ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

DESARRROLLO DE UN MODELO DE RECONOCIMIENTO DE GESTOS DE LA MANO EN TIEMPO REAL UTILIZANDO QNN Y SEÑALES ELECTROMIOGRÁFICAS

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

DANNY SEBASTIÁN DÍAZ PADILLA

danny.diaz@epn.edu.ec

DIRECTOR: MARCO E. BENALCÁZAR, PHD.

marco.benalcazar@epn.edu.ec

CO-DIRECTOR: LORENA I. BARONA, PHD.

lorena.barona@epn.edu.ec

Quito, Junio 2022

CERTIFICATION

I certify that the present work was made by Danny Sebastián Díaz Padilla, under my supervision.

Firmado electrónicamente por:
MARCO ENRIQUE BENALCAZAR PALACIOS

LORENA ISABEL BARONA LOPEZ

Firmado digitalmente por LORENA ISABEL BARONA LOPEZ
Nombre de reconocimiento (DN): cn=LORENA ISABEL BARONA LOPEZ, serialNumber=100821105824, ou=ENTIDAD DE CERTIFICACION DE INFORMACION, o=SECURITY DATA S.A. 2, c=EC
Fecha: 2022.06.24 14:42:29 -05'00'

MARCO ENRIQUE BENALCÁZAR, PHD.	LORENA ISABEL BARONA, PHD.
ADVISOR	CO-ADVISOR

COPYRIGHT STATEMENT

I, Danny Sebastián Díaz Padilla, declare under oath that the work described here is my own; that this work has not previously been submitted for any degree or professional qualification; and that I have consulted all the bibliographic references included in this document.

By means of this declaration, I assign my intellectual property rights over this work to the Escuela Politécnica Nacional, as established by the Intellectual Property Law, and by the current institutional rules and regulations at the time the work was made.

DANNY SEBASTIÁN DÍAZ PADILLA

ACKNOWLEDGMENT

To my mother for always giving me their support and understanding.

To Marco Benalcázar, Lorena Barona and Ángel Valdivieso, for giving me his support and patience during the realization of this project.

Danny Sebastián Díaz Padilla

TABLE OF CONTENTS

RESUMEN

El reconocimiento de gestos de la mano (HGR) ha permitido el desarrollo de formas alternativas de interacción hombre-máquina en los últimos años. Típicamente, se han desarrollado modelos de HGR basados en el aprendizaje supervisado con una gran precisión. Sin embargo, con el tiempo, aparece la necesidad de añadir nuevos gestos. En consecuencia, es necesario establecer un modelo que sea capaz de adaptarse a este cambio. El aprendizaje por refuerzo es un tipo de aprendizaje automático que permite desarrollar agentes capaces de adaptar su comportamiento a entornos dinámicos.

En este trabajo, utilizamos las redes Double Deep-Q (DDQN), un algoritmo de aprendizaje por refuerzo, para construir un agente (un modelo HGR en este caso), basado en señales electromiográficas (EMGs), capaz de reconocer nuevos gestos a lo largo del tiempo. El modelo propuesto es capaz de reconocer 5 gestos de la mano, logrando una precisión del 97,36% en la clasificación y del 94,83% para el reconocimiento. Se puede reentrenar a lo largo del tiempo con nuevas muestras, recalibrando su precisión y manteniéndola constante.

PALABRAS CLAVE: Aprendizaje por refuerzo, Aumento de datos, Doble Deep Q-Learning, Reajuste online, Reconocimiento de gestos de la mano, Olvido catastrófico

ABSTRACT

Hand gesture recognition (HGR) has enabled the development of alternative forms of human-machine interaction in recent years. Typically, HGR models based on supervised learning have been developed with high accuracy. However, over time, the need to add new gestures appears. Consequently, it is necessary to establish a model that is able to adapt to this change. Reinforcement learning is a type of machine learning that permits developing agents capable of adapting their behavior to dynamic environments.

In this work, we use Double Deep-Q Networks (DDQN), a reinforcement learning algorithm, to build an agent (a HGR model in this case), based on electromyography signals (EMGs), capable of recognizing new gestures over time. The proposed model is able to recognize 5 hand gestures, achieving an accuracy of 97.36% for classification and 94.83% for recognition. It can be retrained over time with new samples, recalibrating its accuracy and keeping it constant.

KEYWORDS: Catastrophic forgetting, Data augmentation, Double Deep Q-learning, EMG, Hand gesture recognition, Online readjustment, Reinforcement learning.

1. INTRODUCTION

The problem of hand gesture recognition is to identify the class and the instant of occurrence of a hand movement. Hand gesture recognition enables the development of new, more natural and human-centered forms of human-machine interaction [1]. One technique to develop a HGR is by processing electromyographic signals (EMGs). Which are biomedical signals that measure electrical currents generated in muscles during their contraction representing neuromuscular activities [2]-[4]. To extract the signals is common to use a non-invasive method called Surface electromyography (sEMG). Which gets information by measuring the electric potential field produced by active muscle fibers using electrodes on the skin [5].

1.1. Research question

Using Reinforcement learning, it will be possible to develop a HGR model with online learning capability, based on sEMG, that works in real time identifying the following gestures: fist, wave in, wave out, open and pinch, which works with a classification and recognition accuracy of at least 90%.

1.2. General objective

Develop a real-time hand gesture recognition model, using reinforcement learning and electromyography signals, to recognize 5 hand gestures: fist, wave left (wave in), wave right (wave out), fingers spread (open), and double tap (pinch).

1.3. Specific objectives

- Study the current state of the art related to reinforcement learning, applied to the development of HGR systems based on sEMG.
- Design a reinforcement learning model that meets the following characteristics:
    - Real-time recognition of 5 hand gestures (fist, wave left, wave right, fingers spread, double tap), using sEMG measured with the MYO Armband sensor.
    - Recognition and classification accuracy greater than 90% over the test set of the EMG-EPN 612 dataset.
- Evaluate the proposed model using the EMG-EPN-612 database in terms of: classification accuracy, recognition accuracy, response time, and recalibration. Comparing the obtained results with those obtained from the use of artificial neural networks and support vector machines previously used in the PIGR-19-07 research project.

1.4. General background

There are several models, mostly based on supervised learning, focused on hand gesture recognition in real time. For example: feed-forward artificial neural networks [6]-[8], convolutional neural networks [2,9], and support vector machines [10]-[11]. These models allow real-time recognition (in a response time of less than 300 milliseconds [12]), from four gestures [13] to ten gestures [10]. There is a sub-classification of all these models: general (trained with signals from multiple users) and specific (trained by each user with its own data. Among the best results for the specific models are 95.32% in classification [14] and 94.20% for recognition [15] and for general models 87.53% in classification [16] and 85.08% in recognition [6].

The aforementioned models focus on using only data previously labeled by humans. Basing their learning on a completely offline/static training and do not deal in depth with the variability existing in these signals along the time. When working with sEMG there is interpersonal and intrapersonal variability. That is, how a hand gesture is performed by one person differs in how it is performed by another (interpersonal variability) and also differs from the gesture performed by the same person at a different time (intrapersonal variability) [17].

To mitigate interpersonal and intrapersonal variability it is necessary to build adaptive algorithms that can respond to a dynamically changing environment. One advantage of reinforcement learning is that its algorithms are adaptive [18]. In this work, we propose an HGR model that takes as input surface EMGs, which measure the electric potential produced by active muscle fibers, permitting the recognition of gestures without using data explicitly labeled. The training of this model has two phases: offline (no human interaction) and online (human interaction). For offline training, we use a dataset composed of 612 users. For the online training, we use 100 users.

1.5. Contribution

We propose some improvements regarding previous reinforcement learning models focused on EMGs:

- Experience replay with reserved cache for each class.
- Interpolation of epsilon from 1 to 0.01.
- Re-definition of the Markov Decision Process, so that the next state is the same as the current state, to improve performance.
- Post-processing of impure labels isolated from other groups of labels.

When adding new gestures and evaluating classification and recognition, we have encountered a major problem: “Catastrophic Forgetting”. This is an inevitable feature of models based on connections [19]. We also described ways to mitigate and address this problem in order to get the best result.

2. LITERATURE REVIEW

2.1. Variability

Several works [33,34,36,48] have shown that HGR based on sEMG strongly depends on the subject. It depends on subject-specific characteristics such as muscular mass, skin thickness, strength of the mean voluntary contractions [33] and pattern of muscle synergies [36]. Among different users this dependency produces interpersonal variability.

The EMG signals change according to muscle wasting, changes in the electrode position and force variation [35]. The works [45]–[47] show that the EMG setup is also intrinsically variable, because of: fiber crosstalk, skin perspiration, small movements of the skin-to-electrode interface, power line interference and donning/doffing. Some works have therefore reported covariate shift, a phenomenon that occurs when the probability distribution of the input variables (sEMG data) changes with time [37], both for individual users and across different users [38]. These changes cause an intrapersonal variability, this is, the gesture performed by the same person differs along the time [17].

Minor changes of the EMG traces can hinder pattern recognition, degrading the system performance down, as shown in [48]. Therefore, re-training seems to be mandatory as noted in [34]. That work also shows how different feature representations play a more critical role in achieving a robust performance considering that minor changes. This work focuses on intrapersonal variability, mainly due to the accurate representation of features for specific users because sEMG strongly depends on the subject [33]. Creating a general model considering the variation of several users along the time is not as practical as a specific model that only considers the variation of only the user who will use the HGR system.

To minimize this variance, models must be re-trained every time a user interacts with the HGR system [40]. For each training, we must collect a large dataset to achieve high performance, which is time consuming and inconvenient in terms of usability. If re-training is not performed, the classification accuracy will decrease [39]. For instance, in [49] the performance of the HGR system over five days decreased by an average of 4.1% per day.

2.2. Online learning and Catastrophic forgetting

In online/streaming learning, a model learns online in a single pass loop over any portion of the dataset and it can be evaluated at any point rather than only between large batches. In [34] the user only needs to calibrate the system once a day, and it does not have to track unpredictable long-term changes (over a period of months). A supervised input data is usually considered for updating the model parameters [35]. However, this can be a problem when we need to add new gestures with no labeled data and preserve the accuracy.

We can use special learning techniques to adapt the HGR system to the users. Some adaptative methods, suitable for boosting the learning process, allow the model to leverage the experience gained over many source subjects to reduce the training time of a new target user [50]–[52]. In this way, the learning process does not start every time from scratch and it reduces to a faster refinement of prior knowledge [39].

While applying online learning, besides the variance in sEMG, we have to consider the "Catastrophic forgetting" (CF) problem because it is an inevitable characteristic of "connectionist" models [19]. As noted in [23], updating neural networks incrementally over time causes CF, and the new learning causes a rewrite of existing representations. CF in feed-forward neural networks was first observed in [53] and subsequently studied in [41]. CF is observed when a neural network is first trained on a dataset D1 and subsequently retrained on a disjunct dataset D2. This retraining causes the neural network to forget what it learned from dataset D1 almost immediately, in one or two minibatch steps [43].

Catastrophic Forgetting can be mitigated mainly with: ensemble methods [54,55], dual-memory systems [56]– [59] and regularization approaches as Dropout [60]. Although another work [42] showed that Dropout is not very effective. One influential solution has involved the use of two different complementary learning systems [61], encoding the new information and reactivating it in an interleaved manner. Allowing constant updating of the full scope of semantic representations such that new information does not overwrite older information [44]. In [24] the authors identified three primary mechanisms to mitigate catastrophic forgetting:

1) Replay of previous knowledge.
2) Regularization mechanisms to constrain parameter updates.
3) Expanding the neural network as more data becomes available.

In the special report "The great ai reckoning" [25], several researchers stated that some options to mitigate catastrophic forgetting are:

1) Elastic weight consolidation, also suggested by [19,44]: It is freezing important weights when learning new tasks. This algorithm protects individual network parameters such as synaptic weights by evaluating their importance for prior learning. Weights are regulated by a quadratic loss function that acts like a spring to pull important weights back toward the previous weight value. This method selectively slows the learning rate of protected synaptic weights. It increases the speed of learning in less important synaptic weights [44].
2) Progressive neural networks: It is the creation of a neural network for each task and freeze its connections.
3) Knowledge distillation: It is training several neural networks and averaging their predictions.

One reason for selecting reinforcement learning in this work is because it works with an experience replay, so the replay technique is included. Freezing the model weights and its connections is a good way of not forgetting what has been learned. However, as explained in the report, this means that over time the elasticity of the neural network will be reduced to a point where everything will be frozen and the network cannot assimilate new knowledge. However, preserving some connections produced better results in the mitigation of CF.

3. METHODOLOGY

3.1. Architecture

The architecture proposed is based on [22]. This is composed of 5 stages: data acquisition, pre-processing, feature extraction, classification and post-processing. We represent this in Figure 1.

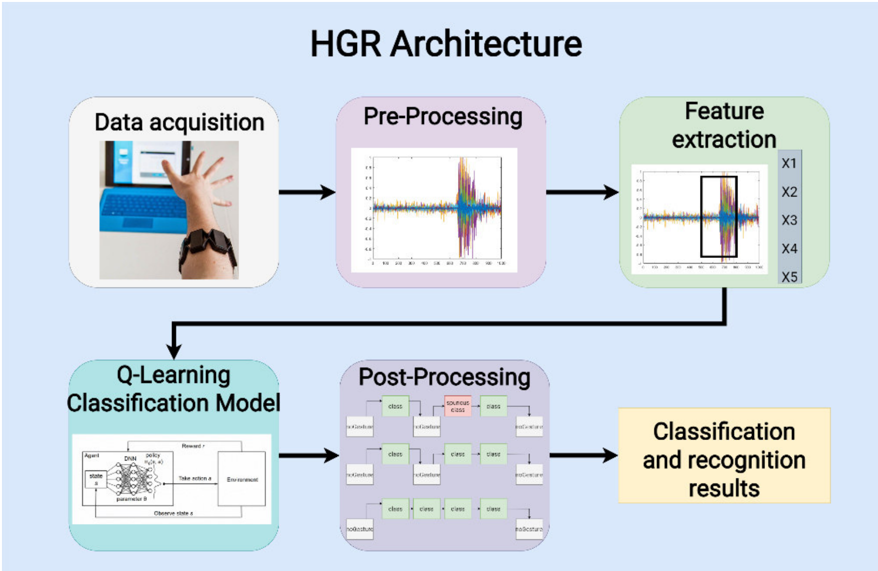

Figure 1: Hand Gesture Recognition architecture based on Q-Learning to learn to classify and recognize EMG signals

3.1.1. Data acquisition

The EMG-EPN-612 [27] database was used. This dataset contains EMGs from 612 users, the measurements were made using the MYO Armband device with a sampling frequency of 200Hz. Half of the users were used for training, validation and testing and the remaining users were used for public evaluation on the [26] web platform. The datasets are composed of 25 repetitions of the following gestures: fist, wave in, wave out, open, pinch and no gesture or relaxation gesture measurements. For each gesture, there is EMG information, ground truth and user information, except for user datasets corresponding to testing hosted on the web platform. The tuning of hyperparameters was done using the first 306 users, and with these hyperparameters the model was evaluated publicly on the web page.

3.1.2. Pre-processing

The energy-based orientation correction method described in [28] was used, because the cuffs are susceptible to electrode rotation. In order to perform the recognition part, it is necessary to make a segmentation process, partitioning the EMG in several windows. Twenty-four windows are generated and labeled using a sliding window of 300 points and stride of 30 points.

3.1.3. Feature extraction

Feature extraction methods are useful to extract relevant features from the EMGs, which can be defined in time, frequency, or time-frequency domains [14]. We generated five features for each of the 8 channels of the EMG signal, resulting in 40 features based. The 5 features generated are: Standard deviation (SD), Absolute envelope (AE), Mean absolute value (MAV), Energy (E) and Root mean square (RMS) [15].

3.1.4. Classification

We use Q-learning for classification which is an off-policy algorithm of reinforcement learning. For any finite Markov Decision Process (MDP), Q-learning finds an optimal policy by maximizing the expected value of the total reward given both an initial state and an action. We can implement this using lookup tables to store the Q values for each state and action of an environment [29]. However, this approach is not suitable for environments with a large state-action spaces. To simplify the representation of this table, we can use an approximation function. An artificial feed-forward neural network serves this purpose. With this neural network we can approximate the value of Q for each state received as input, as if it were a linear regression [29,30]. Finally, for a more stable learning we use Double Deep-Q Network (DDQN), which is an off-policy reinforcement learning algorithm that utilizes double estimation to counteract overestimation problems with traditional Q-learning [62].

Training will be done by reading mini-batches from the experience replay (ER). ER also helps to break correlations and to increase training speed [31,32]. We reserve some space for each action. This is necessary because the model predicts much more no gesture class ("relax" state), producing overfitting. With separated reserved cache we limit their influence on the rest of the gestures. We structured each tuple inside the ER in the following way:

$$E_t = [state_t,\ action_t,\ reward_t,\ state_{t+1},\ terminal\_flag]$$

Where $state_{t+1}$ is equal to $state_t$ because no matter which action selects the agent for a state, it will not change, $action_t$ is the action taken by the agent at $state_t$, $reward_t$ is the reward given for that action and *terminal_flag* indicates if the analyzed signal window is the final one.

The reason for doing this is that for any signal window received by the agent, no matter what action it takes, the next signal window was already segmented from the same signal. The agent has no influence on the future of the first signal window. With this consideration, when applying the Bellman equation, the agent focuses his training on understanding the current window. Also, it is necessary to reduce the gamma discount factor to get better results. We have shown that with these changes, the accuracy improved regarding other reinforcement learning models for HGR. The definition of the reinforcement learning entities is the following:

- **Environment**: For offline training, we defined the environment in code where the label of each window is already known. Here, the reward is obtained by comparing the prediction of a window with the actual label of that window. For online learning, the environment is represented by a human who gives a reward for the final prediction. It is important to emphasize that the human does not give a reward per window, but for the whole signal. Then the reward is assigned to each signal window depending on the muscle activity detected by an external algorithm.
- **Agent**: Is a Double Deep-Q Network that receives a state and predicts which action will give it the best reward.
- **State**: Is the window portion of the entire segmented signal.
- **Episode**: In an episode, the agent will interact with all the windows of a single segmented signal.
- **Action**: Is the choice that the agent must make about which gesture belongs to each window.
- **Reward**: A value of 1 is given when the agent predicts correctly the signal window class and -1 when the agent predicts incorrectly it.

The selected hyper-parameters are shown in Table 1. These hyper-parameters are used in the static model and in the online model. We got better results using regularization by adding a "Dropout" layer after the first activation function.

Table 1: Hyperparameters

Name	Configuration
Sliding window size	300
Sliding window stride	30
Optimizer	ADAM
Input Layer	40 neurons
Hidden Layer 1	40 neurons
Activation Layer 1	ReLU
Dropout Layer	0.1
Hidden Layer 2	40 neurons
Activation Layer 2	ReLU
Output Layer	6 neurons
Epsilon	From 1 to 0.01
Gamma	0.1
Alpha decay	0.1
Loss function	MSE
Batch size	128
Learning rate	0.001
Interval for learn from replay	10 steps
Epochs	5
Experience replay size	600

3.1.5. Post-processing

The agent predicts one gesture class for each window. To know the gesture of the complete signal, it is necessary to merge all the predictions of the windows and reach a consensus on getting the mode. All predicted gestures that do not match the mode are spurious predictions. Through the mode, we can determine the majority class and replace the minority classes (spurious classes) with this one. As we can see in Figure 2, we first replace the minority class (red) with the majority class (green) different from no gesture, then we "fill" the batches where a label or group of labels are isolated from the majority class.

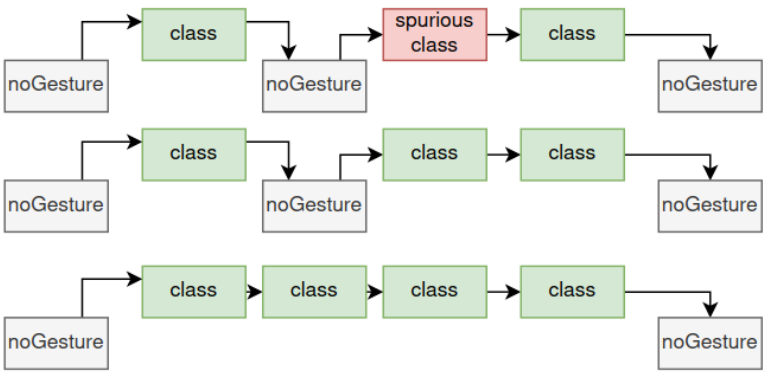

Figure 2: post-processing of the predictions by window.

3.2. Online readjustment

For the readjustment part, a human directly gives feedback. In order to facilitate this interaction, the interface shown in Figure 3 was developed. Usually, in the online training, we simply use a small learning rate and train directly from the experience replay. This method causes the accuracy to remain stable and nearly constant after several iterations. However, the disadvantage is that we need the labeled data. The aim of this work is to re-adjust the model without using explicitly labeled data, taking advantage of reinforcement learning and the correlation breaking property of experience replay. For the evaluation of online learning and the adaptation of the model to a dynamic environment, we have simulated the inclusion of new classes in real time, using the existing gestures in the dataset. In this way, we can evaluate the change of the model's accuracy as it adapts to the environment.

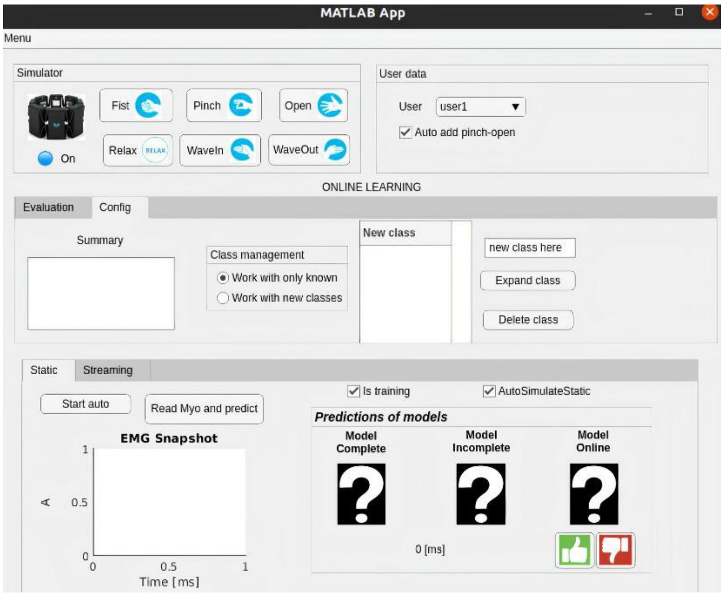

Figure 3: User interface for give feedback to the agent.

We use different approaches to readjust the model. The goal of each approach is to recover the accuracy lost when new classes were included and keep it stable. We start training a model for three gesture classes: “fist”, “waveIn” and “waveOut”. Then two new gesture classes (“pinch” and “open”) will be included, without previous offline training. Samples of these new gestures, got from the dataset of 100 users picked randomly, will gradually be displayed to the agent. The advantage of performing this simulation is that we will properly evaluate the agent’s behavior in front of the pre-stored labeled ``open” and ``pinch” gestures without ever showing them directly. In order to achieve this task, we will make changes only to the output layer and copy the rest of the weights (like transfer learning). This whole process is shown graphically in Figure *4* and it is described in order below:

a) The system receives an electromyographic signal.

b) We use a muscle activity detection algorithm based on frequency sampling. It will give us the ground-truth.

c) We will extract the 40 features for each segmentate window. These features will be sent to the agent to make a prediction and to the system to associate each feature with a label.

d) The agent makes a prediction p, which will be displayed to the user and stored in memory.

e) Using the agent's action prediction, the features and the same predicted label "*P*" will be associated for each respective window of the ground-truth got by the muscle activity algorithm.
f) The interface will display the gesture predicted by the agent.
g) A human should give positive or negative feedback depending on whether the agent got it right or wrong for the gesture performed.
h) Tuples are generated to insert in the experience replay only using the windows where there is muscle activity. Additionally, tuples are generated with the sections marked as "relax" but with a positive reward since there is no muscle activity in those sections.
i) We add the tuples to the experience replay in the corresponding cache sections.
j) A 5-epoch training with a reduced learning rate will be performed on a random mini-batch of the experience replay, including the new insertions.

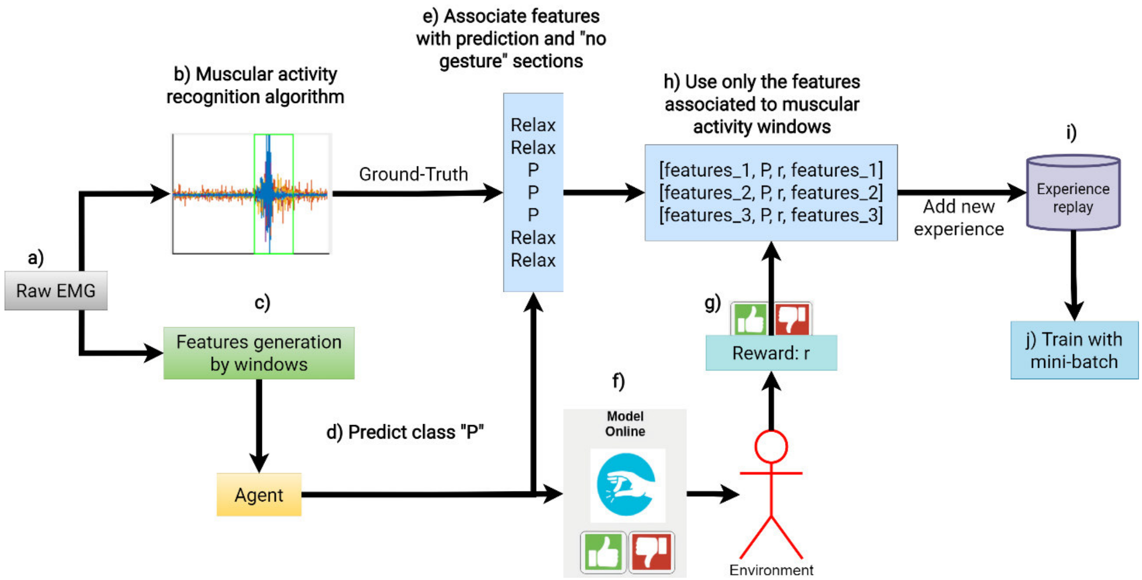

Figure 4: Flow for online learning

If the reward is positive for a gesture that is not known (that is not "fist", "waveIn" or "waveOut"), one can enhance that learning by including a data augmentation of that EMG signal with the label that the agent predicted (Figure 5). The techniques used for creating new data were:

- "moving average box". - Several smoother signals are generated by averaging 10 or more points of the original signal.
- "shifting". - Each signal is shifted to the left or right by 10 or more points.

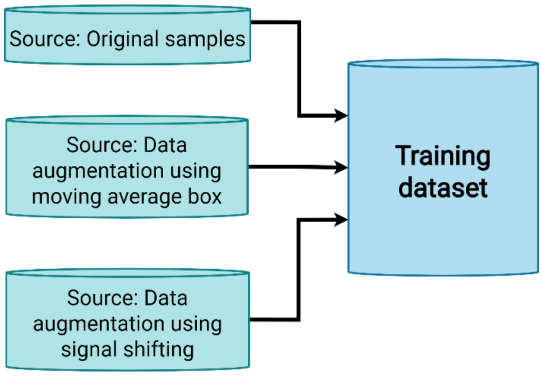

Figure 5: Data augmentation overview

Besides the weight transfer and experience replay techniques, to mitigate catastrophic forgetting, a constant epsilon equal to 0.2 was used. This way, the algorithm keeps "exploring" even after it has been fully trained. We only used this option when the model is being re-adjusted. In production, we should reduce the epsilon value to 0.

4. RESULTS AND DISCUSSION

This section will present the results of the evaluation of the HGR user specific model on the public platform and the online readjustment evaluation.

4.1. Offline/Static results

The model name in the public evaluator is "TestPlatformDQNN-0" and the results were:

- Classification: 97.36 % ± 4.35 %.
- Recognition: 94.83 % ± 5.55 %.

The histograms, provided by the online web platform, for both classification and recognition generated by the platform can be seen in Figure 6 and Figure 7.

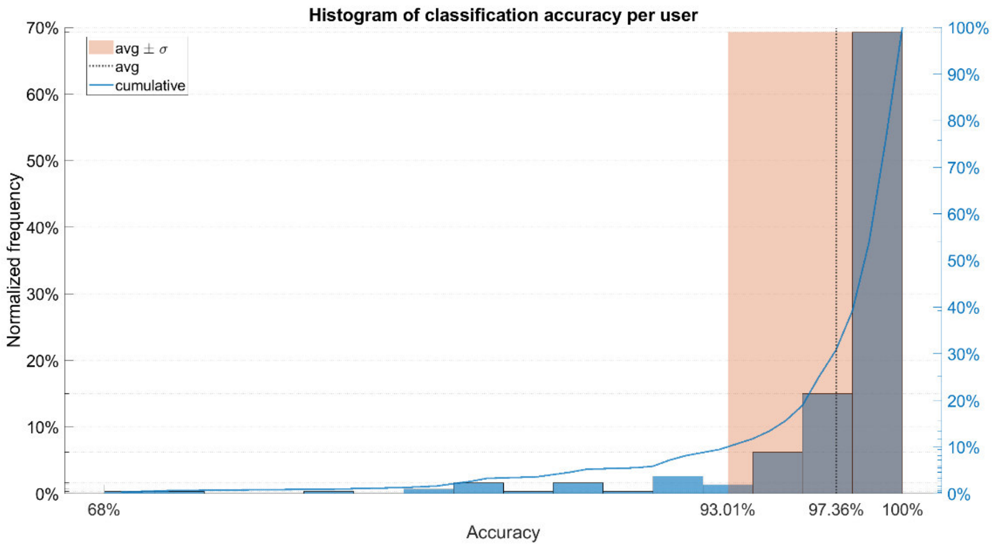

Figure 6: Histogram of classification accuracy.

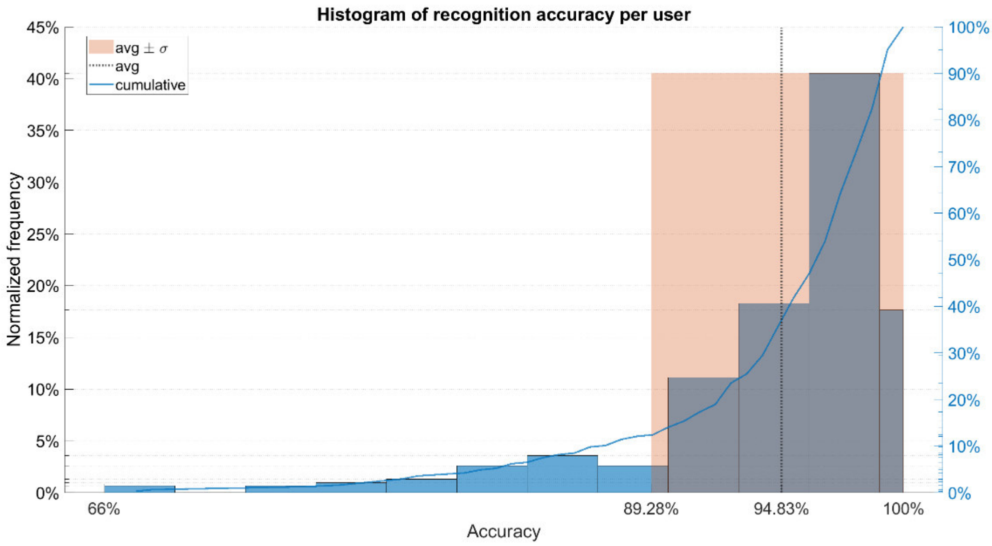

Figure 7:Histogram of recognition accuracy.

We can see the confusion matrix provided by the online web platform in Figure 8.

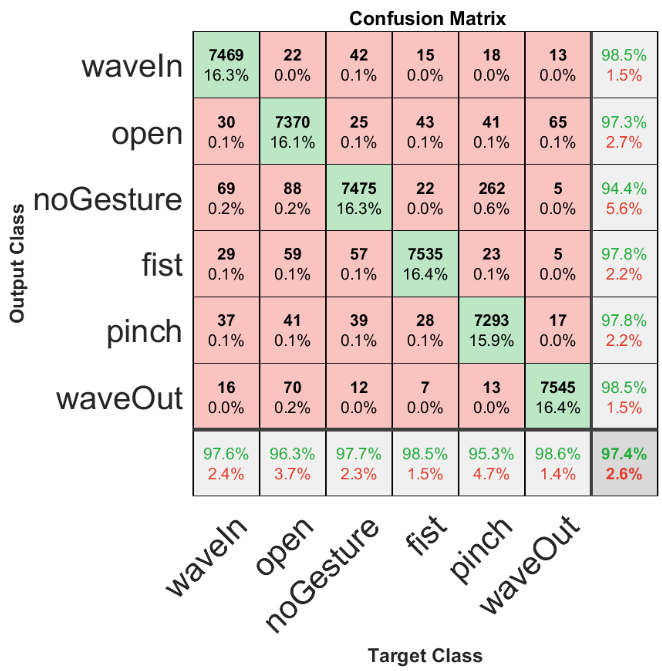

Figure 8: Confusion matrix for static training.

4.2. Online readjustment evaluation

4.2.1. Readjusting using only experience and rewards

When new gestures are added, and we performed no retraining with the known gesture's dataset, the classification and recognition accuracy starts around 15% (Figure 9). This shows that adding a new neuron in the output layer deconstructs all previous connections and weights.

Using only human feedback, a mean accuracy slightly higher than 55% is achieved for classification and recognition after 20 interactions for the 100 users.

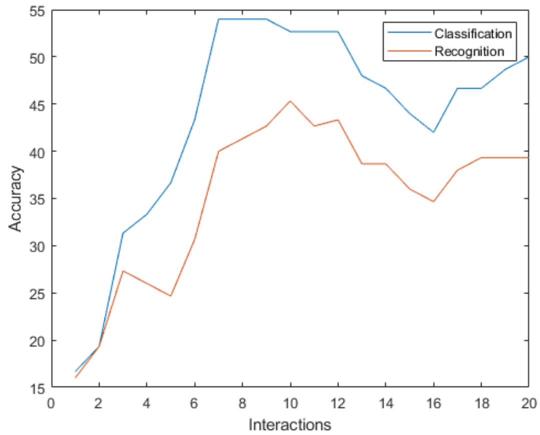

Figure 9: Online readjustment only using experience given from a human.

4.2.2. Readjusting using experience, rewards and retraining

Another approach we can apply to improve the results; it is to perform retraining with the known gesture dataset. In addition, to improve the results even further, we can use data augmentation on all predictions. The initial accuracy got after retraining is 70%. We gradually add gestures from new classes, and the final accuracy ranges between 88% and 90% for both classification and recognition. Model accuracies remain nearly constant after 20 interactions for 100 users (Figure 10).

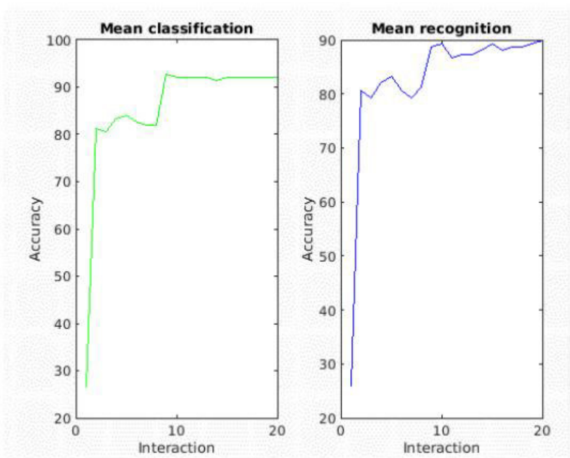

Figure 10: Online readjustment with big batch size retraining.

4.3. Discussion

The confusion matrix (Figure 8) shows a relatively high error (0.6%) in predicting the "noGesture" class instead of the "pinch" class. The classification accuracy of static/offline training is higher compared with other works (Table 2), for user-specific models from the literature. For the neural network results in [14], we should consider that the authors worked with 60 users. The other models, Q-Learning and SVM, worked with the same number of users of this work.

Table 2: Classification and recognition accuracy comparison

Model	Classification	Recognition
Actual model	97.35%	94.82%
Q-learning [22]	90.72%	87.51%
SVM [15]	94.90%	94.20%
Neural Network [14]	96.37%	94.90%

It is worth mentioning the differences between this work and [22] are:

- The ADAM optimizer is used instead of Stochastic Gradient Descent with momentum.
- The mini-batch of the experience replay is shown several times in the training phase.
- We do not assume that an action will change the current state.
- The gamma value (discount factor) is significantly reduced to 0.1.

If we use only experience and not retraining with known dataset when readjusting the model, it may take many interactions to achieve high accuracy, 20 interactions do not seem to be enough (Figure 9). The best way to reach the highest accuracy faster is retraining every four steps with a lower learning rate (Figure 10). The final decision on which model to choose depends on the minimum required accuracy and hardware capacity because retraining can be expensive.

5. CONCLUSIONS

We made a literature review of the current state-of-the-art of reinforcement learning related to online learning to mitigate the catastrophic forgetting problem.

We have developed a reinforcement learning model based on Double Deep-Q Network with 97.35% accuracy in classification and 94.82% in recognition. We have evaluated its robustness and adaptability when including a new class, and we got an average accuracy rate of 89.30% for both classification and recognition. This accuracy is lower than the accuracy reached by training directly with the labeled data, but with the advantage of using only experience and rewards given by a human.

The model can be readjusted after a while, due to the property of experience replay to break correlations. In order to keep the high performance of the model, it is necessary for the human to interact hundreds of times with the agent, which in practical terms is not convenient. Because of the limited number of interactions, it is important to enhance learning with data augmentation. This way, whether the prediction is correct or incorrect, we will inject much more experience into the experience replay that will be used in the real time training. In the end, the model achieves almost a constant accuracy against new unknown classes integration.

6. FUTURE WORK

The muscle activity detection algorithm may cause the recognition to fall in the readjusting phase, due to incorrect segmentation. Developing a more accurate algorithm for this task will be important for future work.

7. REFERENCES

[1] M. Sigalas, H. Baltzakis and P. Trahanias, "Gesture recognition based on arm tracking for human-robot interaction", 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010.

[2] M. B. I. Reaz, M. S. Hussain, and F. Mohd-Yasin, "Techniques of emg signal analysis: detection, processing, classification and applications" Biological procedures online, vol. 8, no. 1, pp. 11–35, 2006.

[3] J. Rodriguez-Falces, J. Navallas, and A. Malanda, "Emg modeling," Computational Intelligence in Electromyography Analysis-A Perspective on Current Applications and Future Challenges, pp. 3–36, 2012.

[4] D. Farina, N. Jiang, H. Rehbaum, A. Holobar, B. Graimann, H. Dietl, and O. C. Aszmann, "The extraction of neural information from the surface emg for the control of upper-limb prostheses: emerging avenues and challenges" IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 22, no. 4, pp. 797–809, 2014.

[5] M. J. Zwarts and D. F. Stegeman, "Multichannel surface emg: basic aspects and clinical utility," Muscle and Nerve: Official Journal of the American Association of Electrodiagnostic Medicine, vol. 28, no. 1, pp. 1–17, 2003.

[6] E. A. Chung and M. E. Benalcázar, "Real-Time Hand Gesture Recognition Model Using Deep Learning Techniques and EMG Signals," 2019 27th European Signal Processing Conference (EUSIPCO), 2019, pp. 1-5.

[7] M. R. Ahsan, M. I. Ibrahimy, and O. O. Khalifa, "EMG signal classification for human computer interaction: A review," Eur. J. Sci. Res., vol. 33, no. 3, pp. 480–501, 2009.

[8] R. H. Chowdhury, M. B. I. Reaz, M. A. Bin Mohd Ali, A. A. A. Bakar, K. Chellappan, and T. G. Chang, "Surface electromyography signal processing and classification techniques," Sensors (Switzerland), vol. 13, no. 9, pp. 12431–12466, 2013.

[9] X. Chen, Y. Li, R. Hu, X. Zhang, and X. Chen, "Hand gesture recognition based on surface electromyography using convolutional neural network with transfer learning method," IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 4, pp. 1292–1304, 2020.

[10] H. Su, S. E. Ovur, X. Zhou, W. Qi, G. Ferrigno, and E. De Momi, "Depth vision guided hand gesture recognition using electromyographic signals," Adv. Robot., vol. 34, no. 15, pp. 985–997, 2020.

[11] B. Crawford, K. Miller, P. Shenoy, and R. Rao, "Real-time classification of electromyographic signals for robotic control," Proc. Natl. Conf. Artif. Intell., vol. 2, pp. 523–528, 2005.

[12] S. Benatti et al., "A sub-10mW real-Time implementation for EMG hand gesture recognition based on a multi-core biomedical SoC," Proc. - 2017 7th Int. Work. Adv. Sensors Interfaces, IWASI 2017, pp. 139–144, 2017.

[13] J. Kim, S. Mastnik, and E. André, "EMG-based Hand Gesture Recognition for Realtime Biosignal Interfacing", 13th International Conference on Intelligent User Interfaces - IUI 08, 2008.

[14] M. E. Benalcázar, Á. L. Valdivieso Caraguay, and L. I. Barona López, "A user-specific hand gesture recognition model based on feed-forward neural networks, emgs, and correction of sensor orientation," Applied Sciences, vol. 10, no. 23, p. 8604, 2020.

[15] L. I. Barona López, Á. L. Valdivieso Caraguay, V. H. Vimos, J. A. Zea, J. P. Vasconez, M. Álvarez and M.E.Benalcázar, “An energy-based method for orientation correction of emg bracelet sensors in hand gesture recognition systems,” Sensors, vol. 20, no. 21, p. 6327, 2020.

[16] A. Jaramillo-Yanez, L. Unapanta, and M. E. Benalcázar, “Short-term hand gesture recognition using electromyography in the transient state, support vector machines, and discrete wavelet transform,” in 2019 IEEE Latin American Conference on Computational Intelligence (LA-CCI), pp. 1–6, IEEE, 2019.

[17] Martens, J., Daly, D., Deschamps, K., Fernandes, R. J. P., and Staes, F. (2015). Intra-individual variability of surface electromyography in front crawl swimming. PloS one, 10(12), e0144998.

[18] J. K. Baher, Abdulhai; Rob, Pringle; Grigoris, “Reinforcement Learning for True Adaptive Traffic Signal Control,” ASCE, pp. 278–285, 2003.

[19] James Kirkpatrick et al. "Overcoming catastrophic forgetting in neural networks". arXiv:1612.00796.

[20] W. Seok, Y. Kim, and C. Park, “Pattern Recognition of Human Arm Movement Using Deep Reinforcement Learning Intelligent information system and embedded software engineering, Kwangwoon University,” pp.917–919, 2018.

[21] Chengjie Song, Chunjie Chen,Yanjie Li1 and Xinyu Wu ”Deep Reinforcement Learning Apply in Electromyography Data Classification”.

[22] Juan Pablo Vasconez, Lorena Isabel Barona Lopez, Angel LeonardoValdivieso Caraguay, Patricio J. Cruz, Robin Alvarez, and Marco E.Benalcazar. ”A Hand Gesture Recognition System using EMG and Reinforcement Learning: a Q-Learning Approach *”.

[23] Tyler L. Hayes, Kushal Kafle, Robik Shrestha, Manoj Acharya, Christopher Kanan ”REMIND Your Neural Network to Prevent Catastrophic Forgetting”. arXiv:1910.02509.

[24] Parisi, G.I., Kemker, R., Part, J.L., Kanan, C., Wermter, S.: Continual lifelong learning with neural networks: A review. Neural Networks (2019).

[25] Tom Chivers ”How DeepMind Is Reinventing the Robot”, part of special report on AI, “The Great AI Reckoning”.

[26]EMGGestureRecognitionEvaluator,https://aplicaciones-ia.epn.edu.ec/webapps/home/session.html?app=EMG%20Gesture%20Recognition%20Evaluator n Evaluator

[27] M. E. Benalcazar, L. Barona, L. Valdivieso, X. Aguas, and J. Zea, “Emg-epn-612 dataset,” Nov. 2020.

[28] L. I. Barona L´opez et al., “An energy-based method for orientation correction of EMG bracelet sensors in hand gesture recognition systems, ”Sensors (Switzerland), vol. 20, no. 21, pp. 1–34, 2020.

[29] Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. MITpress (2018).

[30] Kukker, A., Sharma, R.: Neural reinforcement learning classifier forelbow, finger and hand movements. Journal of Intelligent and FuzzySystems 35(5), 5111-5121 (2018).

[31] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Belle-mare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G.,et al.: Human-level control through deep reinforcement learning. nature518(7540), 529-533 (2015).

[32] Kapturowski, S., Ostrovski, G., Quan, J., Munos, R., Dabney, W.:Recurrent experience replay in distributed reinforcement learning. In:International conference on learning representations (2018).

[33] S. Benatti, F. Montagna, V. Kartsch, A. Rahimi, D. Rossi and L. Benini,"Online Learning and Classification of EMG-Based Gestures on a Parallel Ultra-Low Power Platform Using Hyperdimensional Computing," in IEEETransactions on Biomedical Circuits and Systems, vol. 13, no. 3, pp. 516-528, June 2019.

[34] Gu, Y., Yang, D., Huang, Q., Yang, W., and Liu, H. (2018). Robust EMG pattern recognition in the presence of confounding factors: features,classifiers and adaptive learning. Expert Systems with Applications, 96,208–217.

[35] Kawano, S., Okumura, D., Tamura, H. et al. Online learning method using support vector machine for surface-electromyogram recognition. Artif Life Robotics 13, 483–487 (2009).

[36] Matsubara T, Morimoto J. Bilinear modeling of EMG signals to extract user-independent features for multiuser myoelectric interface. IEEE Trans Biomed Eng. 2013 Aug;60(8):2205-13.

[37] Kaufmann P, Englehart K, Platzner M. Fluctuating emg signals: investigating long-term effects of pattern matching algorithms. Annu Int ConfIEEE Eng Med Biol Soc. 2010;2010:6357-60.

[38] Atzori, M., Gijsberts, A., Castellini, C. et al. Electromyography datafor non-invasive naturally-controlled robotic hand prostheses. Sci Data 1,140053 (2014).

[39] Patricia, N., Tommasit, T. and Caputo, B. (2014). Multi-source Adaptive Learning for Fast Control of Prosthetics Hand. 2014 22nd International Conference on Pattern Recognition.

[40] Sensinger, J. W., Lock, B. A. and Kuiken, T. A. (2009). Adaptive Pattern Recognition of Myoelectric Signals: Exploration of Conceptual Framework and Practical Algorithms. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 17(3).

[41] French, R. (1999). Catastrophic forgetting in connectionist networks. Trends in Cognitive Sciences, 3(4), 128–135.

[42] Pf¨ulb B., Gepperth A., Abdullah S., Kilian A. (2018) CatastrophicForgetting: Still a Problem for DNNs. In: K˚urkov´a V., Manolopoulos Y.,Hammer B., Iliadis L., Maglogiannis I. (eds) Artificial Neural Networksand Machine Learning ICANN 2018. ICANN 2018. Lecture Notes in Computer Science, vol 11139. Springer, Cham.

[43] Schak M., Gepperth A. (2019) A Study on Catastrophic Forgetting in Deep LSTM Networks. In: Tetko I., K˚urkov´a V., Karpov P., Theis F. (eds)Artificial Neural Networks and Machine Learning – ICANN 2019: DeepLearning. ICANN 2019. Lecture Notes in Computer Science, vol 11728.Springer, Cham.

[44] Hasselmo, M. E. (2017). Avoiding Catastrophic Forgetting. Trends in Cognitive Sciences, 21(6), 407–408.

[45] A.Waris et al., "On the robustness of real-time myoelectric control investigations: A multiday fitts law approach," J. Neural Eng., vol. 16,no. 2, pp. 2018, Art. no. 026003.

[46] A. J. Ishak et al., "Design of a wireless surface EMG acquisition system,"in Proc. 24th Int. Conf. Mechatron. Mach. Vis. Pract., Nov. 2017, pp. 1–6.

[47] B. Milosevic et al., “Exploring arm posture and temporal variability in myoelectric hand gesture recognition,” in Proc. 7th IEEE Int. Conf. Biomed. Robot. Biomechatroni., 2018, pp. 1032–1037.

[48] A. Waris et al., “Multiday evaluation of techniques for EMG based classification of hand motions,” IEEE J. Biomed. Health Inf., to be published.

[49] Ams ̈uss, S., Paredes, L. P., Rudigkeit, N., Graimann, B., Herrmann, M.J., and Farina, D. (2013). Long term stability of surface EMG pattern classification for prosthetic control. Conf Proc IEEE Eng Med Biol Soc,2013, 3622-3625.

[50] T. Matsubara, S.-H. Hyon, and J. Morimoto, “Learning and adaptation of a stylistic myoelectric interface: Emg-based robotic control withindividual user differences.” in Proc ROBIO, 2011.

[51] R. Chattopadhyay, N. C. Krishnan, and S. Panchanathan, “Topology preserving domain adaptation for addressing subject based variability insemg signal,” in AAAI Spring Symposium: Computational Physiology,2011.

[52] T. Tommasi, F. Orabona, C. Castellini, and B. Caputo, “Improvingcontrol of dexterous hand prostheses using adaptive learning,” IEEETransactions on Robotics, vol. 29, no. 1, pp. 207–219, 2013.

[53] McCloskey, M., Cohen, N.J.: Catastrophic interference in connectionist networks: the sequential learning problem. Psychol. Learn. Motiv. 24,109–165 (1989).

[54] Fernando, C., et al.: PathNet: Evolution Channels Gradient Descent in Super Neural Networks (2017).

[55] Ren, B., Wang, H., Li, J., Gao, H.: Life-long learning based on dynamic combination model. Appl. Soft Comput. J. 56, 398–404 (2017).

[56] Gepperth, A., Karaoguz, C.: A bio-inspired incremental learning archi-tecture for applied perceptual problems. Cogn. Comput. 8(5), 924–934(2016).

[57] Kemker, R., Kanan, C.: FearNet: Brain-Inspired Model for Incremental Learning, pp. 1–16 (2017).

[58] Rebuffi, S.a., Kolesnikov, A., Sperl, G., Lampert, C.H.: iCaRL : Incre-mental Classifier and Representation Learning, pp. 2001–2010 (2017).

[59] Shin, H., Lee, J.K., Kim, J., Kim, J.: Continual Learning with DeepGenerative Replay (NIPS) (2017).

[60] Goodfellow, I.J.,Mirza,M., Xiao, D., Courville, A., Bengio, Y.: AnEmpirical Investigation of Catastrophic Forgetting in Gradient-BasedNeural Networks (2013).

[61] McClelland, J.L. et al. (1995) Why there are complementary learning systems in the hippocampus and neocortex: insights from the successesand failures of connectionist models of learning and memory. Psychol.Rev. 102, 419–457.

[62] Hasselt H., Guez A., Silver D.: Deep Reinforcement Learning with Double Q-Learning.