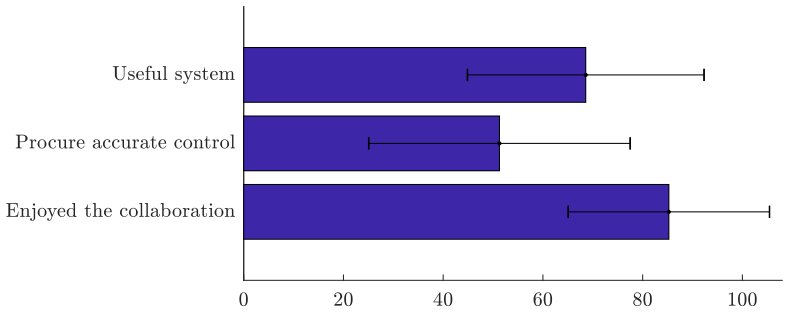

Fig. 11. Average scores related to the perceived connection with the robotic swarm.

is understandable that they are very severe with themselves. From our follow up informal discussions with each participant, they were very critical of their own performance, albeit not of the robots one, but we lacked the time to quantify this with multiple improvised performances (for instance, with and without the robots). However, a strong relationship can be found between the classifier performance and their self-evaluation (Friedman Anova: $p < 0.000001$), the better it performed the more they appreciated the resulting performance.

6.3 Users engagement

On top of evaluating the performance of our classifier, this study aims at quantifying the engagement that a state-based control can produce in the user. As stated earlier, it is less intuitive to interact with a group-like entity than an individual, but expressive motions can be leveraged to initiate a connection. In social robotics, a major metric to evaluate engagement is proxemics; the study of how human position with respect to the others [Levillain et al. 2017]. However, with remote robotic systems, such as drones flying above the user, or robots in remote areas, proxemics can hardly be used. We designed our study to cover a similar situation: the robot playground is on a table, physically

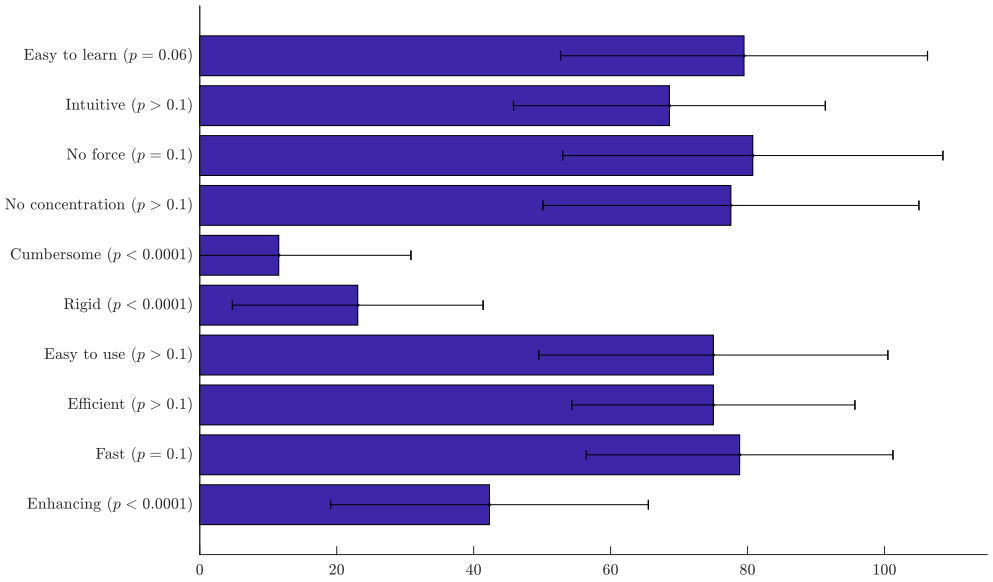

Fig. 12. Average scores related to the perceived usefulness of the system.

separated from the dance floor where the user is located. To assess the engagement of the user we rely on feedback collected with a survey inspired by the work of [Schmidtler et al. 2017]. Originally, this survey (named QUEAD) was intended to evaluate the perceived usability and acceptance of assistive devices. It was found to be well fitted to the use of armband sensors and motion classification. All answers were on a seven-point Likert scale, considered to provide equivalent psychological distance between the scores [Carifio and Perla 2007]. Nevertheless, since this study presents a large number of tied ranks for a relatively small dataset (27 participants), we use a non-parametric evaluation of the parameters' influence (Friedman Anova).

Can the user perceive a connection with the swarm using this system?

Asking the users directly if they felt connected with the system can generate biased results, since they were aware that the hybrid performance aimed at creating a relation between the participants' movement and the swarm motion. Instead, we use three parameters: the perceived usefulness of the system, its accuracy, and the enjoyment associated with the collaboration. Fig. 11 shows that the collaboration was found to be highly enjoyable by the participants ($\tilde{x} = 85\%, \sigma = 20\%$) and the system was mostly perceived as useful ($\tilde{x} = 67\%, \sigma = 24\%$), but the accuracy of the system was not perceived as good ($\tilde{x} = 51\%, \sigma = 26\%$). All three parameters were closely related to the classifier performance ($p < 0.0001$).

What parameters influence the perceived usefulness?

Fig. 12 show the main parameters evaluated for the system and their average score over the 27 participants. These questions were asked specifically about the "control system, consisting of the armbands and the software interpreting their movements". Surprisingly, we find among the most influential parameters of the perceived usefulness, physical

attributes such as if the system is perceived as cumbersome or rigid (not flexible enough to the user will), both with Friedman $p-values$ below 0.0001. If we can attribute most of the scores for these two parameters to the physical design of the Myo armband, it also means the algorithm training was not perceived as cumbersome and rigid either. The third most influential parameter is more obvious: the more the system was perceived as enhancing the user performance, the most useful it was perceived as well ($p < 0.0001$). As for all the other parameters evaluated, one can observe from Fig. 12 that the system was well perceived, but no strong statistical relation was found with the perceived usefulness.

6.4 Comparison with deep learning

In recent years, deep learning-based classifiers have imposed themselves as the state of the art in a vast array of applications. They were employed with great success for the classification of both IMU and EMG based signals [Côté-Allard et al. 2018; Jiang and Yin 2015]. However, they require more processing time and are not yet fit to live training in critical scenarios, such as the ones relevant to this work. Nevertheless, for completeness sake, we compare the performance of the proposed RF classifier and three different deep learning architectures.

As mentioned previously, due to the inherent limitations of the current context, the proposed architectures will have to contend with a limited amount of data for training. Deep learning algorithms are prone to over-fitting, especially on small datasets [Gal and Ghahramani 2016; Srivastava et al. 2014]. As such, in an effort to mitigate over-fitting, Monte Carlo Dropout (MC Dropout) [Gal and Ghahramani 2016], Batch Normalization (BN) [Ioffe and Szegedy 2015] and early stopping strategies are employed.

The following sub-sections present the three different architectures tested. In all cases, as to avoid indirect over-fitting, hyper-parameter optimization and architecture selection was performed employing only the first five users of the evaluation dataset. This subset of datasets will be referred to as the *deep learning validation dataset*. All implementations were implemented with PyTorch [Paszke et al. 2017]. Unless stated otherwise, the non-linear activation is the Parametric Rectified Linear Unit (PReLU) [He et al. 2015] and ADAM [Kingma and Ba 2014] is employed for the optimization of the networks.

The first architecture serves as a direct comparison with the other classifiers tested this far: the exact same 340 features are fed to a fully connected, three-hidden layers network, referred to as *DNN*. Each hidden layer is comprised of 340 nodes plus bias. Between each fully connected layer, the signal goes through Batch Normalization, then the non-linearity (PReLU) and finally MC Dropout is applied to reduce over-fitting.

As previously mentioned, the features employed for classification are calculated from data blocks of 100 *ms* each, with ten of those blocks forming an example of one second. To that effect, the second architecture was built in an effort to leverage this time-dependence intrinsic to the collected data. As such, a recurrent neural network using long short-term memory units(LSTM) [Hochreiter and Schmidhuber 1997] forms the core of this architecture, which can be seen in Fig. 13. This second architecture will be referred to as *LSTM*.

The third architecture tested is a Convolutional Network (ConvNet). The early layers of a Convolutional Network (ConvNet) can be viewed as hierarchical features learners [LeCun et al. 2015]. Consequently, Convnets have established themselves as the state of the art in multiples image-based recognition tasks for their ability to learn features that often supersede those that were handcrafted by experts. The idea behind this selected architecture is to leverage this feature learning ability for the classification of the moods. Previous work using the Myo Armband [Côté-Allard et al. 2018] has shown that ConvNet are particularly well suited for the classification of EMG-based hand gesture recognition. As a result, this architecture, referred to as *ConvNet*, takes heavy inspiration from such work.

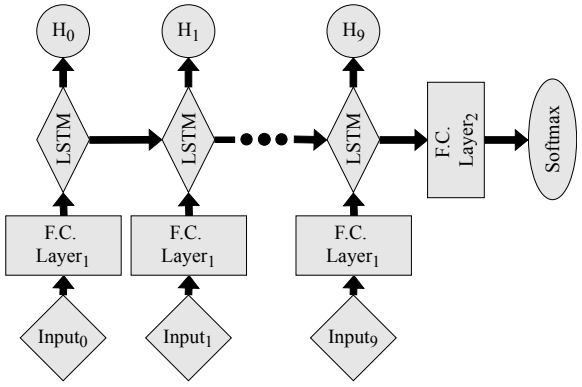

Fig. 13. LSTM's architecture. The shape of the input received by the network is: 10xYx34 (Sequence x Batch Size x Features), yielding ten sub-examples of shape Yx34 referred to in the diagram as Input$_i$ with i ∈ [1..9]. Each fully connected layer contains 50 neurons. Between each fully connected layer, the signal goes through Batch Normalization, then the non-linearity (PReLU) and finally MC Dropout is applied. The single layer LSTM contains 50 features in the hidden state and the hyperbolic tangent is employed for the non-linearity of the cell.

We selected spectrograms for the input of the proposed ConvNet as they are often used for the classification of both EMG and IMU data [Côté-Allard et al. 2018; Renaudin et al. 2012]. Several declinations of the ConvNet architecture were tested using the *deep learning validation dataset* handling the IMU and EMG signals both together and separated. In the end, convolutional layers did appear to learn similar discriminating features than those used for the RF classifier on the EMG data. However, convolutional layers did not seem to be able to extract sufficiently discriminating features for IMU-based data as they systematically led to drastically worse results. Thus the final architecture, presented in Fig. 14, uses convolutional layers only for the EMG-based input while the IMU data is represented by the features described for the other classifiers.

The spectrograms are calculated from the EMG signal with Hann windows of length 29 and an overlap of 20 yielding a matrix of 15x20 (frequency X time) for each of the eight sEMG channels.

Results

Table 1 shows the performances of the three deep learning architectures and the RF classifier. In order to cope with the stochastic nature of the algorithms, the accuracy of each are given as the average of all 27 participants over 20 runs. The Wilcoxon signed rank test [Wilcoxon 1945] is employed to compare the RF classifier against the other deep learning approaches. The null hypothesis is that the medians of the difference between the two group samples are equal (n=27). In all cases, the proposed RF classifier achieves higher accuracy on average than the deep learning approaches and that difference in accuracy is significant ($p < 0.00001$).

6.5 Transfer Learning on moods

Transfer learning (TL) can leverage the information learned from an original task (named the *source* task) to be able to perform a second task (*target* task) more easily. In the context of this paper, we try to learn from the participants so as to more easily predict the performance of a new participant. As such, each participant is considered a separated source-task pair. If transfer learning enhances the performance of the classifier, this would indicate that a generic similar structure exists between the moods expressed across the participants. To test this hypothesis, the TL algorithm

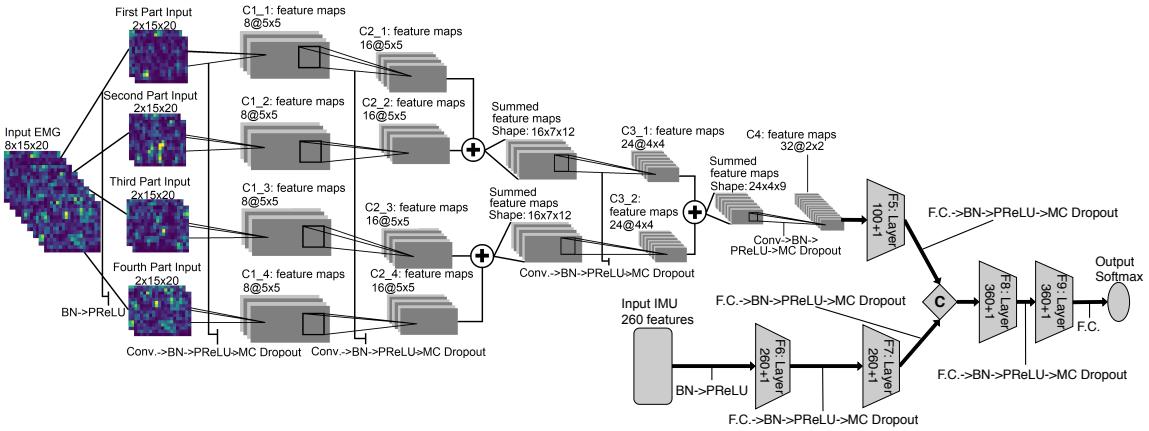

Fig. 14. ConvNet's architecture. In this figure, Conv refers to Convolution and F.C. to Fully Connected layers. The circles with a plus sign correspond to the element-wise summing of the feature maps. The diamond-shaped block indicates the concatenation operation of the feature maps together.

Table 1. Comparison of the three deep learning architectures . As the algorithms presented are stochastic by nature, the values reported are the average accuracy for the 27 participants over 20 runs. The Wilcoxon signed rank test is applied to compare the performances of the *RF Classifier* against the three others. The null hypothesis is accepted when $H_0 = 1$ and rejected when $H_0 = 0$ (with $p = 0.05$).

	RF Classifier	DNN	LSTM	ConvNet
Accuracy (%)	**69.97**	67.04	67.54	65.83
STD (%)	**13.58**	13.60	14.25	13.70
Rank	**1**	2.9	3.1	3
H0	-	0	0	0

presented in [Côté-Allard et al. 2018] was implemented as it was developed for a similar context. For simplicity, the DNN classifier was chosen as the base of our TL implementation.

Fig. 15 presents the TL algorithm architecture. The TL algorithm trains a single network that shares its weights across all participants while learning separate BN statistics for each participant. To learn on the target task, a second identical network with random weights is created while the original network's weights are frozen (except for the BN weights). Fig. 15 illustrates the connections between the two networks. These two linked networks form the target network which is trained normally on the target task.

By applying the described TL to the DNN, the average accuracy across all participants increases from 67.04% to 68.25%. This small, but nonetheless significant accuracy improvement (Wilcoxon Signed Rank test ($p < 0.05$)), indicates similar mood structure between the participants.

7 DISCUSSION AND FUTURE WORKS

Throughout this paper, we explore the scenario of leveraging a wearable sensor device to control a robotic swarm from expressive body motions. We demonstrate an end-to-end system that allows a dancer to create a lexicon of emotionally driven movement sequences that a robotic system is able to react to in real time. The proposed system employs Myo

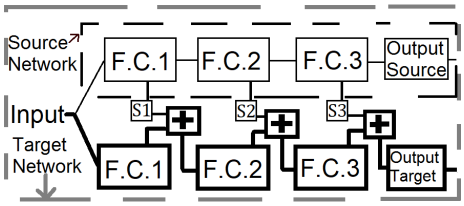

Fig. 15. The TL architecture employed. F.C. blocks correspond to fully connected layers. The source network corresponds to the three hidden layers DNN previously described. The S_i ($i \in [1..3]$) boxes represent a layer that scales its inputs by learned coefficients. Finally, the plus sign corresponds to the element-wise summation operation.

armbands, which are inexpensive, fast to use, non-intrusive, and work well without a line of sight. However, the signal they produce is noisy compared to motion capture or camera systems. Nevertheless, we show that our machine learning approach achieves a robust classification accuracy in real time at a frequency of 1 *Hz* of 94% for three *moods* using only 20 seconds of training data per movement. Testing the system on 27 participants, we achieved an average classification performance of 70% with three to six moods. We note that increasing the size of the windows improves the success rate, at the expense of its applicability in live settings (due to increased latency). We also present that combining EMG with IMU signals produces a small but noticeable increase in accuracy. Among the limitations of our system, we show some acceptable sensitivity to body orientation. We conjecture that the body orientation problem could be minimized by collecting more data for the training set.

The nature of the mood classification we propose in this article adds several challenges with respect to gesture classification: the datasets of this work are comprised of sequences of complex movements of variable size, which are arbitrarily defined by the performer. Far exceeding a one-second duration, the classifier has to predict the current mood after a small subset of the sequence. Additionally, because of the time constraints required by the dancer to perform these moods, the training dataset contains very few samples, making learning for the classifier even more difficult. Furthermore, as the system must remain as minimally intrusive as possible, complete body tracking of the subject is not currently possible. Finally, due to the subjective nature of the task, important differences can exist between the training set and the user's performance, as they tend to vary the speed, amplitude, and direction of their movements. All these factors combined make this type of mood recognition task an exciting challenge and it is our hope that the provided datasets will help the research community to tackle such an important and difficult task. We thought of various scenarios where this controller can be used in critical missions. For instance, a user can be asked to perform a series of calibration tasks beforehand, in order to be able to detect signs of stress or impression of success from his/her body motion. The system deployed in the field can then adapt its behavior based on how the operator "feels". Another more straightforward example is simply to deploy a system in the field, equipped with a series of high-level commands (takeoff, gohome, deploy to cover ground, follow someone, move together) and then ask the operator to design a motion he/she wishes to associate with each command. Since the training takes only minutes, the system would be fully intuitive and adapted to the operator on-site.

We show that the users in our study felt connected to the swarm in terms of enjoyment, and usefulness. The main contributions to the usefulness were the perceived enhancement of their performance together with the subtle physical presence of the controller.

Future experiments include collaboration with other specialists from the art world: object theater professionals to help on the robots motion design and actors to observe other forms of expressive body motions. Most importantly, this

research can lead to applications in critical emergency response missions to detect stress, fear, satisfaction, happiness and surprise, directing an assistive robotic system to the preferable course of action. At first glance, both contexts may seem conceptually far, but we found their requirements to be similar: 1- artists and first responders are already fully cognitively engaged with their tasks and cannot afford to be distracted, 2- they are both physically involved, and 3- they perform in various locations. For instance, the operator of a fleet of robots in an emergency response situation may get overwhelmed by other tasks. If the robotic system can detect signs of stress or high cognitive load from the user body dynamics, it can switch to a safe state, such as hovering in the case of flying robots. We are currently studying how to use these tabletop robots to monitor and command an outdoor fleet of drones. While the movements on the scale of the table are smaller than in this experiment, we are leveraging the findings to create a bond and maintain the user attention towards the swarm.

ACKNOWLEDGMENTS

This work was supported by the NSERC Strategic Partnership Grant 479149-2015 and partially supported by the Research Council of Norway through its Centres of Excellence scheme, project number 262762. The authors would like to thanks the incredible patience of Claudia Tremblay, the performer who made the design dataset possible. Armando Menacci, professor at the Dance School of UQAM showed a great support to our work and helped provide the participants. Finally, the authors would like to acknowledge the generosity of all the artists involved in this work and the enthusiasm they showed toward our research.

REFERENCES

J. Aggarwal and Q. Cai. 1999. Human motion analysis: a review. *Computer Vision and Image Understanding* 73, 3 (1999), 428–440.

J.-H. Ahn, C. Choi, S. Kwak, K. Kim, and H. Byun. 2009. Human tracking and silhouette extraction for humanâĂŞrobot interaction systems. *Pattern Analysis and Application* 12 (2009), 167–177.

Ulysse Côté Allard, François Nougarou, Cheikh Latyr Fall, Philippe Giguère, Clément Gosselin, François Laviolette, and Benoit Gosselin. 2016. A convolutional neural network for robotic arm guidance using sEMG based frequency-features. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2464–2470.

P. K. Artemiadis and K. J. Kyriakopoulos. 2011. A Switching Regime Model for the EMG-Based Control of a Robot Arm. *IEEE Transactions on Systems, Man and Cybernetics–Part B: Cybernetics* 41, 1 (February 2011).

L.A. Blom and L.T. Chaplin. 1982. *The Intimate Act of Choreography*. University of Pittsburgh Press.

Daniel Sundquist Brown, Michael A. Goodrich, Shin-Young Jung, and Sean C. Kerman. 2015. Two Invariants of Human Swarm Interaction. *Journal of Human-Robot Interaction* 5, 1 (2015), 1.

N. Bu, M. Okamoto, and T. Tsuji. 2009. A Hybrid Motion Classification Approach for EMG-Based HumanâĂŞRobot Interfaces Using Bayesian and Neural Networks. *IEEE Transactions on Robotics* 25, 3 (June 2009).

James Carifio and Rocco J. Perla. 2007. Ten Common Misunderstandings, Misconceptions, Persistent Myths and Urban Legends about Likert Scales and Likert Response Formats and their Antidotes. *Journal of Social Sciences* 3, 3 (2007), 106–116.

Jorge Cort, Member Ieee, Sonia Mart, Member Ieee, Timur Karatas, Francesco Bullo, Member Ieee, Jorge Cortés, Sonia Martínez, Timur KarataÅ§, and Francesco Bullo. 2004. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation* 20, 2 (2004), 243–255. arXiv:math/0212212

Sarah Cosentino, Klaus Petersen, Z H Lin, Luca Bartolomeo, Salvatore Sessa, Massimiliano Zecca, and Atsuo Takanishi. 2014. Natural human-robot musical interaction: understanding the music conductor gestures by using the WB-4 inertial measurement system. *Advanced Robotics* 28, 11 (2014), 781–792.

Ulysse Côté-Allard, Cheikh Latyr Fall, Alexandre Drouin, Alexandre Campeau-Lecours, Clément Gosselin, Kyrre Glette, François Laviolette, and Benoit Gosselin. 2018. Deep Learning for Electromyographic Hand Gesture Signal Classification Using Transfer Learning. *arXiv preprint arXiv:1801.07756* (2018).

Angela Loureiro de Souza. 2016. *Laban Movement Analysis—Scaffolding Human Movement to Multiply Possibilities and Choices*. Springer International Publishing, Cham, 283–297.

Griffin Dietz, Jane L. E., Peter Washington, Lawrence H. Kim, and Sean Follmer. 2017. Human Perception of Swarm Robot Motion. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '17*. Denver, 2520–2527.

K. DoHyung, L. Jaeyeon, Y. Ho-Sub, K. Jaehong, and S. Joochan. 2013. Vision-based arm gesture recognition for a long-range humanâĂŞrobot interaction. *Journal of Supercomputer* 65 (2013), 336–352.

Paul Ekman. 1992. Are there basic emotions? (1992), 550–553 pages.

Dominik Endres, Enrico Chiovetto, and Martin A. Giese. 2016. *Bayesian Approaches for Learning of Primitive-Based Compact Representations of Complex Human Activities*. Springer International Publishing, Cham, 117–137.

Ramon A. Suarez Fernandez, Jose Luis Sanchez-lopez, Carlos Sampedro, Hriday Bavle, Martin Molina, and Pascual Campoy. 2016. Natural User Interfaces for Human-Drone Multi-Modal Interaction. *2016 International Conference on Unmanned Aircraft Systems (ICUAS)* (2016), 1013–1022.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*. 1050–1059.

Melvin Gauci, Jianing Chen, Tony J. Dodd, and Roderich Groß. 2014. Evolving aggregation behaviors in multi-robot systems with binary sensors. In *Distributed Autonomous Robotic Systems*, Vol. 104. 355–367.

P. Giguere and G. Dudek. 2011. A Simple Tactile Probe for Surface Identification by Mobile Robots. *IEEE Transactions on Robotics* (2011), 534–544.

Mathieu Le Goc, Lawrence H Kim, Ali Parsaei, Jean-daniel Fekete, Pierre Dragicevic, and Sean Follmer. 2016. Zooids : Building Blocks for Swarm User Interfaces. In *UIST*. Tokyo.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*. 1026–1034.

R. Herrera-Acuña, V. Argyriou, and S. A. Velastin. 2015. A Kinect-based 3D hand-gesture interface for 3D databases. *Journal on Multimodal User Interfaces* 9, 2 (June 2015), 121–139.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*. 448–456.

Wenchao Jiang and Zhaozheng Yin. 2015. Human activity recognition using wearable sensors by deep convolutional neural networks. In *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 1307–1310.

Lawrence H Kim and Sean Follmer. 2017. UbiSwarm: Ubiquitous Robotic Interfaces and Investigation of Abstract Motion As a Display. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3 (2017), 66:1—-66:20.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

Andreas Kolling, Phillip Walker, Nilanjan Chakraborty, Katia Sycara, and Michael Lewis. 2016. Human Interaction with Robot Swarms: A Survey. *IEEE Transactions on Human-Machine Systems* 46, 1 (2016), 9–26.

Masao Kubo, Hiroshi Sato, Tatsuro Yoshimura, Akihiro Yamaguchi, and Takahiro Tanaka. 2014. Multiple targets enclosure by robotic swarm. *Robotics and Autonomous Systems* 62, 9 (2014), 1294–1304.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436.

Florent Levillain, David St-onge, Elisabetta Zibetti, and Giovanni Beltrame. 2018. More than the sum of its parts : assessing the coherence and expressivity of a robotic swarm. In *IEEE International Conference on Robot and Human Interactive Communication*. Nanjing, 583–588.

Florent Levillain, Elisabetta Zibetti, and Sébastien Lefort. 2017. Interacting with Non-anthropomorphic Robotic Artworks and Interpreting Their Behaviour. *International Journal of Social Robotics* 9, 1 (2017), 141–161.

M. V. Liarokapi, P. K. Artemiadis, and K. J. Kyriakopoulos. 2013. Task Discrimination from Myoelectric Activity: A Learning Scheme for EMG-Based Interfaces. In *Proceedings of the IEEE International Conference on Rehabilitation Robotics*. 1–6.

Xilin Liu, Milin Zhang, Andrew Richardson, Timothy Lucas, and Jan Van Der Spiegel. 2016. The Virtual Trackpad: an Electromyography-based, Wireless, Real-time, Low-Power, Embedded Hand Gesture Recognition System using an Event-driven Artificial Neural Network. *IEEE Transactions on Circuits and Systems II: Express Briefs* (2016).

Samuel J. McDonald, Mark B. Colton, C. Kristopher Alder, and Michael A. Goodrich. 2017. Haptic Shape-Based Management of Robot Teams in Cordon and Patrol. *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction - HRI '17* (2017), 380–388.

Luca Mottola, Mattia Moretta, Kamin Whitehouse, and Carlo Ghezzi. 2014. Team-level Programming of Drone Sensor Networks. *SenSys '14 Proceedings of the 12th ACM Conference on Embedded Network Sensor SystemsSystems* (2014), 177–190.

Kristian Nymoen, Mari Romarheim Haugen, and Alexander Refsum Jensenius. 2015. MuMYO – Evaluating and Exploring the MYO Armband for Musical Interaction. *Proceedings of the International Conference on New Interfaces for Musical Expression* (2015), 215–218.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. In *NIPS-W*.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and al. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

Jacques Penders, Lyuba Alboul, Ulf Witkowski, Amir Naghsh, Joan Saez-Pons, Stefan Herbrechtsmeier, and Mohamed El-Habbal. 2011. A Robot Swarm Assisting a Human Fire-Fighter. *Advanced Robotics* 25, 1-2 (2011), 93–117.

Brian Pendleton and Michael Goodrich. 2013. Scalable Human Interaction with Robotic Swarms. *AIAA Infotech@Aerospace (I@A) Conference* (2013), 1–13.

Klaus Petersen, Jorge Solis, and Atsuo Takanishi. 2010. Musical-based interaction system for the Waseda Flutist Robot: Implementation of the visual tracking interaction module. *Autonomous Robots* 28, 4 (2010), 471–488.

A. Phinyomark, S. Hirunviriya, C. Limsakul, and P. Phukpattaranont. 2010. Evaluation of EMG Feature Extraction for Hand Movement Recognition Based on Euclidean Distance and Standard Deviation. In *Proceedings of the IEEE International Conference on Computer Telecommunications and Information Technology*.

A. Phinyomark, C. Limsakul, and P. Phukpattaranont. 2009. A Novel Feature Extraction for Robust EMG Pattern Recognition. *Journal of Computing* 1, 1 (2009), 71–80.

Carlo Pinciroli and Giovanni Beltrame. 2016. Swarm-Oriented Programming of Distributed Robot Networks. *Computer* 49, 12 (2016), 32–41.

Carlo Pinciroli, Andrea Gasparri, Emanuele Garone, and Giovanni Beltrame. 2016. Decentralized Progressive Shape Formation with Robot Swarms. In *13th International Symposium on Distributed Autonomous Robotic Systems (DARS)*. London, 1–13.

Gaëtan Podevijn, Rehan O'Grady, Nithin Mathews, Audrey Gilles, Carole Fantini-Hauwel, and Marco Dorigo. 2016. Investigating the effect of increasing robot group sizes on the human psychophysiological state in the context of human-swarm interaction. *Swarm Intelligence* 10, 3 (2016), 1–18.

Y. Qi. 2012. Random Forest for Bioinformatics. *Ensemble Machine Learning* (2012), 307–323.

Valérie Renaudin, Melania Susi, and Gérard Lachapelle. 2012. Step length estimation using handheld inertial sensors. *Sensors* 12, 7 (2012), 8507–8525.

C. Salter. 2010. *Entangled: Technology and the transformation of performance*. MIT Press, Cambridge, Massachusetts.

A. Sanna, F. Lamberti, G. Paravati, and F. Manuri. 2013. A Kinect-based natural interface for quadrotor control. *Entertainment Computing* 4 (2013), 179–186.

Mithileysh Sathiyanarayanan. 2016. MYO Armband for Physiotherapy Healthcare : A Case Study Using Gesture Recognition Application. *2016 8th International Conference on Communication Systems and Networks (COMSNETS)* (2016), 1–6.

Erik Scheme and Kevin Englehart. 2011. Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use. *Journal of Rehabilitation Research and Development* 48, 6 (2011), 643–660.

Jonas Schmidtler, Klaus Bengler, F. Dimeas, and A. Campeau-Lecours. 2017. A Questionnaire for the Evaluation of Physical Assistive Devices (QUEAD). In *IEEE International Conference on Systems, Man, and Cybernetics*. Banff.

R Simmons and H Knight. 2017. Keep on dancing: Effects of expressive motion mimicry. *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)* (2017), 720–727.

T. R. Song, J. H. Park, S. M. Jung, and J. W. Jeon. 2007. The Development of Interface Device for Human Robot Interaction. In *Proceedings of the International Conference on Control, Automation and Systems*. 640–643.

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research* 15, 1 (2014), 1929–1958.

David St-Onge, Ulysse Côté-Allard, and Giovanni Beltrame. 2018a. Expressive motion with dancers dataset. (2018). https://doi.org/10.21227/H29M1Q

David St-Onge, Ulysse Côté-Allard, and Giovanni Beltrame. Last visited 12/2018b. https://www.youtube.com/watch?v=KNapoKTL4wM. (Last visited 12/2018).

Gentiane Venture, Takumi Yabuki, Yuta Kinase, Alain Berthoz, and Naoko Abe. 2016. *Using Dynamics to Recognize Human Motion*. Springer International Publishing, Cham, 361–376.

Bill Vorn. 2016. I Want to BelieveâĂŤEmpathy and Catharsis in Robotic Art BT - Robots and Art: Exploring an Unlikely Symbiosis. Springer Singapore, Singapore, 365–377.

Phillip Walker and Steven Nunnally. 2012. Neglect Benevolence in Human Control of Swarms in the Presence of Latency. In *Proceedings of the International Conference on Robotics Automation*. 3009–3014.

F. Weichert, D. Bachmann, B. Rudak, and D. Fisseler. 2013. Analysis of the Accuracy and Robustness of the Leap Motion Controller. *Sensors* 13, 5 (2013), 6380–6393.

Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics bulletin* 1, 6 (1945), 80–83.

A. Xu, G. Dudek, and J. Sattar. 2008. A Natural Gesture Interface for Operating Robotic Systems. In *Proceedings of the 2008 IEEE International Conference on Robotics and Automation (ICRA '08)*. Pasadena, California, USA, 3557–3563.

B. Xu, J. Z. Huang, G. Williams, Q. Wang, and Y. Ye. 2012. Classifying Very High-Dimensional Data with Random Forests Built from Small Subspaces. *International Journal of Data Warehousing and Mining* 8, 2 (April–June 2012).