Compression of EMG Signals Using Deep Convolutional Autoencoders

Kimia Dinashi, Ali Ameri, Mohammad Ali Akhaee, Kevin Englehart, *Senior Member, IEEE*, and Erik Scheme, *Senior Member, IEEE*

***Abstract*—Efficient storage and transmission of electromyogram (EMG) data are important for emerging applications such as telemedicine and big data, as a vital tool for further advancement of the field. However, due to limitations in internet speed and hardware resources, transmission and storage of EMG data are challenging. As a solution, this work proposes a new method for EMG data compression using deep convolutional autoencoders (CAE). Eight-channel EMG data from 10 subjects, and high-density EMG data from 18 subjects, were investigated for compression. The CAE architecture was designed to extract an abstract data representation that is heavily compressed, but from which the salient information for classification can be effectively reconstructed. The proposed method attained efficient compression; for CR = 1600, the average PRDN (percentage RMS difference normalized) was 31.5% and the wrist motions classification accuracy (CA) reduced roughly 5%. The CAE substantially outperformed the state-of-the-art high-efficiency video coding and a well-known wavelet-thresholding compression technique. Moreover, by reducing the bit-resolution of the CAE's compressed data from 24 bits to 6 bits, an additional 4-fold compression was achieved without significant degradation of the reconstruction performance. Furthermore, the CAE's inter-subject performance was promising; e.g., for CR = 1600, the PRDN for the inter-subject case was only 2.6% less than that of the within-subject performance. The powerful EMG compression performance with remarkable reconstruction results reflects the CAEs potential as an automatic end-to-end approach with the ability to learn the complete encoding and decoding process. Furthermore, the excellent inter-subject performance demonstrates the generalizability and usability of the proposed approach.**

***Index Terms*—Big data, compression, convolutional autoencoder, deep learning, EMG.**

Manuscript received June 12, 2021; revised October 11, 2021 and January 5, 2022; accepted January 8, 2022. Date of publication January 11, 2022; date of current version July 4, 2022. This work was supported by NSERC Discovery Grants 217354-20 and 04920-21. *(Corresponding author: Ali Ameri.)*

Kimia Dinashi and Mohammad Ali Akhaee are with the Department of Electrical Engineering, University of Tehran, Tehran 111554563, Iran (e-mail: kimiadinashi@gmail.com; akhaee@ut.ac.ir).

Ali Ameri is with the Department of Biomedical Engineering, School of Medicine, Shahid Beheshti University of Medical Sciences, Tehran 1983963113, Iran (e-mail: aliameri86@gmail.com).

Kevin Englehart and Erik Scheme are with the Institute of Biomedical Engineering, University of New Brunswick, Fredericton E3B 5A3, Canada (e-mail: kengleha@unb.ca; escheme@unb.ca).

Digital Object Identifier 10.1109/JBHI.2022.3142034

I. Introduction

THE electromyogram (EMG) signal is a measure of the electrical activity of muscles. Acquisition of EMG signals is common in human motor research, clinical practice (e.g., in neurology, orthopedics, and physiatry), and in myoelectric control of prostheses, robotics, and human-computer interfaces (HCI). With reductions in the size of electrode sensors and instrumentation, it has become common to see the use of high-density EMG recordings with 128 channels (or higher) with a sampling rate of no less than 1 kHz (usually at least 3 kHz), where each sample is digitized with 12, 16 or 24 bits. As an example, for a 128 channel, 2 kHz sampling rate, and 16-bit recording, the bandwidth required to transmit these data (uncompressed) for a real-time control application would be greater than 4 Mbps. If these data are to be transmitted using existing wireless communication protocols, these rates exceed the current state-of-the-art (Bluetooth 5.0, with a maximum theoretical bandwidth of 2 Mbps).

This challenge is more prominent in emerging data-driven applications. For example, telehealth, which enables clinicians to monitor patients from a distance using remotely collected data such as EMG to diagnose or treat them via telecommunications platforms, deals with enormous amounts of data. Maintaining accessibility to such data through cloud-based healthcare data storage thus requires handling big data. Emerging consumer applications such as Facebook's project on HCI with wrist EMG signals [1], will also motivate the collection and storage of massive volumes of EMG data. Moreover, EMG cloud computing, which is especially useful for training deep learning-based myoelectric control models on a remote powerful computer (e.g., Google-Colab), requires transferring large amounts of training EMG data across the internet. In all of these applications, it would be advantageous to first compress EMG data by reducing possible redundancies or removing irrelevant information within the data.

In both academia and industry, the generation and accumulation of large public datasets are highly desirable for use as benchmarks from which to progress analytical methods and build larger and more effective machine learning models. Such big data approaches have played a major role in the advancement of fields such as gait [2], computer vision [3], and speech recognition [3]. However, there has been a lack of public EMG databases until recently that a number of public datasets have been made available. For example, NinaPro [4] database was

created with the goal of facilitating research on myoelectric control of robotic prosthetic hands and human-computer interaction (HCI). It comprises ten datasets from ten different studies with able-bodied or amputee subjects, where the EMG data were collected from the subjects' forearms while performing various wrist and hand movements. Another example is the CapgMyo [5] database, which contains data from one study on EMG-based gesture recognition with 23 intact subjects, wherein high-density EMG signals were collected from the subjects' forearms while eliciting different hand gestures. One of the challenges in establishing EMG databases is attributed to the large size of data, due to high sampling rates (typically 1 or 2 KHz), multiple EMG electrodes, and many subjects. For example, in the second, third, and fourth NinaPro datasets, each subject's data size are roughly 450 MB, 400 MB, and 300 MB, respectively. To facilitate the storage and transmission of large volumes of EMG data for their widespread dissemination, to reduce the burden on existing IT infrastructure, and to overcome limitations in internet speed and hardware resources, some form of data compression is often warranted or even necessary.

Depending on the application, this compression may be *lossless* or *lossy*. With lossless techniques, the exact original data can be reconstructed, but the compression is not efficient. Lossless compression is mainly used in applications where no loss of data can be tolerated. Run Length encoding [6], Huffman coding [7], and Lempel-Ziv-Welch (LZW) [8] are three examples of lossless compression. With Lossy methods, however, significant compression can be achieved, at the expense of losing some information in the data.

Previous studies on EMG data compression have applied widely used transform-based lossy compression techniques including discrete wavelet transform (DWT) coding with optimized basis function [9], [10], embedded zero-trees of DWT coefficients [11], [12], vector quantization of the discrete cosine transform (DCT) and DWT coefficients [13], [14], dynamic bit allocation of the DWT coefficients [15], [16], the multidimensional multiscale parser algorithm [17], a codebook linear prediction [18], the JPEG2000 and H.264/AVC-intra transforms [19], compressed sensing [20], the SPIHT and arithmetic coding [14], entropy coding [16], and video codec algorithm [21].

In contrast to the previous studies described above, this work proposes a fundamentally different strategy by applying a deep learning approach for EMG compression. Deep learning frameworks, especially convolutional neural networks (CNNs), have been shown to perform well in the EMG-based estimation of limb movements [22]–[24] because of their ability to learn features from raw EMG data, eliminating the need for feature engineering. CNNs leverage convolutional layers which are highly effective for series data, images, and videos. They can learn nonlinear transformations by using a nonlinear activation function and multiple layers, and do so more efficiently than through one complex transformation as in other methods. This motivated the consideration of a class of CNNs, i.e., deep convolutional autoencoders (CAEs), for EMG compression as an alternative to conventional transform-based compression techniques. Another inspiring factor was CAEs promising performance in other fields for dimensionality reduction [25] such as in compressing electrocardiogram (ECG) signals [26]. This work proposes a novel application of CAEs for EMG compression, where a CAE is used to automatically learn a lower-dimensional representation of the raw EMG data, preserving relevant information while reducing storage and transmission bandwidth requirements.

II. Methods

A. Datasets

Two datasets were investigated for compression. The first dataset (here labelled EMG-8) was collected in our previous study [27], as briefly described in this section. The second dataset (here labelled EMG-128) was a subset (CapgMyo-DB-a) of the CapgMyo dataset [5].

In the EMG-8 dataset, ten able-bodied subjects (ages: 31.4 $\pm$ 4.1 years, one lefthanded, nine right-handed) participated in this experiment. Subjects signed informed consent forms as approved by the ethics board of Shahid Beheshti University of Medical Sciences (No: IR.SBMU.MSP.REC.1396.101, year: 2017). Eight pairs of surface electrodes (g.HiAmp, g-tec Inc.) were attached around the dominant forearm proximal to the elbow to record eight bipolar EMG channels. A reference electrode was placed on the opposite wrist's ulnar styloid. The EMG data were recorded at 1.2 KHz, with 24-bit resolution, filtered between 5-500 Hz, and notch filtered at 50 Hz using Butterworth filters.

The subjects elicited eight dynamic wrist contractions including flexion, extension, pronation, supination, and their simultaneous combinations: extension and pronation, flexion and pronation, extension and supination, flexion and supination. The experiment comprised a *no motion* trial, as well as eight dynamic trials, corresponding to the eight motions listed above.

The *no motion* trial was 30s in duration and involved maintaining a relaxed posture. Each trial lasted 48s and involved performing the trial-specific wrist motion dynamically, with intensity proportional to a progress bar presented on a computer screen. Each trial included four repetitions of the following cycle: *no motion* (3s), ramping up to the full-range (a subjective comfortable intensity) contraction (3s), maintaining the full range (3s), and ramping down to *no motion* (3s). In case of combined motions trials, the users were prompted to regulate both DoFs simultaneously and proportionally to the same progress bar.

The EMG-128 dataset comprises 128-channel high-density (HD) EMG data from 18 subjects eliciting 8 hand gestures. Surface EMG signals were collected from 18 able-bodied subjects (age:23-26 years) using a wearable device wrapped around the right forearm, consisting of 8 acquisition modules, where each module contained a 2×8 electrode array, with an inter-electrode horizontal distance of 7.5 mm and a vertical distance of 10.05 mm. The 8 modules formed an electrode array of 16×8, i.e., 128 channels in total. The EMG signals were recorded at 1 KHz with 16-bit resolution, and were bandpass filtered between 20-380 Hz. The subjects sat in a chair with their arms rested on a desktop. Each subject elicited 10 repetitions of 8 isometric and isotonic hand gestures; equivalent to gestures 13-20 in the NinaPro dataset [28]. The subjects mimicked hand

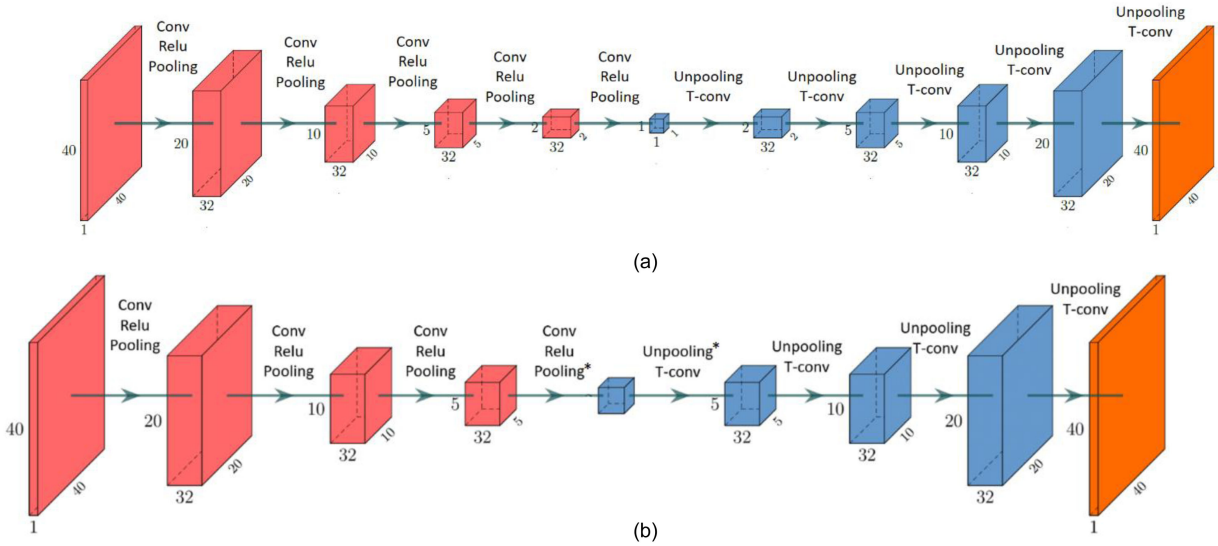

Fig. 1. The CAE architectures for (a) CR = 1600 and (b) CR = 16, 32, 100, 200, where the pooling and unpooling layers shown with $*$ are only used for CR = 100, 200. The *code* size is $5 \times 5 \times 4$, $5 \times 5 \times 2$, $2 \times 2 \times 4$, $2 \times 2 \times 2$, for CR = 16, 32, 100, 200, respectively.

gestures displayed on a screen with their right hand. More details can be found in [5].

B. EMG Compression With the Proposed CAE

An autoencoder is a neural network that is trained to output a copy of its input [25]. The network has a hidden layer that describes a representation of the input data which is called the *code*. Hence, the network can be viewed as comprising two parts: an encoder and a decoder. If the *code* is constrained to have a smaller dimension than the input data, it will contain useful features for data reconstruction. This allows autoencoders to be applied for feature learning and dimensionality reduction tasks [25]. In convolutional autoencoders (CAEs), instead of fully-connected layers, convolutional layers are used that reduce computational complexity, hence allowing for deeper structures. In CAEs, the encoder contains convolutional and pooling layers to compress the data, whereas the decoder expands the data dimensionality through transposed convolutional layers and un-pooling layers. The compressed data are then given as the representation with the lowest dimensionality obtained by extracting the data at the bottleneck of the CAE.

With the EMG-8 dataset, EMG data were segmented for training into 167 ms (200 samples) windows with increments of 40 ms (48 samples). For testing, however, non-overlapping windows (increment = 200 samples) were used. Five compression ratios (CR) of 16, 32, 100, 200, and 1600 were investigated. The CAE architecture was designed according to the desired CR as depicted in Fig. 1. The optimal architectures were found empirically by choosing the number of convolutional layers, the number of pooling layers (which determines the first two dimensions of the size of the *code*) and the number of filters in the last convolutional layer in the encoder part (which determines the third dimension of the size of the *code*). Each 8-channel EMG segment was arranged as a 40×40 matrix to be used as both the input and target of the CAE. For each subject, a CAE was trained and tested with data from the same subject using a 4-fold cross-validation, where 75% of the data were employed for training and the remaining 25% for testing.

All filters in the convolutional and transposed convolutional layers were 3×3, and the input matrices were zero-padded before convolution. Pooling was performed on 2×2 areas with a stride of 2. A mini-batch size of 128 was used. The adaptive moment estimation (Adam) algorithm, with 20 epochs, and an initial learning rate of 0.001, was used for training the network. During training, a randomly sampled 10% of the data were used to validate the network, once per epoch.

The encoder part of the proposed CAE architecture is similar to well-known CNN structures such as AlexNet. The number of convolutional layers, filters, and training parameters were determined empirically to achieve the best performance. For example, the performance improvement found when increasing the number of convolution layers saturated at 4 convolutional layers in the encoder.

Using the EMG-128 dataset, data from 128 channels were segmented into non-overlapping windows of 100 ms (100 samples). Each 128-channel EMG segment were rearranged as a 100×128 matrix, to be used as both the input and target of the CAE. The CR values used with the EMG-8 dataset were not necessarily attainable, due to the input size difference, along with the compressed data size being an integer. Consequently, CRs close to those used with the EMG-8 dataset were chosen for comparison purposes, and included 16, 33, 133, 266, and 1066. The same CAE structure was used with the EMG-128 dataset, except that the number of filters in convolutional layers were 64 (instead of 32) and as with the EMG-8 dataset, CR was regulated by adjusting the number of filters in the last convolutional layer, and the number of pooling layers (Table I). Note that the listed CR values have been rounded to the nearest integer. Analyses were performed with Matlab 2019b on a 3.50 GHz quad-core computer with 7.5 GB RAM and an NVIDIA GTX 1070 GPU.

TABLE I
CAE ARCHITECTURE PARAMETERS FOR EACH CR WITH THE EMG-128 DATASET

	Number of pooling layers	Number of filters in the last conv layer
16	3	4
33	3	2
133	4	2
266	5	4
1066	5	1

To explore the potential to achieve further compression, the bit resolution of the compressed data was reduced via quantization without changing the trained network. For this purpose, because it was observed that, for all subjects, all features in the compressed data were in the range [0,3000] for CR = 16, [0,3500] for CR = 32, [0,4000] for CR = 100, [0,5500] for CR = 200, and [0,8500] for CR = 1600, this range was divided into 2^N uniformly spaced quantization levels, where N was the new bit-resolution.

Initially, this compression was computed on a per-subject basis. However, to explore the generalizability of the approach, the inter-subject performance was also assessed by a leave-one-out approach conducted for each subject, where every subject was tested with a single CAE trained with aggregated data from the other nine subjects.

C. Comparison to a Wavelet-Thresholding Compression Technique

For comparison, a widely used wavelet-thresholding compression algorithm [29] was evaluated. With this method, data were encoded by applying the wavelet transform and setting any wavelet coefficients below a certain threshold to zero. Decoding was then performed by applying the inverse wavelet transform. A level 3 decomposition using Daubechies3 wavelet using a global threshold was employed. The optimal threshold and other parameters were found empirically.

D. Comparison to HEVC

High Efficiency Video Coding (HEVC) [30], known as H.265, is the state-of-the-art standard for video encoding and was consequently employed for comparison to our proposed method. Notably, HEVC has been recently applied to EMG compression [21]. A third version of HEVC was finalized in 2015 and more information can be found at [31]. Here, the reference software for HEVC, known as HM (version 16.23), was used for simulation. All data were fed into HEVC at once. A burst detection approach [21] was applied to prepare the input data for HEVC, where each row of the input EMG matrix contained one EMG burst. Each trial had 4 repetitions of the same movement, where each repetition corresponded to a burst. Thus, 8 trials × 8 channels × 4 repetitions corresponded to 256 rows. Moreover, 16 rows were added to include the EMG data for the *no motion* trial (two rows for each channel). Therefore, the size of the EMG matrix as the input to HEVC was 272 × 15000. HEVC accepts only positive values as input, hence a bias was added to the input matrix. Since HEVC takes 16-bit input data, a quantization preprocessing was conducted to reduce the EMG resolution from 24-bit to 16-bit. Consequenlty, the total CR was obtained by multiplying the HEVC's CR by a factor of 1.5, to account for this preprocessing.

E. Performance Metrics

1) Compression Ratio (*CR*): the ratio of the size of the original data to the size of the compressed data. An efficient compression is expected to have high *CR*.

$$CR = \frac{original\ size}{compressed\ size} \tag{1}$$

2) Percentage root mean square difference (*PRD*): the root mean square (RMS) of the reconstruction error, divided by the RMS of the original data. For a high-quality compression, the *PRD* is expected to be as low as possible.

$$PRD\ (\%) = 100 \times \left(\frac{\sum_{i=0}^{N-1} (S_o(i) - S_r(i))^2}{\sum_{i=0}^{N-1} (S_o(i))^2} \right)^{\frac{1}{2}} \tag{2}$$

where S_o represents the original data, S_r represents the reconstructed data, and N is the number of data samples.

3) *PRD* Normalized (*PRDN*): The RMS of the reconstruction error divided by the standard deviation of the original data. The *PRDN* is expected to be as low as possible, for a high-quality compression.

$$PRDN\ (\%) = 100 \times \left(\frac{\sum_{i=0}^{N-1} (S_o(i) - S_r(i))^2}{\sum_{i=0}^{N-1} (S_o(i) - M)^2} \right)^{\frac{1}{2}} \tag{3}$$

where M is mean of the original data. In this study, *PRD* = *PRDN*, because the DC offset of the EMG signals were removed ($M = 0$).

4) Signal to Noise ratio (*SNR*): The inverse of *PRDN* stated in decibel (dB). A high *SNR* is expected for a quality compression.

$$SNR = 10 \times log \left(\frac{\sum_{i=0}^{N-1} (S_o(i) - M)^2}{\sum_{i=0}^{N-1} (S_o(i) - S_r(i))^2} \right) \tag{4}$$

5) Quality Score (*QS*): the ratio of *CR* to *PRD*. An efficient compression is expected to have high *QS*.

$$QS = \frac{CR}{PRD} \tag{5}$$

6) Classification accuracy (CA): The EMG-based classification accuracy of wrist motions using a convolutional neural network (CNN) classifier trained and tested on the reconstructed data was assessed and compared to that of a CNN trained and tested on the original data. Nine classes, comprising *no motion* and the eight wrist motions were considered. In the active trials (2-9), data were removed when the target was lower than 30% of the full-range to exclude *no motion* data. For each subject, a CNN was trained and tested on data from the same subject, using a 4-fold cross-validation, where 75% of the data were considered for training and the remaining 25% for testing.

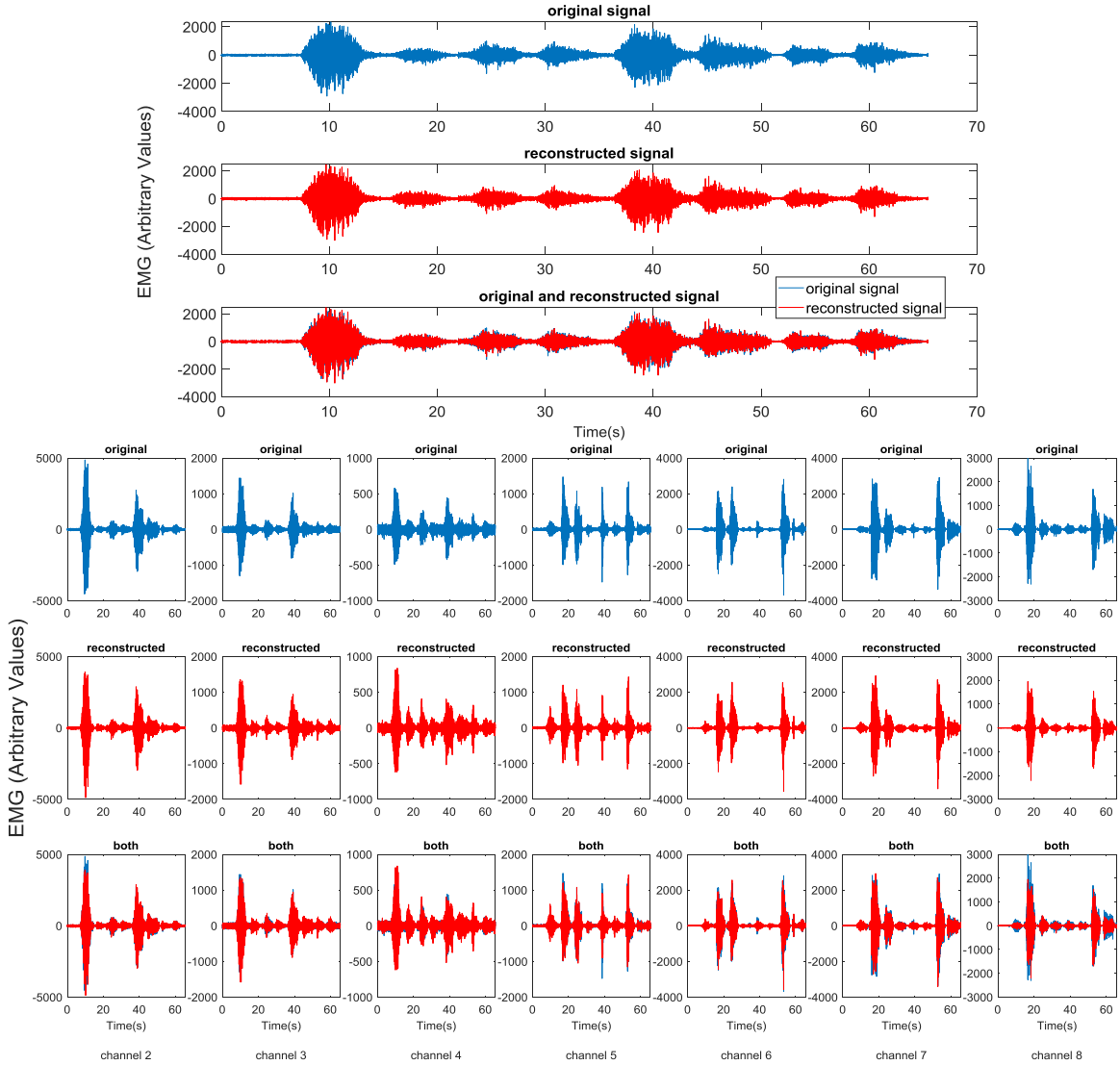

Fig. 2. For CR = 6400, the original and reconstructed signals are plotted for a representative subject. All channels were compressed and reconstructed simultaneously. One channel is shown larger for clarity.

The same CNN architecture of our previous work [32] was employed.

It should be noted that EMG pattern recognition was used here as an example to evaluate the integrity of the reconstructed EMG. Of all of the practical uses of EMG, pattern recognition is one of the most commonly cited and maybe the most unforgiving of distortion, as evidenced by studies in electrode shift, skin conductivity, and limb position [33]. Therefore, the resilience of pattern recognition performance (CA) after compression is an excellent measure of retaining the essence of the EMG signals in the reconstructed data.

F. Statistics

The normality of the data was evaluated with a Shapiro-Wilk test for each performance metric of each method. In all cases, $p>0.05$ and thus normal distributions were assumed. Paired sample t-tests were performed for each metric to compare the CAE method with either the wavelet-thresholding technique or HEVC. A significance level of 0.05 was considered.

III. Results

All results presented here are related to the EMG-8 dataset, unless explicitly stated otherwise. For CR = 6400 (quantized compressed data), the original and reconstructed signals are plotted in Fig. 2, for a representative subject. All eight channels were compressed and reconstructed, simultaneously. One channel is plotted larger for clarity. For each subject, for CR = 6400, the average processing time for the network training was 61s, and for testing (encoding + decoding) was 6.8 ms, for each window of the 8-channel data.

Table II lists the performance metrics averaged across all subjects, along with standard deviation (N = 10) for the proposed CAE (without quantization) and the wavelet-thresholding method for different CRs (note that HEVC will be included in

TABLE II
CAE AND WAVELET-THRESHOLDING (WT) METHODS PERFORMANCE METRICS AVERAGED ACROSS ALL SUBJECTS, ALONG WITH STANDARD DEVIATIONS (N = 10) FOR DIFFERENT CRS

	PRDN		SNR		QS		CA (Orig: 90.5 ± 1.3)	
	CAE	WT	CAE	WT	CAE	WT	CAE	WT
16	25.7 ± 0.5	39.4 ± 4.1	27.2 ± 0.4	18.7 ± 2.1	0.6 ± 0.01	0.4 ± 0.05	88.4 ± 1.3	65.2 ± 4.0
32	26.1 ± 0.5	53.3 ± 3.9	26.8 ± 0.4	12.6 ± 1.5	1.2 ± 0.02	0.6 ± 0.05	88.2 ± 1.3	55.9 ± 3.9
100	29.0 ± 0.5	73.2 ± 3.6	24.7 ± 0.4	6.3 ± 1.0	3.4 ± 0.1	1.4 ± 0.1	87.1 ± 1.3	40.1 ± 3.1
200	30.8 ± 0.7	82.2 ± 3.3	23.6 ± 0.5	3.9 ± 0.8	6.5 ± 0.2	2.4 ± 0.1	86.7 ± 1.5	32.8 ± 2.5
1600	31.5 ± 0.4	94.1 ± 2.6	23.1 ± 0.3	1.2 ± 0.6	50.9 ± 0.7	17.0 ± 0.5	85.5 ± 1.2	23.2 ± 1.9

TABLE III
CAE PERFORMANCE METRICS FOR SINGLE-CHANNEL EMG COMPRESSION AVERAGED ACROSS ALL SUBJECTS, ALONG WITH STANDARD DEVIATIONS (N = 10) FOR DIFFERENT CRS

	PRDN	SNR	QS
16	26.3 ± 0.2	26.7 ± 0.2	0.6 ± 0.01
32	27.1 ± 0.3	26.1 ± 0.2	1.2 ± 0.01
128	29.5 ± 0.4	24.4 ± 0.2	4.3 ± 0.05
256	29.6 ± 0.3	24.3 ± 0.2	8.6 ± 0.09

quantization-based results in Table V, as it has built-in quantization). In all cases, the proposed method significantly outperformed ($p<0.001$) the wavelet-thresholding approach. For context, the baseline classification accuracy of the original data, before any compression, was 90.5%±1.3.

Table III presents the performance metrics for applying the CAE (without quantization) to a single channel of EMG. For each subject, the results were averaged across all eight channels. A window size of 213ms (256 samples) was used to form 16 × 16 EMG matrices as the CAE input and target. CR values with multi-channel input data were not necessarily attainable with single-channel input data, due to different input matrix sizes, along with the compressed data size being an integer. Consequently, CR values of 16, 32, 128, and 256 were explored (note that CR = 256 represents compressing each window to only one sample). For similar CR values, Table III results are comparable to those of multi-channel in Table II (see discussion).

Table IV(A) lists the CAE's PRDN after quantization of the compressed data which reduced the bit resolution of the compressed data from 24-bit to either 4, 6 or 8-bit. For comparison, the impact of direct quantization of EMG data alone (without applying CAE or any other compression method), have also been included in Table IV(B) (see Discussion).

In Table IV(A), the 4-bit quantization of the CAE's compressed data caused degradation ($p<0.001$) in PRDN, whereas an 8 or 6-bit quantization did not appreciably ($p>0.05$) deteriorate PRDN. Clearly, the choice of quantization resolution depends on the given tolerance for performance degradation. The 6-bit quantization (resulting in a 4-fold increase of CR) was selected here to attain maximum possible CR without noticeable performance reduction. Table V presents the results for 6-bit quantization of the compressed data for the CAE, along with those for wavelet-thresholding, where the wavelet coefficients have been quantized, as well as HEVC performance with 6-bit quantization. Fig. 3 depicts the CAE results (6-bit quantization) for individual subjects.

For the EMG-128 dataset, the performance metrics (averaged across all subjects) along with standard deviation (N = 18) for the CAE, without and with quantization (6-bit) of the compressed data, are listed in Table VI(A) and (B), respectively. As described previously, the CR values are different here, due to the different input sizes. The listed CR values have been rounded to the nearest integer. Note that the EMG-128 dataset resolution was 16-bit, therefore a 6-bit quantization of the compressed data increased the CR by a factor of 2.67. These findings demonstrate comparable performance to that of the EMG-8 dataset.

Fig. 4 shows the PRDN for the inter-subject case (CR = 1600), where a CAE trained on aggregated data from the other nine subjects and was tested on data from the remaining user. For comparison, the within-subject PRDN are also plotted, where the training and testing data were from the same subject. For CR = 1600, the within-subject and inter-subject PRDN were 31.5±0.4 and 34.1±1.6, respectively, and the difference between them was significant ($p<0.001$).

Table VII lists the number of learnable parameters (weights and biases) in the decoder part of the CAE for each CR. Table VIII shows the CA when training a CNN classifier on the original data but testing it on the reconstructed data. In this case, the results were significantly lower ($p<0.001$) than when training and testing on the reconstructed data alone, as previously shown in Table II. Table IX shows the CA, for training and testing a CNN classifier on the compressed data itself (for CR = 1600, an LDA classifier was used as the input size was 1). Again, these results were significantly lower ($p<0.001$) than when using the reconstructed data for classification as listed in Table II.

Table X lists the processing time for each method for CR = 6400. Wavelet-thresholding and HEVC do not require algorithm training, therefore only testing times are reported. For comparison, test times (total processing time for encoding and decoding) were computed for an 8-channel EMG segment of 167 ms (200 samples), i.e., 1600 samples in total. The CAE training time was computed for training a CAE on approximately 4000 input matrices of size 8 × 200.

IV. DISCUSSION

A deep CAE-based approach was proposed for compressing EMG signals, and proved to be highly efficient. For CR = 1600 (without quantization), it achieved a *PRDN* of 31.5%, and the

TABLE IV

PRDN AVERAGED ACROSS ALL SUBJECTS, ALONG WITH STANDARD DEVIATIONS (N = 10) FOR (A) THE CAE WITH 4, 6, AND 8-BIT COMPRESSED DATA, (B) DIRECT QUANTIZATION OF EMG DATA ALONE, USING 4,6, AND 8-BIT WITHOUT APPLYING ANY COMPRESSION METHOD

(A)

CR before quantization	4-bit	6-bit	8-bit
16	28.1 ± 0.8	26.6 ± 0.5	26.5 ± 0.5
32	28.8 ± 0.8	27.2 ± 0.6	27.1 ± 0.6
100	31.0 ± 0.7	29.8 ± 0.6	29.7 ± 0.6
200	32.1 ± 0.5	31.1 ± 0.7	31.1 ± 0.6
1600	32.0 ± 0.4	31.5 ± 0.4	31.5 ± 0.4

(B)

4-bit	6-bit	8-bit
83.7 ± 11.2	22.2 ± 1.3	5.8 ± 0.2

TABLE V

CAE, WAVELET-THRESHOLDING (WT), AND HEVC WITH 6-BIT COMPRESSED DATA PERFORMANCE METRICS AVERAGED ACROSS ALL SUBJECTS, ALONG WITH STANDARD DEVIATIONS (N = 10)

	PRDN			SNR		
	CAE	WT	HEVC	CAE	WT	HEVC
64	26.6 ± 0.5	42.8 ± 2.6	39.8 ± 6.8	26.5 ± 0.4	17.0 ± 1.3	18.7 ± 3.6
128	27.2 ± 0.6	54.5 ± 3.4	49.4 ± 3.0	26.1 ± 0.4	12.2 ± 1.3	14.2 ± 1.2
400	29.8 ± 0.6	73.4 ± 3.3	72.0 ± 2.5	24.2 ± 0.4	6.2 ± 0.9	6.6 ± 0.7
800	31.1 ± 0.7	82.3 ± 3.1	81.3 ± 2.2	23.4 ± 0.4	3.9 ± 0.8	4.2 ± 0.6
6400	31.5 ± 0.4	94.2 ± 2.3	96.3 ± 0.8	23.1 ± 0.3	1.2 ± 0.5	0.7 ± 0.2

	QS			CA (Orig: 90.5 ± 1.3)		
	CAE	WT	HEVC	CAE	WT	HEVC
64	2.4 ± 0.05	1.5 ± 0.1	1.7 ± 0.3	86.9 ± 1.6	61.3 ± 3.4	61.2 ± 6.1
128	4.7 ± 0.1	2.4 ± 0.2	2.6 ± 0.2	86.4 ± 1.8	55.5 ± 3.2	49.6 ± 2.7
400	13.4 ± 0.3	5.5 ± 0.3	5.6 ± 0.2	85.6 ± 1.4	40.1 ± 3.3	40.8 ± 3.4
800	25.7 ± 0.5	9.7 ± 0.4	9.9 ± 0.3	85.3 ± 1.5	32.9 ± 2.3	41.6 ± 4.5
6400	203.3 ± 2.7	68.0 ± 1.7	66.4 ± 0.6	85.5 ± 1.9	23.1 ± 1.6	38.9 ± 12.9

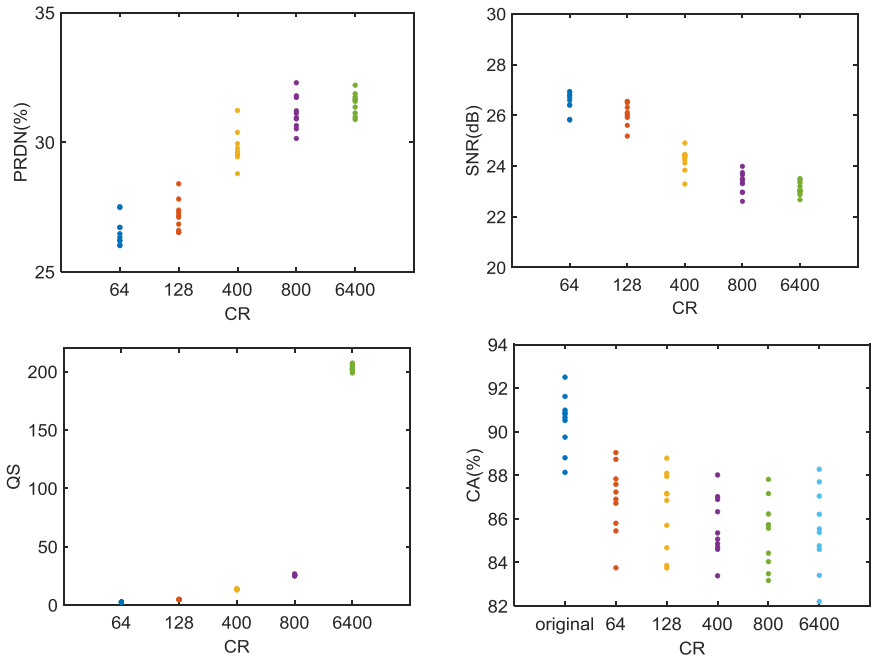

Fig. 3. The CAE with 6-bit resolution compressed data performance metrics for individual subjects. Each subject is shown with a point.

TABLE VI
EMG-128 DATASET: THE CAE PERFORMANCE METRICS AVERAGED ACROSS ALL SUBJECTS, ALONG WITH STANDARD DEVIATIONS (N = 18)

(A)

	PRDN	SNR	QS
16	24.0 ± 3.5	28.8 ± 3.2	0.7 ± 0.1
33	25.4 ± 4.3	27.7 ± 3.7	1.3 ± 0.3
133	27.1 ± 3.8	26.4 ± 3.4	5.1 ± 1.1
266	29.9 ± 4.5	24.5 ± 3.8	9.2 ± 2.3
1066	34.5 ± 6.2	21.7 ± 4.5	32.6 ± 10.1

(B)

	PRDN	SNR	QS
44	25.5 ± 5.4	27.7 ± 4.2	1.8 ± 0.4
89	26.7 ± 5.6	26.8 ± 4.3	3.5 ± 0.8
355	27.8 ± 4.6	26.0 ± 3.8	13.2 ± 3.0
709	30.0 ± 4.6	24.4 ± 3.8	24.5 ± 6.2
2837	34.5 ± 6.2	21.7 ± 4.6	86.6 ± 26.9

(A) Without Quantization, (B) with 6-bit Quantization of the Compressed Data.

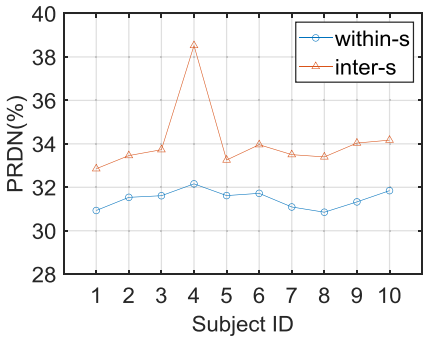

Fig. 4. For CR = 1600, the PRDN for the inter-subject and within-subject are plotted for individual subjects.

TABLE VII
NUMBER OF LEARNABLE PARAMETERS (WEIGHTS AND BIASES) IN THE DECODER PART OF THE CAE FOR EACH CR

	Number of weights and biases
16	19969
32	19393
100	19969
200	19393
1600	28353

classification accuracy of the wrist motions was reduced by only 5.0% when using the reconstructed data, with respect to that of the original data. As context, for CR = 1600, this means that an EMG data file with size 11.6MB would be reduced to 6.2 KB. These results show that the proposed CAE was able to automatically learn the end-to-end process of encoding and decoding from the EMG data. During the network training, the weights, and hence the associations between feature maps of different layers, were learned to minimize the reconstruction error. For compression, the model exploits EMG temporal redundancy, by learning temporal correlation within EMG segments (167 ms). Also, the model uses redundancy between different EMG channels, by learning inter-channel correlations within

TABLE VIII
CA (%) FOR A CNN CLASSIFIER TRAINED ON THE ORIGINAL DATA AND TESTED ON THE RECONSTRUCTED DATA

	CA (Orig: 90.5 ± 1.3)
16	82.8 ± 2.4
32	81.8 ± 3.0
100	78.8 ± 3.1
200	76.8 ± 3.9
1600	75.0 ± 3.9

TABLE IX
CA (%) FOR A CNN CLASSIFIER TRAINED AND TESTED ON THE COMPRESSED DATA (FOR CR = 1600, AN LDA CLASSIFIER WAS USED)

	CA (Orig: 90.5 ± 1.3)
16	77.7 ± 1.3
32	76.9 ± 1.5
100	71.4 ± 2.2
200	65.5 ± 3.8
1600	28.4 ± 2.6

TABLE X
PROCESSING TIME AVERAGED ACROSS ALL SUBJECTS, ALONG WITH STANDARD DEVIATION (N = 10) ARE SHOWN FOR CAE, WAVELET-THRESHOLDING, AND HEVC, FOR CR = 6400

	CAE	WT	HEVC
Train	61 s ± 1.0 s	-	-
Test	6.8 ms ± 0.5 ms	3.9 ms ± 0.4 ms	11.3 ms ± 1.0 ms

EMG matrices. For instance, for CR = 1600, each EMG segment (8 × 200 = 1600 samples) was encoded with a single sample, incorporating useful information from all channels. The dimensionality reduction is performed by the pooling layers which decrease the size of the feature maps and thereby remove redundant features learned through the convolutional layers.

Further compression was achieved by reducing the bit resolution, through quantization of the compressed data. It was observed that reducing the bit resolution of the compressed data to 6-bit, did not significantly deteriorate the performance metrics, despite increasing the CR 4-fold, as the EMG data were recorded with 24-bit resolution. With quantization, CR = 6400 was achieved while demonstrating PRDN and CA performances similar to those of CR = 1600 (without quantization). The impact of changing bit resolution of EMG signals on pattern recognition has previously been investigated in [34], where they found that reducing the bit resolution from 16 to 8, resulted in no degradation of the pattern recognition accuracy, which is consistent with our results.

The CAE architecture was found empirically. The performance was found to be robust to modest variations in the network structure, e.g., moderate changes in the number of filters in convolutional layers did not change the performance appreciably. Consequently, it is expected that small changes in EMG preprocessing, which may not significantly alter the shape of the signal, would not require changing the network architecture. Also, EMG data recorded with different systems may require slight structural modifications.

Table V shows that with increasing CR 100-fold (from 64 to 6400), the CAE reconstruction performance degraded relatively slightly (PRDN degraded from 26.6% to 31.5%, and SNR reduced from 26.5 dB to 23.1 dB), which in turn resulted in only subtle reduction of CA from 86.9% to 85.5%. This highlights the CAE outstanding performance in larger CRs. Due to relatively small change in PRDN with increasing CR 100-fold, QS as the ratio of CR to PRDN grew substantially (from 2.4 to 203.3). The same reasoning applies to the CAE results without quantization (Table II) which shows the same trend.

A comparison between Tables II and III indicates comparable performance between single-channel and multi-channel EMG data. The maximum CR in both cases represent compressing the input EMG matrix into only one sample. The reason for higher maximum CR with multi-channel EMG is the larger input size (1600 vs. 256). Thus, greater CRs will be possible for single-channel EMG, by increasing the EMG segment length. Moreover, the HD-EMG results (Table VI) demonstrate performance comparable to that of 8-channel EMG, and indicate the CAE compatibility with various dimensionality of data.

The proposed approach substantially outperformed HEVC and the wavelet-thresholding technique, especially for higher CR, where the performance difference between the CAE and other two methods increased considerably. In the literature on EMG compression, the state-of-the-art HEVC and well-known coding techniques such as applying wavelet, DCT, or other transforms have been employed for dimensionality reduction with mediocre success (CR$<$50). The powerful compression efficacy of the proposed approach, in comparison to conventional methods, indicates the superiority of CAEs for EMG dimensionality reduction. CAEs have been previously employed for ECG compression [26], which attained CR $=$ 32.25 with PRDN $=$ 31.2%. Nevertheless, our proposed method shows vast improvements for multi-channel EMG, achieving CR $=$ 6400 with PRDN $=$ 31.5%.

A comparison between Tables II and V results showed that for the CAE and wavelet-thresholding, a 6-bit quantization of the compressed data did not noticeably ($p>0.05$) degrade the performance metrics. However, direct quantization of EMG data alone (without applying any compression), presented in Table IV(B), showed significant degradation of the PRDN ($p<0.001$) with respect to the original data (PRDN $=$ 0). Interestingly, for 4-bit quantization of EMG data, the PRDN was 83.7 (Table IV(B)), whereas for 4-bit quantization of the CAE's compressed data, along with compression, the PRDN was substantially superior (between 28.14 and 31.99 for different CRs as shown in Table IV(A)). Histogram studies revealed that the CAE's compressed data were spread much more uniformly across their range, whereas the EMG data were mostly concentrated around low amplitudes. Consequently, quantization of the CAE's compressed data was drastically more efficient than quantization of EMG data. The results also demonstrate the efficacy of quantization of wavelet coefficients, as 6-bit quantization did not appreciably ($p>0.05$) change the performance of wavelet-thresholding.

The prevalence of multiple channels of EMG has likely further benefited the CR performance here, because the CAE does not compress each channel individually, and instead leverages the inter-channel correlations to conduct both spatial and temporal compression of the input data matrices.

The PRDN for the inter-subject case (CR $=$ 1600) degraded by only 2.6% with respect to that of the within-subject (Fig. 4). Such a high inter-subject performance is consistent with the expectation that, despite the common use of within-subject pattern recognition models for myoelectric control, similar features are learned across all subjects. Therefore, for all subjects, the same network weights can be used for compression with high efficiency. This improves the usability and generalizability of the proposed approach, enabling the application of a pretrained compression model without the need for retraining.

Table VII lists the number of learnable parameters in the decoder part of the CAEs. The decoder weights and biases must be known for data reconstruction; hence they must be stored or transmitted to the receiver, along with the compressed data. However, this process is needed only once for a user, and consequently, its impact on true CR can be neglected in realistic scenarios; e.g., the size of 3 seconds of 8-channel EMG data is 28800 samples, higher than the number of parameters for the largest CAE investigated (Table VII). Moreover, for the inter-subject case, no CAE parameters are transmitted to the receiver as these parameters are already known, due to the use of a pretrained CAE.

A comparison of the results in Tables II and VIII, indicates that classification of the reconstructed data degrades if the classifier was trained with the original data. This was expected but suggests that there are some meaningful differences between the original and reconstructed EMG data. Similarly, a comparison of the results in Tables II and IX shows that the classification of the data in its compressed state was significantly lower than once it is reconstructed. This was anticipated as the abstract representation was not learned with the goal of minimizing the classification loss. Nevertheless, once reconstructed, it is apparent that the classification-relevant information in the EMG is retained.

The recommended EMG segment length in the literature, for pattern recognition, is 100-250ms, to allow the learning of informative EMG patterns without noticeable delays. Similarly, the CAE needs to learn useful EMG features for compression, hence the segment length is recommended to be at least 100ms. On the other hand, the segment length must be kept short to avoid performance degradation due to limited training samples, unless the dataset is very large. Moreover, real-time compression applications necessitate short segments. Considering these requirements, here we have selected EMG segment lengths in the range 100-250 ms.

The advantage of the proposed method over previous compression techniques is achieving significantly higher performance especially for larger CRs as listed in Tables II and V. The disadvantage of this method however, is the time and computational cost of training the CAE, and the need for sufficient data to train the CAE. As a solution to avoid training, a pretrained CAE (with data from previous users) can be leveraged (Fig. 4).

The total computer processing time for encoding and decoding of each 8-channel EMG segment (167 ms) when CR $=$ 6400 was only 6.8 ms, similar to that of wavelet-thresholding (3.9 ms)

and HEVC (11.3 ms). This fast CAE processing in Matlab suggests the future viability of near real-time applications such as myoelectric control via cloud computing and online monitoring of EMG or other biomedical data in order to perform diagnosis or treatment via telemedicine.

V. Conclusion

A deep CAE-based approach was proposed for compression of EMG signals that substantially improved the compression efficacy with respect to previous methods. The CAE can automatically learn the encoding and decoding processes from raw data, yielding high inter-subject performance, enabling the translation of the model to new users. Because of low computational complexity and consequently fast processing, this approach may be beneficial for reducing storage and network traffic demands.

References

[1] Facebook Reality Labs, "Inside Facebook reality labs: Wrist-based interaction for the next computing platform," 2021. Accessed: Jan. 22, 2021. [Online]. Available: https://tech.fb.com/inside-facebook-reality-labs-wrist-based-interaction-for-the-next-computing-platform/

[2] A. Phinyomark, G. Petri, E. Ibáñez-Marcelo, S. T. Osis, and R. Ferber, "Analysis of big data in gait biomechanics: Current trends and future directions," *J. Med. Biol. Eng.*, vol. 38, no. 2, pp. 244–260, 2018.

[3] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic, "Deep learning applications and challenges in big data analytics," *J. Big Data*, vol. 2, no. 1, pp. 1–21, 2015.

[4] M. Atzori *et al.*, "Characterization of a benchmark database for myoelectric movement classification," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 23, no. 1, pp. 73–83, Jan. 2015.

[5] W. Geng, Y. Du, W. Jin, W. Wei, Y. Hu, and J. Li, "Gesture recognition by instantaneous surface EMG images," *Sci. Rep.*, vol. 6, no. 1, pp. 1–8, 2016.

[6] T. J. Lynch, *Data Compression. Techniques and Applications*. New York, NY, USA: Van Nostrand/Reinhold, 1985.

[7] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proc. IRE*, vol. 40, no. 9, pp. 1098–1101, 1952.

[8] T. A. Welch, "A technique for high-performance data compression," *Computer*, vol. 17, no. 6, pp. 8–19, 1984.

[9] L. Brechet, M.-F. Lucas, C. Doncarli, and D. Farina, "Compression of biomedical signals with mother wavelet optimization and best-basis wavelet packet selection," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 12, pp. 2186–2192, Dec. 2007.

[10] J. P. L. M. Paiva, C. A. Kelencz, H. M. Paiva, R. K. H. Galvao, and M. Magini, "Adaptive wavelet EMG compression based on local optimization of filter banks," *Physiol. Meas.*, vol. 29, no. 7, pp. 843, 2008.

[11] J. A. Norris, K. B. Englehart, and D. F. Lovely, "Myoelectric signal compression using zero-trees of wavelet coefficients," *Med. Eng. Phys.*, vol. 25, no. 9, pp. 739–746, 2003.

[12] P. Wellig, Z. Cheng, M. Semling, and G. Moschytz, "Electromyogram data compression using single-tree and modified zero-tree wavelet encoding," in *Proc. 20th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. Vol. 20 Biomed. Eng. Towards Year 2000 Beyond (Cat. No. 98CH36286)*, Hong Kong, China, 1998, vol. 3, pp. 1303–1306.

[13] T. Grönfors, M. Reinikainen, and T. Sihvonen, "Vector quantization as a method for integer EMG signal compression," *J. Med. Eng. Technol.*, vol. 30, no. 1, pp. 41–52, 2006.

[14] E. P. Ntsama, W. Colince, and P. Ele, "Comparison study of EMG signals compression by methods transform using vector quantization, SPIHT and arithmetic coding," *SpringerPlus*, vol. 5, no. 1, 2016, Art. no. 444.

[15] P. de A Berger, A. d. O. Francisco, J. C. do Carmo, and A. F. da Rocha, "Compression of EMG signals with wavelet transform and artificial neural networks," *Physiol. Meas.*, vol. 27, no. 6, 2006, Art. no. 457.

[16] M. H. Trabuco, M. V. C. Costa, and F. A. de Oliveira Nascimento, "S-EMG signal compression based on domain transformation and spectral shape dynamic bit allocation," *Biomed. Eng. Online*, vol. 13, no. 1, 2014, Art. no. 22.

[17] B. Eddie Filho, E. A. da Silva, and M. B. de Carvalho, "On EMG signal compression with recurrent patterns," *IEEE Trans. Biomed. Eng.*, vol. 55, no. 7, pp. 1920–1923, Jul. 2008.

[18] E. S. Carotti, J. C. De Martin, R. Merletti, and D. Farina, "Compression of multidimensional biomedical signals with spatial and temporal codebook-excited linear prediction," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 11, pp. 2604–2610, Nov. 2009.

[19] M. V. C. Costa, J. L. A. d. Carvalho, P. d. A. Berger, A. F. d. Rocha, and F. A. d. O. Nascimento, "Compression of surface electromyographic signals using two-dimensional techniques," *Recent Adv. Biomed. Eng*. Vienna: InTech, Oct. 2009, pp. 17–38.

[20] A. M. Dixon, E. G. Allstot, D. Gangopadhyay, and D. J. Allstot, "Compressed sensing system considerations for ECG and EMG wireless biosensors," *IEEE Trans. Biomed. Circuits Syst.*, vol. 6, no. 2, pp. 156–166, Apr. 2012.

[21] M. H. Trabuco, M. V. Costa, B. Macchiavello, and F. A. d. O. Nascimento, "S-EMG signal compression in one-dimensional and two-dimensional approaches," *IEEE J. Biomed. Health Informat.*, vol. 22, no. 4, pp. 1104–1113, Jul. 2017.

[22] A. Phinyomark and E. Scheme, "EMG pattern recognition in the era of big data and deep learning," *Big Data Cogn. Comput.*, vol. 2, no. 3, 2018, Art. no. 21.

[23] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, "A deep transfer learning approach to reducing the effect of electrode shift in EMG pattern Recognition-based control," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 28, no. 2, pp. 370–379, Feb. 2020.

[24] M. Atzori, M. Cognolato, and H. Müller, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Front. Neurorobot.*, vol. 10, 2016, Art. no. 9.

[25] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*. Cambridge, U.K.: MIT Press, 2016.

[26] O. Yildirim, R. San Tan, and U. R. Acharya, "An efficient compression of ECG signals using deep convolutional autoencoders," *Cogn. Syst. Res.*, vol. 52, pp. 198–211, 2018.

[27] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, "Regression convolutional neural network for improved simultaneous EMG control," *J. Neural Eng.*, vol. 16, no. 3, 2019, Art. no. 036015.

[28] M. Atzori *et al.*, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Sci. Data*, vol. 1, no. 1, pp. 1–13, 2014.

[29] D. L. Donoho and J. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 81, no. 3, pp. 425–455, 1994.

[30] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[31] Fraunhofer Heinrich Hertz Institute, "High efficiency video coding (HEVC)," 2021. Accessed: Jan. 22, 2021. [Online]. Available: https://hevc.hhi.fraunhofer.de

[32] A. Ameri, M. A. Akhaee, E. Scheme, and K. Englehart, "Real-time, simultaneous myoelectric control using a convolutional neural network," *PLoS One*, vol. 13, no. 9, 2018, Art. no. e0203835.

[33] E. Scheme and K. Englehart, "Electromyogram pattern recognition for control of powered upper-limb prostheses: State of the art and challenges for clinical use," *J. Rehabil. Res. Develop.*, vol. 48, no. 6, pp. 643–659, 2011.

[34] A. W. Wilson, Y. G. Losier, P. A. Parker, and D. F. Lovely, "A bus-based smart myoelectric electrode/amplifier—System requirements," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 10, pp. 3290–3299, Nov. 2011.