Evaluation of Simple Microphone-based Mechanomyography (MMG) Probe Sets for Hand Stiffness Classification

Submitted: 06th February 2019; accepted: 20th May 2019

Igor Zubrycki, Grzegorz Granosik

DOI: 10.14313/JAMRIS/2-2019/16

Abstract:
We describe simple to build mechanomyography sensors, with one or two channels, based on electret microphones. We evaluate their application as a source of information about the operator's hand stiffness, which can be used for changing a robot's gripper stiffness during teleoperation. We explain a data acquisition procedure for further employment of a machine-learning. Finally, we present the results of three experiments and various machine learning algorithms. Support vector classification, random forests, and neural-network architectures (fully-connected artificial neural networks, recurrent, convolutional) were compared in two experiments. In first and second, two probes were used with a single participant, with probes displaced during learning and testing to evaluate the influence of probe placement on classification. In the third experiment, a dataset was collected using two probes and seven participants. As a result of the single-probe tests, we achieved a (binary) classification accuracy of 94%. During the multi-probe tests, large cross-participant differences in classification accuracy were noted, even when normalizing per-participant.

Keywords: *MMG, acoustic myography, teleoperation, mechanomyography*

1. Introduction

In this paper, we evaluate how simple mechanomyography probes can be used for hand stiffness classification. We are particularly interested in the use of such microphone-based probes to complement our teleoperation system for robotic gripper teleoperation based on wearable sensor/haptic glove and a vision system. The teleoperation environment provides unique challenges for teleoperation devices: the need for real-time data acquisition and processing, ease of use even during stressful situations, robustness and low cost.

Robots may be teleoperated in an environment where the operator has limited perception capability and therefore has to proceed cautiously. In such situations, a person would change the stiffness of his hand to limit contact forces during unexpected collisions or when interacting with unknown objects [10]. In this paper, we present a design and results of research using a mechanomyography probe for estimating the operator's hand stiffness.

Various grippers can change own's stiffness using a designated mechanical structure [5] or through sensor-based control mechanism (impedance control) [16]. The control mechanism could automatically select such value, but in an unknown environment (such as rescue or exploratory robotics), where teleoperation is the dominating mode of the control, such value should be somehow connected to the actions of the operator.

However, the cognitive load of the operator overwhelmed by the number of controls of the teleoperation interface can negatively influence his effectiveness [13]. Instead of "classic teleoperation," immersive teleoperation (where the operator's body behavior controls various parameters of the teleoperated robot) can lessen the cognitive load and be more "intuitive." Myosignals could be, in such a scenario, used for setting the stiffness parameters of the gripper.

Myosignals have been already used to change the stiffness of multi-finger grippers, to improve their robustness to crashes [3]. Also, such signals were used to change the impedance of an industrial manipulator in teleoperation mode [4]. They were also used for prosthetics control. Such prosthetics could have one or more degrees of freedom through using one or multiple myosignal probes [19]. Myosignals could trigger predefined motions [19] or, after calibration, be used to control the movement of each joint or closure of grip [17]. Wołczowski et al. lead a multi-year effort on creating a biosignal acquisition for the control of a multifunctional hand prosthesis, based on probes capable of both MMG and EMG measurement [32].

Various types of myosignals could be used, but electromyography is the most widely used, due to a long history and very advanced probing schemes achieved. It depends on measuring very small voltages; therefore, changes in sensor placement and skin impedance can lead to large differences in amplitude, shape, and signal delay [22]. Mechanomyography, which depends on measuring mechanical vibrations of muscles is less dependant on such issues [36].

While less prevalent than electromyography, there were several applications of mechanomyography probes for medical and rehabilitation uses [8]. As there are several challenges regarding how mechanomyographic signals could be acquired, there is no one standard method for signal acquisition and analysis. A review of mechanomyography sensor development is presented in [15, 20]. Signals were read using accelerometers, piezometers or microphones, or combinations of such. Similarly, a signal analysis could be done using multiple methods; in time or frequency domains.

Our goal was to design an inexpensive but usable

setup, based on electret microphone connected to a standard PC for classifying hand stiffness based on the mechanomyographic signal.

Mechanomyography has several features that make it an interesting alternative to electromyography. Mainly, the simplicity of setup where probes do not need to be as precisely placed as in electromyography (while still position and relative sensor orientation can have an impact as described in [18]), as well as, robustness to changes in air or skin moisture or occurrence of strong radio / EM signal [9]. This is because of the nature of low-frequency audio signals which are sensed in mechanomyography [36].

The design described in this paper is strongly influenced by the results of Silva et al. from PRISM laboratory, from their long-term project on using mechanomyography for prosthetics control [29]. The team created a two-sensor (accelerometer, microphone) per probe set in which accelerometer was used to estimate and possibly eliminate the influence of motion on the sensor readouts. Motion artifacts are important factors for mechanomyography, with a more significant effect on accelerometer based MMG. Posatskiy and Chau presented such a comparative study of microphones and accelerometers and also found that for both types of sensors non-vanishing motion artifact harmonics were present [26]. Especially for telemanipulation, where the operator moves his limbs, such influence is important, and in this paper, we investigate the effect of dynamic motions on hand-stiffness classification accuracy.

The apparent ease of setup makes it convenient to use for a teleoperation system, as operators installing the probes would not need prolonged training. Also in hectic situations as the ones occurring in rescue situations, possible probe displacement should not result in misclassifications.

In Section 2 we present the design of two types mechanomyographic probes. We describe their design goals and properties as well as fabrication details. In subsection 2.2 we describe how the data acquired and processed.

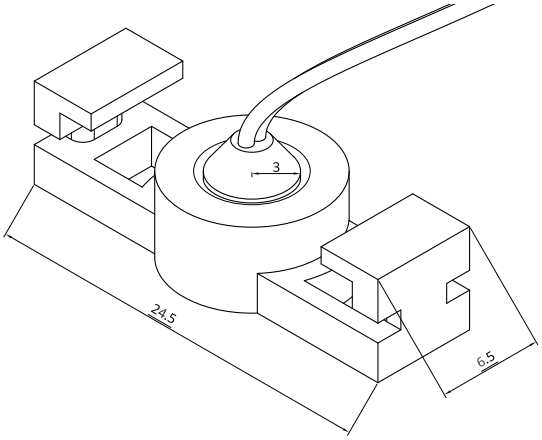
In Section 3 we evaluate the machine learning algorithms for classifying hand stiffness based on the data acquired using the probes described in Section 2. Particularly, we present details of three experiments two of which evaluated the use of plastic and silicone probes and classifier robustness to sensor placement. In third, we evaluated multi-subject stiffness classification using two-probe set.

In Section 4 we present how the classification pipeline can be integrated into a Robot Operating System network, particularly for the teleoperation using operator's arm and hand movement. We finish the paper with summary and conclusions.

2. Data Acquisition Using Mechanomyographic Probe

2.1. The Design of Mechanomyographic Probe

For the acquisition of a mechanomyographic signal (MMG) we manufactured three sets of probes, each consisting of microphones in a harness. We used a capacitive, electret microphone, which did not require phantom power. Microphones Marnsnaska DZ0289 have (declared) 2.2 kOhm impedance and sensitivity of -52 dB with bandwidth 30 Hz - 15 kHz. The lower value is declared by the manufacturer, and comes from measuring the microphone in hearing range, other works reported lower actual values [23]. Watakabe evaluated the properties of a condenser microphone (Panasonic/ Matsushita WM-034B) when attached to a cylinder with an air chamber, finding that with the correct combination of length and diameter of the cylinder, the cut-off frequency could be reduced to 2- 4 Hz, for diameters of the cylinder over 10 mm [31].



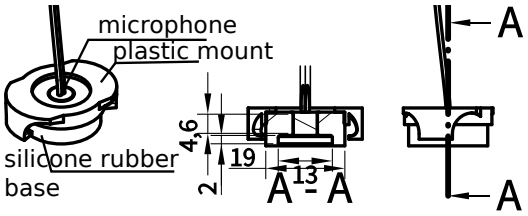
(a) A draft design of mechanomyographic probe

(b) Photograph of mechanomyographic probe

Fig. 1. Mechanomyographic probe

Due to use of electret microphone, it could be directly connected to a PC's audio card (in our case, built in motherboard audio card, Asus Z97-K). To maximize the simplicity of the setup we did not use additional amplifiers or preamplifiers. A dedicated Python application with GUI recorded the audio signal (see Fig. 3).

The first version of the mechanomyographic probe uses 3D FDM printed harness (Fig. 1a) (PLA material), in the second one the harness is made from rubber silicone (DRagonSkin, shore 30A), with an additional plastic harness for mounting the rubber part. The mold for rubber silicone was also made using the



(a) Silicone-rubber based mechanomyographic probe

(b) Photograph of silicone-based mechanomyographic probe

Fig. 2. Silicone-rubber based mechanomyographic probe

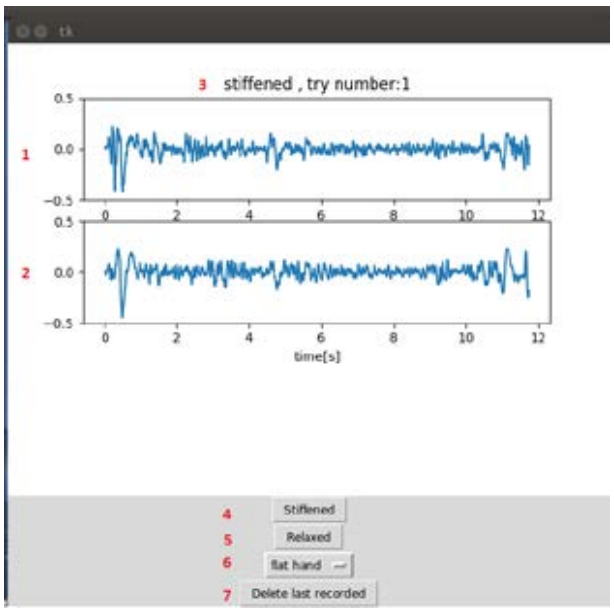


Fig. 3. A graphical user interface for recording mechanomyography data

FDM process, and its design is available on the project's website.

We designed the probe to have a cylinder shaped air column between the microphone membrane and the skin. In the case of the plastic probe, the air column is closed by the seal formed by the skin while in case of the silicone-rubber probe, we closed the air column with additional silicone membrane on the bottom of the device. Such a closed air column is needed to register low frequencies of the MMG signal and to enable free movement of the microphone's membrane.

The air column has a diameter of 13 mm and a height of 2 mm, which is a combination with good signal-to-noise ratio [28]. The motivation behind designing two probes was to evaluate whether simplistic, easy to make plastic probe could deliver adequate results. The main issue with the plastic probe is the lack of dampening as because of the relative softness of the silicone forms a low pass filter, attenuating high frequencies while the closed air column provides near-optimal signal amplification parameters [27].

The probes do not include an additional accelerometer as in [27]. Authors motivated the addition of such a device to filter the influence of the limb movement on the signal but did not provide an algorithm to do so. Such setup would require an additional real-time board which could record two streams of data in parallel which would greatly complicate the setup.

Both probes are mounted using elastic harnesses and placed over a selected muscle group. No further preparation is needed.

Additionally, we designed a simple two-probe set, with two probes of the second version attached to the cloth, 13 cm apart (Fig. 4). Using the PLA material to print the harness, the material can be easily melted to weld to the cloth (linen). A rubber harness was attached between the two probes. The two-probe harness was designed to enable easy setup for the operator, as he/she only needs to slide the harness onto the forearm. However, due to a constant distance between the probes, there is the risk that the probes cannot be placed accurately over a particular muscle group. Cooperation of this setup with multiple users is investigated in one of the experiments.

Even with two probes, they can still be connected to a (stereo) audio input either by soldering the GND lines together and soldering the positive rails to selected parts of a stereo audio jack or by using a Y-type microphone adapter that combines the two microphone inputs into two channels. Even with this setup, no additional circuitry is used.

2.2. Data Acquisition and Preparation

The audio signal received by the probe's microphone is analyzed online using a dedicated Python program with PyAudio, pyWavelets and Scipy libraries (program available on the project repository [35]).

In standard setting the audio data is acquired with 16 kHz frequency (16384 samples per second). We use either Discrete Fourier Transform or Wavelet Packet Decomposition to form feature vectors for classification. As the interesting signal is in the range between 5 - 50 Hz [20] and to reduce the computational cost we can downsample the signal. The signal is twice downsampled using the *decimate* SciPy function [2] which consists of low-pass filtering with FIR filter with Hamming window, and decimating (i.e., keeping every 8th sample) to selected frequency (256 samples per second).

On computer with i7-4790 processor, the computation time for decimating 1s one-channel recording

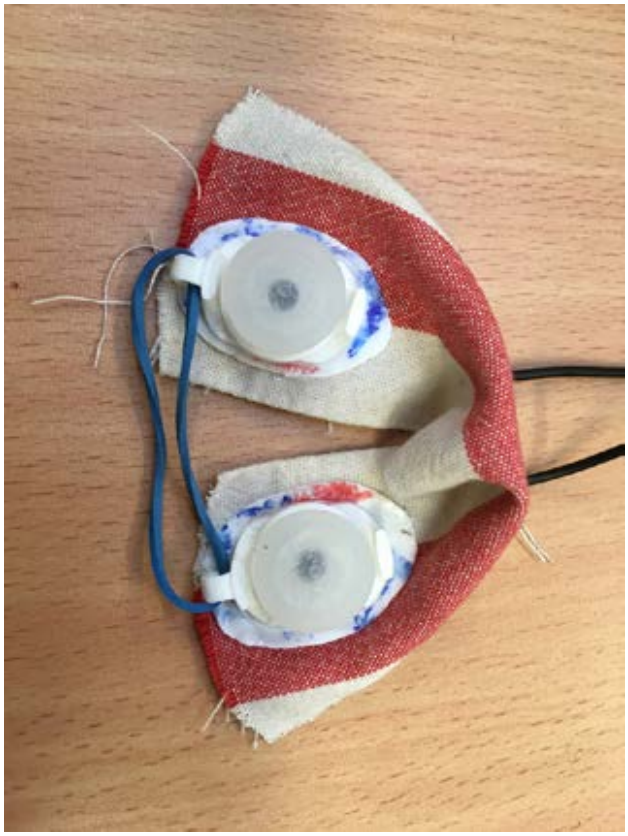
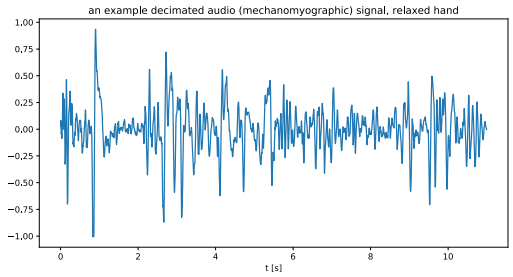
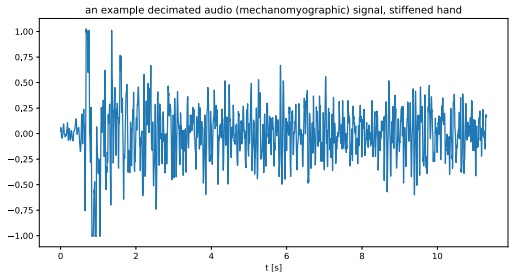


Fig. 4. A two-probe set using the silicone probes



(a) An example of mechanomyographic signal for a relaxed hand



(b) An example of mechanomyographic signal for stiffened hand

Fig. 5. Examples of mechanomyographic signal

registered with 16kHz frequency is around $750\mu s$. In case of using magnitudes of Fourier coefficients calculating the coefficients for the original signal is faster than downsampling and calculating the coefficients from the downsampled signal ($157\mu s \pm 1\mu s$ vs $798\mu s \pm 4$).

However, in case of using features from Wavelet Packet Decomposition, without downsampling a deeper level of decomposition would be needed (12th instead of 6th) to acquire desired resolution of frequency sampling. For a 1s recording, decimation and wavelet packet decomposition on 6th level takes around $1.8ms \pm 5.35\mu s$, compared to 12th level wavelet packet decomposition $76.3ms \pm 811\mu s$.

An example of downsampled (decimated) signal for a relaxed and stiffened hand is shown in Fig. 5a and Fig. 5b.

For the experiments described below, a signal window consists of varying lengths of signals, acquired from 0.25 to 1s.

3. Evaluation of Machine Learning Algorithms for Classifying Hand Stiffness

The assumed use of the system is for teleoperation. What is important in practice, during the interaction with the teleoperation system, an operator will typically grasp virtual (non-existing) objects with some light feedback provided by a haptic feedback subsystem. Therefore, we can assume that the operator never produces a significant force in the arm that would not be happening in the same time as a contra force – there will not be a significant force output on the environment by the operator's hands. In that case, a single mechanomyographic probe may be enough, while in case of a hand coming into contact with the environment the signals from pairs of muscle groups would need to be registered [14]. However, the operator will move the hand during the teleoperation, so the system needs to recognize the stiffness even when the hand is moving.

There are several issues in classifying hand stiffness using mechanomyography, all being the result of multiple and non-stationary signal sources registered in the audio signal of the probe. There is stiffness information in the myographic signal [14] but it also has information about the motion of the limb, which is difficult to decouple [27]. Also, the mechanomyographic signal is nonstationary – stiffening the hand results in a momentary peak in signal (see a peak at 1s at Fig. 5b), followed by a signal with slightly different (compared to relaxed hand) amplitude and frequency. Muscle fatigue also influences the vibrations of muscles.

As in this case, the mechanomyographic signal is de facto an audio signal, background noises (from, for example, movement of probe cables, operators clothes) are also registered. Conversely, because the signal is registered by a microphone, motion artifacts have less influence on the recording that in case of using an accelerometer.

Also, although less than in the electromyographic signal, the positioning of the probe could influence the registered signal. We assume that the probe is mounted by the operator, who could not be a specialist in good signal acquisition or anatomy. Therefore, we cannot guarantee that the probe will always be put in the

same place only that it will be in some area around the particular muscle group.

All previously described issues make a simple signal filtering to acquire the stiffness classification infeasible, and most previous approaches are data-driven, machine learning based, where new data are classified using a model that was created using previously acquired and labeled data (supervised learning). Instead of calibrating the sensor, based on the data, a machine learning algorithm captures the pattern in the data facilitating classification.

3.1. Acquiring Data for Supervised Learning

Labeled data for supervised learning was acquired during two experiments using the previously described mechanomyographic probes. In both experiments, the probe was placed in 7 different parts of the arm, on an area of around 35 mm in diameter, on flexor carpi radialis and flexor carpi ulnaris muscles. The exact placement of the probes is illustrated in Fig.6. In both experiments, the operator was asked to move the hand using 20 gestures, typical for normal hand movement, a list of which is presented in Tab. 1. The choice of gestures was made such that there were both static poses and dynamic actions – possibly making it more difficult to classify the stiffness due to the motion artifacts. We did not try to classify the gestures, but in the third experiment, they were recorded to enable better interpretation of the results.

Actions were done twice, once having a relaxed hand, the second time having it stiffened. Five of the positions were used in training (1,3,4,5,6 see Fig.6) while points 2 and 7 were used in tests.

In all three experiments, the goal was to classify the mechanomyographical signal for a binary label – whether the hand was stiff or relaxed. Participants were instructed to stiffen their hand so that the fingers would not (significantly) move when there would be an external force and to have near none resistance to force (just to compensate gravity) when having the hand relaxed. We did not measure the actual mechanical stiffness of the hand.

In the first experiment, a plastic probe was used, and data were acquired with an effective acquisition frequency of 1024 samples per second and a frame length of 0.5s. For each vector of samples, magnitudes of discrete Fourier coefficients were calculated and registered (a vector of length 512). The aim of the Experiment was assessing the feasibility to use a simple sensor with computationally fast data process (even on microcontrollers or DSP's) to acquire a usable signal.

In the second experiment, data were collected using the silicone-rubber probe, with an audio signal being registered with a frequency of 16 kHz and labeled with the frequency of 2 Hz (i.e., every half-second). A single audio recording was registered for one action.

Each hand gesture motion recording, for each person, was split into 0.5 s labeled recordings. In the first experiment, as the data was labeled every 500 ms, 3220 half-second training and 689 (around 20%) testing label-recording pairs were acquired, which adds up to about 30 minutes of the registered signal.

In the second experiment, 6944 half-second training and 2962 (about 27%) testing label-recording pairs were acquired, adding up to around 80 minutes of a signal.

Tab. 1. Hand motions used in data acquisition for mechanomyography experiments

number	activity
1	flat hand
2	flat hand rotated 90 degrees
3	closed fist
4	closed fist rotated 90 degrees
5	spherical grasp
6	3 finger grip
7	wedge grip
8	pointing
9	4 straightened fingers
10	2 straightened (freedom gesture)
11	a cylindrical grasp
12	radial palmar grasp
13	finger wave motion
14	tapping fingers
15	a "scissor" gesture
16	flexing hand in wrist
17	a hand laying on table
18	grabbing a box (transverse volar grip)
19	grabbing a bottle (cylindrical palmar prehension)
20	using a keyboard

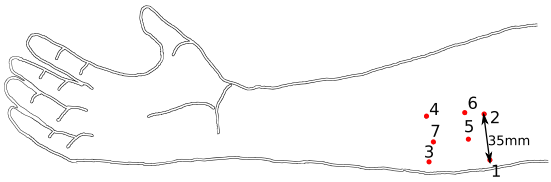


Fig. 6. Sensor placement during the experiments

3.2. Experiment 1

For the first experiment, three classes of algorithms were investigated. These were: Support Vector Machines (SVM) with Linear Kernel, Extreme Random Trees and Neural Networks.

For SVM and Extreme Random Trees, we used Scikit-Learn implementations and Keras for Neural Networks [1, 25]. For Fully-Connected Neural Network we used a three-layer Neural Network, with one hidden layer, batch normalization, and dropout during training, see Fig. 9. In all cases, a backpropagation learning algorithm was used. Similar algorithms were used in other papers for electro or mechanomyography analysis [6].

Feature vector was constructed from magnitudes of Fourier coefficients, corresponding to frequency band 0-50 Hz. Also, a recursive feature elimination (RFE) method was used to select a subset of features that had the highest importance. The method trains the classifier using an initial set of features, computes the *feature importance* (a ranking classifier) for the

classifier and removes the feature with the smallest ranking criterion. The procedure is repeated until a desired (reduced) number of features is achieved [12].

For each classification method and each type of feature vector a hyperparameter optimization was used, as all selected algorithms are highly dependent on selected hyperparameters (C parameter for SVM with linear kernel, number of estimators, min sample split, a maximum number of features for extreme random trees, number of neurons on a layer and regularization parameters for a neural network).

A grid search with cross-validation (5-split) was used for optimization. For each algorithm and each hyperparameter, we created a list of values to evaluate and then we took the Cartesian product of these possible values. Then, for each hyperparameter combination, the machine learning algorithm was taught on a 4/5ths of the test data, while tested on the remaining 1/5th (of the test set), repeating five times on a different split. Subsequently, a mean precision value, taken from evaluating the taught algorithms on particular test sets was used as a return value for hyperparameter combination.

A model with the best parameters was then tested on the test set.

3.3. Experiment 2

In the second experiment, in addition to the same features as in Experiment 1, that is constructed from magnitudes of FFT coefficients and eliminated using RFE feature selection, we evaluated other signal transforms.

Namely, we used spectral analysis using Wavelet Packet decomposition, and we directly used the decimated signal in the time domain. Spectral analysis and direct method were used motivated by the previously researched nonstationarity of the signal [7].

For analysis using Wavelet Packets, we used the methodology described in [30], for analysis of EEG signal. We used the same parameters, that is Daubechies wavelet 4 (db4) wavelet family and 6th level of wavelet packet decomposition. For calculations, PyWavelets library for Python language was used [21].

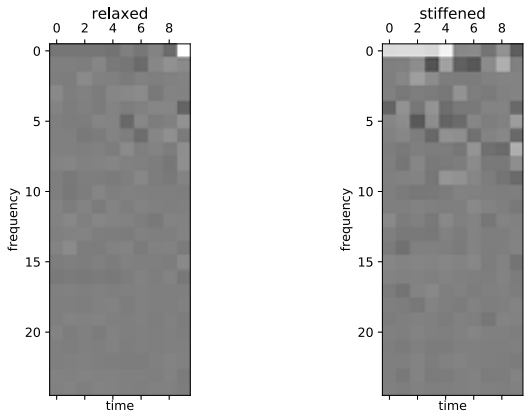
Wavelet packet decomposition, for a signal of length 256 and 6th level of decomposition, produces 64 vectors of length 7. Sorted by frequency, the first 25 of those represents frequencies between 0-50 Hz.

Several combinations of features were investigated, each set creating a feature vector. Same types of feature vectors were selected for a shorter time window 0.25s.

- feature vector constructed from a concatenation of 25 first (frequency sorted) vectors from the 6th level of Wavelet Packet decomposition (length 25*7) – *all decomposition coefficients*
- feature vector constructed from a concatenation of a vector where each element is a mean of elements of one of the vectors of Wavelet Packet decomposition – *average coefficients* from [30], with a vector where each element is a norm of one of the vectors of Wavelet Packet decomposition, first 25 vectors were used (length 50) – *subband energy vector* [30],
- subset of features constructed from concatenation of *subband energy vector* and *average coefficients* and reduced by feature elimination using Recursive feature elimination (RFE). The scikit-learn implementation of the RFE algorithm was used [25].

Additionally, other architectures of neural networks were used, namely convolutional (see Fig. 10b), and recurrent (see Fig.10c) neural networks, which required different input data formats:

- in case of convolutional neural networks, the input was a matrix (64x7) where each row was one of the vectors of Wavelet Packet decomposition, forming an "image" of the spectrogram (see Fig.7) – *matrix of decomposition coefficients*. This approach is similar to the representation used by Wołczowski and Zdunek in [33] – each measurement can be represented as a 3D tensor $y \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, where I_1 is the scale dimensionality, I_2 is time shift dimensionality, I_3 is the number of sensors (1 or 2).
- for a recurrent network, a raw decimated time series vector was used, of length 256.



(a) Relaxed hand **(b)** Stiffened hand

Fig. 7. Image of matrix made from vectors of Wavelet Packet Decomposition. Corresponding time series are presented in Fig. 5

Similarly to the first experiment, for each method and feature set combination, a hyperparameter optimization based on cross-validation (5-fold) was used. The model with the best set of hyperparameters was evaluated on the test set. Also for all algorithms, the speed of execution was measured. This was done using a categorization of a random vector by a selected method, using *timeit* Python function, which repeated execution of this function several times to estimate mean execution time and standard deviation.

3.4. Multi-subject Stiffness Classification Using Two-probe Set

In this experiment, subjects were equipped with two probes using the two-probe set (see Fig. 4). Seven healthy subjects, 4 male, 3 female, 5 right- and 2 left-handed, age 22-33, performed an experimental proto-

Fig. 8. A two-probe set placement for the multi-participant experiment

col. For all subjects, the right hand was used, and there were no changes for the setup of the probe. The whole procedure lasted between 10 and 30 minutes per participant. All subjects participated voluntarily and gave written consent to the procedures. Experimenters placed the probe over the flexor digitorum superficialis and extensor digitorum (if not possible due to the differences in subject forearm diameter the probes were placed so they would be on two sides of brachioradialis muscle), see Fig 8. All subjects repeated the same set of actions as in previous experiments (see Tab. 1), which were recorded. Overall, 664 gesture recordings were collected, consisting of 7445 seconds of data.

Data were processed through the same pipeline as in previous experiments, separate for both channels (i.e., low-pass filtering and decimation and 6th level wavelet pocket features).

Two machine learning architectures – convolutional neural networks and extra trees classifier – were used. For each participant, five conditions were evaluated. In first, learning was done using single-channel recordings of the other participants and test was done on this participant. In the second, two channels were used. In third, also two channels were used, but for each participant, data was scaled used ScikitLearn robust scaler: i.e., the median was removed, and data was scaled to interquartile (1st quartile and 3rd quartile) range per each channel. In forth condition, only same-subject recordings were used, using 5-fold data splitting, i.e., data was split 5 times into test-train sets, and the result is the mean precision value. In the fifth condition, only same-subject recordings were used (with 5-fold data splitting), but only one mechanomyography channel (sensor on the anterior side of the arm).

3.5. Results and Discussion

Results of the Experiment one (plastic probe) and comparison to results for the same features but for silicone probe (from Experiment two) are presented in table 2. Rest of results for experiment two (single silicone probe, features constructed from Wavelet Packet Decomposition or raw time vector) are presented in table 3. Results of the third experiment (multi-participant experiment) are presented in Fig. 11.

The Table 2 shows that nearly every method gave significantly better results for a silicone-probe than a plastic one. Still, an 89% accuracy of classification can be achieved with a simple plastic probe, using Support Vector Classification (with Linear Kernel) with a feature vector being all magnitudes of FFT coefficients. This feature set is also the best for silicone probe, with 91% accuracy. What is important, this method is much faster than Extra Trees Classifier and Neural Network, with time for single classification being around 30 us for SVM and around 2 ms for Extra Trees Classifier and Neural Network. The preparation of the signal, that is calculation of magnitudes of Fourier coefficients takes approximately $916 \pm 5ns$ (for sampling frequency of 1 kHz).

We also assume that 94% result of classification for Extra Trees Classifier is probably erroneously good, as mean result on cross-validated data was only 74%.

The Table 3, presents that overall best result is achieved where a convolutional neural network is used on a 1s time frame, with each wavelet packet decomposition vector constructing one row of the input matrix, achieving a classification result of 94.5%. Using a longer time frame gives better results, but for a 0.25s time frame, 90% accuracy is still possible. Interestingly, using a raw timeframe and recurrent neural network would result in the classification with the accuracy of 89% but with the significantly longer calculation times – 18 ms compared with classification times of less than 2ms and transformation times (for signal decimation and wavelet packet decomposition) of 4 ms.

In the third experiment models based on two-probe input had a better result (i.e., two-channel cross-participant results were better than one channel cross-participant results, and two-channel same-subject results were better than one channel same-subject results) but the multi-person experiment with the two-probe set has also shown significant differences in results between participants. Best results were achieved when models were trained using same-subject data even when using only one of the channels. This is similar to the results of other groups. Particularly, Youn and Kim presented that for estimating elbow flexion forces from MMG signal same-subject validation tests were significantly better than those of the cross-subject validation test [34]. Per-participant data normalization did not significantly improve the cross-participant results.

The comparably more reduced performance in all the cases can be explained by the fact that the current design of the two probe harness does not allow accurate placement of two probes depending on the forearm size and that the participants were not trained. Also, there were fewer recordings per-participant than in the previous experiments, which can explain a worse performance of neural-network-based algorithms, which tend to require large amounts of data and over-learn otherwise.

4. Integration With a ROS-Based System

The mechanomyography-based stiffness classifi-

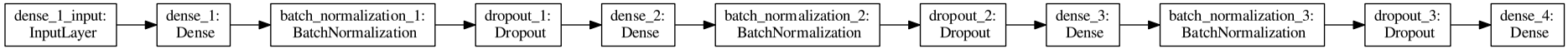
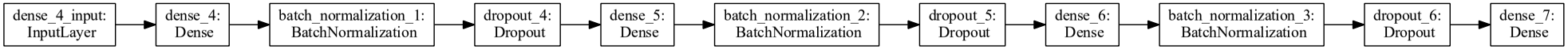
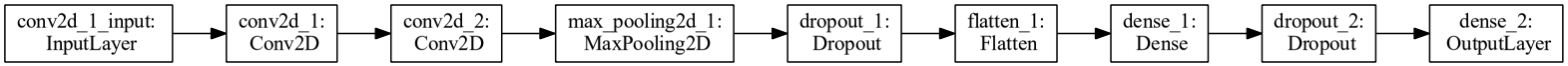


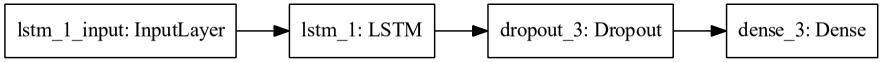
Fig. 9. Fully-Connected Artificial Neural Network architecture used in Experiment 1



(a) Fully-Connected Artificial Neural Network



(b) Convolutional Neural Network



(c) Recursive Neural Network (with Long-Short Memory unit [11])

Fig. 10. Artificial Neural Network architectures used in the Mechanomyography Experiment 2

cation device can be integrated with a robotic system using the ROS (Robot Operating System). The program does online data processing and classification, results of which are available as ROS topics.

ROS node publishes the recognized category, the estimated probability of classification (if applicable: in SVM with Linear Kernel and Neural Network) and a decomposed (wavelet packet, FFT) or raw signal. In an actual system, the short 0.25s time frame is used.

The proposed integration as a part of a teleoperation system is illustrated in Fig. 12. Operator's hand pose and position are tracked by a vision system and a sensor glove. From this data, the desired gripper pose is calculated. Simultaneously, machanomygraphic probe, placed on the operator's arm, measures the muscle signals. A data processing pipeline, described in previous sections, is used to classify the stiffness. The stiffness class is then used by the gripper controller, working in a stiffness control mode, to switch the stiffness to either hard or soft. After calibration, we plan to use the probability of the label to further adjust the stiffness of the gripper.

Also, other ROS nodes can subscribe to stiffness, probability of the label, as well as, raw data or processed data topics.

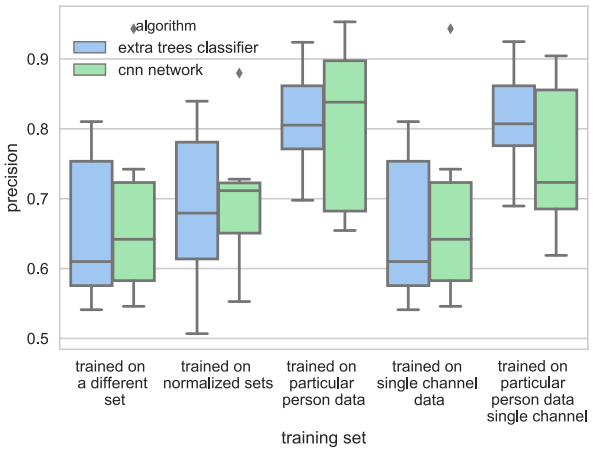


Fig. 11. Comparison of precision results for multiple participants while training on different datasets

5. Conclusions

In this paper, we presented a concept of a simple and inexpensive device to measure mechanomyographic signal for estimation of hand's stiffness using machine learning methods. The device with the best classification method was robust to probe displacement, showing an over 90% accuracy of classification with 4 Hz speed and for a 15 mm displacement of the probe. The system is capable of achieving 94% accuracy if the algorithms could run on a graphical card and 1 Hz frequency of classification and when calibrated for a particular person. Results achieved are comparable to those of other authors with 90% achieved for mechanomyography (1s window of acquisition) [15] and 95% for electromyography [24].

The prototyping of the device can be extremely low-cost, with a material cost of around 10 PLN (less than 3 USD) for plastic probes and 20 PLN (5 USD) for silicone probes, and requires only a simple FDM 3D printer.

A described device is limited by its use of only one or two channels, but the signal acquisition is very simple and can run on a standard PC with a built-in audio card. We put the software code and CAD files for the devices in open source repository [35]. The device software is prepared as a ROS Package to enable re-use and integration with ROS-based systems.

As further steps, we plan to use the mechanomyographic signals to enable regression of stiffness, similarly as Hoppner et al. did for electromyographic signal [14]. Also, we plan to improve the two-probe setup to enable better fit a particular muscle-group pair. This, however, needs to be followed by an easy-to-use protocol for teleoperators so they can fix the sensor themselves.

ACKNOWLEDGEMENTS

Authors would like to thank Malgorzata Pruszkiewicz and Hubert Kowalczyk for their help in data collection.

AUTHORS

Igor Zubrycki* – Lodz University of Technology, Stefanowskiego 18/22, 90-924 Lodz, e-mail: igor.zubrycki@p.lodz.pl, www: www.robotyka.p.lodz.pl.

Grzegorz Granosik – Lodz University of Technology, Stefanowskiego 18/22, 90-924 Lodz, e-mail: granosik@p.lodz.pl, www: www.robotyka.p.lodz.pl.

*Corresponding author

REFERENCES

[1] "Home – keras documentation". https://keras.io, 2015.

[2] "scipy.signal.decimate — scipy v1.1.0 reference guide". https://docs.scipy.org/doc/scipy-1.1.0/reference/generated/scipy.signal.decimate.html, 2019.

[3] A. Ajoudani, S. B. Godfrey, M. Bianchi, M. G. Catalano, G. Grioli, N. Tsagarakis, and A. Bicchi, "Exploring Teleimpedance and Tactile Feedback for Intuitive Control of the Pisa/IIT SoftHand", *IEEE Transactions on Haptics*, vol. 7, no. 2, 2014, 203–215, 10.1109/TOH.2014.2309142.

[4] A. Ajoudani, N. G. Tsagarakis, and A. Bicchi, "Tele-Impedance: Preliminary results on measuring and replicating human arm impedance in tele operated robots". In: *2011 IEEE International Conference on Robotics and Biomimetics*, 2011, 216–222, 10.1109/ROBIO.2011.6181288.

[5] L. A. Al Abeach, S. Nefti-Meziani, and S. Davis, "Design of a Variable Stiffness Soft Dexterous Gripper", *Soft Robotics*, vol. 4, no. 3, 2017, 274–284, 10.1089/soro.2016.0044.

Tab. 2. Classification results for Experiment 1. Results for probe one (plastic probe), in bracket results for the second probe (silicone-rubber based)

Classifier type	data source	hyperparameters	accuracy	ex. time
SVM with Linear Kernel	RFE selected features from magnitudes of FFT coefficients	C=1 (C=0.1)	0.77 (0.88)	31.3 $\pm 3.09us$
	all (0-50 Hz) magnitudes of FFT coefficients	C=10 (C=0.1)	0.8996 (0.91)	30.2 $\pm 2.02us$
Extra Trees Classifier	RFE selected features from magnitudes of FFT coefficients	n of est 40, min sample split 5, max feat 5 (n of estimators 40, min samples split 5, max features 6)	0.81 (0.88)	2.29 ms $\pm 8.06us$
	all (0-50 Hz) magnitudes of FFT coefficients	n of est 40, min sample split 5, max feat 16 (n of est 40, min sample split 2, max feat 16)	0.94 *0.74* (0.88)	2.29 ms $\pm 11.6us$
fully-connected Artifitial Neural Network	RFE selected features	3 layers, 10 neurons per layer	0.69 (0.81)	2.11 ms $\pm 13.2us$
	all (0-50 Hz)magnitudes of FFT coefficients		0.73 (0.85)	2.14 ms $\pm 37.7us$

Tab. 3. Results of classification for Experiment 2. Results for 1s timeframe, and in bracket results for a time frame of 0.25s

Classifier type	data source	hyperparameters	accuracy	ex. time
SVM with Linear Kernel	RFE selected features	C=10 (C=10)	0.901 (0.877)	33.2 $\pm 1us$
	average of coefficients +subband energy vector	C=10 (C=1)	0.8996 (0.86)	29.2 $\pm 2.65us$
	all decomposition coefficients	C=10 (C=100)	0.56 (0.59)	27.2 $\pm 2.0us$
Extra Trees Classifier	RFE selected features	n est 40, min s split 5, max feat 5 (n est 40, max feat 5, min s split 10)	0.901 (0.88)	2.2 ms $\pm 0.03ms$
	average coefficients +subband energy vector	n est 40, min s split 5, max feat 7 (n of est: 40, max feat 7, min s split 10)	0.904 (0.88)	2.17 $\pm 0.02ms$
	all decomposition coefficients	n est 40, min s split:5, max feat 12 (n est 40, max feat: 12, min s split 5)	0.88 (0.85)	2.23 $\pm 0.01ms$
fully-connected ANN	RFE selected features		0.91 (0.89)	1.87 $\pm 0.3ms$
	average of coefficients +subband energy wector		0.914 (0.89)	1.9 $\pm 0.3ms$
	all decomposition coefficients	see Fig. 10b	0.88 (0.88)	1.81 $\pm 0.1ms$
Convolutional NN	matrix of decomposition coefficients		0.945 (0.90)	1.54 $\pm 0.2ms$
Recurrent NN	downsampled (256/s) time signal	see Fig.10c	0.892 (0.88)	18.2 $\pm 3.9ms$

[6] A. H. Al-Timemy, G. Bugmann, J. Escudero, and N. Outram, "Classification of Finger Movements for the Dexterous Hand Prosthesis Control With Surface Electromyography", *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 3, 2013, 608–618, 10.1109/JBHI.2013.2249590.

[7] T. W. Beck, J. M. DeFreitas, M. S. Stock, and M. A. Dillon, "An examination of mechanomyographic signal stationarity during concentric isokinetic, eccentric isokinetic and isometric muscle actions", *Physiological Measurement*, vol. 31, no. 3, 2010, 339–361, 10.1088/0967-3334/31/3/005.

[8] T. W. Beck, T. J. Housh, J. T. Cramer, J. P. Weir, G. O.

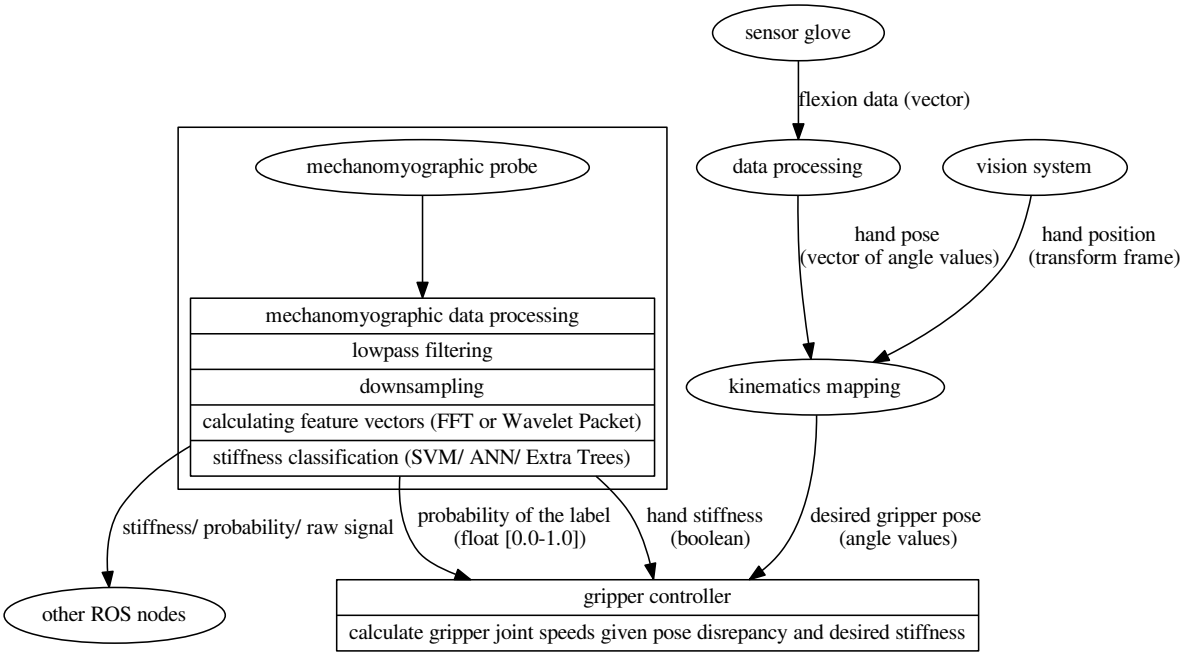


Fig. 12. Proposed integration of stiffness estimation node in the teleoperation system

Johnson, J. W. Coburn, M. H. Malek, and M. Mielke, "Mechanomyographic amplitude and frequency responses during dynamic muscle actions: a comprehensive review", *BioMedical Engineering OnLine*, vol. 4, no. 1, 2005, 67, 10.1186/1475-925X-4-67.

[9] S. Day, "Important Factors in surface EMG measurement", *Bortec Biomedical Ltd publishers*, 2002, 1–17.

[10] D. W. Franklin, G. Liaw, T. E. Milner, R. Osu, E. Burdet, and M. Kawato, "Endpoint Stiffness of the Arm Is Directionally Tuned to Instability in the Environment", *Journal of Neuroscience*, vol. 27, no. 29, 2007, 7705–7716, 10.1523/JNEUROSCI.0968-07.2007.

[11] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to Forget: Continual Prediction with LSTM", *Neural Computation*, vol. 12, no. 10, 2000, 2451–2471, 10.1162/089976600300015015.

[12] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene Selection for Cancer Classification using Support Vector Machines", *Machine Learning*, vol. 46, no. 1, 2002, 389–422, 10.1023/A:1012487302797.

[13] S. G. Hart and L. E. Staveland. "Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research". In: P. A. Hancock and N. Meshkati, eds., *Advances in Psychology*, volume 52 of *Human Mental Workload*, 139–183. North-Holland, January 1988.

[14] H. Höppner, M. Große-Dunker, G. Stillfried, J. Bayer, and P. van der Smagt, "Key Insights into Hand Biomechanics: Human Grip Stiffness Can Be Decoupled from Force by Cocontraction and Predicted from Electromyography", *Frontiers in Neurorobotics*, vol. 11, 2017, 17, 10.3389/fnbot.2017.00017.

[15] M. A. Islam, K. Sundaraj, R. B. Ahmad, N. U. Ahamed, and M. A. Ali, "Mechanomyography Sensor Development, Related Signal Processing, and Applications: A Systematic Review", *IEEE Sensors Journal*, vol. 13, no. 7, 2013, 2499–2516, 10.1109/JSEN.2013.2255982.

[16] E. Jezierski, "Low cost impedance controller for robotic gripper drive with DC motor". In: *2015 20th International Conference on Methods and Models in Automation and Robotics (MMAR)*, 2015, 806–811, 10.1109/MMAR.2015.7283979.

[17] N. Jiang, K. B. Englehart, and P. A. Parker, "Extracting Simultaneous and Proportional Neural Control Information for Multiple-DOF Prostheses From the Surface Electromyographic Signal", *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 4, 2009, 1070–1080, 10.1109/TBME.2008.2007967.

[18] P. Kaczmarek, T. Mańkowski, and J. Tomczyński, "Towards sensor position-invariant hand gesture recognition using a mechanomyographic interface". In: *2017 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, 2017, 53–58, 10.23919/SPA.2017.8166837.

[19] K. Kiguchi and Y. Hayashi, "An EMG-Based Control for an Upper-Limb Power-Assist

Exoskeleton Robot", *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 4, 2012, 1064–1071, 10.1109/TSMCB.2012.2185843.

[20] E. Krueger, E. M. Scheeren, G. N. Nogueira-Neto, V. L. da Silveira Nantes Button, and P. Nohama, "Advances and perspectives of mechanomyography", *Revista Brasileira de Engenharia Biomédica*, vol. 30, no. 4, 2014, 384–401, 10.1590/1517-3151.0541.

[21] G. Lee, F. Wasilewski, R. Gommers, K. Wohlfahrt, A. O'Leary, and H. Nahrstaedt. "Pywavelets - wavelet transforms in python", 2006.

[22] R. Lopez and T. C. Davies, "The effect of surface electromyography placement on muscle activation amplitudes and timing". In: *2016 IEEE EMBS International Student Conference (ISC)*, 2016, 1–4, 10.1109/EMBSISC.2016.7508618.

[23] M. Ma. *MMG sensor for muscle activity detection : low cost design, implementation and experimentation : a thesis presented in fulfilment of the requirements for the degree of Masters of Engineering in Mechatronics, Massey University, Auckland, New Zealand*. Thesis, Massey University, 2010.

[24] M. A. Oskoei and H. Hu, "Evaluation of Support Vector Machines in Upper Limb Motion Classification Using Myoelectric Signal". In: *14th International Conference on Biomedical Engineering: ICBME 2008*, 2008.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in python", *Journal of Machine Learning Research*, vol. 12, 2011, 2825–2830.

[26] A. O. Posatskiy and T. Chau, "The effects of motion artifact on mechanomyography: A comparative study of microphones and accelerometers", *Journal of Electromyography and Kinesiology*, vol. 22, no. 2, 2012, 320–324, 10.1016/j.jelekin.2011.09.004.

[27] J. Silva and T. Chau, "Coupled microphone-accelerometer sensor pair for dynamic noise reduction in MMG signal recording", *Electronics Letters*, vol. 39, no. 21, 2003, 1496, 10.1049/el:20031003.

[28] J. Silva, T. Chau, S. Naumann, W. Helm, and A. A. Goldenberg, "Optimization of the signal-to-noise ratio of silicon-embedded microphones for mechanomyography". In: *CCECE 2003 - Canadian Conference on Electrical and Computer Engineering. Toward a Caring and Humane Technology*, vol. 3, 2003, 1493–1496, 10.1109/CCECE.2003.1226187.

[29] J. Silva, W. Heim, and T. Chau, "A Self-Contained, Mechanomyography-Driven Externally Powered Prosthesis", *Archives of Physical Medicine and Rehabilitation*, vol. 86, no. 10, 2005, 2066–2070, 10.1016/j.apmr.2005.03.034.

[30] W. Ting, Y. Guo-zheng, Y. Bang-hua, and S. Hong, "EEG feature extraction based on wavelet packet decomposition for brain computer interface", *Measurement*, vol. 41, no. 6, 2008, 618–625, 10.1016/j.measurement.2007.07.007.

[31] M. Watakabe, K. Mita, K. Akataki, and Y. Itoh, "Mechanical behaviour of condenser microphone in mechanomyography", *Medical and Biological Engineering and Computing*, vol. 39, no. 2, 2001, 195–201, 10.1007/BF02344804.

[32] A. Wołczowski, M. Błędowski, and J. Witkowski, "The System for EMG and MMG Signals Recording for the Bioprosthetic Hand Control", *Journal of Automation, Mobile Robotics and Intelligent Systems*, vol. 11, no. 3, 2017, 22–29, 10.14313/JAMRIS_3-2017/25.

[33] A. Wołczowski and R. Zdunek, "Electromyography and mechanomyography signal recognition: Experimental analysis using multi-way array decomposition methods", *Biocybernetics and Biomedical Engineering*, vol. 37, no. 1, 2017, 103–113, 10.1016/j.bbe.2016.09.004.

[34] W. Youn and J. Kim, "Feasibility of using an artificial neural network model to estimate the elbow flexion force from mechanomyography", *Journal of Neuroscience Methods*, vol. 194, no. 2, 2011, 386–393, 10.1016/j.jneumeth.2010.11.003.

[35] I. Zubrycki. "A ROS node for aquisition, learning and sharing mechanomyografy data from audio signal: AdoHaha/mechanomiography_node", April 2018.

[36] J. M. Zuniga, T. J. Housh, C. L. Camic, C. Russell Hendrix, H. C. Bergstrom, R. J. Schmidt, and G. O. Johnson, "The effects of skinfold thicknesses and innervation zone on the mechanomyographic signal during cycle ergometry", *Journal of Electromyography and Kinesiology*, vol. 21, no. 5, 2011, 789–794, 10.1016/j.jelekin.2011.05.009.