# Bioformers: Embedding Transformers for Ultra-Low Power sEMG-based Gesture Recognition

Alessio Burrello*, Francesco Bianco Morghet†, Moritz Scherer‡, Simone Benatti§,
Luca Benini*‡, Enrico Macii¶, Massimo Poncino†, Daniele Jahier Pagliari†
* DEI, Università di Bologna, Bologna, Italy
† Department of Control and Computer Engineering, Politecnico di Torino, Turin, Italy
‡ Integrated Systems Laboratory, ETH Zurich, Switzerland
§ Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, Italy
¶Interuniversity Department of Regional and Urban Studies and Planning, Politecnico di Torino, Turin, Italy
Emails: name.surname@unibo.it, name.surname@polito.it, simone.benatti@unimore.it, scheremo@iis.ethz.ch

***Abstract*—Human-machine interaction is gaining traction in rehabilitation tasks, such as controlling prosthetic hands or robotic arms. Gesture recognition exploiting surface electromyographic (sEMG) signals is one of the most promising approaches, given that sEMG signal acquisition is non-invasive and is directly related to muscle contraction. However, the analysis of these signals still presents many challenges since similar gestures result in similar muscle contractions. Thus the resulting signal shapes are almost identical, leading to low classification accuracy. To tackle this challenge, complex neural networks are employed, which require large memory footprints, consume relatively high energy and limit the maximum battery life of devices used for classification. This work addresses this problem with the introduction of the Bioformers. This new family of ultra-small attention-based architectures approaches state-of-the-art performance while reducing the number of parameters and operations of 4.9×. Additionally, by introducing a new inter-subjects pre-training, we improve the accuracy of our best Bioformer by 3.39%, matching state-of-the-art accuracy without any additional inference cost.**

**Deploying our best performing Bioformer on a Parallel, Ultra-Low Power (PULP) microcontroller unit (MCU), the GreenWaves GAP8, we achieve an inference latency and energy of 2.72 ms and 0.14 mJ, respectively, 8.0× lower than the previous state-of-the-art neural network, while occupying just 94.2 kB of memory.**

***Index Terms*—Transformers, sEMG, Gesture Recognition, Deep Learning, Embedded Systems**

## I. Introduction

In the last few years, thanks to the availability of increasingly powerful yet energy-efficient devices, there has been a consistent trend towards moving computations to the edge, therefore eliminating the need of relying on a centralized computational unit [1]. One of the fields that benefits the most from this new paradigm is personalized healthcare. Many applications, such as heart rate (HR) monitoring [2], have been moved to wearable devices, reducing overall energy consumption thanks to the drastic reduction of raw data communication. Further, thanks to edge computing, new applications such as closed-loops brain stimulations, which require reliable, low-latency processing, become viable [3].

Human Machine Interfaces (HMIs) can be extremely useful in personalized healthcare, enabling new types of interactions for impaired patients, for example, through hand gestures [4]. A reliable approach for gestures recognition exploits surface electromyographic signal (sEMG), which has been demonstrated to be strongly correlated to the different arms positions and muscle contractions. In this approach, a pre-defined set of gestures is selected, and training data are collected letting subjects perform the different gestures following a pre-defined pattern during the recording sessions. Then, a classification algorithm is trained to distinguish the different hand gestures based on the sEMG signals gathered. Currently, Deep Learning (DL) algorithms [5]–[7] are the state-of-the-art for this task.

However, most of these advanced algorithms are too computationally expensive to be deployed on memory-constrained edge devices, occupying a too large memory footprint or necessitating a high amount of operations, hampering the battery lifetime of edge devices [5], [7], [8]. For this reason, several recent works focus on designing lightweight yet accurate DL models that can be deployed on edge devices. These works typically target platforms with less than 1MB memory and with a power envelope in the order of tens of mW [9]. Such tight constraints are typically met leveraging optimizations at different levels, which affect the model architecture (e.g., Neural Architecture Search and pruning [2]) and the bit-width used for storing and processing its parameters and intermediate outputs (quantization). These combined optimizations have recently enabled the deployment of several highly accurate models for healthcare applications at the edge [2].

However, there is a clear pace difference in terms of DL model innovation between the cloud and the edge. On large scale classification problems, such as natural language processing (NLP) and, more recently, computer vision (CV), Transformer networks are quickly becoming state-of-the-art, outclassing all competitors in terms of accuracy. These results are obtained thanks to models such as BERT [10], GPT-3 [11] and the VisionTransformer (ViT) [12], which include hundreds of millions, or even billions of parameters. In contrast, most successful examples of DL deployment at the edge are based on Convolutional Neural Networks (CNNs) [13], [14].

In this work, we aim to demonstrate that Transformers are also suitable for smaller-scale problems and that they can achieve state-of-the-art performance in a TinyML gesture recognition scenario with lower computational complexity compared to CNNs. Furthermore, inspired by the large-scale pre-

This work was supported in part by the EU Grant Bonsapp (g.a. no. 101015848).

training typically applied to transformers for CV and NLP, we design a new pre-training strategy for sEMG-based gesture recognition. Specifically, the contribution of our work is threefold:

- We introduce a novel DL architecture for gesture recognition, the *Bioformer*, which exploits the attention mechanism to reduce computational complexity while achieving state-of-the-art gestures recognition results.
- We introduce an inter-subject pre-training step to improve deep learning architectures' representation capability in gesture recognition. While this task is typically patient-specific, we show that employing data from other subjects aids the network to extract more significant and generalizable features.
- We demonstrate the advantages of using an initial 1D-convolutional layer to aggregate raw signals in a series of projections to feed the transformer network. Specifically, this layer increases accuracy and reduces complexity simultaneously.

Testing our architecture on the Ninapro DB6 dataset, which includes eight grasp gestures from 10 subjects, our best network achieves 62.34% accuracy, further improved to 65.73% thanks to the inter-subject pre-training. Quantized to 8bits, it occupies as little as 94.2 kB, which is 4.9× lower than previous state-of-the-art CNN, TEMPONet [15], [16], achieving 65.0% on the same task. Deployed on the GAP8 multi-core MCU, the same Bioformer only consumes 0.139 mJ per inference, being 8.0× more efficient than TEMPONet.

## II. Background & Related Work

### A. Surface Electromyographic Signal

EMG signals [17] originate from the electrical activity that occurs during a muscular contraction, ranging from $10\,\mu V$ to $1\,mV$ with a bandwidth of $\sim 2\,kHz$ for standard applications, even though it is possible to acquire EMG data up to $\sim 10\,kHz$ in Motor Unit Action Potential Analysis. EMG activity is acquired by conductive plates (i.e. electrodes) placed on the skin surface that collect the underlying electrical muscular activity. A major issue of signal acquisition is related to the skin-electrode interface, which is prone to high variability and can degrade signal quality. Also, electrode re-positioning and user adaptation [15], as well as motion artefacts and floating ground noise, represent major causes of signal degradation and variability.

### B. Related Work

In the last few years, sEMG-based hand gesture recognition has gained traction in academia and commercial applications. Early approaches rely on conventional ML methods, such as Support Vector Machine (SVM), Random Forests (RFs), Linear Discriminant Analysis (LDA) or shallow artificial neural networks (ANN) [18]–[20]. Even though SoA recognition accuracy is above 90%, most experiments are limited to a single-session setup, failing to cope with the inter-session accuracy drop observed when gesture inference is made on sessions never seen during training [19]. In fact, one of the major challenges in sEMG-based gesture recognition is managing the performance drop due to electrode multiday donning-doffing [20], [21].

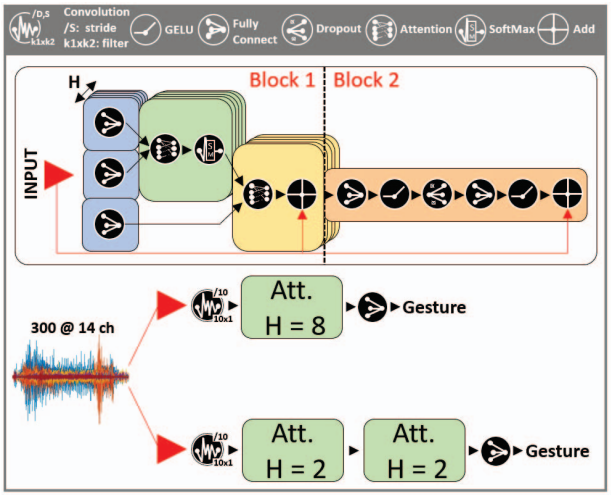



Fig. 1: In the upper part, the basic MHSA layer used inside our architectures. In the lower part, the two Bioformers architectures that we propose as benchmarks.

To tackle this issue, some approaches rely on extending the training datasets or increasing the electrode count, obtaining larger sets of features that improve algorithm convergence [18]. However, performance drops are still consistent, and the lack of generalization hinders the deployment of these solutions in real-world scenarios. Multi-session training is a promising strategy explored by virtue of the release of several multi-session sEMG datasets, such as the Non-Invasive Adaptive hand Prosthetics Database 6 (NinaPro DB6, 10 sessions, 8 classes) [21] and the University of Bologna - INAIL (Unibo-INAIL) database (8 days × 4 arm postures, 6 gestures) [20]. These works show increased accuracy recognition over time but strongly rely on domain-specific knowledge and hand-crafted features. To increase model robustness and eliminate the need for hand-crafted feature extraction, DL-based models represent a viable solution, also prompted by the availability of relatively large public multi-session datasets. For instance, [22] propose Convolutional Neural Networks (CNN), which outperform a SVM in classification accuracy across several subjects on a public dataset. More recently, Temporal Convolutional Networks (TCNs), variants of a CNN widely used for time-series analysis [7], [8], are gaining traction also in sEMG signal processing, showing high accuracy in multi-session problems. Even though DL-based approaches tackle the EMG variability problem successfully, they are based on large models. Hence, their deployment on real-time, resource-constrained edge platforms, such as wristbands or smartwatches, is still a non-trivial task.

### C. Attention & Transformers

In 2017, [23] demonstrated the possibility to exploit a neural network solely based on the *attention* mechanism to improve the performance of different language modelling tasks. In DL, the concept of attention refers to layers that generate input-dependent synaptic weights, resulting in a variable relation among inputs that depends on the relative importance from the

point of view of the target task (i.e., paying "more attention" to the most important inputs). In [23], the authors exploited in particular the so-called Multi-Heads Self-Attention (MHSA) blocks that analyze the relationship of different parts of the input data among themselves. Similarly, in our work, we employ the MHSA as the basic building block of our architectures.

Given a tensor $\mathbf{X}$, with dimension $S \times C$, where $S$ is the *input (or sequence) length* and $C$ the number of *channels*, the MHSA produces an output of the same shape $S \times C$. It comprises two main blocks, as shown on the top of Fig. 1. The first one uses a set of parallel and independent heads, $H$, all of which perform a series of three operations on the input data (Blue, green and yellow rectangles in Fig.1). The first operation projects the sequence $\mathbf{X}$ into three separate *projection spaces* each of size $P$, using three trainable linear layers. These projections are called *queries* $\mathbf{Q}$, *keys* $\mathbf{K}$ and *values* $\mathbf{V}$, and are computed as:

$$\mathbf{Q} = \mathbf{X}\mathbf{W}_{\text{query}} \qquad \mathbf{K} = \mathbf{X}\mathbf{W}_{\text{key}} \qquad \mathbf{V} = \mathbf{X}\mathbf{W}_{\text{value}} \tag{1}$$

where $\mathbf{W}_{\text{query}}$ , $\mathbf{W}_{\text{key}}$ and $\mathbf{W}_{\text{value}}$ are all matrices of size $C \times P$. In the second and third operations (green and yellow blocks), the *scaled dot-product attention* combines $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$ as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \doteq \underset{\text{over keys}}{\text{SoftMax}}\left(\frac{\mathbf{Q}\mathbf{K}^{\text{T}}}{\sqrt{P}}\right)\mathbf{V}. \tag{2}$$

Noteworthy, the $\frac{\mathbf{Q}\mathbf{K}^{\text{T}}}{\sqrt{P}}$ block represents the so-called attention matrix, which is used to weigh each element of the *values* tensor with respect to all others based on the relative importance.

The second block comprises two linear layers (orange rectangle in Fig. 1) that process independently all heads produced as output by the scaled dot-product attention relative to the same sequence element, and project them first to a hidden space and then back to the space $\mathbb{R}^C$.

## III. Material and Methods

This paragraph introduces Bioformer, a Vision Transformer (ViT) [12] inspired architecture, which significantly reduces the computation complexity for sEMG-based gesture recognition, while reaching an accuracy comparable with the state-of-the-art. Further, we propose a new pre-training protocol to feed more data to our transformer. Finally, we provide few details about the experimental setup and the deployment of the networks.

### A. Bioformer: Network Topology

Our network comprises three modules. First, the input signal is projected onto a space of dimension $N \times 64$ by means of a 1D-convolutional layer. We use padding = 0 and stride equal to the filter dimension to aggregate non-overlapping windows of the input signal. Similarly to what is done in ViT for images, the idea is to create a series of $N$ tokens of dimension 64 that encode the input information. We tested [1, 5, 10, 20, 30] for the filter dimension. Note that the higher is the dimension, the smaller is the number $N$ of produced tokens. Therefore, the computation complexity of the following attention blocks' reduces for larger filters. Compared to standard transformers [23], tuning the dimension of this first layer increases the flexibility of the architecture, allowing to trade-off the total number of operations and the accuracy.

Then, the output of the 1D convolutional layer is processed by the self-attention blocks, described in Sec. II-C. In the rest of the paper, we focus on two variants of the Bioformer architecture, both of which exhibit good accuracy on sEMG-based gesture recognition. The parameters of the two networks are all identical, except for the number of heads and the number of layers (depth). The first network comprises one attention layer with eight heads, while the second one consists of two attention layers with two heads each. These two parameters have been chosen after performing a grid search on depth $\in \{1, 2, 3, 4\}$ and heads $\in \{1, 2, 4, 8\}$. We chose the architectures with the best trade-off of accuracy vs. parameters. The hidden space has dimension 128, while each head has a dimension $P$ of 32. Similarly to [12], a "class token" is concatenated after the QKV projection step, adding one sample to the sequence length ($(N + 1) \times 64$). The outputs corresponding to the class token are used to produce the network prediction. Having a dedicated token for prediction, instead of simply using the last (or first) input token, has been shown to yield higher accuracy in [12]. Intuitively, this solution gives higher flexibility to the class token output, which can learn to "pay attention" to relevant elements in the input sequence from the point of view of the classification.

The lower section of Fig. 1 summarizes these two network architectures.

### B. Bioformer: Training

The standard training for sEMG gesture recognition, regardless of the employed dataset, is subject-specific, given that the movements and muscle contractions associated with different gestures can differ significantly from one subject to another [7], [15]. On the other hand, it is known that performing a pre-training step on data similar to the ones used for the final training is highly beneficial for DL models, and in particular for Transformers [10]. For instance, typically, many state-of-the-art image recognition networks do not randomly initialize their weights, but go through a first training on the Imagenet dataset before fine-tuning on their target dataset, especially if the latter is small. The Imagenet pre-training allows the network to start fine-tuning from a set of well-initialized weights, which can already extract meaningful features for generic images. Similar pre-training + fine-tuning protocols are also key elements of most successful transformer models, e.g., in NLP [10], [11]. Noteworthy, pre-training not only helps to speed up the fine-tuning convergence but also yields higher final accuracy.

Based on these assumptions, this work introduces a new two-step training procedure for sEMG-based gesture recognition. Compared to the standard approach, we first perform an inter-subjects pre-training, in which data relative to all subjects available in the training dataset are employed. Then, we proceed with subject-specific fine-tuning, which is common to all state-of-the-art approaches. Despite the task being strictly subject-dependent, one can intuitively imagine that the sEMG signal features that are useful for gesture classification should be similar for all patients. Indeed, using this protocol, we observe that feeding more data to the network during pre-training is

beneficial for accuracy. To clarify better our proposed protocol, we report below the training procedure that we use for subject 1 of the 10-subjects Ninapro DB6 training dataset. First, we train the network for 100 epochs with data coming from patients 2-10, excluding subject 1, on which we want to test the final model. In this step, the model adjusts the weights to extract general features that classify the gestures of the other nine patients as accurately as possible. Then, we perform 20 epochs of fine-tuning using only the training data of subject 1. During this fine-tuning, the recording sessions of the patients are separated between train and test set, following the classical sequential training protocol used by other state-of-the-art approaches for this task, which mimics a real scenario, using sessions 1-5 for training and 6-10 for testing: sessions 1-5 are used, with their golden labels, for training. Then, the trained model is deployed and used to predict gestures belonging to the remaining sessions.

For the pre-training step, we use Adam optimizer with a linear warmup of the learning rate from 1e-7 to 5e-4; for the fine-tuning step, a fixed learning rate of 1e-4 is used, with a reduction of 10× after 10 epochs.

### C. Experimental Setup & Dataset

To validate our new architectures, we employ the public sEMG-based hand gesture recognition dataset called Non-Invasive Adaptive hand Prosthetics Database 6 (NinaPro DB6) [21], which has been explicitly realized to investigate the degradation of sEMG-based hand gesture recognition accuracy over time. The dataset includes 10 non-amputee subjects (3 females, 7 males, average age $27 \pm 6$ years), who have been asked to undergo 10 gathering sessions. The 10 sessions are distributed over 5 days, one in the morning, one in the afternoon, each including 12 repetitions of the gestures for each patient. The gestures considered include the rest position and seven grasps, covering hand movements typically done during daily activities. Each grasp repetition lasts approximately 6 s, followed by 2 s of rest. The array of sensors is composed of 14 Delsys Trigno sEMG Wireless electrodes, placed on the high half of the forearm, simulating the amputation of the lower half of the forearm. Each sensor gathers the data at a sampling rate of 2 kHz. The dataset is divided into windows of 150 ms (i.e., 300 samples) with a slide between them of 15 ms.

For training and validating the model using floating-point (fp32) arithmetic, we employed Python3.7 together with Pytorch1.8.1. We then perform few epochs of quantization aware training (QAT) to shift from fp32 to integer (int8) arithmetic. We follow the steps described in I-BERT [24] to replace the floating-point operators that compose MHSA layers with their int8 counterparts. Finally, we deployed the resulting quantized models on the GAP8 MCU, using the optimized kernels described in [25]. GAP8 is a commercial MCU from GreenWaves [9], comprising a controller unit called Fabric Controller (FC), composed of a single RISC-V core, which manages the peripherals and orchestrates the program execution, and a cluster of 8 identical RISC-V cores (with a shared 64 kB scratchpad memory) which can be activated to execute and accelerate intensive workloads. A 512kB main memory is shared between FC and cluster.

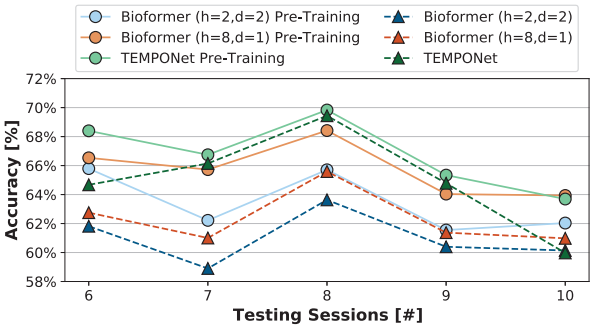



Fig. 2: Performance variation on the different testing sessions.

## IV. Experimental Results

In this section, we first demonstrate the performance of our networks on the Ninapro DB6. Then, we perform an ablation study to demonstrate i) the pre-training impact on the Bioformer (h = 8, d = 1) and ii) the influence of the filter dimension of the initial convolutional layer. Finally, we discuss the complexity of our architectures and their latency and energy when deployed on the GAP8 MCU.

### A. Ninapro DB6 benchmark

Fig. 2 reports the accuracy of our two Bioformers and of the state-of-the-art TEMPONet [15]. Each point corresponds to one of the five testing sessions, and the reported accuracy is the average across patients. Higher session numbers correspond to tests farther in time from the training period. Compared to the reference TCN, our Bioformers achieve slightly lower accuracy both with and without pre-training. Bioformers without pre-training achieve a 2.7%-3.9% lower accuracy on average. However, the accuracy difference w.r.t. TEMPONet decreases for sessions that are farther in time from the training and therefore more dissimilar from it. In particular, the h=8, d=1 Bioformer outperforms TEMPONet on testing session 10 (+ 0.48%). This result suggests that thanks to the reduced number of parameters (reported below) our architecture is more prone to well generalize on more dissimilar data, a key factor for a task where the data show high variability over time. Noteworthy, applying the pre-training is beneficial both for the proposed Bioformers and for TEMPONet. However, the accuracy difference between the two types of models decreases, confirming the superior capability of Transformer-based architectures to take benefit from pre-training with large amounts of data. On the different sessions, we observe an average gain of 3.39%, 2.48%, and 1.80% for Bioformer (h=8, d=1), Bioformer (h=2, d=2), and TEMPONet, respectively.

Overall, our best architecture (i.e., the one with 8 heads) achieves an average 65.73% accuracy, which is 0.73% better than the previous state-of-the-art TEMPONet, and 1.07% lower than the new pre-trained TEMPONet.

### B. Ablation Study: pre-training & Patch Dimension

In this paragraph, we detail i) the benefit of applying our new training approach and ii) the impact of the filter dimension of the initial 1D convolutional layer in Bioformers.

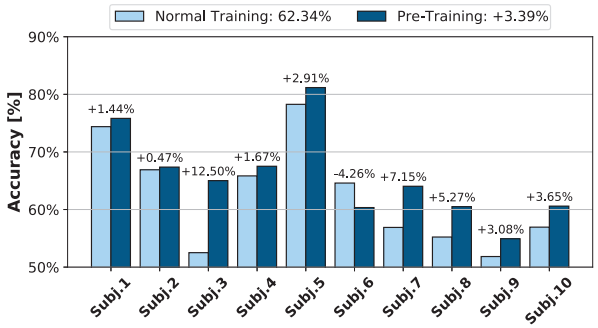



Fig. 3: Accuracy per subject with intra- and inter- patient training data.

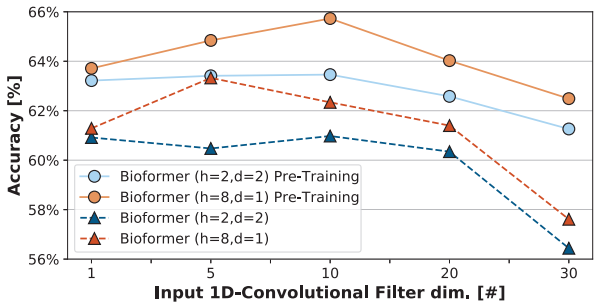
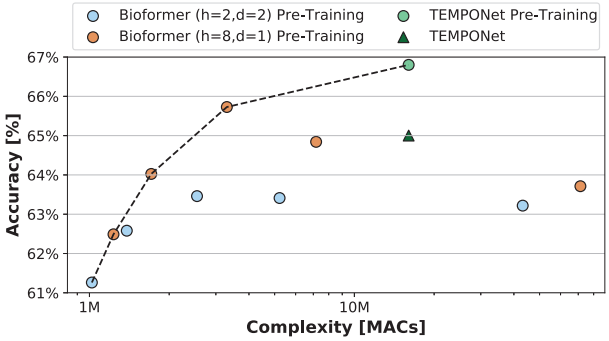



Fig. 4: Performance using [1,30] filter dimensions for the front-end convolutional layer. Increasing filter dimension reduces both the number of parameters and the number of operations.

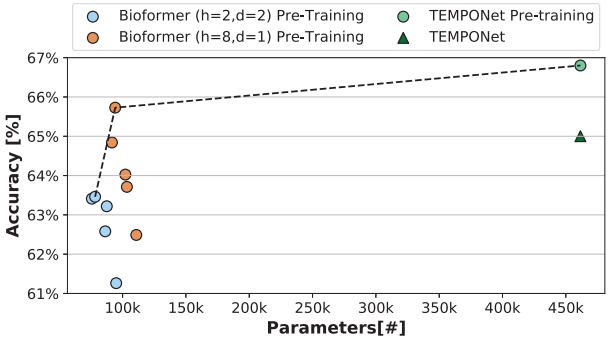
Fig. 3 details the performance change between standard training and our two-steps training for each subject. We can notice that the most significant advantages are obtained for subjects that present the lower accuracy before pre-training. On subjects whose starting accuracy is lower than 60%, the average accuracy improvement is 6.33%, while on the other five subjects, it is just 0.45%, leading to an overall average improvement of 3.39%. Solely the Subj.6's accuracy get worse with our new training. This could be caused by the lower learning rate used in the subject-specific fine-tuning that does not allow the network to converge to the global minimum.

In Fig. 4, we show the impact of the filter dimension of the first convolutional layer. Note that 1D convolution is always applied in a non-overlapping fashion in our networks. Therefore, a wider filter implies a smaller input signal for the attention block. Each solid line represents a Bioformer on which we applied the two-step training (pre-training and fine-tuning), whereas the dashed lines correspond to networks trained with the standard procedure. For most models, a filter dimension equal to 10 results in the best accuracy, despite its lower complexity compared to 1 and 5 (the resulting input sequence length is 30 instead of 60 and 300 for filter sizes 5 and 1[1], respectively). Furthermore, despite the resulting lower accuracy, increasing the filter dimension beyond 10 can be beneficial from the deployment point of view, given the reduction in the

[1]When a filter size of 1 is applied, the 1D-convolutional layer becomes a fully-connected embedding layer.




(a) Accuracy vs parameters.




(b) Accuracy vs MAC operations.

Fig. 5: Pareto spaces

algorithm's complexity, whose number of operations depends almost linearly on the sequence length. For instance, changing the dimension from 10 to 20, employing the Bioformer (h=8, d=1) only causes a drop of 1.70% of accuracy, while reducing the total number of operations by a factor 1.93×, and the energy by 2×, with a potentially very significant impact on the battery life of the device executing the inference.

### C. Deployment on GAP8

Fig. 5 shows different Bioformer architectures as well as TEMPONet in the N. of Operations versus accuracy and N. of parameters vs accuracy planes. While the pre-trained TEMPONet reaches the highest accuracy, all other Pareto points are populated by Bioformers. The different points plotted for the same Bioformer refer to different filter sizes of the initial 1D Convolutional layer. In the complexity versus accuracy space, we identified two key architectures of our Bioformers. Our most accurate model (h=8, d=1, filter = 10) outperforms the state-of-the-art TEMPONet and is only 1.07% less accurate than the pre-trained TEMPONet, but showing an impressive 4.9× operations reduction. The lightest Bioformer (h=2, d=2, filter = 10) on the Pareto frontier, instead, reduces the required number of operations of an additional factor 3.3× (16.17× lower than TEMPONet), at the cost of an additional 4.47% accuracy drop. Furthermore, in the lowermost graph, we can observe that all our models have a comparable number of parameters. In fact, the modification of the filter dimension of the 1D convolution only impacts the number of parameters of the first layer and

TABLE I: Performance of the quantized Pareto architectures on the GAP8 MCU. Bio1 corresponds to Bioformer (h=8, d=1), Bio2 to Bioformer (h=2, d=2).
Abbreviations: Lat.: latency, E.: energy, Q.Acc.: quantized accuracy.

| Network | Memory | MMAC | Lat.[ms] | E.[mJ] | Q. Acc. |
|---|---|---|---|---|---|
| MCU: GAP8, 100 MHz @ 1V, 51 mW | | | | | |
| Bio1, wind=30 | 110.8 kB | 1.2 | 1.03 | 0.052 | 61.09% |
| Bio1, wind=20 | 102.1 kB | 1.7 | 1.37 | 0.070 | 63.14% |
| Bio1, wind=10 | 94.2 kB | 3.3 | 2.72 | 0.139 | 64.69% |
| Bio2, wind=30 | 92.2 kB | 1.0 | 1.55 | 0.079 | 60.19% |
| Bio2, wind=10 | 78.3 kB | 2.5 | 4.82 | 0.246 | 62.43% |
| TEMPONet [15] | 461 kB | 16.0 | 21.82 | 1.11 | 61.00% |

of the linear layers contained in MHSA blocks, with a limited impact on the total model size.

The results of deploying some of these Pareto architectures on GAP8 are shown in Table I. The average power consumption while using the 8-cores cluster to execute the Bioformer inference is 51 mW @ 100 MHz. For TEMPONet, we report its statistic in an identical setup, to allow for a fair comparison between the two models. Note that the accuracy of these models, as reported in Table I, is the one obtained after the quantization-aware fine-tuning.

After quantization, our most accurate model yet achieves 64.69% accuracy, consuming an impressively lower 8.0× energy compared to TEMPONet. Additionally, this model can even fit a smaller MCU since it only requires 94.2 kB.

The Bioformer with the lowest latency further reduces the energy compared to TEMPONet by 17.3×, with an accuracy reduction of only 3.60% and a comparable memory footprint (110.8 kB). Overall, considering this last model, a 150 ms window classified every 15 ms costs 52 $\mu$J and has a latency of 1.02 ms, while for the remaining time, the GAP8 SoC only collects data. In this step, we can idle the 8-core cluster accelerator using its embedded hardware synchronization unit [26] and therefore reduce the power consumption to the 10 mW consumed by the Fabric Controller. This yields an average power consumption over time of 12.81 mW. Using a small $1000\,\mathrm{mAh}$ battery, we can continuously perform sEMG-based gesture recognition for a lifetime of $\sim 257h$, 4.77× higher compared to using TEMPONet for this task ($\sim 54h$ with the same battery [15]).

## V. Conclusions

We have shown that Transformers can achieve close to state-of-the-art performance on sEMG-based gesture recognition while strongly reducing the complexity and the memory footprint required for deployment on edge nodes. Further, we have introduced a new pre-training procedure that yields up to a 3.39% accuracy improvement for both Transformer and TCN-based models.

On Ninapro DB6, our most accurate Bioformer obtains 65.73% accuracy, better than the previous state-of-the-art accuracy (65.00% of TEMPONet [15]) and only 1.07% lower than TEMPONet trained with our new protocol. Simultaneously, this Bioformer reduces the number of MACs and memory footprint compared to TEMPONet by 4.9×. Deployed on GAP8, it consumes just 0.139 mJ with a latency of 2.72 ms.

## References


[1] S. Madakam *et al.*, "Internet of things (iot): A literature review," *Journal of Computer and Communications*, vol. 3, no. 05, p. 164, 2015.

[2] M. Risso *et al.*, "Robust and energy-efficient ppg-based heart-rate monitoring," in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2021, pp. 1–5.

[3] F. T. Sun *et al.*, "Responsive cortical stimulation for the treatment of epilepsy," *Neurotherapeutics*, vol. 5, no. 1, pp. 68–74, 2008.

[4] R. Meattini *et al.*, "An semg-based human–robot interface for robotic hands using machine learning and synergies," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2018.

[5] Y. Hu *et al.*, "A novel attention-based hybrid cnn-rnn architecture for semg-based gesture recognition," *PloS one*, vol. 13, no. 10, 2018.

[6] P. Tsinganos *et al.*, "Deep learning in emg-based gesture recognition," in *PhyCS*, 2018.

[7] P. Tsinganos *et al.*, "Improved gesture recognition based on semg signals and tcn," in *ICASSP 2019*. IEEE, 2019, pp. 1169–1173.

[8] J. L. Betthauser *et al.*, "Stable Electromyographic Sequence Prediction during Movement Transitions using Temporal Convolutional Networks," *International IEEE/EMBS Conference on Neural Engineering*, 2019.

[9] E. Flamand *et al.*, "GAP-8: A RISC-V SoC for AI at the Edge of the IoT," in *2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*. IEEE, 2018, pp. 1–4.

[10] J. Devlin *et al.*, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[11] T. B. Brown *et al.*, "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020.

[12] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[13] A. Howard *et al.*, "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1314–1324.

[14] A. Burrello *et al.*, "Predicting hard disk failures in data centers using temporal convolutional neural networks," in *Euro-Par 2020: Parallel Processing Workshops*, vol. 12480. Nature Publishing Group, 2020, p. 277.

[15] M. Zanghieri *et al.*, "Robust real-time embedded emg recognition framework using temporal convolutional networks on a multicore iot processor," *IEEE transactions on biomedical circuits and systems*, vol. 14, no. 2, pp. 244–256, 2019.

[16] M. Zanghieri *et al.*, "Temporal variability analysis in semg hand grasp recognition using temporal convolutional networks," in *2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*. IEEE, 2020, pp. 228–232.

[17] C. J. De Luca *et al.*, "The use of surface electromyography in biomechanics," *Journal of applied biomechanics*, 1997.

[18] P. Kaufmann *et al.*, "Fluctuating emg signals: Investigating long-term effects of pattern matching algorithms," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*. IEEE, 2010, pp. 6357–6360.

[19] M. Atzori *et al.*, "Electromyography data for non-invasive naturally-controlled robotic hand prostheses," *Scientific Data*, vol. 1, p. 140053, dec 2014. [Online]. Available: http://www.nature.com/articles/sdata201453

[20] B. Milosevic *et al.*, "Exploring Arm Posture and Temporal Variability in Myoelectric Hand Gesture Recognition," *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*, vol. 2018-August, pp. 1032–1037, 2018.

[21] F. Palermo *et al.*, "Repeatability of grasp recognition for robotic hand prosthesis control based on sEMG data," in *2017 International Conference on Rehabilitation Robotics (ICORR)*. IEEE, jul 2017, pp. 1154–1159. [Online]. Available: https://ieeexplore.ieee.org/document/8009405/

[22] M. Atzori *et al.*, "Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands," *Frontiers in Neurorobotics*, vol. 10, 09 2016.

[23] A. Vaswani *et al.*, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[24] S. Kim *et al.*, "I-bert: Integer-only bert quantization," *arXiv preprint arXiv:2101.01321*, 2021.

[25] A. Burrello *et al.*, "A microcontroller is all you need: Enabling transformer execution on low-power iot endnodes," in *2021 IEEE International Conference on Omni-Layer Intelligent Systems (COINS)*, 2021, pp. 1–6.

[26] F. Glaser *et al.*, "Hardware-accelerated energy-efficient synchronization and communication for ultra-low-power tightly coupled clusters," in *2019 DATE Conference*, March 2019, pp. 552–557.