Efficient in Vivo Neural Signal Compression Using an Autoencoder-Based Neural Network

Daniel Valencia, Patrick P. Mercier, *Senior Member, IEEE*, and Amir Alimohammad

***Abstract*—Conventional in vivo neural signal processing involves extracting spiking activity within the recorded signals from an ensemble of neurons and transmitting only spike counts over an adequate interval. However, for brain-computer interface (BCI) applications utilizing continuous local field potentials (LFPs) for cognitive decoding, the volume of neural data to be transmitted to a computer imposes relatively high data rate requirements. This is particularly true for BCIs employing high-density intracortical recordings with hundreds or thousands of electrodes. This article introduces the first autoencoder-based compression digital circuit for the efficient transmission of LFP neural signals. Various algorithmic and architectural-level optimizations are implemented to significantly reduce the computational complexity and memory requirements of the designed in vivo compression circuit. This circuit employs an autoencoder-based neural network, providing a robust signal reconstruction. The application-specific integrated circuit (ASIC) of the in vivo compression logic occupies the smallest silicon area and consumes the lowest power among the reported state-of-the-art compression ASICs. Additionally, it offers a higher compression rate and a superior signal-to-noise and distortion ratio.**

***Index Terms*—Application specific integrated circuits, neural engineering, neural networks.**

I. Introduction

THE field of intra-cortical brain-computer interfaces (BCIs) has been rapidly evolving over the past decade. BCIs effectively translate (decode) recorded neural signals into a quantifiable representation for augmenting or enhancing the user's working experience. The input to a neural decoding algorithm is a specific representation of the neural activity and the output is either a continuous variable or a discrete selection. For example, the former can represent the kinematic variables to control a computer cursor or a robotic limb [1], [2], while the latter may represent the particular mental state of a user (e.g., sleeping or alert) [3] or the end-goal of a planned movement [4], [5]. The application of novel machine learning (ML)-based decoding algorithms has enabled increasingly complex BCI applications, such as thought-to-text [6] and thought-to-speech synthesis [7].

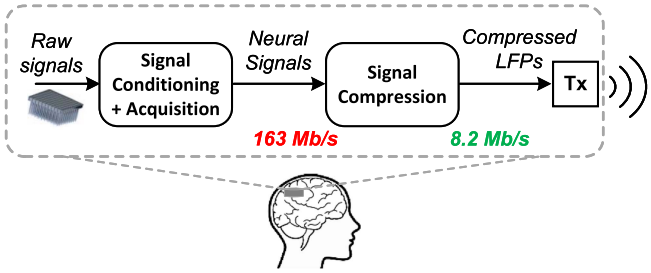

Fig. 1. Intracortical recording and wireless transmission of compressed neural signals, employing 1000 recording channels.

Intracortical neural recording systems have continuously advanced from the widely-employed Utah Array [8], supporting up to a hundred recording sites, to high-density recording electrodes, such as Neuralink [9] and Neuropixels [10], supporting hundreds of recording sites per implantable recording shank and hence, thousands of recording channels. An increasing number of recording channels will inevitably impose a higher data rate requirement. For example, a neural recording system with 1000 recording electrodes, sampled at 20 kS/s with 16-bit resolution, would require a data rate of 320 Mbps. The state-of-the-art wireless transmission of neural data has a mean energy dissipation of 6.7 pJ/bit [11], [12], [13], [14], [15], which would imply 2.14 mW of power for wireless transmission alone. In accordance with the the Food and Drug Administration [16], considering neural tissue-specific absorption rate, the limit for a safe wireless power transfer is 7.7 mW. Thus, transmitting raw neural signals would consume over 27% of the available power budget. By employing in vivo neural signal processing and compression, the data rate requirement can be drastically reduced. For example, as shown in Fig. 1, if in vivo compression is able to reduce the data rate by a factor of 20, the wireless transmission requires only 0.1 mW of power, i.e., approximately 1.4% of the available power budget. Therefore, efficient realization of in vivo signal compression becomes crucial for BCIs employing high-density microelectrode arrays (MEAs).

Compression schemes are generally divided into two categories. Lossless methods involve reducing the dynamic range of the neural signals and encoding the signals as variable bit-rate data streams [17], [18], [19]. One of the commonly-employed

Manuscript received 12 September 2023; revised 27 December 2023; accepted 25 January 2024. Date of publication 29 January 2024; date of current version 29 May 2024. This work was supported by the National Science Foundation under Award 2007131. This paper was recommended by Associate Editor S. Sonkusale. *(Corresponding author: Daniel Valencia.)*

Daniel Valencia is with the Department of Electrical, Computer Engineering, San Diego State University, San Diego, CA 92182 USA, and also with the University of California San Diego, La Jolla, CA 92093 USA (e-mail: dlvalencia@sdsu.edu).

Patrick P. Mercier is with the University of California San Diego, La Jolla, CA 92093 USA (e-mail: pmercier@ucsd.edu).

Amir Alimohammad is with the Department of Electrical, Computer Engineering, San Diego State University, San Diego, CA 92182 USA (e-mail: aalimohammad@sdsu.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TBCAS.2024.3359994.

Digital Object Identifier 10.1109/TBCAS.2024.3359994

lossless compression methods for LFPs is to exploit the temporal redundancy and spatial correlation of LFPs. The temporal difference $x_d[n] = x[n] - x[n-1]$ reduces the dynamic range of the signals and consequently, the required number of bits per sample to about a half [17]. With a reduced numerical range, neural signals can then be represented using Huffman coding [20], which encodes more commonly occurring samples using fewer bits. The combination of temporal difference and Huffman coding is considered as a lossless compression scheme, which generally offers a compression rate on the order of 2 to 5 [17], [18], [19].

While lossless methods provide perfect reconstruction, lossy methods can significantly increase the data compression rate by employing spatial downsampling. The underlying principle for employing lossy methods in the context of neural signals is the relatively large amount of spatial correlation among neural recording channels. Considering that the state-of-the-art neural decoding algorithms are relatively robust to noise and signal perturbations [21], [22], by tolerating relatively small signal errors, employing a lossy compression scheme might be a more viable approach for BCI applications. Sources of noise include instrumentation perturbations due to electrode micro-motions, bit errors during wireless transmission, and quantization noise caused by finite numerical resolutions. Compressed sensing (CS) [23] is a lossy scheme used for neural signals [24], [25], [26], where signal $\mathbf{x} \in \mathbb{R}^n$ is multiplied with a sensing matrix $\Phi \in \mathbb{R}^{m \times n}$ to produce $\mathbf{y} \in \mathbb{R}^m$, effectively compressing the signal by a factor of m/n. A variety of CS-based algorithms are employed in silico to reconstruct the original signal $\mathbf{x}$ from the compressed (encoded) signal $\mathbf{y}$ [24], [25], [26].

In this work, we propose the novel application of a ML-based compression scheme based on autoencoders (AEs). Compared to the state-of-the-art neural signal compression circuits, the designed AE-based compression scheme offers a greater compression rate, higher signal-to-noise and distortion ratio, smallest silicon area, and lowest power consumption. The rest of this article is organized as follows. Section II discusses the motivation toward employing the local field potentials over neural action potentials. Section III discusses the algorithm and performance of the designed autoencoder-based compression scheme. For an efficient realization of the in vivo autoencoder, various algorithmic and architectural optimization techniques are employed and presented in Section IV. The architecture of the designed compression hardware is presented and discussed in Section V and its implementation characteristics are compared against the relevant compression circuits. Finally, Section VI makes some concluding remarks.

II. Local Field Potential-Based BCIs

Compared to the non-invasive electro-encephalography (EEG), the invasive intra-cortical recording modality offers the highest temporal and spatial resolutions in which the neural activity can be represented, as either the excitation of individual neurons, called single-unit activities (SUAs), or an ensemble of neurons, called multi-unit activities (MUAs). Applications employing SUAs or MUAs conventionally perform in vivo spike detection [27] to reduce the wireless data rate requirements. Neurons are known to fire relatively infrequently with respect to the sampling rate, on the order of 40 Hz [28]. Additionally, due to the physiological refractory period of neurons, spiking activity is usually represented at the millisecond level with the required bandwidth of at most one kHz per channel. SUAs can be obtained by spike sorting, which can be viewed as a clustering process where spikes fired from the same neurons are grouped together [29]. Some BCIs employ in vivo circuitry to classify spike waveforms and transmit only 2–3 bits per spike class [30], [31], [32], [33], while others avoid in vivo spike sorting and instead transmit the MUA spike waveforms, which requires about 2 to 3 milliseconds of data per spike waveforms. By transmitting only spiking activity, BCIs employing in vivo spike classification have a compression rate of 1000 – 6000 [30], [31], [32], [33] at the expense of spike sorting computations, while BCIs that transmit the entire spike waveforms have compression rates of 2–44 [34] at the cost of greater data transmissions. Compared to the SUA-based neural signal processing, MUA-based decoding, however, does not require computationally-intensive spike sorting [30], [31], [32], [33]. It has been shown that the overall decoding performance degradation is negligible when employing MUAs [35]. In addition, transmitting only MUA events (e.g., spike counts) drastically reduces wireless transmission rates. For example, if neural signals are sampled at 10–30 kS/s with a 10–16 b resolution, the required wireless data rate is 100–480 Kbps per recording channel. With a 96-channel Utah Array, the staggering transmission rate is 9.6–46 Mbps. However, MUA features are commonly represented as spike counts over an interval of 1–25 milliseconds. Most implementations of spike detection impose a biologically plausible spike refractory period of one millisecond [21], [36] and hence, one millisecond spike bins would require the data rate of only one Kbps per recording channel, yielding a data rate of at most 96 Kbps for a 96-channel Utah Array, resulting in at least a 100 times data rate reduction. Therefore, MUAs have been widely employed in both clinical trials and therapeutic applications of BCIs.

Neural activities can alternatively be represented by the local field potentials (LFPs), which are formed by the aggregate synaptic activities of populations of neurons. Compared to the SUAs and MUAs, LFPs represent slower variations in the neural signal's voltage and have lower spatial resolutions (LFPs: 0.5 mm, MUAs: 0.1 mm, and SUAs: 0.05 mm) [37], [38], [39]. Due to recording variations and instabilities, such as electrode drift and neuron drop-out [40], as well as potential scarring between the electrode-tissue interface over a relatively long period of time [41], LFPs are considered more stable compared to the SUAs and MUAs. While many of the MUA-based BCIs focus on decoding the neural activity in the motor cortex, LFP-based BCIs decode neural activities of the cognitive regions of the brain, such as the posterior parietal cortex, which allows higher-level cognitive decoding than that of the lower-level continuous motor control [5], [42]. Various, studies have shown that reliable neural decoding can be performed using LFP signals [5], [42], [43]. A study reported in [43] found that various movement intentions, such as the imagined end-point, kinematic trajectory, and type of movement, can be predicted reliably from the LFP signals.

From the perspective of implantable integrated circuits for neural recording, the acquisition and processing of LFPs offers an opportunity for significantly reducing the in vivo power consumption. For example, neural signals are often sampled at a rate of 10–30 kS/s to provide the necessary temporal resolution for detecting action potentials (spikes) [21]. The acquisition of neural signals consists of low-noise amplifiers and analog-to-digital converters (ADCs), most of which employ 10–16 b resolutions and have a nominal power consumption of 0.25 μW–7.3 μW per recording channel. Because the LFP frequencies of interest are often up to a few hundred hertz (i.e., 0.1–300 Hz), the sampling rate can be reduced significantly compared to that required for spike-based processing, typically between 1–2 kS/s. A five fold reduction in the sampling rate and signal bandwidth would reduce the power consumption of the in vivo signal acquisition and conditioning [21]. Also, to extract SUAs or MUAs from the recorded neural signals, additional in vivo neural signal processing, such as adaptive threshold estimation and spike detection, which consumes between 0.6 and 1.78 μW of power per recording channel, is required [22], [44]. Thus, a reasonable estimate for processing MUAs is approximately 8 μW of power per channel. LFPs require no threshold estimation or additional processing aside from signal filtering to discard signal components above 300 Hz. While the in vivo neural signal processing for LFPs is more power efficient than that for SUAs/MUAs, the data rate requirements, however, are considerably higher. For example, considering LFPs sampled at 2 kS/s with a 16-bit resolution, its wireless transmission would require 32 Kbps/channel, while the spike counts over one millisecond bins would require only one Kbps/channel. Therefore, the direct transmission of continuous LFPs would require over 3 Mbps for a 96-channel electrode array. The data rate would increase as higher density electrodes are implanted for emerging BCI applications.

In practical systems, various algorithmic and architectural-level schemes can be employed to efficiently manage the in vivo BCI power consumption. For instance, in an asynchronous (self-paced) BCI paradigm, the user's desire to engage in the BCI task can be detected with an intention estimation logic. When the user is not actively involved in the BCI task, a significant portion of signal acquisition and processing can be powered down, leading to substantial power savings [22]. While the topic of user's intention estimation is beyond the scope of this work, its potential to effectively disable the majority of in vivo circuitry, along with the designed signal compression scheme, could significantly reduce the energy dissipation of LFP-based BCIs.

III. Autoencoder-Based Compression

LFPs are formed by the accumulated synaptic activity of populations of neurons and hence, they can be readily detected by recording channels that are relatively close to one another. Fig. 2 shows the inter-channel correlation over 30 seconds of a neural recording from a non-human primate performing self-paced reaching tasks [45]. One can see that various sub-regions within the recording array are highly correlated. Most

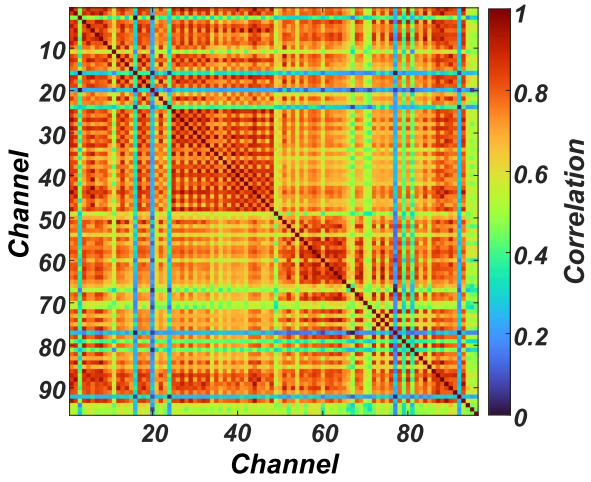

Fig. 2. Inter-channel correlation over 30 seconds of a neural recording.

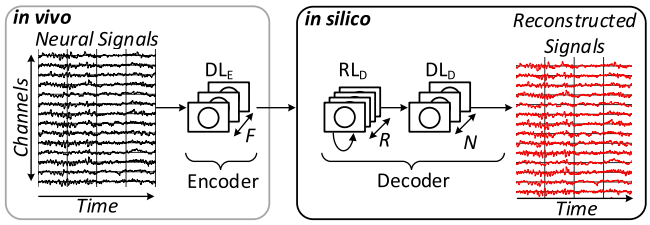

Fig. 3. Block diagram of the designed autoencoder-based compression scheme.

lossy compression methods aim to exploit this inherent spatial correlation to perform a spatial downsampling of the LFPs. By exploiting the spatial correlation, the LFPs can be represented using either a subset of the channels or a linear transformation of the original signal.

An autoencoder (AE) is a neural network, consisting of an encoder network, which reduces the spatial dimension of the input data, and a decoder network, which is trained to reconstruct the original input. Conventional AEs employ mirrored network architectures such that the encoder and decoder have the same number and type of layers, but in the reverse order. Due to their data-driven approach to learn an optimal low-dimensional representation of the input signal, AEs have been previously employed for spike waveform feature extraction [46], [47] and for compressing spike waveforms [34].

The block diagram of the designed and implemented AE-based compression architecture is shown in Fig. 3. The LFPs are derived from the recorded neural signals using a second-order Butterworth low-pass filter with a cutoff frequency at 300 Hz. The encoder network is relatively small and is feasible to be realized as an implantable circuit in vivo for compressing filtered neural signals before wireless transmission, while the decoder is implemented in silico to reconstruct the neural signals for subsequent processing. The encoder network consists of a single dense layer $\mathsf{DL_E}$ and the decoder consists of a recurrent layer $\mathsf{RL_D}$ and a dense layer $\mathsf{DL_D}$. The encoder layer reduces the input dimension from N to F, where N denotes the number of channels in the recorded signal and F denotes the number of units in the

dense layer. The recurrent layer $\mathsf{RL_D}$ accepts the F-dimensional outputs of the encoder, learns temporal information within the encoded signals, and provides an R-dimensional spatially upsampled output. The decoder's dense layer $\mathsf{DL_D}$ performs the final spatial upsampling of R to N and reconstructs the input signals. The units in the $\mathsf{DL_E}$ employ Tanh activation functions, while the $\mathsf{DL_D}$ performs a linear regression to reconstruct the channel activity. The compression rate of the designed model is $CR = Nw_i/Fw_e$, where w_i and w_e denote the resolution of the input data and the output resolution of the $\mathsf{DL_D}$, respectively. For example, for a 96-channel Utah Array with $F = 10$ units, 16-bit data samples and 10-bit outputs, the compression rate is 15.36. Note that conventional digital acquisition (DAQ) systems typically employ 16 bits of resolution for sampling neural signals. In practice, the lower-order bits of the digitized neural signals may be discarded if they fall below the noise floor of the amplifier [19]. For instance, with a given signal amplifier featuring V_1 $\mu\mathrm{V}_{\mathrm{RMS}}$ input referred noise and an ADC resolution of V_2 μ V per bit, the least significant $\log_2(V_1/V_2)$ bits can be discarded. It is important to note that our analyses do not involve such resolution reduction, as the focus is on the design and implementation of a parameterizable auto-encoder-based compression scheme that can be synthesized onto a custom resolution, independent of the specification of the employed DAQ system.

The designed AE-based network is trained with the Python Tensorflow framework using two publicly available datasets. Dataset I [45] consists of neural recordings from a Macaque monkey while performing a self-paced point-to-point reaching task. Dataset II [48] consists of neural recordings from two monkeys K and L while performing an object reach and grasp task. Dataset I is sampled at $f_s = 24.4$ kS/s and is filtered using an anti-aliasing low-pass filter at 7.5 kHz, built into the recording system. Dataset II is sampled at $f_s = 30$ kS/s and is filtered using a high-pass filter at 0.3 Hz and a low-pass filter at 7.5 kHz. Dataset I consists of 30 recordings acquired over 7 months and Dataset II consists of two recordings (one for each monkey) during a single recording session. Both datasets are downsampled to 2 kS/s by applying a low-pass filter with the cutoff frequency at 1 kHz followed by decimation. The model was trained using the Adam optimizer and the mean absolute error between the input signals and the reconstructed signals was considered as the loss function.

Lossy compression methods often report reconstruction error using the signal-to-noise and distortion ratio (SNDR) metric, defined as $\mathrm{SNDR} = 20\log_{10}\frac{||\mathbf{x}||_2}{||\mathbf{x}-\hat{\mathbf{x}}||_2}$, where $\mathbf{x}$ and $\hat{\mathbf{x}}$ denote the original and reconstructed neural signals, respectively, and $||\cdot||_2$ denotes the L2-norm [49]. The performance of the designed model was evaluated using the R2 score by analyzing the similarity between the true and reconstructed LFPs. The R2 score describes the ability of the reconstruction model to capture the variance present in the data and is given as $\mathrm{R2} = 1 - \frac{\sum_i(\hat{y}_i-\bar{y})^2}{\sum_i(y_i-\bar{y})^2}$, where $\hat{y}_i$ and y_i denote the i-th reconstructed and true outputs, respectively, and $\bar{y}$ denotes the mean. While the compression rate increases with a smaller F, the increased spatial downsampling may adversely impact the reconstruction quality. Using the first

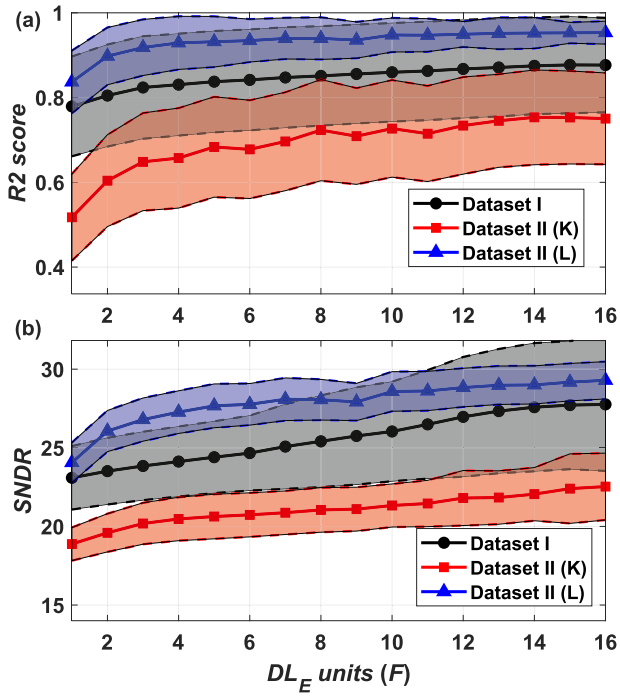

Fig. 4. Mean and standard deviation of (a) R2 scores and (b) SNDRs for the reconstructed LFP signals analyzed over various numbers of $\mathsf{DL_E}$ units F.

recording from Dataset I and both recordings from Dataset II, the model was trained by varying F between 16 and 1 and R between 256 and 64 with 16 evenly spaced intervals of 13. This approach was employed to ensure compensation between smaller values of F and larger values of R in the decoder. As the recordings of the two datasets are from three different animals (Dataset I with one animal, and Dataset II with two different animals), the model was trained separately on each dataset. The training was performed on the first 80% of each recording session, with the next 10% used for validation and the final 10% for testing. The training, validation, and testing sets were then split into time spans of 160 ms, which corresponds to 320 time steps with $f_s = 2$ kS/s. The training was performed for up to 1000 epochs using early stopping on the validation set to prevent overfitting. Fig. 4(a) shows the mean R2 scores over various number of units F for the first recording session of Dataset I and both recordings of Dataset II, and the shaded areas show the standard deviation across the recording channels. It was found that the model with $F = 8$ and $R = 128$ provides a median R2 score of 0.852 ± 0.04 trained on each of the 30 recordings in Dataset I, 0.72 ± 0.23 and 0.93 ± 0.09 over the Monkey K and L recordings of Dataset II, respectively. It was found that the encoder dimensionality F has a stronger impact on the overall performance of the model compared to that of the decoder dimensionality R. While the decoder dimensionality could be further increased, values of R beyond 128 did not yield considerable performance gains. Fig. 4(b) shows the mean of SNDR. Similarly to those of the R2 score, an increasing number of units F shows a small increase in SNDR. The median SNDR over each of the 30 recordings in Dataset I was 22.79 ± 2.19, and 21.19 ± 2.89 and 28.89 ± 2.66 over the Monkey K and Monkey L recordings, respectively.

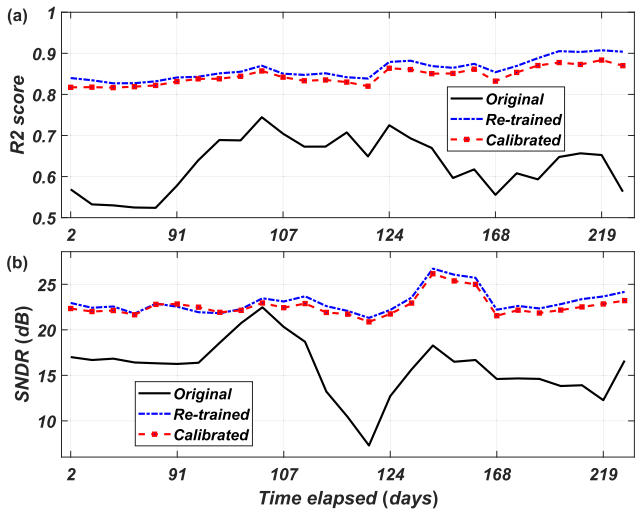

Fig. 5. Mean of the (a) R2 scores and (b) SNDRs of the original, re-trained, and calibrated models over all recordings of Dataset I, respectively.

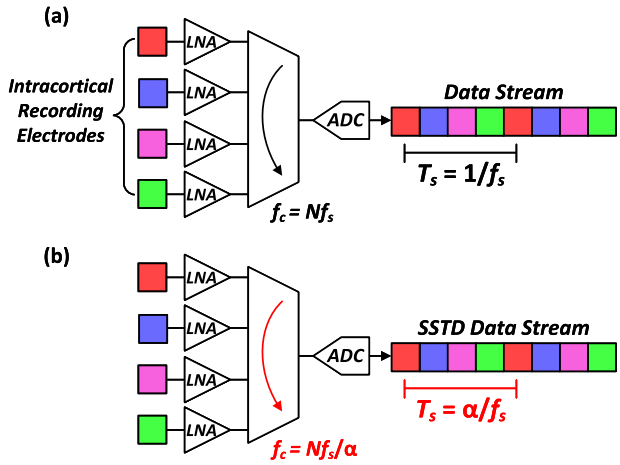

Fig. 6. Analog front-end configuration for realizing staggered spatio-temporal downsampling (SSTD).

As shown in Fig. 4, the performance of the model is data-dependent, as is in general for the ML-based algorithms. One important characteristic of the ML-based compression models is their ability to generalize to future data. Fortunately, Dataset I contains several months of recordings, which allowed the generalization of the baseline model trained on the first recording session to be tested on all subsequent recordings. Fig. 5(a) and (b) show that the designed model provides a variable performance, with the R2 scores between 0.5 and 0.73 and the SNDR between 5.26 and 22.93, which is due to the changes in the statistics of the neural signals per recording channel. For a relatively steady performance, the original model is either re-trained or calibrated using a small amount of new data. Fig. 5(a) and (b) show a negligible difference between the re-trained model and the calibrated version of the original model, however, the calibrated model only requires 10% of the new data whereas the re-trained model requires 80% of the new data for training. Additionally, the calibrated model retains the weights of the encoder network and only adjusts the weights of the decoder. In addition to the R2 score, SNDR is a more commonly employed metric to quantify the performance of lossy compression algorithms. The range of acceptable R2 scores depends on the underlying application (i.e., neural decoding, which is outside the scope of this work). However, modern ML-based decoding algorithms for spike-based BCIs are robust to about 10% of input spiking errors [21], [22]. Although the signal modality of our design is LFPs, it can be inferred that relatively high R2 scores, where the model can account for at least 70% of the variance of the LFP signals, may be sufficient for reliable neural decoding.

One important consideration is that the employed calibration requires the acquisition of uncompressed LFP signals to create the new ground truth data. Implantable BCI systems generally operate in one of the two modes. In the "active" state, the compression circuitry is enabled to reduce the transmitted data rate while the user is engaged in the BCI task. In the "training" state, there is no need for real-time transmission of compressed neural signals. During the "training" state, raw LFP waveforms can be recorded and transmitted for the calibration or re-training of the AE decoder network. Recently employed BCIs, such as NeuraLink [9] and miniaturized implantable recording motes [50], [51], [52], utilize near-field communication technology to establish communication with in silico microprocessors. This allows for higher communication bandwidths and long-term data storage during the "training" phase.

IV. In Vivo Encoder Optimizations

Because the encoder network will be realized in vivo, it is imperative to investigate potential architectural optimizations for reducing the memory requirements, computational complexity, and data resolution of the encoder output. Since the $\mathrm{DL_E}$ requires one weight value per output node per input channel, the total memory for the $\mathrm{DL_E}$ is $F(N+1)$. One method for lowering the total number of parameters is to reduce F, however, as discussed in Section III, this will result in degraded reconstruction quality. An alternative is to lower the number of input channels N, however, instead of selecting a subset of channels to process, the spatial correlation among the channels is exploited to reduce the input dimension of the network. For an efficient signal conditioning, we propose the staggered spatio-temporal downsampling (SSTD) recording configuration, as shown in Fig. 6. Conventional analog front-end (AFE) circuitry for neural recording typically employs a shared ADC among various recording sites along with low-noise amplifiers (LNAs) [53]. To accurately sample each signal, the switching frequency of the multiplexer f_c is toggled at the rate of Nf_s, where N denotes the number of recording sites sharing the ADC, and f_s denotes the sampling rate of the underlying neural signal. Due to the relatively high inter-channel correlation within recorded neural signals, each channel is temporally downsampled by a factor α. This downsampling is achieved by reducing the switching frequency f_c by a factor α, providing a temporally downsampled data stream for each recording site. Lowering the input dimension of the network by a folding factor of α reduces the total memory requirement to $F(N/\alpha+1)$. The decoder network is still trained to reproduce the original input dimension N and

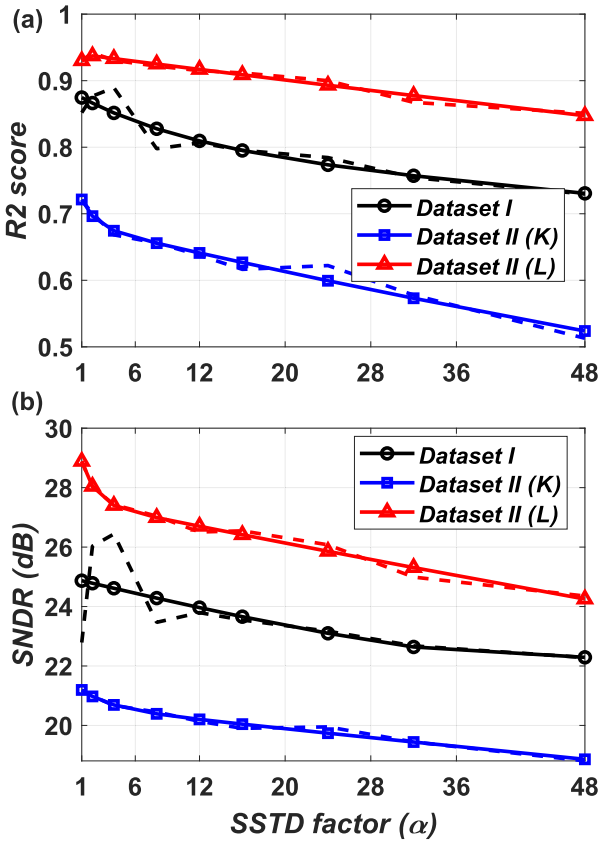

Fig. 7. Variations of (a) R2 scores and (b) SNDR for the first recording of Dataset I, the Monkey K recording of Dataset II, and the Monkey L recording of Dataset II, over various SSTD factors. The dashed lines show the metrics for raw data, while the marked-up lines depict the trend.

predicts the samples in between the temporally downsampled data along each input channel, i.e., the samples for times $2\Delta s$ to $(\alpha-1)\Delta s$ for each channel.

Fig. 7(a) and (b) shows the mean R2 scores for the recording session I of Dataset I and the two recordings of Dataset II. It can be seen that both the R2 performance and SNDR are data dependent. Further, the selection of α depends on the dataset for achieving a particular range of performance metrics. It should be noted, however, that the variation of performance metrics with respect to α is relatively minor, and the dataset itself has a larger impact on the performance variation. Since the performance of the ML-based algorithms is generally data dependent, employing the SSTD configuration introduces a larger statistical variation on Dataset I compared to that of Dataset II, and the change is relatively linear with respect to the folding factor. We chose to employ a folding factor of $\alpha = 12$, which reduces the memory requirement of the in vivo encoder from 776 to only 72 parameters, a 90% reduction. Lowering the input dimension of the network by a folding factor $\alpha = 12$ also reduces the computational complexity of the model. The original input consists of a multiplication of an input vector $\mathbf{x} \in \mathbb{R}^{1\times N}$ and a weight matrix $\mathbf{W} \in \mathbb{R}^{N\times F}$, requiring $F \times N$ multiplications and $F \times (N-1)$ additions, while lowering the input dimension by a factor of 12, the total number of multiplications and additions will be dropped to $F \times N/12$ and $F \times (N/12-1)$,

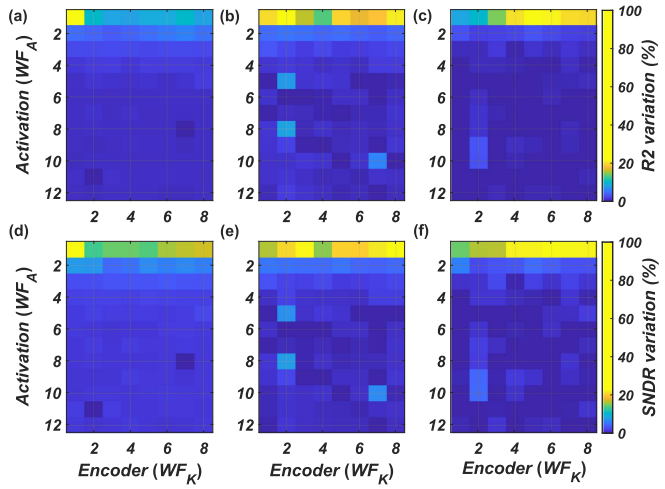

Fig. 8. Variations of the R2 score for (a) Dataset I, (b) Dataset II (K), and (c) Dataset II (L), along with the variations of the SNDR for (d) Dataset I, (e) Dataset II (K), and (f) Dataset II (L) over different encoder resolutions $\mathrm{WF_K}$ and HardTanh output resolutions $\mathrm{WF_A}$.

respectively. Another opportunity for optimizing the in vivo encoder is the realization of the Tanh activation function, defined as:

$$y = \frac{e^{-z} - e^{z}}{e^{-z} + e^{z}}, \tag{1}$$

where z denotes the accumulated weighted input to the activation function. To avoid implementing the exponential and division operators directly, the piecewise linear approximation is commonly employed [54]. An alternative approach that does not require linear approximation parameters is employing the HardTanh function, defined as:

$$y = \begin{cases} -1 & \text{if } z \le -1, \\ z & \text{if } -1 < z < 1, \\ 1 & \text{if } z \ge 1, \end{cases} \tag{2}$$

, where z denotes the weighted input to the activation function, and requires only two comparators to determine whether the output of the function should saturate at +1 or -1.

Another consideration for efficient in vivo hardware realization is that the encoder network parameters and activations are represented in the fixed-point format. The baseline model is first trained employing the SSTD configuration and the HardTanh activation function. After an initial training, the encoder and decoder networks are split into two separate networks. During the initial training, the encoder network's parameters are constrained to [-1, 1) and hence, are represented in the fixed-point format $\mathrm{Q(1.WF_K)}$, where 1 and $\mathrm{WF_K}$ denote the number of integer and fraction bits, respectively. The output of the HardTanh function is quantized into $\mathrm{Q(2.WF_A)}$ format. The decoder network is then re-trained to account for the slight variations introduced by the encoder quantization. Fig. 8 shows the variations in the R2 score and SNDR over values of $\mathrm{WF_K}$ and $\mathrm{WF_A}$ for Datasets I and both recordings of Dataset II. It can be seen that the HardTanh output resolution $\mathrm{WF_A}$ has a stronger impact on the performance variation compared to that of the

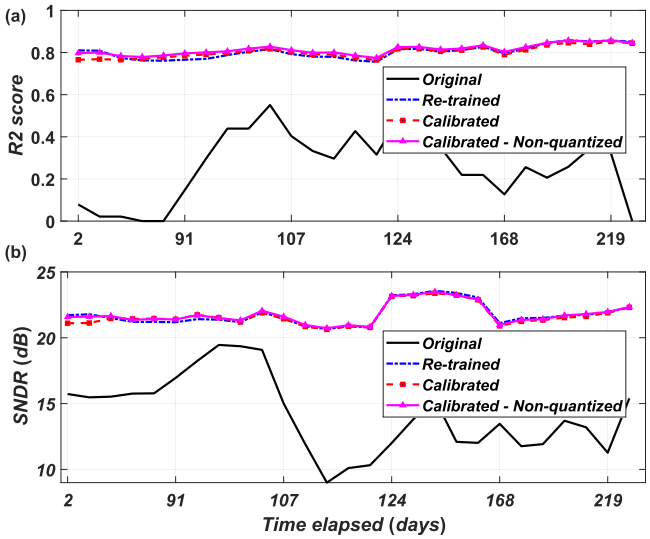

Fig. 9. Mean (a) R2 scores and (b) SNDR of the original, re-trained, and calibrated models using encoder quantization and SSTD over all recordings of Dataset I, respectively. The performance of the calibrated SSTD configuration without quantization is also shown.

encoder paramter resolution WF_K. It was found that WF_K=4 and WF_A=8 results in a 6.9% variation in the median of the R2 score values for Dataset I, a negligible 0.03% variation for the Monkey K recording of Dataset II, and a 0.4% variation for the Monkey L recording of Dataset II compared to the SSTD model with $\alpha = 12$. Thus, the chosen numerical quantization have a negligible impact on the reconstruction quality of the model. It's worth noting that, as shown in Fig. 8, smaller values for WF_A and WF_K result in a relatively lower performance variation, but these values may be viable when primarily focusing on a single dataset and might be unviable for a generalized realization.

Fig. 9(a) and (b) shows the mean of the R2 score and SNDR values, respectively, of the original model, the re-trained model, and the calibrated model, over the 30 recordings of Dataset I with the encoder quantization and SSTD. The performance of the SSTD configuration without quantization is also shown. It is shown that the calibration with 10% of the training data achieves a performance comparable to that of the re-trained model as well as that of the model without employing quantization. One can see that the employed in vivo optimizations have a negligible impact on the performance degradation of the model.

V. Hardware Architecture and Implementation Results

Fig. 10 shows the top-level block diagram of the designed and implemented in vivo encoder network. It consists of an array of normalization units, multipliers, adder trees, and HardTanh activation function units. The normalization units are used to normalize the input data based on the training set. The number of normalization units D defines the number of input channels to process simultaneously and is equal to the number of electrode channels divided by the folding factor α. Each input channel requires a set of normalization parameters min and range prior to applying the DL_E weights, where min and range denote the

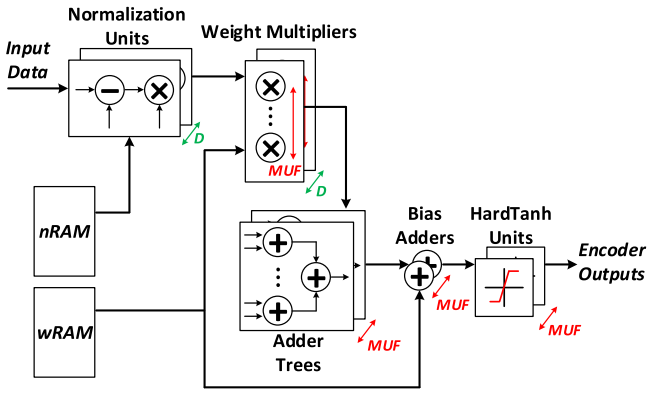

Fig. 10. Top-level block diagram of the designed encoder network.

TABLE I
The ASIC Characteristics of the Designed and Synthesized in Vivo Encoder Over Various Values of MUF

MUF	Freq. (kHz)	Area (mm^2)	Power (μW)
1	16	1.25	10
2	8	1.37	7.6
4	4	1.94	7.35
8	2	2.11	10.4

channel's minimum and the range of amplitude values stored in the normalization parameter memory nRAM, respectively. The model parameters are stored in the weight parameter memory wRAM. Each bank of weight multipliers and its associated adder tree computes the accumulation of the weighted inputs. The weighted sums are biased with the adders and then the HardTanh activation function is applied.

The multiplier unfolding factor parameter MUF denotes the number of weighted sums computed per clock cycle. For example, with MUF = 1, it would take eight clock cycles to compute the eight outputs of the in vivo encoder. For a throughput equal to the sampling rate f_s, a clock frequency equal to $8f_s$ is required. To find an optimal value for MUF, the encoder network is synthesized with α=12 for the MUF values of 1, 2, 4, and 8. The logic synthesis was performed using Synopsys Design Compiler and the place and route was performed with Cadence Innovus in a standard 180-nm CMOS process. To estimate the power consumption of the design, the post-routed netlist was simulated using Synopsys Verilog Compiler Simulator (VCS) by applying a testing subset of the Dataset I to the post-routed netlist. Table I gives the dynamic power consumption and area utilization of the designed in vivo encoder over various values of MUF. Reducing the operating frequency to f_s would naturally reduce the dynamic power consumption, however, it was found that MUF = 4 consumes the least power due to requiring half the number of adder trees and hardware of those for MUF = 8. The ASIC layout of the synthesized in vivo encoder with MUF = 4 is shown in Fig. 11 and is estimated to occupy 1.94 mm^2 of silicon area in a standard 180-nm CMOS process.

Table II gives the ASIC characteristics and implementation results of various previously published intra-cortical neural signal compression designs. Both lossless and lossy compression modalities have been reported. Since the main focus is on the

TABLE II
CHARACTERISTICS AND IMPLEMENTATION RESULTS OF IN VIVO NEURAL SIGNAL COMPRESSION ASICS

Work	Algorithm	Tech. (nm)	Supply (V)	Area/ch. (mm^2)†	Power/ch. (μW)†	CR	SNDR (dB)
[17]	DRR, Huffman coding (Lossless)	180	1.8	0.039	3.57	4.58	–
[18]	DRR (Lossless)	130	1.2	0.008	15.8	2	–
[25]	CS (Lossy)	180	1.2	0.008	3.55	4	$\simeq 14$
[55]	CS (Lossy)	130	1.2	0.087	31.003	10	–
[56]	CS (Lossy)	180	–	–	4.8	8	9.78
Ours	AE (Lossy)	180	1.8	0.002	0.076	19.2	Dataset I: 15 ± 3 Dataset II: 19 ± 3

† Normalized to a 180-nm CMOS process with a 1.8 V supply as described in [57], accounting only for the in vivo digital compression circuitry.

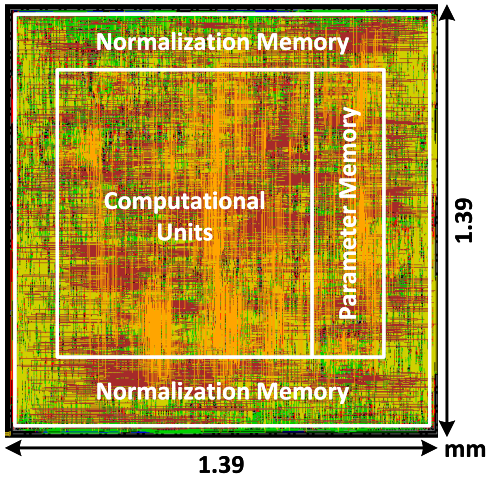

Fig. 11. 1.94 mm^2 ASIC layout of the synthesized in vivo encoder in a standard 180-nm CMOS process.

efficient design and implementation of the compression circuits, the ASIC characteristics and implementation results provided in Table II exclusively take into account the silicon area and power consumption of the in vivo compression circuits to ensure a fair comparison. While there are several techniques for lossless compression, the most commonly employed methods for compressing neural signals involves the dynamic range reduction (DRR) and variable wordlength encoding. In [17], the spatial and temporal correlation of neural signals were exploited to reduce the dynamic range of LFPs. First, a temporal difference was applied to the signal followed by removing the spatial average of groups of 16 recording channels. Then, Huffman encoding was applied to compress the LFPs to 2 or 3 bits prior to transmission. With the input LFPs represented using 11 bits, this resulted in an average compression rate of 4.58. The reported design compressed temporally-sparse spike waveforms by performing spike detection and transmitting only spike events rather than the continuous waveforms. For a fair comparison, our design is compared with the digital LFP compression hardware, which was reported to consume 3.59 μW of power per channel from a 1.8 V supply operating at 2 kHz. A similar lossless compression scheme was reported in [18] where the dynamic range of the signals was reduced by applying common-average referencing and subtracting this baseline from all channels prior to transmission. However, rather than encoding the signals, the baseline was transmitted for in silico reconstruction, resulting in an average compression rate of 2. The compression circuit was estimated to consume 6.4 μW of power per channel from a 1.2 V supply when operating at 400 kHz. In [25], an analog-based realization of a CS-based compression scheme was reported. The sensing matrix consists of samples uniformly selected from the Bernoulli distribution and achieves a compression rate of up to 16. The design was implemented in a 180-nm technology operating at 4 kHz and consuming 0.95 μW of power from a 1.2 V supply. In [55], another CS-based compression ASIC was presented, employing a novel Manhattan distance cluster-based sensing matrix for compressing multi-channel neural signals, achieving the compression rate of 10. The design was implemented in a 130-nm CMOS process and the simulated power consumption was 12.5 μW per channel from a 1.2 supply voltage. The CS-based compression system-on-chip reported in [56] was estimated to consume 4.8 μW per channel and to achieve a compression rate of 8.

As given in Table II, the designed and implemented autoencoder-based compression scheme consumes the lowest power per channel, while achieving a significantly higher compression rate than the previously reported designs. Compared to the compressed sensing methods for spatial dimensionality reduction, the designed and implemented autoencoder-based compression scheme offers reduction in power due to the spatio-temporal downsampling that drastically reduces the complexity of the encoder network. Additionally, the units in the encoder network learn to share parameters among different input channels, which has a small impact on its reconstruction quality, as was shown in Section IV. For practical BCI applications in which subsequent decoding and processing may tolerate reconstruction and wireless transmission errors reasonably well, the lossy methods with a greater compression ratio offer a more viable approach. Nevertheless, even though the lossless methods do not achieve relatively high compression ratios, their true signal reconstruction capability is a valuable attribute for carefully analyzing neural signals in silico.

Table III gives the power consumption of various neural signal processing ASICs employing alternative signal modalities. In [27] MUAs were obtained by performing spike detection. In [32] the data rate was further reduced by obtaining SUAs using spike detection and in vivo spike sorting. In [34] SUAs were obtained by compressing detected spike waveforms using an in vivo AE. Waveform reconstruction and spike sorting were

TABLE III
POWER CONSUMPTION OF THE IN VIVO NEURAL SIGNAL PROCESSING ASICS EMPLOYING MUA, SUA, AND LFP NEURAL SIGNALS

Work	Neural signal	Method	Power/ch. (μW)† in vivo digital signal processing	Power/ch. (μW) AFE
[27]	MUAs	Spike detection	0.64	7.4
[32]	SUAs	Spike sorting	2.02	
[34]			4.09	
Ours	LFPs	AE-based compression	0.076	1.4‡

† Normalized to a 180-nm CMOS process with a 1.8 V supply.
‡ Assuming the reduced sample rate and bandwidth requirements for LFPs over spike-based recordings [21], accounting only for the in vivo digital signal processing circuitry.

performed in silico. Evidently, employing more in vivo neural signal processing would consume more power per recording channel. The analog front end (AFE) of the neural recording circuitry, which consists of low-noise amplifiers and analog-to-digital converters, typically consume an average of 7.4 μW of power per channel [58]. The power consumption of the LNAs depend on various factors, such as the acceptable input-referred noise and whether chopper-stabilization is required to mitigate dominant flicker noise [59]. Additionally, the power consumption of the amplifier is also proportional to the width of the frequency band of interest [21], which for LFPs is about five times smaller than that of spikes. Conventional AFEs often employ successive approximation register (SAR) ADCs, whose power consumption is relatively linearly proportional to the sampling rate f_s [60], [61]. Therefore, it is reasonable to assume that reducing both the amplifier bandwidth and ADC sampling rate by a factor of five would yield a similar reduction in power consumption, approximately $7.4/5 = 1.5$ μW per channel. Therefore, the compression of LFPs would provide a more scalable recording scheme for BCIs employing high-density MEAs.

Intracortical BCIs must operate within strict low-power constraints to prevent heating of the brain tissues beyond 1 ° C, as outlined by the Food and Drug Administration. A maximum power budget of 7.7 mW has been reported for neural implants with wireless power transfer, based on 15-mm receiving antennas implanted on the cortical surface of the brain with an RF operating frequency of 2.02 GHz. Further analyses of different antenna sizes at various brain depths and their maximum power budgets are discussed in [16]. Fig. 12(a) shows the total power consumption for detecting MUAs and compressing LFPs with the designed AE-based digital circuit (including that of the AFE and wireless transmission), and the implant's power budget. Given a wireless transmission power of 158 pJ/bit [62], it becomes evident that the proposed compression would enable simultaneous recording from over 3000 channels, a significant increased compared to spike-based counterparts. Fig. 12(b) shows the supported data rates for transmitting MUAs, LFPs, and compressed LFPs over various number of channels. Considering the highest Bluetooth data rate of 2 Mbps, it can be seen that the transmission of raw LFPs is only feasible for fewer than 100 recording channels. The proposed AE-based

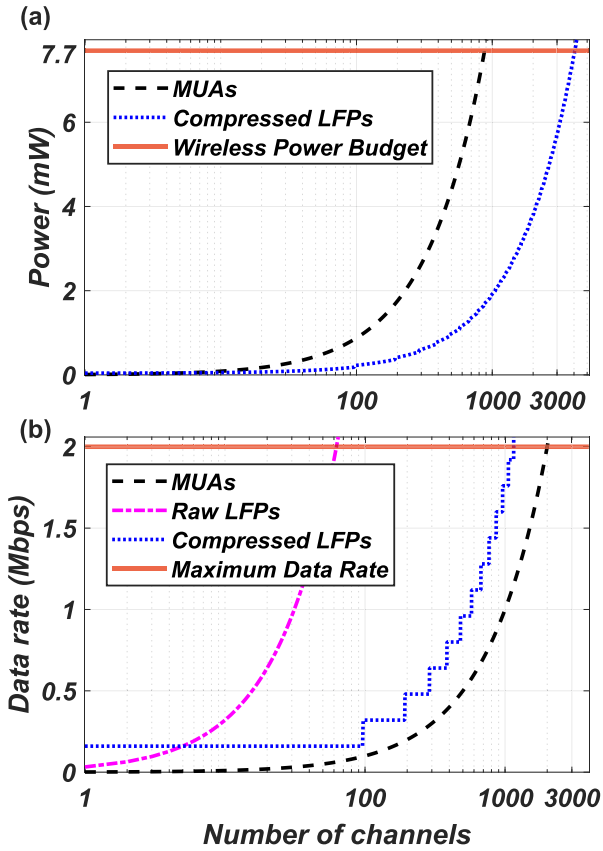

Fig. 12. (a) The total power consumption for detecting MUAs and compressed LFPs (including that of the AFE and wireless transmission) and (b) the supported data rate for MUAs, raw LFPs, and compressed LFPs over various number of channels.

compression scheme would make it possible to significantly increase the number of recording channels, approaching that of MUAs. Accounting for both the 7.7 mW implant's power budget and also the maximum data rate of 2 Mbps, the maximum number of channels to transmit raw LFPs is only 62, whereas that for MUAs with one millisecond bins is 878. By employing the designed AE-based compression, however, the maximum number of recording channels can be increased to 1152, an 18 times improvement over that for raw LFPs and a 1.3 times improvement over that for MUAs. In the above comparison, we assumed the conventional independent processing of multi-channel signals. To enhanced the overall density of intracortical electrodes, modern realizations of recording ICs decrease the electrode pitch by suppressing the sensitivity around the recording electrodes to avoid multiple detections of the same action potential. This isolation effectively reduces the overall processing of the neuronal data and in vivo power consumption. [44], [63].

VI. CONCLUSION

This article presents the design and implementation of an autoencoder-based compression scheme for in vivo compression of neural signals. Various optimization schemes were employed for the efficient hardware realization of the designed circuits.

Using two widely-employed neural datasets, we discussed and composed the reconstruction performance of the designed compression scheme against other state-of-the-art designs. We presented an application-specific integrated circuit of the designed in vivo encoder in a standard 180-nm CMOS process, estimated to occupy 0.02 mm^2 per channel and consume 0.076 μW of power per channel from a 1.8 V supply. Compared to the recently reported compression integrated circuits, the designed and synthesized compression architecture occupies the least silicon area, consumes the least power, offers the highest compression rate of over 19 times, and achieves a mean reconstruction quality of 17 dB over two datasets.

VI. Data Availability

The data presented in this study are openly available in Zenodo at https://doi.org/10.5281/zenodo.3854034, reference [45] (Dataset I) and in G-Node at https://doi.org/10.12751/g-node.f83565, reference [48] (Dataset II).

Acknowledgment

The authors would like to thank O'Doherty et al. [45] and Brochier et al. [48] for providing open access to Dataset I and II, respectively.

References

[1] G. H. Mulliken, S. Musallam, and R. A. Andersen, "Decoding trajectories from posterior parietal cortex ensembles," *J. Neurosci.*, vol. 28, no. 48, pp. 12913–12926, 2008.

[2] J. E. Downey et al., "Blending of brain-machine interface and vision-guided autonomous robotics improves neuroprosthetic arm performance during grasping," *J. Neuroeng. Rehabil.*, vol. 13, no. 1, pp. 1–12, 2016.

[3] J. J. Williams, R. N. Tien, Y. Inoue, and A. B. Schwartz, "Idle state classification using spiking activity and local field potentials in a brain computer interface," in *Proc. IEEE 38th Int. Conf. Eng. Med. Biol. Soc.*, 2016, pp. 1572–1575.

[4] H. Scherberger, M. R. Jarvis, and R. A. Andersen, "Cortical local field potential encodes movement intentions in the posterior parietal cortex," *Neuron*, vol. 46, no. 2, pp. 347–354, 2005.

[5] M. Filippini, A. Morris, R. Breveglieri, K. Hadjidimitrakis, and P. Fattori, "Decoding of standard and non-standard visuomotor associations from parietal cortex," *J. Neural Eng.*, vol. 17, no. 4, 2020, Art. no. 046027.

[6] F. R. Willett, D. T. Avansino, L. R. Hochberg, J. M. Henderson, and K. V. Shenoy, "High-performance brain-to-text communication via handwriting," *Nature*, vol. 593, no. 7858, pp. 249–254, 2021.

[7] G. K. Anumanchipalli, J. Chartier, and E. F. Chang, "Speech synthesis from neural decoding of spoken sentences," *Nature*, vol. 568, no. 7753, pp. 493–498, 2019.

[8] R. R. Harrison, "The design of integrated circuits to observe brain activity," *Proc. IEEE*, vol. 96, no. 7, pp. 1203–1216, Jul. 2008.

[9] E. Musk et al., "An integrated brain-machine interface platform with thousands of channels," *J. Med. Internet Res.*, vol. 21, no. 10, 2019, Art. no. e16194.

[10] N. A. Steinmetz et al., "Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings," *Science*, vol. 372, no. 6539, 2021, Art. no. eabf4588.

[11] H. Miranda and T. H. Meng, "A programmable pulse UWB transmitter with 34% energy efficiency for multichannel neuro-recording systems," in *Proc. IEEE Custom Integr. Circuits Conf.*, 2010, pp. 1–4.

[12] A. Hennessy and A. Alimohammad, "Design and implementation of a digital secure code-shifted reference UWB transmitter and receiver," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 64, no. 7, pp. 1927–1936, Jul. 2017.

[13] A. Hennessy and A. Alimohammad, "Compact digital implementation of a non-coherent IR-UWB transmitter and receiver," *IET Commun.*, vol. 12, no. 20, pp. 2616–2622, 2018.

[14] M. Song, Y. Huang, H. J. Visser, J. Romme, and Y.-H. Liu, "An energy-efficient and high-data-rate IR-UWB transmitter for intracortical neural sensing interfaces," *IEEE J. Solid-State Circuits*, vol. 57, no. 12, pp. 3656–3668, Dec. 2022.

[15] F. Jiang, M. Lin, X. Chen, L. Zhang, and X. Sheng, "A novel low power high speed Gbps UWB transmitter design using fractional DTC and high spectrum efficiency intra-chip PPM," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 71, no. 1, pp. 111–115, Jan. 2024.

[16] Y. Zhao, L. Tang, R. Rennaker, C. Hutchens, and T. S. Ibrahim, "Studies in RF power communication, SAR, and temperature elevation in wireless implantable neural interfaces," *PLoS One*, vol. 8, no. 11, 2013, Art. no. e77759.

[17] S.-Y. Park, J. Cho, K. Lee, and E. Yoon, "Dynamic power reduction in scalable neural recording interface using spatiotemporal correlation and temporal sparsity of neural signals," *IEEE J. Solid-State Circuits*, vol. 53, no. 4, pp. 1102–1114, Apr. 2018.

[18] Y. Khazaei, A. A. Shahkooh, and A. M. Sodagar, "Spatial redundancy reduction in multi-channel implantable neural recording microsystems," in *Proc. 42nd IEEE Conf. Eng. Med. Biol. Soc.*, 2020, pp. 898–901.

[19] A. Cuevas-López, E. Pérez-Montoyo, V. J. López-Madrona, S. Canals, and D. Moratal, "Low-power lossless data compression for wireless brain electrophysiology," *Sensors*, vol. 22, no. 10, 2022, Art. no. 3676.

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

[21] N. Even-Chen et al., "Power-saving design opportunities for wireless intracortical brain–computer interfaces," *Nature Biomed. Eng.*, vol. 4, no. 10, pp. 984–996, 2020.

[22] D. Valencia, G. Leone, N. Keller, P. P. Mercier, and A. Alimohammad, "Power-efficient in vivo brain-machine interfaces via brain-state estimation," *J. Neural Eng.*, vol. 20, no. 1, 2023, Art. no. 016032.

[23] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[24] S. Schmale, B. Knoop, J. Hoeffmann, D. Peters-Drolshagen, and S. Paul, "Joint compression of neural action potentials and local field potentials," in *Proc. IEEE Asilomar Conf. Signals, Syst. Comput.*, 2013, pp. 1823–1827.

[25] M. Shoaran, M. H. Kamal, C. Pollo, P. Vandergheynst, and A. Schmid, "Compact low-power cortical recording architecture for compressive multichannel data acquisition," *IEEE Trans. Biomed. Circuits Syst.*, vol. 8, no. 6, pp. 857–870, Dec. 2014.

[26] B. Sun and W. Zhao, "Compressed sensing of extracellular neurophysiology signals: A review," *Front. Neurosci.*, vol. 15, 2021, Art. no. 682063.

[27] D. Valencia, P. P. Mercier, and A. Alimohammad, "In vivo neural spike detection with adaptive noise estimation," *J. Neural Eng.*, vol. 19, no. 4, 2022, Art. no. 046018.

[28] D. A. Henze, Z. Borhegyi, J. Csicsvari, A. Mamiya, K. D. Harris, and G. Buzsaki, "Intracellular features predicted by extracellular recordings in the hippocampus in vivo," *J. Neuriophysiol.*, vol. 84, no. 1, pp. 390–400, 2000.

[29] M. S. Lewicki, "A review of methods for spike sorting: The detection and classification of neural action potentials," *Netw.: Computation Neural Syst.*, vol. 9, no. 4, 1998, Art. no. R53.

[30] D. Valencia and A. Alimohammad, "An efficient hardware architecture for template matching-based spike sorting," *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 3, pp. 481–492, Jun. 2019.

[31] D. Valencia and A. Alimohammad, "A real-time spike sorting system using parallel OSort clustering," *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 6, pp. 1700–1713, Dec. 2019.

[32] D. Valencia and A. Alimohammad, "Neural spike sorting using binarized neural networks," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 206–214, 2021.

[33] D. Valencia and A. Alimohammad, "Partially binarized neural networks for efficient spike sorting," *Biomed. Eng. Lett.*, vol. 13, no. 1, pp. 73–83, 2023.

[34] J. Thies and A. Alimohammad, "Compact and low-power neural spike compression using undercomplete autoencoders," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 27, no. 8, pp. 1529–1538, Aug. 2019.

[35] S. Todorova, P. Sadtler, A. Batista, S. Chase, and V. Ventura, "To sort or not to sort: The impact of spike-sorting on neural decoding performance," *J. Neural Eng.*, vol. 11, no. 5, 2014, Art. no. 056005.

[36] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski, *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*. Cambridge, U.K.: Cambridge Univ. Press, 2014.

[37] S. Katzner, I. Nauhaus, A. Benucci, V. Bonin, D. L. Ringach, and M. Carandini, "Local origin of field potentials in visual cortex," *Neuron*, vol. 61, no. 1, pp. 35–41, 2009.

[38] G. Buzsáki, C. A. Anastassiou, and C. Koch, "The origin of extracellular fields and currents—EEG, ECoG, LFP and spikes," *Nature Rev. Neurosci.*, vol. 13, no. 6, pp. 407–420, 2012.
[39] S. Saha et al., "Progress in brain computer interface: Challenges and opportunities," *Front. Syst. Neurosci.*, vol. 15, 2021, Art. no. 578875.
[40] J. E. Downey, N. Schwed, S. M. Chase, A. B. Schwartz, and J. L. Collinger, "Intracortical recording stability in human brain–computer interface users," *J. Neural Eng.*, vol. 15, no. 4, 2018, Art. no. 046016.
[41] J. W. Salatino, K. A. Ludwig, T. D. Kozai, and E. K. Purcell, "Glial responses to implanted electrodes in the brain," *Nature Biomed. Eng.*, vol. 1, no. 11, pp. 862–877, 2017.
[42] S. Musallam, B. Corneil, B. Greger, H. Scherberger, and R. A. Andersen, "Cognitive control signals for neural prosthetics," *Science*, vol. 305, no. 5681, pp. 258–262, 2004.
[43] T. Aflalo et al., "Decoding motor imagery from the posterior parietal cortex of a tetraplegic human," *Science*, vol. 348, no. 6237, pp. 906–910, 2015.
[44] Y. Chen et al., "A 384-channel online-spike-sorting IC using unsupervised geo-osort clustering and achieving 0.0013 mm ^{2}/ch and 1.78 μ W/ch," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2023, pp. 486–488.
[45] J. E. O'Doherty, M. M. B. Cardoso, J. G. Makin, and P. N. Sabes, "Nonhuman primate reaching with multichannel sensorimotor cortex electrophysiology," *Zenodo*, May 2020, doi: 10.5281/zenodo.3854034.
[46] E.-R. Ardelean, A. Coporîie, A.-M. Ichim, M. Dînşoreanu, and R. C. Mureşan, "A study of autoencoders as a feature extraction technique for spike sorting," *PLoS One*, vol. 18, no. 3, 2023, Art. no. e0282810.
[47] J. Eom et al., "Deep-learned spike representations and sorting via an ensemble of auto-encoders," *Neural Netw.*, vol. 134, pp. 131–142, 2021.
[48] T. Brochier et al., "Massively parallel recordings in macaque motor cortex during an instructed delayed reach-to-grasp task," *Sci. Data*, vol. 5, no. 1, pp. 1–23, 2018.
[49] W. H. Greub, *Linear Algebra*, vol. 23. Berlin, Germany: Springer Science & Business Media, 2012.
[50] N. Ahmadi et al., "Towards a distributed, chronically-implantable neural interface," in *Proc. Int. IEEE/EMBS Conf. Neural Eng.*, 2019, pp. 719–724.
[51] M. M. Ghanbari et al., "A sub-mm^3 ultrasonic free-floating implant for multi-mote neural recording," *IEEE J. Solid-State Circuits*, vol. 54, no. 11, pp. 3017–3030, Nov. 2019.
[52] J. Lee et al., "Neural recording and stimulation using wireless networks of microimplants," *Nature Electron.*, vol. 4, no. 8, pp. 604–614, 2021.
[53] F. Hashemi Noshahr, M. Nabavi, and M. Sawan, "Multi-channel neural recording implants: A review," *Sensors*, vol. 20, no. 3, 2020, Art. no. 904.
[54] D. Valencia, S. F. Fard, and A. Alimohammad, "An artificial neural network processor with a custom instruction set architecture for embedded applications," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 67, no. 12, pp. 5200–5210, Dec. 2020.
[55] N. Li, M. Osborn, G. Wang, and M. Sawan, "A digital multichannel neural signal processing system using compressed sensing," *Digit. Signal Process.*, vol. 55, pp. 64–77, 2016.
[56] X. Liu et al., "A fully integrated wireless compressed sensing neural signal acquisition system for chronic recording and brain machine interface," *IEEE Trans. Biomed. Circuits Syst.*, vol. 10, no. 4, pp. 874–883, Aug. 2016.
[57] A. Stillmaker, Z. Xiao, and B. Baas, "Toward more accurate scaling estimates of CMOS circuits from 180 nm to 22 nm," VLSI Computation Lab, ECE Department, University of California, Davis, CA, USA, Tech. Rep. ECE-VCL-2011-4, vol. 4, 2011.
[58] J. Li, X. Liu, W. Mao, T. Chen, and H. Yu, "Advances in neural recording and stimulation integrated circuits," *Front. Neurosci.*, vol. 15, 2021, Art. no. 663204.
[59] S. Mondal, C.-L. Hsu, R. Jafari, and D. Hall, "A dynamically reconfigurable ECG analogfront-end with a 2.5 $\times$ data-dependent power reduction," in *Proc. IEEE Custom Integr. Circuits Conf.*, 2017, pp. 1–4.
[60] M. Yip and A. P. Chandrakasan, "A resolution-reconfigurable 5-to-10-bit 0.4-to-1 v power scalable SAR ADC for sensor applications," *IEEE J. Solid-State Circuits*, vol. 48, no. 6, pp. 1453–1464, Jun. 2013.
[61] H. Wang, X. Wang, A. Barfidokht, J. Park, J. Wang, and P. P. Mercier, "A battery-powered wireless ion sensing system consuming 5.5 nW of average power," *IEEE J. Solid-State Circuits*, vol. 53, no. 7, pp. 2043–2053, Jul. 2018.
[62] J. Rosenthal and M. S. Reynolds, "A 158 pj/bit 1.0 mbps bluetooth low energy (BLE) compatible backscatter communication system for wireless sensing," in *Proc. Conf. Wireless Sensors Sensor Netw.*, 2019, pp. 1–3.
[63] M. Jang et al., "A 1024-channel 268 nw/pixel 36$\times$36 μm 2/ch data-compressive neural recording IC for high-bandwidth brain-computer interfaces," in *Proc. IEEE Symp. VLSI Technol. Circuits*, 2023, pp. 1–2.

Daniel Valencia is currently working toward the Ph.D. degree in the joint doctoral program with San Diego State University and University of California, San Diego, CA, USA. He is currently a graduate Researcher with the VLSI Design and Test Laboratory, Department of Electrical and Computer Engineering, San Diego State University, San Diego. His research interests include field-programmable gate arrays, brain-computer interfacing, and VLSI architectures for neural signal processing.

Patrick P. Mercier (Senior Member, IEEE) received the B.Sc. degree in electrical and computer engineering from the University of Alberta, Edmonton, AB, Canada, in 2006, and the S.M. and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 2008 and 2012, respectively. He is currently a Professor of electrical and computer engineering with the University of California San Diego, San Diego, CA, USA, where he is also the co-Director of the Center for Wearable Sensors and the Site Director of the Power Management Integration Center. His research interests include the design of energy-efficient microsystems, focusing on the design of RF circuits, power converters, and sensor interfaces for miniaturized systems and biomedical applications. Prof. Mercier has authored or coauthored 200 peer-reviewed papers, including 26 ISSCC papers, 34 JSSC papers, and several papers in high-impact journals, such as *Science*, *Nature Biotechnology*, *Nature Biomedical Engineering*, *Nature Electronics*, *Nature Communications*, and *Advanced Science*. He was the recipient of numerous awards, including a Natural Sciences and Engineering Council of Canada (NSERC) Julie Payette fellowship in 2006, NSERC Postgraduate Scholarships in 2007 and 2009, an Intel Ph.D. Fellowship in 2009, the 2009 IEEE International Solid-State Circuits Conference (ISSCC) Jack Kilby Award for Outstanding Student Paper at ISSCC 2010, Graduate Teaching Award in Electrical and Computer Engineering at UCSD in 2013, Hellman Fellowship Award in 2014, Beckman Young Investigator Award in 2015, DARPA Young Faculty Award in 2015, UC San Diego Academic Senate Distinguished Teaching Award in 2016, Biocom Catalyst Award in 2017, NSF CAREER Award in 2018, National Academy of Engineering Frontiers of Engineering Lecture in 2019, San Diego County Engineering Council Outstanding Engineer Award in 2020, ISSCC Author Recognition Award in 2023, and ECE Teacher of the Year award in 2023. He was an Associate Editor for the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION(TVLSI), IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS(TBioCAS), and IEEE SOLID-STATE CIRCUITS LETTERS. He is currently a member of the Executive Committee of ISSCC, and has served on the technical program committees for ISSCC, CICC, and the VLSI Symposium. Prof. Mercier was the co-editor of *Ultra-Low-Power Short Range Radios* (Springer, 2015) *Power Management Integrated Circuits* (CRC Press, 2016), and *High-Density Electrocortical Neural Interfaces* (Academic Press, 2019).

Amir Alimohammad received the Ph.D. degree from the University of Alberta, Edmonton, AB, Canada. He is currently a Professor with the Department of Electrical and Computer Engineering, San Diego State University, San Diego, CA, USA. His research interests include digital VLSI design for brain-computer interfacing and wireless communication. He is the Editor for the board of *Neuroprosthetics at Frontiers in Neuroscience* and *Neurology*. He is also a Member of the National Science Foundation Center for Neurotechnology.