MASTER THESIS

AUTOMATIC EMG ENVELOPE DETECTION FOR LEG MUSCLES USING DEEP LEARNING FROM A MULTI-ELECTRODE EMBEDDED GARMENT

Orla Rispens - s1905724

FACULTY OF ENGINEERING TECHNOLOGY
DEPARTMENT OF BIOMECHANICAL ENGINEERING

EXAMINATION COMMITTEE
Prof. Dr. Ir. M. Sartori
Dr. Ir. M.I. Refai
Dr. Ing. G. Englebienne

DOCUMENT NUMBER
BE - <NUMBER>

UNIVERSITY OF TWENTE.

27-08-2024

I. Preface and acknowledgments

Before you lies the master thesis: 'Automatic EMG envelope detection for leg muscles using deep learning from a multi-electrode embedded garment'. Which was written as a part of my master's program 'Biomedical Engineering' at the University of Twente. Performing this assignment has been my main focus since November 2023.

During my internship, I noticed that I had not challenged myself enough, which led me to choose this subject for my master's thesis. I had not previously done any courses on deep learning, and gaining knowledge on this subject has been an amazing challenge for the past months. During my thesis, I improved my Matlab programming, worked on my skills in the movement lab, and wrote my first article. After many months of hard work, I am proud to present this thesis.

This endeavor would not have been possible without my daily supervisor dr. ir. Mohammed Irfan Refai with whom I had countless meetings, shared ideas, and gave me great feedback on many versions of my thesis. I also would like to express my deepest appreciation to prof. dr. ir. Massimo Sartori for his invaluable feedback and a very inspiring meeting. Additionally, I'm extremely grateful to dr. ing. Gwenn Englebienne who generously provided his knowledge and expertise in the field of deep learning.

A special thanks to dr. ir. Donatella Simonetti for answering my many questions during meetings and via e-mails about her research. Many thanks to Jeanine Lodeweges-de Vries, who helped me with all the administrative steps and planned the graduation colloquium. Thanks should also go to Yolanda Assink-Strokappe for answering my questions and planning meetings with my supervisors. I am also grateful to Hinke Gouma for helping me with the administrative steps that had to be taken. I am also thankful to Michiel Ligtenberg for showing me how all the equipment in the lab worked.

I would like to acknowledge the test subjects who participated in this research for their time and efforts. I would be remiss in not mentioning Isa Bakker, Bjorn Rispens and Nilé Hovius for proofreading my thesis. Lastly, I'd like to thank all of my friends and family for their continued support during the completion of my thesis.

Orla Rispens

Enschede, August 27, 2024

CONTENTS

I Preface and acknowledgments i

Abstract 1

II Introduction 1

III Methods 2

III-A Convolutional Neural Network . . . 3

III-B Encoder-Decoder Network . . . 3

III-C EMG clustering using NNMF . . . 3

III-D Experimental data lower leg . . . 3

III-E Experimental data upper leg . . . 3

III-E1 Electrode embedded wearable garment . . . 4

III-E2 Experimental procedures . . . 4

III-E3 Manual selection of muscle electrodes . . . 5

III-E4 Data processing . . . 5

III-F Training of networks and performing NNMF clustering . . . 6

III-F1 CNN and EDN . . . 6

III-F2 Clustering using NNMF . . . 6

III-G Validation procedures . . . 6

IV Results 7

IV-A Lower leg . . . 7

IV-B Upper leg . . . 7

V Discussion 10

V-A Limitations and future perspective . . . 11

VI Conclusion 12

Appendix 14

A Use of AI . . . 14

B Results . . . 15

B1 Statistics of lower leg . . . 15

B2 Results upper leg . . . 20

C Muscle synergies of the lower leg by NNMF and EDN . . . 21

C1 Introduction . . . 21

C2 Methods . . . 21

C3 Results . . . 21

C4 Discussion and conclusion . . . 22

D Locality mask upper leg . . . 24

E Euclidean distances . . . 25

Automatic EMG envelope detection for leg muscles using deep learning from a multi-electrode embedded garment

Rispens, O.C.
Faculty of Engineering Technology
University of Twente
Enschede, Netherlands
o.c.rispens@student.utwente.nl

Abstract—**Surface electromyography (sEMG) can be used to determine the activity of a muscle. However, applying electrodes in the right positions is time-consuming and requires extensive anatomical knowledge. A multi-electrode embedded garment was created by Simonetti et al. for the lower leg, to speed up electrode placement. Muscle-specific EMG envelopes were extracted using a non-negative matrix factorization (NNMF)-based clustering method for lower leg muscles. In the present study, the garment and the NNMF-based clustering method were adapted to the upper leg. Additionally, two new approaches were created to extract muscle-specific EMG envelopes: one using a convolutional neural network (CNN), and the other an encoder-decoder network (EDN). Validation was performed using two datasets: for the upper leg EMG recordings of 8 subjects were made, and for the lower leg three subjects. Before recordings, two electrodes of the garment for five lower or upper leg muscles were manually selected based on the SENIAM guidelines. The identified EMG envelopes by manual selection, NNMF-based clustering, the CNN, and the EDN were gait-cycle averaged. The performance of NNMF-based clustering and the CNN and EDN were compared by calculating the R^2 value between each and manual selection. The median overall R^2 value of both CNN (0.9) and EDN (0.9) for the lower leg was significantly larger ($p<0.01$) than that of NNMF (0.8), indicating that the EMG envelopes created by CNN and EDN more closely resembled those made through manual selection. For the upper leg, no significant results could be reported due to the small number of subjects, but the NNMF-based method had more R^2 values in the 'very weak' (<0.2) and 'weak' (0.2-0.39) categories and less in the 'very strong' (>0.8) than CNN and EDN. NNMF-based clustering, CNN, and EDN generally found muscle-specific EMG envelopes that resembled those found by manual selection. To conclude, the newly created CNN and EDN-based methods outperformed the previously created NNMF-based methods and all methods found satisfactory EMG envelope results. These methods could play a great role in accelerating the electrode placement process.**

II. Introduction

Surface electromyography (sEMG) has many applications, including diagnosing gait disorders, supporting rehabilitation techniques, and health monitoring systems [1]. For instance, lower limb exoskeletons used in rehabilitation can aid in relearning to walk by performing repetitive movements [2, 3]. To control exoskeletons, obtaining information about intended movements is essential. This can be derived from sEMG signals, which measure motor unit action potentials (MUAPs) at the skin surface, indicating muscle activity about 10 milliseconds before movement starts. The number and intensity of MUAPs correlate with muscle contraction intensity, thus reflecting the level of muscle activity [4]. By preprocessing the EMG signals with filtering and rectification steps, an EMG envelope is created; the greater the muscle force, the higher the EMG envelope reaches [5].

The measurement of sEMG follows guidelines established by the SENIAM project [6]. However, this process requires extensive anatomical knowledge and precise electrode placement, making it a time-consuming process [7]. These drawbacks pose challenges for patients using exoskeletons for daily physical therapy or as permanent mobility solutions. Therefore, to effectively use EMG measurements in diagnostic and therapeutic contexts, electrode placement should be accelerated and simplified.

One solution to accelerate electrode placement is the Myo armband, designed for the lower arm. This device consists of eight electrodes embedded in an easy-to-use bracelet that is worn on the lower arm. This design allows for rapid donning and doffing. It has been extensively used in research on hand gesture classification: features are extracted from the measured EMG signals, which are used as input for various machine-learning methods [8–10]. However, while the bracelet is effective for the upper limb, it may slide down the leg during walking due to repetitive muscle contractions.

To overcome this challenge, Farina et al. [11] proposed an alternative approach, which involved using a garment embedded with four 5x5 high-density (HD) EMG grids. This garment can additionally be used to classify gestures, through the use of linear discriminant analysis. A garment could be better suited for lower limb use since it would not slide down the lower limb. Both the Myo armband and the garment successfully expedite electrode placement, potentially making them useful in exoskeleton applications. However, research on both devices focused on classifying trained gestures rather than extracting muscle-specific sEMG signals. These muscle-specific signals, appearing 10 ms before movement, are useful for exoskeletons to provide timely support [12].

Ohiri et al. [13] created a garment using embedded electrodes at known muscle locations to extract EMG envelopes. However, a manual check was needed to ensure the electrodes were on the muscles, after which placement on some calf muscles was still not sufficiently precise, causing inaccurate sEMG readings [13]. Generally, garments can shift during wear or after donning and doffing, causing electrodes and muscles to misalign, thereby reducing sEMG quality. Additionally, anatomical differences between subjects complicate the design of a garment that fits all subjects. This could lower sEMG quality for some subjects [14] One attempt to address these issues involved drawing the electrodes directly on the skin, eliminating shifting and conforming to individual muscle shapes [15]. However, this method is even more time-consuming than using the SENIAM guidelines for electrode placement.

To solve the issues regarding variety in muscle location, Simonetti et al. [16] developed an electrode-embedded garment, covering the entire lower leg. This electrode-embedded garment was comfortable and easy-to-wear. Instead of targeting specific muscles, electrodes were distributed uniformly around the lower leg. Non-negative matrix factorization (NNMF), a clustering method that enforces positive results only, identified clusters of electrodes with similar activation patterns. The clusters were assigned to muscles using a lower leg-specific algorithm. This method produced consistent muscle-specific EMG envelopes, across various movement tasks, without targeting specific muscles with electrode placement. However, a disadvantage lies in the algorithm used to assign clusters to muscles. It was created for lower leg muscles; a new algorithm would have to be created for different sets of muscles. This requires time and effort from researchers. Furthermore, when the garment is worn on a new day, electrodes may be in a slightly different position relative to the muscles, causing incorrect readings.

The EMG signals measured using the electrode-embedded garment need different processing methods to address electrode shift issues and to facilitate adaptations to multiple muscle sets. Analyzing signals from numerous uniformly distributed electrodes is already achieved in HD-EMG, which uses multiple closely spaced electrodes on a specific muscle [17]. Methods used to analyze HD-EMG may also apply to the analysis of EMG measurements made from an electrode-embedded garment such as the one from Simonetti et al., as both comprise many EMG electrodes. Analysis of HD-EMG was done by Yu et al. [18] using a convolutional neural network (CNN), which employs convolutional layers to extract features, and pooling layers to retain the most important ones [19]. They used this CNN to determine wrist torques [18]. Similarly, Simpetru et al. [20] used convolutional layers on three HD-EMG electrode grid recordings to determine hand joint angles.

Yu et al. and Simpetru et al. both demonstrated a deep learning network consisting of convolutional layers that successfully analyzed HD-EMG signals. Such networks may also work for large numbers of electrodes that are spaced further apart. Both studies used a regression network, which provides numerical outputs [21]. A regression network is also needed to find muscle-specific envelopes, as the envelopes are numerical values. Regression deep learning networks, comprising convolutional layers, have not previously been used to find muscle-specific EMG envelopes, but the works on HD-EMG signals show its potential, which will be explored in this research.

To summarize, there is a need for accelerated EMG electrode placement. Current solutions do address the placement of electrodes by using electrode-embedded garments. However, the possibility of issues caused by electrode shifts is still present and the used methods are not yet easily adaptable to multiple muscle sets. Considering these needs and research gaps, the central research question considered here is: How can a deep learning network with convolutional layers extract muscle-specific EMG envelopes for lower leg muscles from a garment with uniformly distributed electrodes and how can such a network be easily adapted to function for other muscle sets?

To answer this question, two types of networks were developed to test the feasibility of using deep learning networks for more widely spaced electrode grids to identify muscle-specific EMG envelopes: a regression CNN and an encoder-decoder network (EDN). Both use the same input and output, namely the measurements from numerous electrodes and the muscle-specific EMG envelopes. The EDN comprises an encoder, which uses convolutional and pooling layers to scale down the input while retaining key features, and a decoder, which uses transposed convolutional layers to scale the image size back up. In between is a bridge, where only the key features necessary to create the output remain [21].

Two datasets will be used to test these networks. The first dataset, recorded by Simonetti et al. [16], consists of recordings from seven movement tasks from eight subjects. The second was recorded for this research using a newly designed electrode-embedded garment for the upper leg, involving measurements with three subjects. The recorded EMG signals from both datasets will be analyzed using the NNMF-based method obtained from Simonetti et al. in addition to the newly created deep learning networks. Afterward, the quality of the resulting muscle-specific EMG envelopes of all three methods will be compared.

III. Methods

In this section, firstly, the three methods used to extract muscle envelopes; CNN (III-A), EDN (III-B), and NNMF (III-C) will be explained. Secondly, the two datasets (one for the lower leg (III-D), the other for the upper leg (III-E)) on which the analysis took place are explained: used subjects and the tasks each subject had to perform. Also, the created garment for the upper leg will be presented (III-E1). Thirdly, the preprocessing of the acquired data (III-E4), and then the analysis using CNN, EDN, and NNMF is explained (III-F). Lastly, the analysis to determine the best method is explained (III-G).

As mentioned, EMG envelopes are estimated using a CNN, an EDN, and NNMF-based clustering. Each method follows a similar process: data is collected from a healthy individual using an electrode-embedded garment on either the upper or lower leg. After collection, the data is preprocessed into EMG envelopes and split into ”training” and validation data. The training of the networks and the NNMF clustering takes place on the training data. Finally, the trained networks or identified clusters (in the case of NNMF) are applied to the validation data to assess the quality. The entire process is graphically shown in the flowchart in Fig. 1.

A. Convolutional Neural Network

A CNN consists of convolutional layers that extract features and pooling layers that only keep the most important features. The architecture of the CNN used in this study is shown in Fig. 2. It starts with an image of a certain size, each pixel in this image represents the value of the EMG envelope measured by one of the electrodes at a certain point in time. In the present study, the image size is 8x8, but this could easily be changed to a higher number. The network further consists of the following layers:

- Convolutional layer, filter size 2x2 with padding, filter amount of 40
- Batch normalization layer and Sigmoid activation layer
- Max-pooling layer with size 2x2 and stride 1x1
- Convolutional layer, filter size 2x2 with padding, filter amount of 40
- Batch normalization layer and Sigmoid activation layer
- Max-pooling layer with size 2x2 and stride 2x2
- 2 Fully connected layers of 1x1x16
- Fully connected layer of 1x1x5

The training input was provided by the measured EMG signal from the garment, while the training output was provided by manually selected electrodes from upper/lower leg muscles.

B. Encoder-Decoder Network

Similarly to the CNN, the EDN consists of convolutional and pooling layers. However, EDN also uses transposed convolutional layers, enabling sizing up. The architecture of the EDN used in this study is shown in Fig. 3. The input is again an image of size 8x8. All layers are shown below: all (transposed) convolutional layers followed by a rectified linear unit activation layer:

- Encoder
  - Convolutional layer, filter size 2x2 with padding, filter amount of 40
  - Max-pooling layer with size 2x2 and stride 2x2
  - Convolutional layer, filter size 2x2 with padding, filter amount of 40
  - Max-pooling layer with size 2x2 and stride 2x2
  - Convolutional layer, filter size 2x2 with padding, filter amount of 40
  - Max-pooling layer with size 2x1 and stride 1x1
- Bridge
  - 2 convolutional layers, filter size 2x2 with padding, filter amount of 40
- Decoder
  - 3 consecutive transposed convolutional layers, filter size 1x2 with padding, filter amount of 40
  - Fully connected layer of 1x1x5

The training output and input were the same as those of the CNN.

C. EMG clustering using NNMF

The process of EMG clustering using NNMF consists of the following four steps.

1) A similarity map is created for each electrode using k-nearest neighbor clustering and Euclidean distances. This indicates the resemblance of the sEMG signal between electrodes.
2) Two filters are applied to the similarity map of each electrode to retain exclusively the values for electrodes with spatial proximity to the electrode in question, setting the rest to zero. One filter is based on how close electrodes were to the electrode in question, which had to be no more than two electrodes apart; the other filter ensures that the electrodes had to be adjacent to each other, for them to be on top of the same muscle.
3) All similarity maps are combined into one large matrix, to which NNMF is applied to find clusters of closely related electrodes.
4) These clusters are assigned to specific muscles using a designated algorithm.

D. Experimental data lower leg

Measurements made by Simonetti et al. [16] were used as the dataset for the lower leg. The methods used to record this data are explained in the article by Simonetti et al. (2022) [16]. Measurements are used from eight healthy subjects, with recorded movement tasks detailed in table I. During each movement task, 64 electrodes continually measured sEMG signals. Additionally, two electrodes were manually selected according to the SENIAM guidelines on top of each of the following muscles:

- Tibialis anterior (TA)
- Peroneus longus (PL)
- Gastrocnemius medialis (GM)
- Gastrocnemius lateralis (GL)
- Soleus (SO)

Preprocessing of the recorded EMG signals and the manually selected electrodes was performed as described by Simonetti et al. (2022) [16].

E. Experimental data upper leg

For the upper leg, new data was recorded. A description of the electrode-embedded garment is given below, followed by an explanation of the experimental procedures, the manual selection method, and the preprocessing steps.

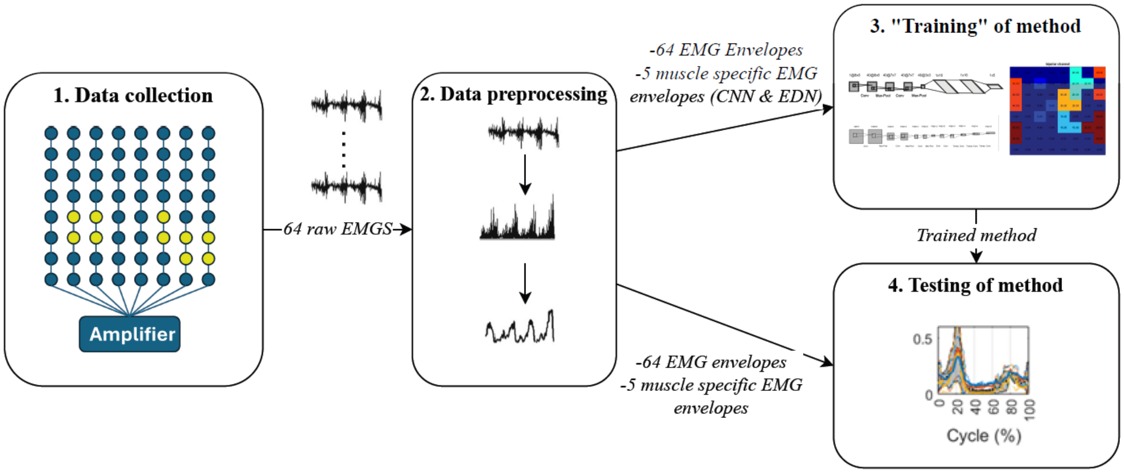


Fig. 1: Flowchart of the general method used to process all data. ***Box 1)*** *A simplified diagram of data collection. The circles represent electrodes, with yellow circles indicating manually selected electromyography (EMG) electrodes. The manual selection process is explained in detail in section III-E3.* ***Box 2)*** *The preprocessing steps consist of filtering the EMG data until 64 EMG envelopes remain. This occurs separately for the manually selected EMG channels, after the filtering steps 5 EMG envelopes remain.* ***Box 3)*** *The 'training': for the convolutional neural network (CNN) and encoder-decoder network (EDN) this consists of training the deep learning network. For the NNMF training consists of finding the muscle clusters.* ***Box 4)*** *For validation, the EMG envelopes found by the CNN, EDN, or NNMF-based clustering were averaged over gait cycles and compared with manually selected EMG envelopes.*

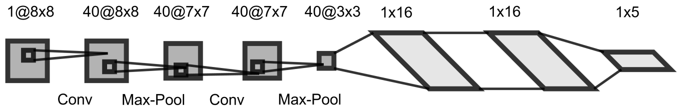


Fig. 2: Architecture of the convolutional neural network. All filters are set to 1 for visual simplicity. In reality, 40 filters were present. Conv stands for convolutional layer, Max-Pool for max-pooling layers.

TABLE I: List of the recorded movement tasks that are part of the lower leg dataset.

Task	Acronym	Recording time (s)
Forward locomotion		
1 km/h	FW1	35
3 km/h	FW3	30
5 km/h	FW5	25
Backward locomotion		
1 km/h	BW1	30
3 km/h	BW3	25
Running at 7 km/h	RN	25
Sidestep	SS	40

TABLE II: List of the recorded tasks for the upper leg. SD stands for 'same day', and ND for 'new day'. All recordings lasted 30 seconds. FW3 was recorded twice: once for training, and the other for validation.

Day 1	
Session 1	
Forward locomotion	
1.8 km/h	FW1.8
3 km/h (twice)	FW3
4.2 km/h	FW4.2
3 km/h: stiff knee gait	SK3
Backward locomotion	
1.8 km/h	BW1.8
Side lunge	SL
Session 2	
Forward locomotion	
3 km/h	SD1
Session 3	
Forward locomotion	
3 km/h	SD2
Day 2	
Forward locomotion	
3 km/h	ND

1) Electrode embedded wearable garment: An electrode-embedded garment was created to simplify the placement of the 64 electrodes on the leg. It consists of two main parts. Firstly, eight grids of eight electrodes (10 mm in diameter) were screen-printed from Ag/AgCl. These grids were made of a 26 cm long flexible printed circuit board (PCB material). Custom cables were created to connect these electrode grids to the SAGA manufactured by TMSi [22]. Secondly, a pair of thermal leggings (Thermobroek voor skiën Dames BL 100 zwart, Decathlon, The Netherlands) was used. The electrodes were attached to these, using small Velcro stickers and, by making small cuts electrode grids were weaved through the leggings. The electrode-embedded wearable garment is shown in Fig. 4.

2) Experimental procedures: Data was recorded from 3 healthy subjects (age = 24± 2 years, height = 179 ± 4 cm, weight = 80±10 kg, 1 male, 2 female). Recordings were performed on two separate days, see table II for all recordings.

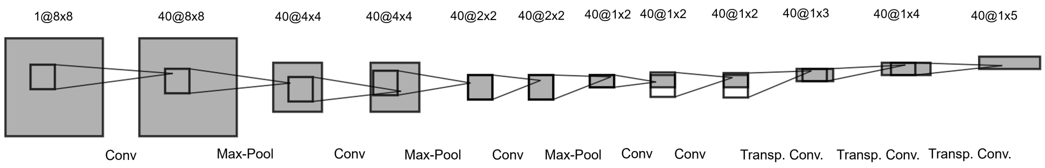


Fig. 3: Architecture of the encoder-decoder network. All filters are set to 1 for visual simplicity. In reality, 40 filters were present. Conv stands for convolutional layer, Max-Pool for max-pooling layers, and Transp. Conv. for transposed convolutional layers.

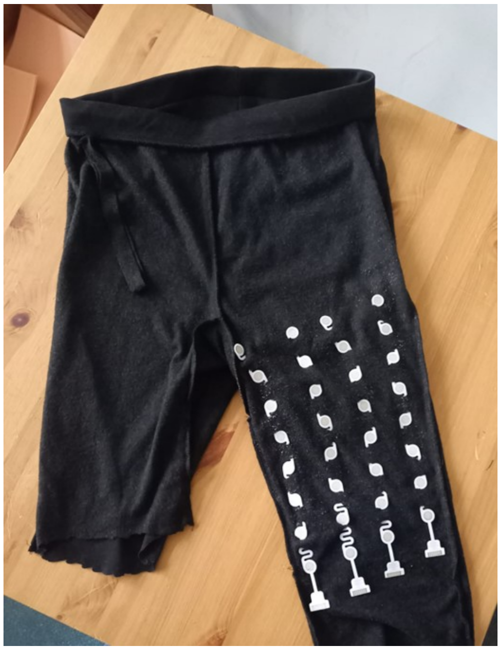


Fig. 4: The wearable electrode-embedded garment. Only the backside is shown, but electrodes are also present at the front. The leggings are cut short to increase subject comfort.

This research was approved by the Natural Sciences & Engineering Sciences (NES) Ethics Committee of the University of Twente with reference number 240046. Each subject signed an informed consent form. Firstly, the right upper leg of the subject was prepared for wearing the garment: NuPrep gel was used to scrub the right upper leg and salt water was applied to wet the leg to improve the electrode impedance. The created 8x8 electrode grid-embedded garment was donned, and electrode locations for specific muscles were noted according to the SENIAM guidelines. The patella was also prepared with NuPrep gel and a gel electrode was attached, functioning as a ground electrode. Reflective markers were placed on both shoes of the subject at the heel and the first toe. Marker data was recorded at 100 Hz using optical motion capture cameras (Qualisys Oqus, Sweden). EMGs were recorded at 2048 Hz using a multi-channel amplifier (SAGA, TMSi, The Netherlands).

3) Manual selection of muscle electrodes: Real muscle EMG envelopes were needed to train the deep learning networks and to validate the results of the deep learning networks and the NNMF-based clustering method. Two electrodes on top of each muscle according to the SENIAM guidelines were selected, from which muscle-specific EMG envelopes were created as described in section III-E4 [6]. The locations of muscles for subject 1 are shown in figure 5.

The following muscles were manually selected:

- Semitendinosus (ST)
- Biceps femoris (BF)
- Vastus lateralis (VL)
- Rectus femoris (RF)
- Vastus medialis (VM)

These specific upper leg muscles were selected as they are all present in the SENIAM guidelines, in addition to having a large influence on knee torque [6].

4) Data processing: All data processing was performed using MATLAB (Matlab2022b, MathWorks, Natick (MA), USA) software.

Kinematic data: The data collected from the reflective markers was low-pass filtered at 6 Hz with a second-order Butterworth filter. Amplification of EMG data: Raw EMG data was amplified with a gain of 23 (SAGA 64+, TMSi, The Netherlands). Re-referencing: The means of the electrodes in columns 1-4 and 5-8 were calculated, excluding noisy electrodes (with standard deviation larger than 5 times the median standard deviation of all electrodes), or silent electrodes (with standard deviation smaller than 0.001 mV). The mean in columns 1-4 was subtracted from all electrode measurements in these columns, and the same was done in columns 5-8. Manual selection: Following the re-referencing step, the two manually selected electrodes from each muscle were subtracted. EMG envelopes: All EMG signals (re-referenced signals of the 64 EMG electrodes and the 5 resulting signals after subtracting the manually selected electrodes) were high-pass filtered at 20 Hz using a second-order Butterworth filter. Afterward, the signals were full-wave rectified and low-pass filtered at 6 Hz using a fourth-order Butterworth filter. Normalization: The EMG envelopes of each session were normalized separately. The maximum value across all tasks per session was determined for each electrode/ muscle and normalization was performed using these values. Reconstruction of missing electrodes: Six electrode recordings were constantly equal to zero, potentially due to a faulty cable. Bilinear interpolation was performed using the four neighboring electrodes to reconstruct the electrode recordings.

F. Training of networks and performing NNMF clustering

For the lower leg, FW1 was used to train the CNN and EDN and to perform NNMF-based clustering. FW1 was chosen as Simonetti et al. used this because it worked best for lower leg clustering. It was used for CNN and EDN as well to be consistent. For the upper leg, FW3 was used. This movement task was chosen because this was the most comfortable walking speed, so this would be the most suitable for future use.

1) CNN and EDN: The deep learning networks were trained using the 64 EMG envelopes, restructured to an 8x8 grid, as input and the 5 muscle-specific EMG envelopes from the manual selections as output. Both the upper and lower leg networks were trained in the same way. The trained networks were then used to predict the output for all other tasks and the other recordings of FW3 or FW1. The training was done with 1 epoch, mini-batch size 32, and an Adam optimizer.

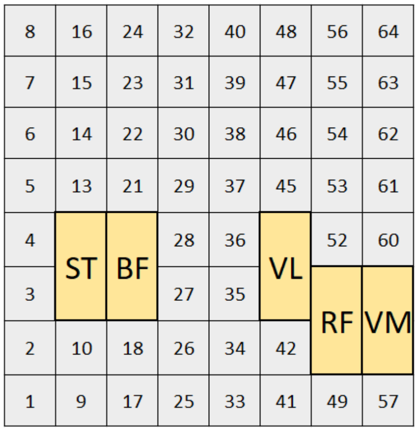
Only the encoder part was also run for the EDN to find muscle synergies as key components from which to build the muscle-specific EMGs. This is explained in more detail in appendix C

The electrodes that were the most important for each muscle output were determined by calculating the gradient of each muscle output for each electrode input. This was done by adding a value of 1e-6 to each electrode one by one for each point in time during a gait cycle and then recording how much change this added value to each electrode caused in each muscle output. The larger the change, the more important the electrode was for that muscle output: the two electrodes that invoked the largest change were selected for each muscle.

2) Clustering using NNMF: The NNMF clustering was performed using three gait cycles (right heel strike to the next right heel strike) of the 64 EMG envelopes of the same tasks used for deep learning training. The steps referred to below are explained in section III-C. There were some differences between the upper and lower leg in the second and fourth steps of the process.

In the second step, filters were applied to the similarity maps to retain electrode values close to each electrode. This filter included all eight closest neighbors for the lower leg, while only the two electrodes above and the two below were included for the upper leg. Figure D.1 in appendix D shows the upper leg mask. This distinction was made, as the surface area of the upper leg was larger, causing a larger space between electrodes; electrodes from different columns were never on the same muscles.

For the fourth step, which consisted of an algorithm to appoint identified clusters to muscles, the lower leg algorithm was copied from Simonetti et al. (2023) [22]. For the upper leg, a new version was created. The electrodes with numbers 1 through 32 were always on the back of the leg, with number 1 being on the medial side. This ensured that the ST muscle was always associated with the cluster with the lowest electrode numbers. The rest of the sequence follows the anatomy of the upper leg muscles: ST, BF, VL, RF, and VM. Fig. 5 shows the typical locations of muscles in the electrode grid.



8	16	24	32	40	48	56	64
7	15	23	31	39	47	55	63
6	14	22	30	38	46	54	62
5	13	21	29	37	45	53	61
4	ST	BF	28	36	VL	52	60
3			27	35		RF	VM
2	10	18	26	34	42		
1	9	17	25	33	41	49	57

Fig. 5: Locations of muscles in the 8x8 electrode grid. ST = semitendinosus, BF = biceps femoris, VL = vastus lateralis, RF = rectus femoris and VM = vastus medialis

The clusters that result from NNMF include a few electrodes per muscle and their corresponding weights. These electrode EMG envelopes, multiplied by their associated weight, were added, resulting in muscle-specific EMG envelopes. This process was repeated for all other tasks and the other recordings of FW3 or FW1.

G. Validation procedures

The manual, CNN, EDN and NNMF-based clustering EMG envelopes of all tasks and muscles were gait-cycle averaged using gait cycles ranging from heel strike to heel strike. For the upper and lower leg results, R^2 values were calculated for all tasks between the averaged EMG envelopes derived from manually selected electrodes and those derived using CNN, EDN, or NNMF-based clustering. Using these R^2 values, box plots were created to compare performance, after which statistical analyses were performed to determine whether there were significant differences in performance. Specifically, for the lower leg, the following comparisons of performance are made:

- Between CNN, EDN and NNMF overall
- Between methods on muscles
- Between methods on tasks
- Between tasks within methods

For the upper leg, the following comparisons are made:

- Between methods overall
- Between methods on muscles
- Between methods and within methods on tasks of the same session
- Between methods and within a method on tasks of new sessions

Additionally, for the upper leg, the percentage of R^2 values across all muscles and tasks above 0.7 is calculated for each method.

Lastly, the Euclidean distances between the average location of the manually selected electrodes and the electrodes identified for each muscle using NNMF-based clustering and CNN and EDN (the top two electrodes according to the gradients) were calculated and compared.

IV. Results

A. Lower leg

All gait cycle-averaged EMG envelopes for Subject 1 are shown in Fig. 6. Generally, the estimated EMG envelopes remain within the manually selected EMG envelopes. Some exceptions include the PL muscle for BW3 where all methods skip the peaks and the underestimation of GM envelope peaks for the FW trials. Detailed statistical comparisons are provided in Appendix B. All statistical comparisons were performed using a Kruskal-Wallis test, followed by a multiple comparison test with Dunn-Sidák critical value, to account for the non-normality of the data. A summary of key results is described below, comparing the used methods overall. A comparison of performance between muscles and methods is given and lastly, the performance between tasks and methods is compared.

Firstly, Fig. 7a shows the R^2 value comparisons for methods, taking the median of tasks and muscles for all subjects. The median R^2 of CNN (0.90) and EDN (0.90) are similar (p=0.99), while the median R^2 of NNMF (0.80) significantly differs from those of CNN ($p<0.01$) and EDN ($p<0.01$).

Secondly, Fig. 7b compares the performance of the methods across different muscles, combining R^2 values from all tasks and subjects. The following observations were made regarding the performance of different methods on the muscles:

- The medians R^2 values were higher for plantar flexor (PF) muscles than for dorsiflexor (DF) muscles for CNN (PF median=0.93, DF median=0.86, $p<0.01$) and EDN (PF median=0.92, DF median=0.81, p=0.05).
- For the GL muscle, the median R^2 value of NNMF (0.64) is significantly lower than those of CNN (0.92, $p<0.01$) and EDN (0.93, $p<0.01$).
- For the GM muscle, the median of R^2 values of NNMF (0.96) is significantly higher than that of EDN (0.90, p=0.03).
- No significant differences exist for the TA, PL, and SO muscles.

Thirdly, Fig. 7c compares the performance of the methods across different tasks, combining R^2 values from all muscles and subjects. Differences can be observed in two ways: within a task or a method. Firstly, there was no significant difference in how well NNMF, CNN, and EDN performed on all tasks, for example, NNMF did not perform better on FW1 than on RN or SS tasks. There were some differences when comparing how well each task was performed by the methods. Significant differences were observed for FW3: NNMF (0.79) scores significantly lower than CNN (0.90, p=0.03), FW5: NNMF (0.67) scores significantly lower than CNN (0.88, $p<0.01$), and for BW3: the NNMF median (0.81) was significantly lower than that of CNN (0.90, p=0.02) and EDN(0.90, p=0.03).

Lastly, the Euclidean distances between the manually selected electrodes and those identified by NNMF-based clustering, CNN, and EDN are compared in figure E.1 for muscles in Appendix E and figure 8 for subjects. The only significant difference was found for subject 1, where CNN (2.69) has significantly larger median distances than NNMF (1.58, p=0.050) and EDN (1.50, p=0.01). It can be seen that the distances vary greatly. The locations of muscles in the electrode grid are shown in figure E.2 in Appendix E.

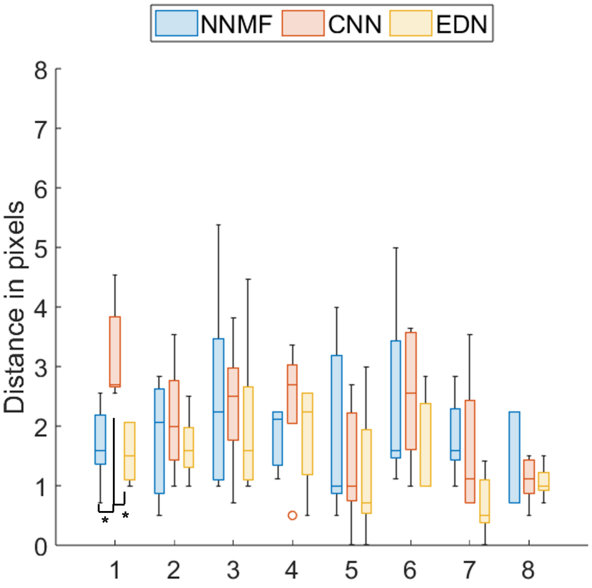


*Fig. 8: A box plot is shown of Euclidean distances in pixels (with one pixel representing the distance between two electrodes) on the y-axis, calculated between the manually selected electrodes of lower leg muscles and those selected by non-negative matrix factorization clustering, convolutional neural network, and encoder-decoder network. The numbers shown on the x-axis indicate subjects. The horizontal lines indicate median values, calculated between all muscles. The bottom and top edges indicate the 25th and 75th percentiles and the whiskers extend to the most extreme values not considered outliers. Circles indicate outliers. *indicates that p<0.05.*

B. Upper leg

Due to the low number of test subjects (n=3), no significant results can be reported. However, repeatability was assessed by using data from more than one subject. Gait cycle-averaged EMG envelopes for Subject 1, containing all tasks and muscles, are shown in Fig. 9.

In Fig. 10 all values of R^2 from all subjects, movement tasks except sessions, and muscles, are divided into the following categories [23]:

- <0.2: very weak

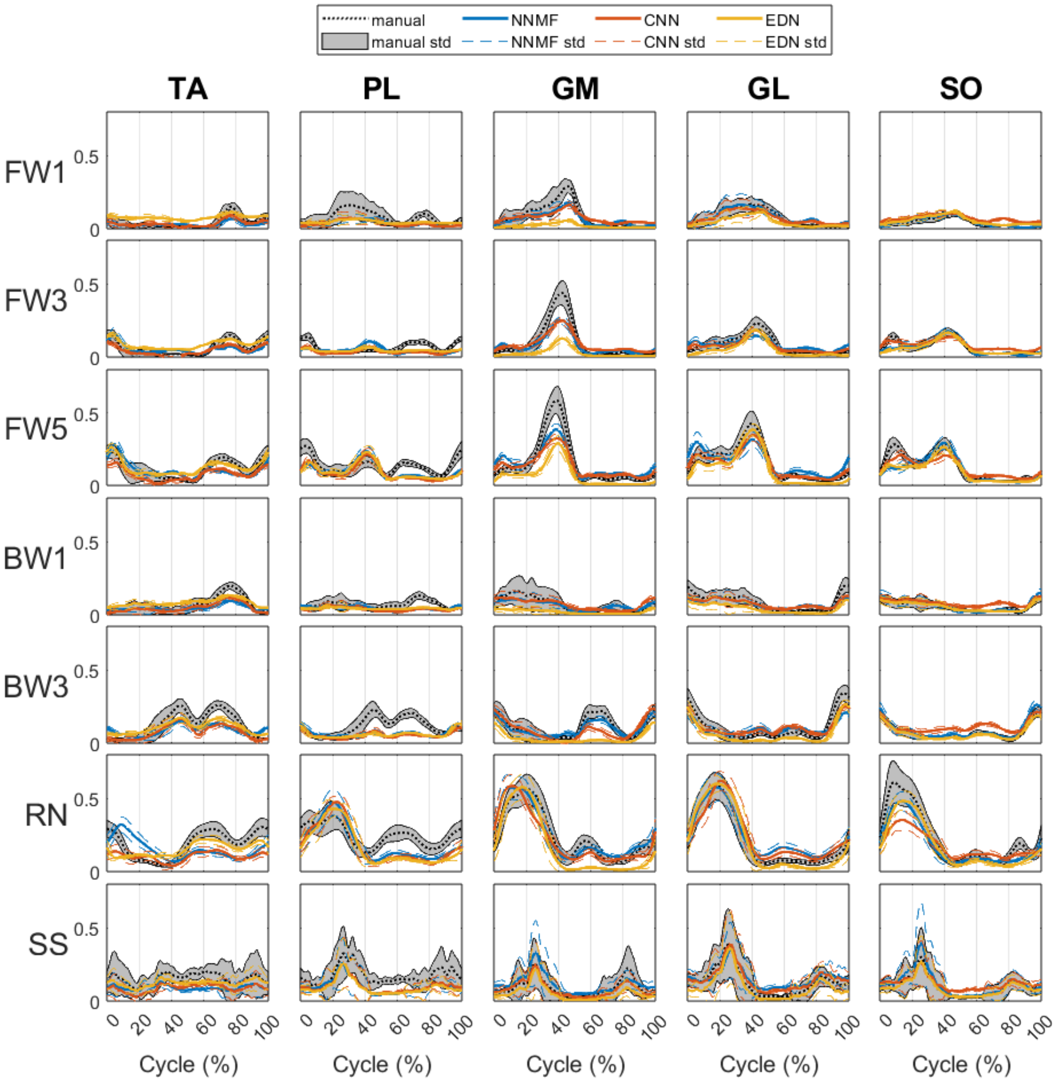


Fig. 6: Gait cycle-averaged electromyography (EMG) envelopes for five lower leg muscles across all tasks for subject 1. Gait cycles were defined as right heel strike to the next right heel strike. EMG envelopes were created using manual selection, the convolutional neural network (CNN), encoder-decoder network (EDN), and non-negative matrix factorization (NNMF) clustering. The manually selected EMG envelope plus and minus the standard deviation is shown in light gray. The standard deviation of CNN, EDN, and NNMF are shown in dashed lines of the same colors.

- 0.2-0.39: weak
- 0.4-0.59: moderate
- 0.6-0.79: strong
- >0.8: very strong

It can be seen that EDN has the most R^2 values in the 'very strong' category, while NNMF has the least. NNMF has the most R^2 values in the 'very weak' category. Table B.5 in appendix B2 shows the median R^2 values taken per subject for each movement task and muscle.

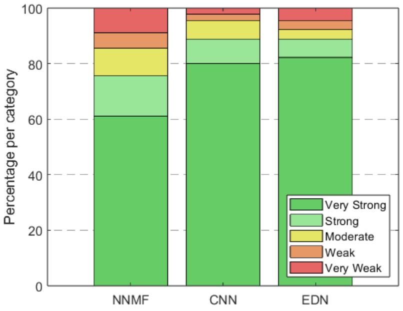


Fig. 10: R^2 values are divided up into categories: very weak (<0.2), weak (0.2-0.39), moderate (0.4-0.59), strong (0.6-0.79), and very strong (>0.8). R^2 values were taken from movement tasks FW1.8, FW3, FW4.2, SK, and SL, all muscles, and all subjects for the upper leg.

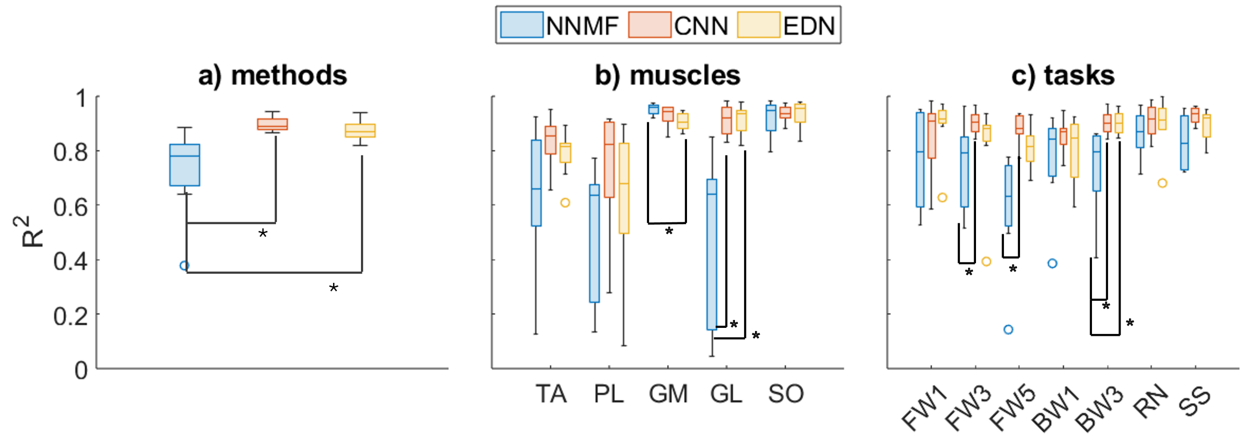


*Fig. 7: In all subfigures box plots are shown of R^2 values calculated between the gait-cycle averaged manually selected electromyography envelopes of lower leg muscles and non-negative matrix factorization clustering, convolutional neural network, and encoder-decoder network. The y-axis represents the R^2 values. The horizontal lines indicate median values, the bottom, and top edges indicate the 25^{th} and 75^{th} percentiles, and the whiskers extend to the most extreme values not considered outliers. Circles indicate outliers. *indicates that p<0.05. In (a) the performance of the methods is compared overall. The median is taken for all tasks and muscles, and the resulting values per subject are shown. In (b) the performance of the methods on muscles is compared. The median across all tasks is taken, and the resulting values per subject are shown. In (c) the performance of the methods on the tasks is compared. The median across all muscles is taken, and the resulting values per subject are shown.*

Fig. 11 shows similar categories, however here the performance on sessions is compared between the methods. It can be seen that all methods perform best on FW3, but performance on SD and ND is not clearly different. For CNN and EDN the percentage in the 'very strong' category is similar between SD and ND, for NNMF the amount in the 'very strong' category goes down from 60% to 40%. However, all methods see an increase of values in the 'very weak' category for SD compared to ND. For CNN and EDN, 80% of R^2 values for ND are in either the 'very strong' or 'strong' category, for NNMF this is 60%.

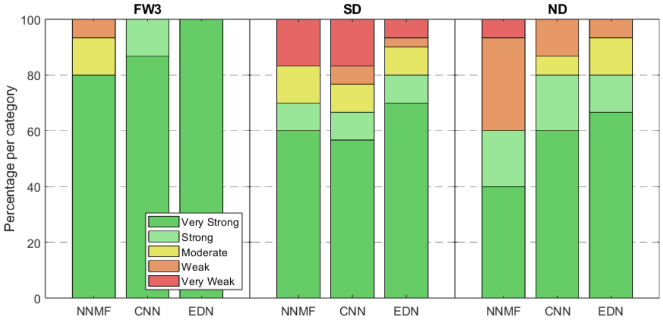


Fig. 11: R^2 values are divided up into categories: very weak (<0.2), weak (0.2-0.39), moderate (0.4-0.59), strong (0.6-0.79), and very strong (>0.8). Each subplot shows R^2 values taken from one of the sessions: FW3, SD, and ND, which are mentioned at the top of each subplot. All muscles and all subjects are combined for each session.

Fig. 12 shows the performance of the methods on each subject separately. All movement tasks excluding sessions are used. It can be seen that performance of CNN and EDN is comparable for subjects 1 and 3, but less for subject 2. NNMF performance differs for each subject, with performance on 1 being best and 3 being worst.

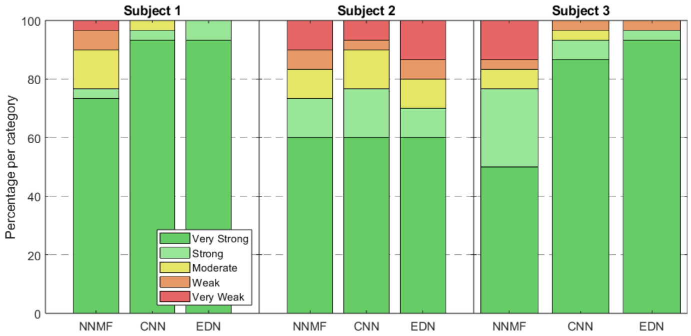


Fig. 12: R^2 values are divided up into categories: very weak (<0.2), weak (0.2-0.39), moderate (0.4-0.59), strong (0.6-0.79), and very strong (>0.8). Each subplot shows percentages of R^2 values from one of the subjects, which are mentioned at the top of each subplot. All muscles and the movement tasks FW1.8, FW3, FW4.2, SK, and SL are combined per subject.

Lastly, the Euclidean distances between the manually selected electrodes and those identified by NNMF-based clustering, CNN, and EDN are compared in figure E.3 for muscles and in figure E.4 for subjects, both in Appendix E. No significant differences can be reported, but the distances do vary greatly. The locations of muscles in the electrode grid are shown in figure E.5 in Appendix E.

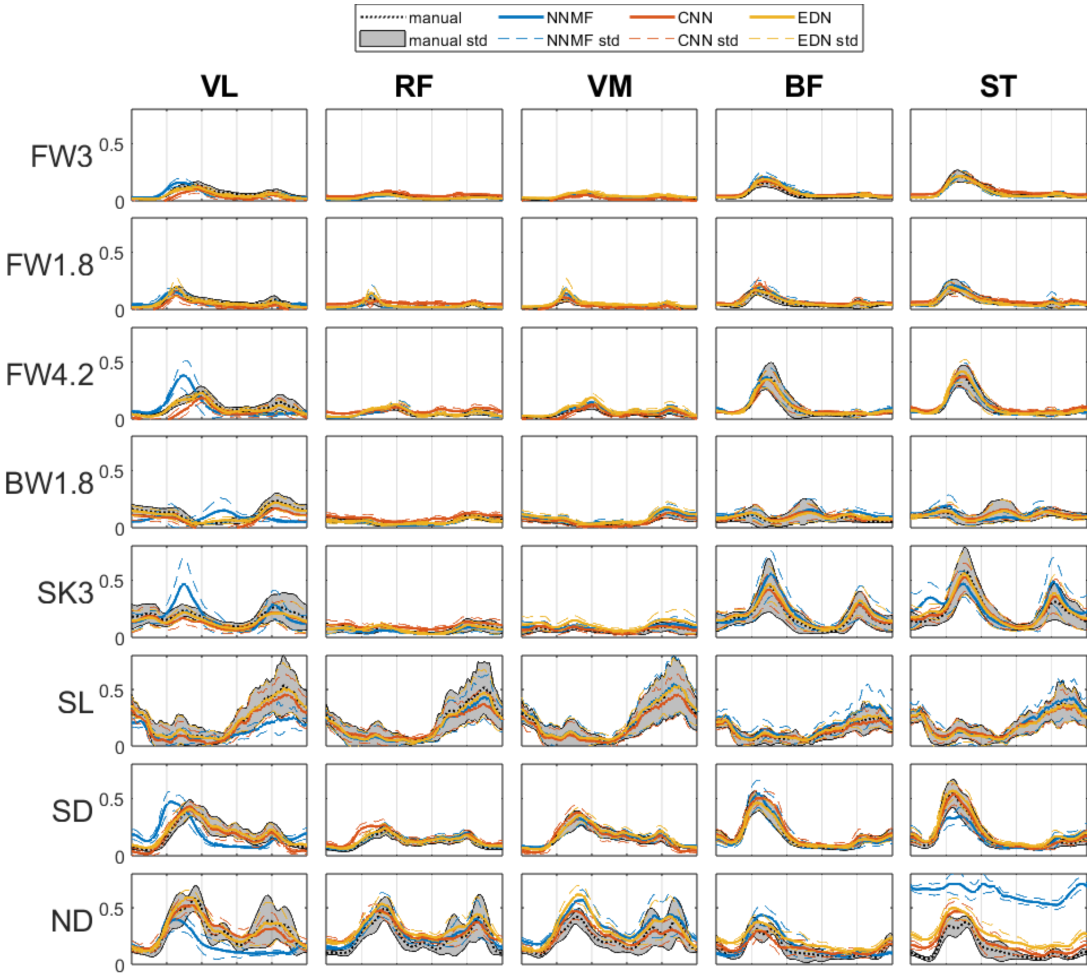


Fig. 9: Gait cycle-averaged electromyography (EMG) envelopes for five upper leg muscles across all tasks for subject 1. Gait cycles were defined as right heel strike to the next right heel strike. EMG envelopes were created using manual selection, the convolutional neural network (CNN), the encoder-decoder network (EDN), and non-negative matrix factorization (NNMF) clustering. The manually selected EMG envelope plus and minus the standard deviation is shown in light gray. The standard deviation of CNN, EDN, and NNMF are shown in dashed lines of the same colors.

V. Discussion

Two methods, a CNN and an EDN, were developed to extract muscle EMG envelopes for the lower leg, using a garment previously created by Simonetti et al. (2023) [22]. These methods, along with NNMF-based clustering, were adapted for the upper leg, using a newly developed garment consisting of thermal leggings with an embedded 8x8 electrode grid. The CNN and an EDN could be adapted without altering their architecture, while NNMF-based clustering needed changes in the cluster to muscle algorithm and in the locality masks.

For the lower leg, CNN and EDN outperformed NNMF, indicating their potential for producing ankle torques, as previously demonstrated with NNMF [16]. Furthermore, extraction of EMG envelopes for the dorsiflexor (TA) was less successful than for plantar flexors. Simonetti et al. [16] hypothesized this was due to the smaller surface area of TA and that increasing electrode density might address this issue. Surprisingly, very few significant differences were observed between R^2 values of tasks. Maybe training on one training epoch successfully prevented overfitting the EDN and CNN. For NNMF it may be because the garment did not move in between tasks, so clusters also remained roughly in the same position relative to the muscles.

For the upper leg, Fig. 9 shows that CNN and EDN EMG envelopes generally matched manually selected EMG envelopes and over 80% of R^2 values for CNN and EDN were in the 'very strong' category, suggesting that CNN and EDN can accurately identify muscle-specific EMG envelopes. NNMF had about 20% fewer 'very strong' R2 values, indicating CNN and EDN may outperform NNMF, though more subjects are needed to confirm this. Performance was lower in new sessions (SD and ND), highlighting a need for solutions to improve inter-session performance. consistency. This will be further elaborated on in section V-A. Testing on three subjects revealed that even with more noise (found present in Subject 2), over 70% of $R2$ values for Subject 2 were in the 'strong' or 'very strong' categories.

For both the upper and lower leg a large variance in distance exists between the manually selected electrodes and those selected by NNMF-based clustering, CNN, and EDN. Despite this large spread in distances, the resulting EMG envelopes represent the manually selected EMG envelopes well. This might be caused by the fact that the surface area of muscles is usually larger than just the locations indicated by the SENIAM guidelines. So despite the distance between the electrodes, the measured signals might be similar.

When comparing CNN and EDN with the NNMF framework created by Simonetti et al. [22], adapting the former to a new set of muscles is easier as no changes to the architecture of the networks are required. In contrast, the NNMF method required a new algorithm to assign clusters to muscles. Additionally, NNMF-based clustering needs precise garment positioning for these algorithms to function. On the other hand, CNN and EDN require labeled training data, also necessitating knowledge of garment orientation during training. The need for training data will be further elaborated on in section V-A.

The garment created for the upper leg was easy and quick to don, offering advantages over methods like Ershad et al.'s [15] ink-drawing electrodes, which are time-consuming to apply and require extensive anatomical knowledge. A comparison can also be made to the Myo bracelet, even though it was designed for the upper limb. The Myo bracelet is arguably easier to put on, but for the lower leg such a bracelet would not be feasible due to the different heights of locations of muscles. Additionally, it has a risk of sliding down during walking, while the thermal leggings and the lower-leg garment do not have this issue.

A. Limitations and future perspective

The primary limitation is the need to create labeled training data, which is time-consuming and requires professionals to follow the SENIAM guidelines to perform manual selection. To mitigate this, transfer learning can be used, which involves training a machine learning model on several tasks or samples and then adapting it to a specific task or sample [24]. This approach can be applied by training the networks on a large number of subjects to create a generic network, which can then be fine-tuned for specific subjects. Transfer learning has been successfully used by Zhang et al. [25] for inter-subject prediction of ankle joint torque from surface EMG signals using an NMS solver-informed ANN. The network was trained on nine subjects, after which transfer learning was performed to fit the model to a new subject. Also, Liu et al. [26] demonstrated its potential by predicting 3D hand poses through measurements using the Myo bracelet and recurrent neural networks. The network was trained on just one subject, after which minimal training data was used to transfer it to a new subject.

A second option is exploring unsupervised machine learning techniques, such as autoencoders, which use the same input as output and are typically used for dimensionality reduction or learning features [27]. Autoencoders have been used to find muscle synergies from sEMG signals, but the identified signals still need to be assigned to synergies based on known shapes [28–30]. Applying this to the current problem would require knowledge of typical EMG envelope shapes for proper assignment. Furthermore, the EMG envelopes of VL, RF, and VM muscles might look very similar, making it difficult for an autoencoder to separate all underlying muscle EMG envelopes.

Performance of all methods was lower for new sessions (ND and SD) than for FW3. Two methods could be considered to improve inter-session quality. Firstly, transfer learning can address this issue, as demonstrated by Ameri et al. [31], who improved inter-subject CNN classification accuracy by fine-tuning the network, which was trained with normal HD-EMG data, with 2,5 cm shifted HD-EMG signals. Similarly, Lee et al. [32] used transfer learning in hand gesture classification using a domain adversarial neural network. They used labeled data from a previous day and needed only unlabeled data from the target day to update the network. This way they eliminated the need for labeled training data for a new session and increased classification accuracy during a new session.

Another solution is using data augmentation, which is when a deep learning network is trained on input data with added noise or shifted input data [21]. Data augmentation has successfully been used in HD-EMG research. Firstly, Sun et al. [33] used a CNN to classify hand gestures, increasing classification accuracy from 20% (trained on the original HD-EMG data) to 84.6% (trained on augmented HD-EMG data). Secondly, Chamberland et al. [34] used augmented HD-EMG data to train their CNN to increase hand gesture classification accuracy by 25.67%. Both studies demonstrate how useful augmented training data is in classification tasks with HD-EMG, it must be researched whether this also applies to regression tasks with more widely spaced EMG electrodes.

When using the current garment, the legs must be shaved, scrubbed, and wetted with salt water to reduce impedance. Dry electrodes or advanced noise filtering could eliminate this requirement. Additionally, six electrodes showed zero readings during upper leg measurements, causing the need for reconstruction. This reconstruction may have had some influence on the results. The reason for the zeroed measurements must be found and the underlying problem must be addressed. A combined garment for the upper and lower leg should be created, as donning and attaching two separate garments is not practical. This new garment should include the gluteal muscles to extract hip torque. Lastly, electrodes in the garments do not provide adequate density at the TA muscle. Also, on the upper leg, the SENIAM locations of muscles are quite low, which means that electrode density must be increased in the lower part of the upper leg, and decreased in the upper part.

The manual selection procedure was never validated against electrode placement performed as described by the SENIAM guidelines. Validation should be conducted. The categories used for R^2 values of the upper leg are not specific for gait-cycle averaged EMG envelopes, but just a general way to assess correlation, meaning that the practical applicability is uncertain.

VI. Conclusion

Two new methods to extract muscle EMG envelopes for the upper and lower leg were developed: one based on a CNN, the other on an EDN. An NNMF-based clustering method, previously created for the lower leg, was adapted to the upper leg. The newly created CNN and EDN methods significantly outperformed the NNMF-based method for the lower leg. Due to low subject amounts, the results of the upper leg were not significant, but NNMF-based clustering did have more measurements in the weaker categories.

The distance between the locations of muscles found by manual selection and those found by NNMF-based cluster, CNN, and EDN, varied greatly, but resulting EMG envelopes generally resembled those found by manual selection.

For the upper leg, a new multi-electrode-embedded garment was created. Together with the previously created lower leg garment, this provides a good basis from which to make a full lower limb garment.

Future studies should focus on validating the manual selection method used and on eliminating the need for training data.

References

1. Al-Ayyad, M., Owida, H. A., Fazio, R. D., Al-Naami, B. & Visconti, P. Electromyography Monitoring Systems in Rehabilitation: A Review of Clinical Applications, Wearable Devices and Signal Acquisition Methodologies. *Electronics 2023, Vol. 12, Page 1520* **12,** 1520. doi:10.3390/ELECTRONICS12071520 (7 Mar. 2023).
2. Bruni, M. F. *et al.* What does best evidence tell us about robotic gait rehabilitation in stroke patients: A systematic review and meta-analysis. *Journal of Clinical Neuroscience* **48,** 11–17. doi:10.1016/J.JOCN.2017.10.048 (Feb. 2018).
3. Park, Y.-H., Lee, D.-H. & Lee, J.-H. A Comprehensive Review: Robot-Assisted Treatments for Gait Rehabilitation in Stroke Patients. *Medicina 2024, Vol. 60, Page 620* **60,** 620. doi:10.3390/MEDICINA60040620 (4 Apr. 2024).
4. Su, D., Hu, Z., Wu, J., Shang, P. & Luo, Z. Review of adaptive control for stroke lower limb exoskeleton rehabilitation robot based on motion intention recognition. *Frontiers in Neurorobotics* **17.** doi:10.3389/FNBOT.2023.1186175 (2023).
5. Garcia, M. A. C. & Vieira, T. M. M. Surface electromyography: Why, when and how to use it. *Revista Andaluza de Medicina del Deporte* **4,** 17–28 (1 2011).
6. Merletti, R., Hermens, H. & Kadefors, R. European community projects on surface electromyography. *Annual International Conference of the IEEE Engineering in Medicine and Biology-Proceedings* **2,** 1119–1122. doi:10.1109/IEMBS.2001.1020387 (2001).
7. Baud, R., Manzoori, A. R., Ijspeert, A. & Bouri, M. Review of control strategies for lower-limb exoskeletons to assist gait. *Journal of NeuroEngineering and Rehabilitation* **18.** doi:10.1186/S12984-021-00906-3 (1 Dec. 2021).
8. Vásconez, J. P., López, L. I. B., Ángel Leonardo Valdivieso Caraguay & Benalcázar, M. E. A comparison of EMG-based hand gesture recognition systems based on supervised and reinforcement learning. *Engineering Applications of Artificial Intelligence* **123,** 106327. doi:10.1016/J.ENGAPPAI.2023.106327 (Aug. 2023).
9. Côté-Allard, U. *et al.* Deep Learning for Electromyographic Hand Gesture Signal Classification Using Transfer Learning. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **27,** 760–771. doi:10.1109/TNSRE.2019.2896269 (4 Apr. 2019).
10. Sosin, I., Kudenko, D. & Shpilman, A. Continuous Gesture Recognition from sEMG Sensor Data with Recurrent Neural Networks and Adversarial Domain Adaptation. *2018 15th International Conference on Control, Automation, Robotics and Vision, ICARCV 2018,* 1436–1441. doi:10.1109/ICARCV.2018.8581206 (Dec. 2018).
11. Farina, D., Lorrain, T., Negro, F. & Jiang, N. High-density EMG E-textile systems for the control of active prostheses. *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology Society.* **2010,** 3591–3593. doi:10.1109/IEMBS.2010.5627455 (2010).
12. Sitole, S. P. & Sup, F. C. Continuous Prediction of Human Joint Mechanics Using EMG Signals: A Review of Model-Based and Model-Free Approaches. *IEEE Transactions on Medical Robotics and Bionics* **5,** 528–546. doi:10.1109/TMRB.2023.3292451 (3 Aug. 2023).
13. Ohiri, K. A. *et al.* E-textile based modular sEMG suit for large area level of effort analysis. *Scientific Reports 2022 12:1* **12,** 1–14. doi:10.1038/s41598-022-13701-4 (1 June 2022).
14. Etana, B. B. *et al.* A review on the recent developments in design and integration of electromyography textile electrodes for biosignal monitoring. *Journal of Industrial Textiles* **53.** doi:10.1177/15280837231175062 (Jan. 2023).
15. Ershad, F. *et al.* Customizable, reconfigurable, and anatomically coordinated large-area, high-density electromyography from drawn-on-skin electrode arrays. *PNAS Nexus* **2,** 1–14. doi:10.1093/PNASNEXUS/PGAC291 (1 Jan. 2023).
16. Simonetti, D., Koopman, B. & Sartori, M. Automated estimation of ankle muscle EMG envelopes and resulting plantar-dorsi flexion torque from 64 garment-embedded electrodes uniformly distributed around the

human leg. *Journal of Electromyography and Kinesiology* **67,** 102701. doi:10.1016/J.JELEKIN.2022.102701 (Dec. 2022).

17. Drost, G., Stegeman, D. F., van Engelen, B. G. & Zwarts, M. J. Clinical applications of high-density surface EMG: A systematic review. *Journal of Electromyography and Kinesiology* **16,** 586–602. doi:10.1016/J.JELEKIN.2006.09.005 (6 Dec. 2006).
18. Yu, Y., Chen, C., Sheng, X. & Zhu, X. Wrist Torque Estimation via Electromyographic Motor Unit Decomposition and Image Reconstruction. *IEEE Journal of Biomedical and Health Informatics* **25,** 2557–2566. doi:10.1109/JBHI.2020.3041861 (7 July 2021).
19. Xiong, D., Zhang, D., Zhao, X. & Zhao, Y. Deep Learning for EMG-based Human-Machine Interaction: A Review. *IEEE/CAA Journal of Automatica Sinica* **8,** 512–533. doi:10.1109/JAS.2021.1003865 (3 Mar. 2021).
20. Simpetru, R. C. *et al.* Accurate Continuous Prediction of 14 Degrees of Freedom of the Hand from Myoelectrical Signals through Convolutive Deep Learning. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS* **2022-July,** 702–706. doi:10.1109/EMBC48229.2022.9870937 (2022).
21. Prince, S. J. *Understanding Deep Learning* (The MIT Press, 2023).
22. Simonetti, D. *et al.* Automated spatial localization of ankle muscle sites and model-based estimation of joint torque post-stroke via a wearable sensorised leg garment. *Journal of Electromyography and Kinesiology* **72,** 102808. doi:10.1016/j.jelekin.2023.102808 (Oct. 2023).
23. Papageorgiou, S. N. On correlation coefficients and their interpretation. *Journal of Orthodontics* **49,** 359. doi:10.1177/14653125221076142 (3 Sept. 2022).
24. Hosna, A. *et al.* Transfer learning: a friendly introduction. *Journal of Big Data* **9,** 1–19. doi:10.1186/S40537-022-00652-W (1 Dec. 2022).
25. Zhang, L., Zhu, X., Gutierrez-Farewik, E. M. & Wang, R. Ankle Joint Torque Prediction Using an NMS Solver Informed-ANN Model and Transfer Learning. *IEEE Journal of Biomedical and Health Informatics* **26,** 5895–5906. doi:10.1109/JBHI.2022.3207313 (12 Dec. 2022).
26. Liu, Y., Zhang, S. & Gowda, M. A Practical System for 3-D Hand Pose Tracking Using EMG Wearables With Applications to Prosthetics and User Interfaces. *IEEE Internet of Things Journal* **10,** 3407–3427. doi:10.1109/JIOT.2022.3223600 (4 Feb. 2023).
27. Bengio, Y. Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning* **2,** 1–127. doi:10.1561/2200000006 (1 Nov. 2009).
28. Buongiorno, D. *et al.* Task-Oriented Muscle Synergy Extraction Using An Autoencoder-Based Neural Model. *Information 2020, Vol. 11, Page 219* **11,** 219. doi:10.3390/INFO11040219 (4 Apr. 2020).
29. Lee, J. H. Characteristics of muscle synergy extracted using an autoencoder in patients with stroke during the curved walking in comparison with healthy controls. *Gait Posture* **107,** 225–232. doi:10.1016/J.GAITPOST.2023.10.009 (Jan. 2024).
30. Spüler, M., Irastorza-Landa, N., Sarasola-Sanz, A. & Ramos-Murguialday, A. Extracting muscle synergy patterns from EMG data using autoencoders. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **9887 LNCS,** 47–54. doi:10.1007/978-3-319-44781-0_6 (2016).
31. Ameri, A., Akhaee, M. A., Scheme, E. & Englehart, K. A Deep Transfer Learning Approach to Reducing the Effect of Electrode Shift in EMG Pattern Recognition-Based Control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **28,** 370–379. doi:10.1109/TNSRE.2019.2962189 (2 Feb. 2020).
32. Lee, D. *et al.* EMG-based hand gesture classifier robust to daily variation: Recursive domain adversarial neural network with data synthesis. *Biomedical Signal Processing and Control* **88,** 105600. doi:10.1016/J.BSPC.2023.105600 (Feb. 2024).
33. Sun, T., Libby, J., Rizzo, J. R. & Atashzar, S. F. Deep Augmentation for Electrode Shift Compensation in Transient High-density sEMG: Towards Application in Neurorobotics. *IEEE International Conference on Intelligent Robots and Systems* **2022-October,** 6148–6153. doi:10.1109/IROS47612.2022.9981786 (2022).
34. Chamberland, F. *et al.* Novel Wearable HD-EMG Sensor With Shift-Robust Gesture Recognition Using Deep Learning. *IEEE Transactions on Biomedical Circuits and Systems* **17,** 968–984. doi:10.1109/TBCAS.2023.3314053 (5 Oct. 2023).
35. Torricelli, D. *et al.* Muscle synergies in clinical practice: Theoretical and practical implications. *Biosystems and Biorobotics* **10,** 251–272. doi:10.1007/978-3-319-24901-8_10 (2016).
36. Bizzi, E. & Cheung, V. C. The neural origin of muscle synergies. *Frontiers in Computational Neuroscience* **7,** 1–6. doi:10.3389/FNCOM.2013.00051 (Apr. 2013).

Appendix

A. Use of AI

During the preparation of this work the author used ChatGPT by OpenAI in order to review the grammar, spelling and use of signal words of all sections. After using this tool/service, the author reviewed and edited the content as needed and takes full responsibility for the content of the work.

During the preparation of this work the author used Grammarly in order to review the grammar, spelling and use of signal words of all sections. After using this tool/service, the author reviewed and edited the content as needed and takes full responsibility for the content of the work.

B. Results

1) Statistics of lower leg: The statistical methods used on the results of the lower leg are further elaborated in this section.

Firstly, it should be noted that all R^2 values were not normally distributed, as determined by performing a Kolmogorov-Smirnov test. Due to non-normality, a non-parametric test was selected for the statistical analysis. The Kruskal-Wallis test was chosen, where the null hypothesis is that each data group comes from the same distribution. A p-value lower than 0.05 indicates that the null hypothesis can be rejected, suggesting that the groups come from different distributions.

It was claimed that the overall median R^2 values of CNN and EDN are comparable, while the median of NNMF is lower. The following median R^2 values were found:

- NNMF = 0.80
- CNN = 0.90
- EDN = 0.90

A Kruskal-Wallis test was performed, followed by a multiple comparison test to determine the specific p-values for each method. The following p-values were found:

- NNMF vs CNN = 0.0018: significant difference, CNN $>$ NNMF
- NNMF vs EDN = 0.0049: significant difference, EDN $>$ NNMF
- CNN vs EDN = 0.99: no significant difference

For muscles, it was claimed that dorsiflexors (DF) have lower R^2 values than plantar flexors (PF). The following medians were reported for each group:

- NNMF, DF = 0.69
- NNMF, PF = 0.88
- CNN, DF = 0.81
- CNN, PF = 0.95
- EDN, DF = 0.70
- EDN, PF = 0.94

The p-values resulting from Kruskal-Wallis tests are:

- NNMF = 0.0012: significant difference, PF>DF
- CNN = 0.0016: significant difference, PF>DF
- EDN = 0.0011: significant difference, PF>DF

Further claims were made about individual muscles. Table B.1 shows the median values for each method, for all muscles. A Kruskal-Wallis test was performed, followed by a multiple comparison test, resulting in the p-values shown. The significance of the results and which method scored significantly higher are indicated.

TABLE B.1: In this table the median R^2 values of all muscles for all methods are shown. Per muscle also the p-values between all methods are shown with the added conclusion on whether the result is significant and what this means.

Tibialis Anterior				
Method	*Median*	*Between*	*p-value*	*significant?*
NNMF	0,66	NNMF vs CNN	0,14	no
CNN	0,86	NNMF vs EDN	0,71	no
EDN	0,81	CNN vs EDN	0,68	no

Peroneus Longus				
Method	*Median*	*Between*	*p*	*significant?*
NNMF	0,64	NNMF vs CNN	0,13	no
CNN	0,82	NNMF vs EDN	0,74	no
EDN	0,68	CNN vs EDN	0,58	no

Gastrocnemius Medialis				
Method	*Median*	*Between*	*p*	*significant?*
NNMF	0,96	NNMF vs CNN	0,48	no
CNN	0,94	NNMF vs EDN	0,03	yes: NNMF>EDN
EDN	0,90	CNN vs EDN	0,42	no

Gastrocnemius Lateralis				
Method	*Median*	*Between*	*p*	*significant?*
NNMF	0,64	NNMF vs CNN	0,00	yes: CNN >NNMF
CNN	0,92	NNMF vs EDN	0,01	yes: EDN>NNMF
EDN	0,93	CNN vs EDN	1,00	no

Soleus				
Method	*Median*	*Between*	*p*	*significant?*
NNMF	0,95	NNMF vs CNN	1,00	no
CNN	0,94	NNMF vs EDN	0,96	no
EDN	0,96	CNN vs EDN	0,97	no

Lastly, comparisons between tasks were made by combining R^2 values of all muscles and subjects. There were few significant differences between performance on tasks. Comparisons were made in two ways: firstly, between tasks within one method (results shown in table B.2), and secondly, between methods within each task (results shown in table B.3).

TABLE B.2: In this table the medians are shown of all tasks for all methods. Within each method, the performance on the tasks is compared: a p-value is calculated to identify whether there is a significant difference between the two tasks. If there is a difference, it is indicated which of the two is larger.

NNMF					
Task	*Median*	*Comparison between*		*p-value*	*significant*
BW1	0,68	BW1	BW3	1,00	no
BW3	0,72	BW1	FW1	1,00	no
FW1	0,80	BW1	FW3	1,00	no
FW3	0,64	BW1	FW5	0,61	no
FW5	0,61	BW1	RN	1,00	no
RN	0,78	BW1	SS	1,00	no
SS	0,76	BW3	FW1	1,00	no
		BW3	FW3	1,00	no
		BW3	FW5	0,88	no
		BW3	RN	1,00	no
		BW3	SS	1,00	no
		FW1	FW3	1,00	no
		FW1	FW5	0,61	no
		FW1	RN	1,00	no
		FW1	SS	1,00	no
		FW3	FW5	0,95	no
		FW3	RN	1,00	no
		FW3	SS	1,00	no
		FW5	RN	0,19	no
		FW5	SS	0,59	no
		RN	SS	1,00	no

CNN					
Task	*Median*	*Comparison between*		*p-value*	*significant*
BW1	0,71	BW1	BW3	0,98	no
BW3	0,78	BW1	FW1	0,99	no
FW1	0,82	BW1	FW3	0,97	no
FW3	0,82	BW1	FW5	1,00	no
FW5	0,74	BW1	RN	0,74	no
RN	0,84	BW1	SS	0,15	no
SS	0,85	BW3	FW1	1,00	no
		BW3	FW3	1,00	no
		BW3	FW5	1,00	no
		BW3	RN	1,00	no
		BW3	SS	0,99	no
		FW1	FW3	1,00	no
		FW1	FW5	1,00	no
		FW1	RN	1,00	no
		FW1	SS	0,98	no
		FW3	FW5	1,00	no
		FW3	RN	1,00	no
		FW3	SS	0,99	no
		FW5	RN	1,00	no
		FW5	SS	0,70	no
		RN	SS	1,00	no

EDN					
Task	*Median*	*Comparison between*		*p-value*	*significant*
BW1	0,70	BW1	BW3	0,82	no
BW3	0,79	BW1	FW1	0,48	no
FW1	0,82	BW1	FW3	1,00	no
FW3	0,80	BW1	FW5	1,00	no
FW5	0,74	BW1	RN	0,50	no
RN	0,83	BW1	SS	0,88	no
SS	0,84	BW3	FW1	1,00	no
		BW3	FW3	1,00	no
		BW3	FW5	0,50	no
		BW3	RN	1,00	no
		BW3	SS	1,00	no
		FW1	FW3	0,93	no
		FW1	FW5	0,21	no
		FW1	RN	1,00	no
		FW1	SS	1,00	no
		FW3	FW5	1,00	no
		FW3	RN	0,94	no
		FW3	SS	1,00	no
		FW5	RN	0,22	no
		FW5	SS	0,60	no
		RN	SS	1,00	no

TABLE B.3: In this table the medians are shown of all tasks for all methods. Within each task, the performance of the methods is compared: a p-value is calculated to find out whether there is a significant difference between the two methods for that task. If there is a difference, it is indicated which of the two is larger.

BW1					
Method	*Median*		*Comparison between*	*p-value*	*significant*
NNMF	0,83		NNMF vs CNN	0,86	no
CNN	0,87		NNMF vs EDN	0,99	no
EDN	0,85		CNN vs EDN	0,96	no

BW3					
Method	*Median*		*Comparison between*	*p-value*	*significant*
NNMF	0,81		NNMF vs CNN	0,02	yes, CNN>NNMF
CNN	0,90		NNMF vs EDN	0,03	yes, EDN>NNMF
EDN	0,90		CNN vs EDN	1,00	no

FW1					
Method	*Median*		*Comparison between*	*p-value*	*significant*
NNMF	0,81		NNMF vs CNN	0,71	no
CNN	0,91		NNMF vs EDN	0,49	no
EDN	0,92		CNN vs EDN	0,98	no

FW3					
Method	*Median*		*Comparison between*	*p-value*	*significant*
NNMF	0,79		NNMF vs CNN	0,03	yes, CNN>NNMF
CNN	0,90		NNMF vs EDN	0,47	no
EDN	0,88		CNN vs EDN	0,47	no

FW5					
Method	*Median*		*Comparison between*	*p-value*	*significant*
NNMF	0,67		NNMF vs CNN	0,00	yes: CNN>NNMF
CNN	0,88		NNMF vs EDN	0,08	no
EDN	0,81		CNN vs EDN	0,30	no

RN					
Method	*Median*		*Comparison between*	*p-value*	*significant*
NNMF	0,85		NNMF vs CNN	0,36	no
CNN	0,92		NNMF vs EDN	0,33	no
EDN	0,91		CNN vs EDN	1,00	no

SS					
Method	*Median*		*Comparison between*	*p-value*	*significant*
NNMF	0,78		NNMF vs CNN	0,05	no
CNN	0,94		NNMF vs EDN	0,49	no
EDN	0,92		CNN vs EDN	0,60	no

Table B.4 shows the pixel distances found by all methods for the lower leg muscles.

TABLE B.4: This table shows the pixel distances for all muscles and subjects between manually selected electrodes and those found by non-negative matrix factorization-based clustering, the convolutional neural network, and the encoder-decoder network.

Method	**Subject**					
NNMF		*TA*	*PL*	*GM*	*GL*	*SO*
	1	2,06	1,58	0,71	1,58	2,55
	2	2,55	2,83	0,50	1,00	2,06
	3	1,12	2,24	2,83	1,00	5,38
	4	2,24	1,31	1,12	2,12	2,24
	5	0,50	1,00	4,00	2,92	1,00
	6	1,58	1,58	1,12	5,00	2,92
	7	1,58	1,00	2,12	2,83	1,58
	8	0,71	2,24	0,71	2,24	2,24
CNN		*TA*	*PL*	*GM*	*GL*	*SO*
	1	2,69	4,53	2,55	2,69	3,61
	2	2,00	1,00	3,54	1,58	2,50
	3	2,69	0,71	2,50	3,81	2,12
	4	3,35	2,55	2,92	0,50	2,69
	5	0,00	2,69	1,00	1,00	2,06
	6	1,00	2,55	3,64	1,80	3,54
	7	1,12	0,71	0,71	3,54	2,06
	8	1,50	1,12	1,00	1,41	0,50
EDN		*TA*	*PL*	*GM*	*GL*	*SO*
	1	1,00	2,06	1,12	1,50	2,06
	2	1,58	1,80	2,50	1,41	1,00
	3	1,58	1,12	2,06	4,47	1,00
	4	2,24	2,55	1,41	0,50	2,55
	5	0,00	1,58	0,71	0,71	3,00
	6	1,00	2,24	1,00	1,00	2,82
	7	0,50	0,00	1,41	0,50	1,00
	8	1,12	1,50	0,71	1,00	1,00

2) Results upper leg: Table B.5 shows median values from the three subjects for each muscle during each task, resulting in 45 R^2 values per method. Table B.6 shows the pixel distances for all methods, muscles, and subjects.

TABLE B.5: This table shows median R^2 values of the three subjects for each task and muscle for each method.

Method	**Task**					
NNMF		*VL*	*RF*	*VM*	*BF*	*ST*
	FW3	0,87	0,99	0,99	0,87	0,88
	FW1.8	0,73	0,97	0,99	0,68	0,95
	FW4.2	0,78	0,94	0,99	0,91	0,92
	BW1.8	0,28	0,91	0,99	0,35	0,20
	SK3	0,63	0,74	0,96	0,08	0,03
	SL	0,92	0,94	1,00	0,96	0,93
	SD1	0,49	0,95	0,94	0,83	0,72
	SD2	0,52	0,95	0,98	0,77	0,85
	ND	0,36	0,92	0,88	0,66	0,32
CNN		*VL*	*RF*	*VM*	*BF*	*ST*
	FW3	0,97	0,97	0,96	0,99	0,96
	FW1.8	0,88	0,83	0,93	0,94	0,87
	FW4.2	0,94	0,97	0,88	0,99	0,96
	BW1.8	0,91	0,97	0,95	0,90	0,60
	SK3	0,92	0,83	0,92	0,86	0,47
	SL	0,95	0,97	0,98	0,89	0,97
	SD1	0,15	0,49	0,67	0,92	0,87
	SD2	0,94	0,91	0,95	0,96	0,37
	ND	0,95	0,85	0,78	0,78	0,70
EDN		*VL*	*RF*	*VM*	*BF*	*ST*
	FW3	0,94	0,93	0,97	0,94	0,97
	FW1.8	0,90	0,96	0,98	0,87	0,95
	FW4.2	0,97	0,98	0,99	0,96	0,96
	BW1.8	0,94	0,97	0,99	0,65	0,80
	SK3	0,97	0,86	0,96	0,84	0,83
	SL	0,96	0,99	1,00	0,97	0,99
	SD1	0,52	0,96	0,98	0,95	0,55
	SD2	0,95	0,99	0,96	0,95	0,61
	ND	0,95	0,95	0,81	0,67	0,44

TABLE B.6: This table shows the pixel distances for all upper leg muscles and subjects between manually selected electrodes and those found by non-negative matrix factorization-based clustering, the convolutional neural network, and the encoder-decoder network.

Method	**Subject**					
NNMF		*VL*	*RF*	*VM*	*RF*	*ST*
	1	2,83	2,00	3,00	1,00	1,41
	2	3,00	3,00	2,00	2,00	0,00
	3	3,61	2,24	1,41	2,24	2,24
CNN		*VL*	*RF*	*VM*	*RF*	*ST*
	1	0,00	0,00	1,12	1,50	1,50
	2	1,12	1,50	1,58	4,00	5,00
	3	2,55	0,71	0,00	0,71	4,12
EDN		*VL*	*RF*	*VM*	*RF*	*ST*
	1	0,00	1,00	0,00	0,71	1,41
	2	0,00	0,71	1,58	0,50	0,50
	3	1,12	0,00	0,00	0,00	1,41

C. Muscle synergies of the lower leg by NNMF and EDN

1) Introduction: Many different muscles contribute to movements such as walking and balancing. Controlling all of these muscles simultaneously would require great computational power. To alleviate this, it is thought that muscles can be divided into modules, which work together to achieve certain movements. These modules are known as muscle synergies [35, 36]. For the lower leg, these synergies are divided into plantar flexion (PF) and dorsiflexion (DF) groups. They have been previously extracted by Clark et al., Ivanenko et al., and Gonzalez-Vargas et al. Simonetti et al., [16] demonstrated that muscle synergies could be extracted using NNMF-based clustering on a uniformly distributed electrode grid around the lower leg. Since in this research NNMF-based clustering similar to the method of Simonetti et al. is performed, it is hypothesized that the clusters also represent muscle synergies.

An EDN, such as the one created in this research, consists of an encoder and a decoder part. At the bridge in between, only the most important key features remain, from which the output can be created. These key features are hypothesized to reflect muscle synergies, as it should be possible to recreate the muscle-specific EMG envelopes from these synergies.

2) Methods: To demonstrate that the key features reflect muscle synergies, the encoder part of the EDN is used as a separate network. As input to this network, recordings from movement task FW1 are given. The output, a 2x40xN (with N representing the number of points in time included) array, is averaged over the second dimension to leave a 2xN array. This array is gait-cycle averaged, with gait-cycles ranging from heel-strike to heel-strike. The resulting synergies are visually inspected to appoint them to the PF or DF group. Additionally, two clusters were extracted from three gait cycles of the FW1 task using NNMF. The clusters were also gait cycle averaged over the three gait cycles. These clusters were automatically sorted into either the PF or DF groups. The mean and standard deviation are taken across all subjects of the resulting synergies for NNMF and the encoder. The R^2 is calculated between both averaged synergies and the previously extracted synergies from Clark et al., Ivanenko et al., and Gonzalez-Vargas et al to compare the shapes.

3) Results: In Fig. C.1 the resulting PF and dorsiflexion synergies from EDN and NNMF are shown with those extracted by Clark et al., Ivanenko et al., and Gonzalez-Vargas et al. Table C.1 shows the R^2 values, and table C.2 shows RMSE values, both calculated between the synergies by Clark et al., Ivanenko et al., and Gonzalez-Vargas et al. and those from NNMF clustering and the encoder part of the EDN. It is evident that for R^2, NNMF scores higher for all reference synergies. For RMSE, all values are higher for EDN than NNMF, except for the PF synergy from Ivanenko. In Fig. C.2 the synergies from EDN and NNMF are plotted, accompanied by their associated standard deviations.

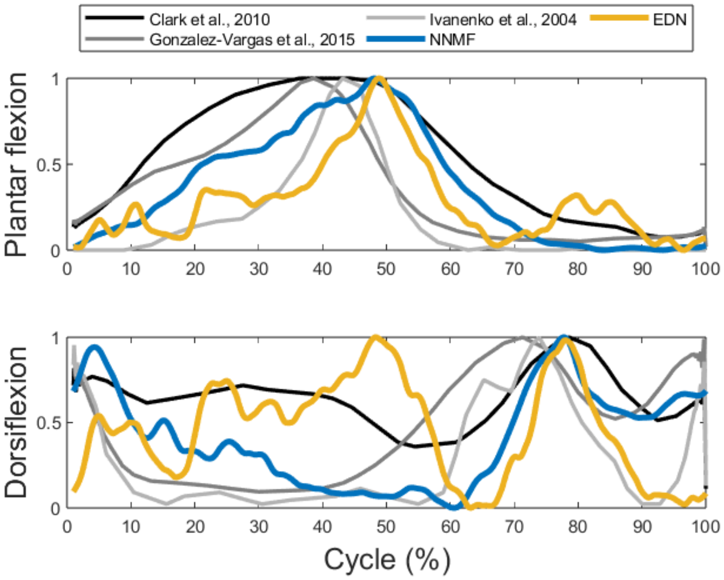


Fig. C.1: Muscle synergies of plantar flexion and dorsiflexion from Clark et al., Ivanenko et al., and Gonzalez-Vargas et al. compared with muscle synergies extracted using non-negative matrix factorization and the encoder-decoder network.

TABLE C.1: R^2 values calculated between the synergies by Clark et al., Ivanenko et al., and Gonzalez-Vargas et al. and those from non-negative matrix factorization clustering and the encoder part of the encoder-decoder network.

	EDN PF	**NNMF PF**		**EDN DF**	**NNMF DF**
Clark et al.	0,47	0,90		0,10	0,55
Gonzalez-Vargas et al.	0,22	0,57		0,17	0,19
Ivanenko et al.	0,57	0,67		0,01	0,31

TABLE C.2: Root mean square error values calculated between the synergies by Clark et al., Ivanenko et al., and Gonzalez-Vargas et al. and those from non-negative matrix factorization clustering and the encoder part of the encoder-decoder network.

	EDN PF	**NNMF PF**		**EDN DF**	**NNMF DF**
Clark et al.	0,36	0,21		0,33	0,31
Gonzalez-Vargas et al.	0,30	0,23		0,50	0,32
Ivanenko et al.	0,21	0,27		0,48	0,30

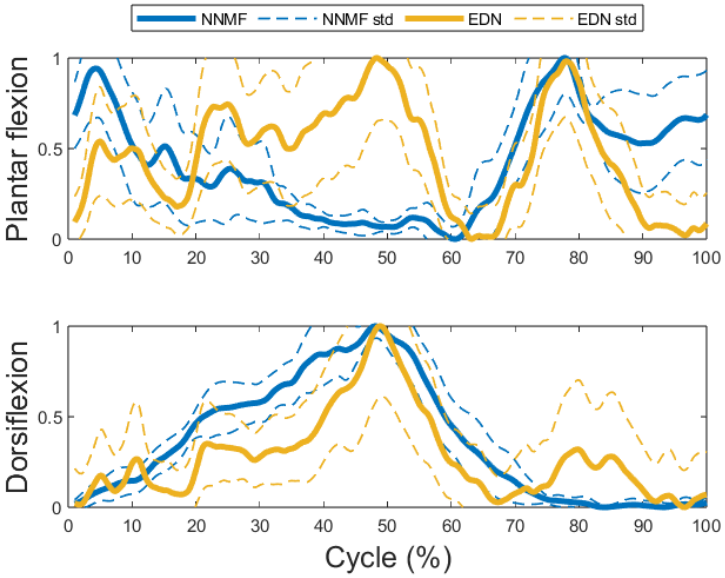


Fig. C.2: Muscle synergies of plantar flexion and dorsiflexion extracted using non-negative matrix factorization and the encoder-decoder network, including the standard deviations.

4) Discussion and conclusion: From the resulting R^2 and RMSE values, it can be concluded that the EDN synergies resemble previously extracted synergies much less than the NNMF synergies.

The EDN synergies do not show a sharp distinction between DF and PF synergies, while the others all do. This is shown by the following:

- For the PF synergy, a small peak exists at around 80% of the gait cycle. A similar (larger) peak exists at 80% of the DF synergy gait cycle.
- For the DF synergy a large peak exists at about 50% of the gait cycle, which has similar timing to the peak at 50% of the PF synergy.

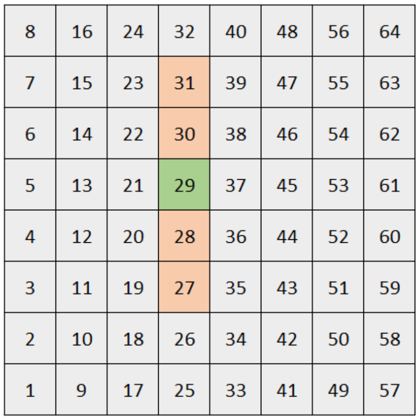
It can not be said with certainty why such a distinction is not very present. It could be because the EDN has 40 filters, and maybe there is a distinction between filters that give more PF-centered or DF-centered results. In future research, an EDN with a bridge consisting of one filter should be considered.

The standard deviation of both NNMF and EDN is quite large, indicating that large differences exist between subjects. This means that the resulting muscle synergies are not always the same. This could be due to natural differences between individuals, but also, in the case of the EDN, it might be caused by differences caused by network training.

Another thing to note is that large differences can also be seen between the previously reported synergies. The exact reason for these differences is unclear. Maybe the synergy extraction method partially determines the shape, or the groups used were not large enough to find the same synergy.

To conclude, the synergies extracted by EDN do not fully resemble the separate synergies, while the synergies extracted by NNMF largely do. However, the synergies extracted by the EDN show peaks with similar timing as the reference synergies, but they are not split into DF and PF synergies as distinctly.

D. Locality mask upper leg



8	16	24	32	40	48	56	64
7	15	23	31	39	47	55	63
6	14	22	30	38	46	54	62
5	13	21	29	37	45	53	61
4	12	20	28	36	44	52	60
3	11	19	27	35	43	51	59
2	10	18	26	34	42	50	58
1	9	17	25	33	41	49	57

Fig. D.1: The mask used to ensure locality of electrodes, the whole figure represents an 8x8 electrode grid, with the numbers representing the electrode numbers. The red squares are considered ”close enough” to the green one to potentially form a cluster

E. Euclidean distances

In this appendix, some figures are shown which are related to the euclidean distances that were calculated between the manually selected electrodes and those selected by NNMF-based clustering, the CNN, and the EDN.

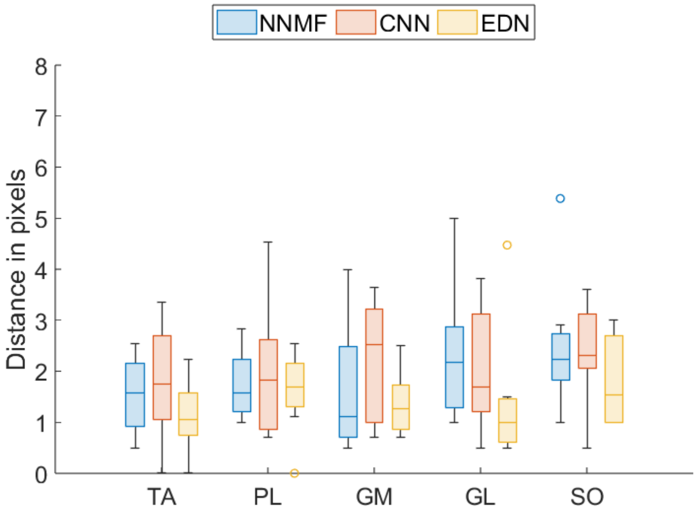


*Fig. E.1: In this figure a box plot is shown of Euclidean distances in pixels (with one pixel representing the distance between two electrodes) calculated between the manually selected electrodes of lower leg muscles and those selected by non-negative matrix factorization clustering, convolutional neural network, and encoder-decoder network. The y-axis represents the Euclidean distances in pixels. The x-axis represents the muscles. The horizontal lines indicate median values, calculated between all muscles. The bottom and top edges indicate the 25th and 75th percentiles and the whiskers extend to the most extreme values not considered outliers. Circles indicate outliers. *indicates that p<0.05.*

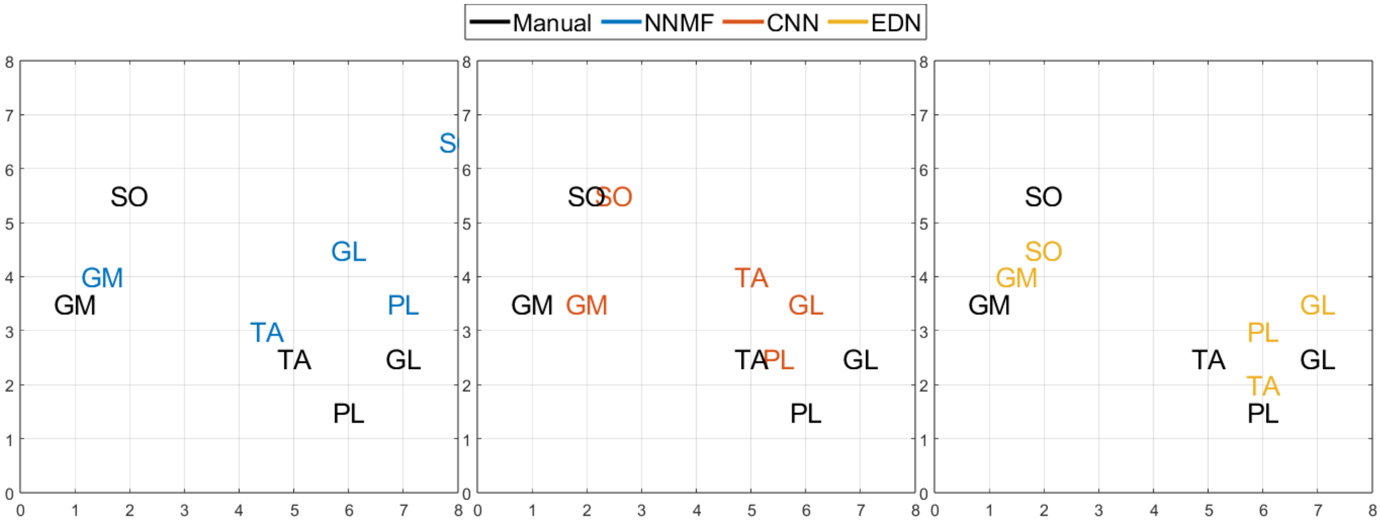


Fig. E.2: In this figure three subplots are shown, each showing the average location of the manually selected electrodes and the average location of the electrodes selected by either non-negative matrix factorization-based clustering, the convolutional neural network, or the encoder-decoder network for each lower leg muscle. The locations are shown in an 8x8 grid, with each number on the y-axis representing a row and each on the x-axis representing a column of electrodes. The locations are shown for just subject 1.

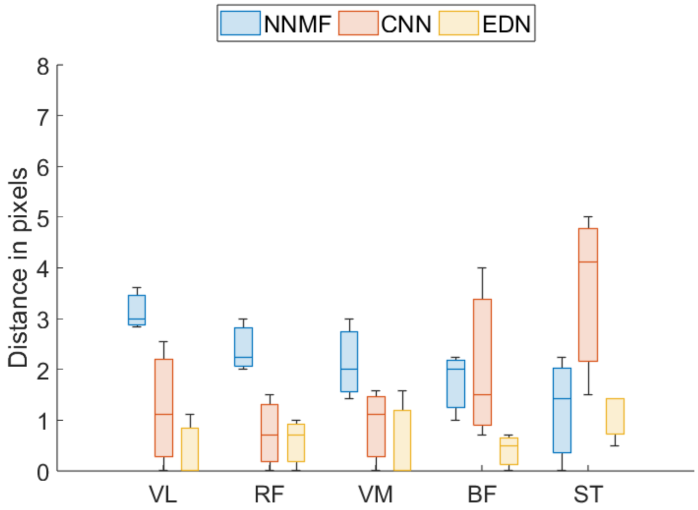


*Fig. E.3: In this figure a box plot is shown of Euclidean distances in pixels (with one pixel representing the distance between two electrodes) calculated between the manually selected electrodes of upper leg muscles and those selected by non-negative matrix factorization clustering, convolutional neural network, and encoder-decoder network. The y-axis represents the Euclidean distances in pixels. The x-axis represents the muscles. The horizontal lines indicate median values, calculated between all muscles. The bottom and top edges indicate the 25th and 75th percentiles and the whiskers extend to the most extreme values not considered outliers. Circles indicate outliers. *indicates that p<0.05.*

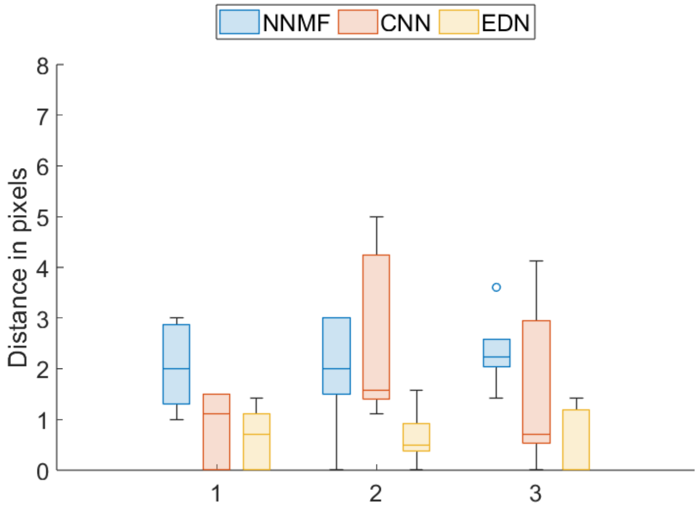


*Fig. E.4: In this figure a box plot is shown of Euclidean distances in pixels (with one pixel representing the distance between two electrodes) calculated between the manually selected electrodes of upper leg muscles and those selected by non-negative matrix factorization clustering, convolutional neural network, and encoder-decoder network. The y-axis represents the Euclidean distances in pixels. The numbers on the x-axis represent the subjects. The horizontal lines indicate median values, calculated between all muscles. The bottom and top edges indicate the 25th and 75th percentiles and the whiskers extend to the most extreme values not considered outliers. Circles indicate outliers. *indicates that p<0.05.*

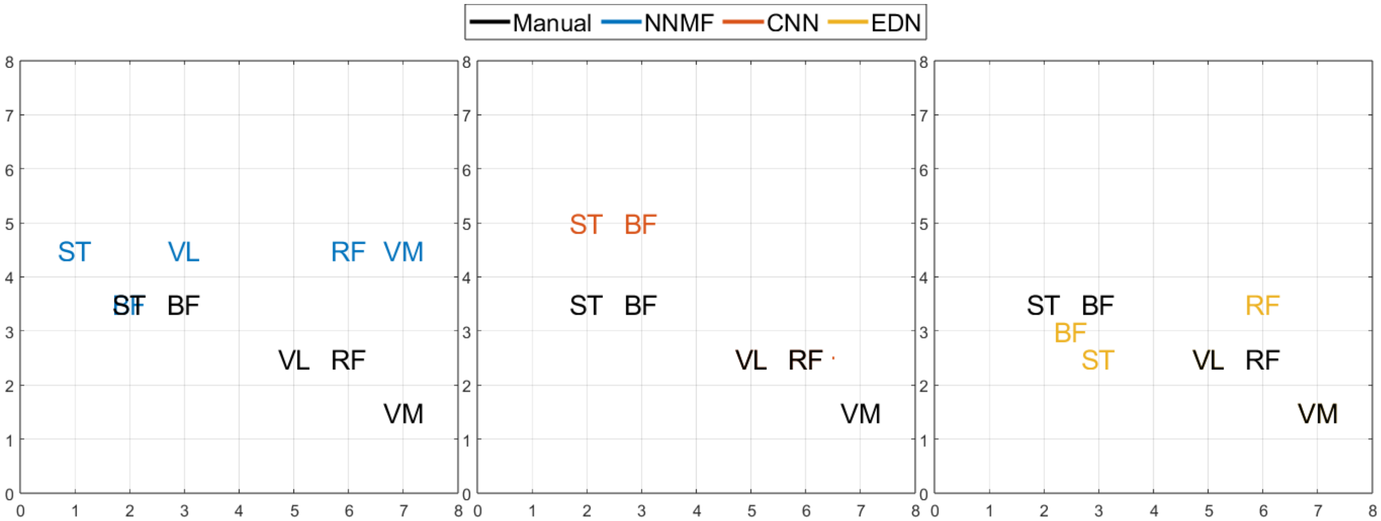


Fig. E.5: In this figure three subplots are shown, each showing the average location of the manually selected electrodes and the average location of the electrodes selected by either non-negative matrix factorization-based clustering, the convolutional neural network, or the encoder-decoder network for each upper leg muscle. The locations are shown in an 8x8 grid, with each number on the y-axis representing a row and each on the x-axis representing a column of electrodes. The locations are shown for just subject 1.