A Data-Compressive Wired-OR Readout for Massively Parallel Neural Recording

Dante G. Muratore*, Pulkit Tandon*, Mary Wootters*†, E.J. Chichilnisky‡§, Subhasish Mitra*†, Boris Murmann*
Department of *Electrical Engineering, †Computer Science, ‡Neurosurgery and Ophthalmology,
and §Hansen Experimental Physics Laboratory, Stanford University, USA

***Abstract*—This paper describes an architecture for the massively parallel digitization of neural action potentials. The scheme achieves simultaneous data compression and channel multiplexing through wired-OR interactions within an array of single-slope A/D converters. The achieved compression is lossy but effective at retaining the critical samples belonging to action potential spikes. Simulation results using *ex-vivo* experimental data from a 512-channel array show compression rates up to ∼73x while maintaining ≥90% reconstruction coverage for parasol cells in the primate retina.**

***Index Terms*—Neural Interfaces, Compression Algorithm, Brain-Machine Interfaces, A/D conversion.**

I. Introduction

Multi-channel action potential recording systems are widely used in neuroscientific studies and emerging clinical applications [1], [2]. While first-generation interfaces had limited electrode counts of ≤100, present research systems target significantly larger and denser arrays for single-cell specificity. Fig. 1(a) shows an example of a 512-electrode array (60 μm pitch) that is used for an *ex-vivo* study of the retina using tethered electronics. To advance scientific discovery and clinical applications, future systems must support parallel *in-vivo* recording from tens of thousands of electrodes within the form factor and power budget of a fully implanted device. However, meeting these requirements poses a number of significant engineering challenges [3].

Assuming 100,000 channels and a heat dissipation limit of 10 mW, the per-channel power budget is only 100 nW. Even if this was achievable for the A/D interface, moving the immense amount of data on-chip and transmitting it off-chip would be another major hurdle (e.g., 10 bits x 20 kS/s x 100,000 = 20 Gb/s). Due to both of these issues, today's large-scale interfaces are limited to sub-array digitization. For example, the readout of [4] supports 59,760 electrodes, but only 2,048 are simultaneously addressable.

Researchers have investigated a wide range of options to combat these challenges. In applications that require only binary spike train information, large power and data reductions are possible via on-chip thresholding [5], [6], [7]. However, this precludes off-chip spike sorting and thus achieving single-cell resolution. To achieve data reduction without sacrificing fine-grain information, on-chip spike sorting [8] and compression [9] have been considered. However, these techniques do not alleviate the issues with massive multiplexing and digitization at the front-end. On the other hand, the works

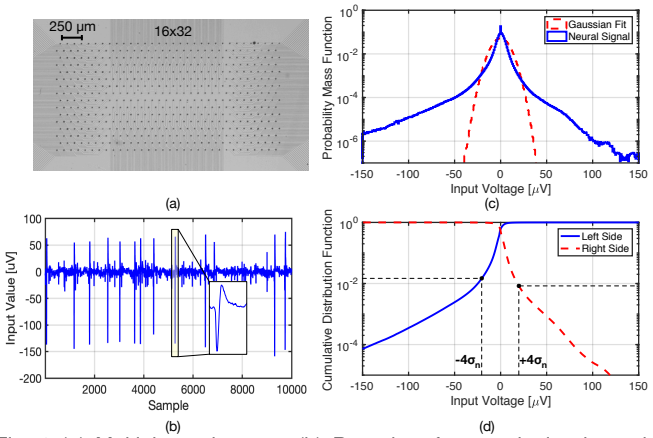

Fig. 1 (a) Multielectrode array. (b) Raw data from a single electrode. (c) Probability mass function and (d) cumulative distribution function of 100,000 samples from 512 electrodes after offset removal.

of [10], [11], [12] aim to improve the efficiency of active analog multiplexing, but do not address the post-digitization data deluge. An intriguing technique where both multiplexing and compression occur simultaneously via analog channel superposition was proposed in [13]. However, the scalability of this approach is limited by the noise summation of the superimposed channels.

The scheme proposed in this paper is philosophically aligned with [13]. It integrates multiplexing and compression within the array circuitry and does not expend compute energy to eliminate unwanted data. As we describe in more detail below, the compression is lossy, but effective at retaining critical samples belonging to the spikes of the action potential waveforms (see Fig. 1(b)), which are necessary for spike sorting and single-cell detection. Our approach exploits the fact that spikes occur at a low rate and hence make up only a small portion of the signal's probability mass. This is illustrated in Figs. 1(c)-(d), which show the probability mass function (PMF) and cumulative distribution function (CDF) of experimental data from a 16x32 electrode array. The PMF is large around zero and shows a long tail due to the sparse spike activity (a normal distribution is plotted for reference). As an example, if we assume that meaningful spikes exceed four times the noise standard deviation, we see that only about 2.5% of the samples suffice to represent such activity. It should therefore be possible to achieve compression rates on the order of 40x by discarding unwanted samples near the baseline.

Section II describes the proposed architecture, which inte-

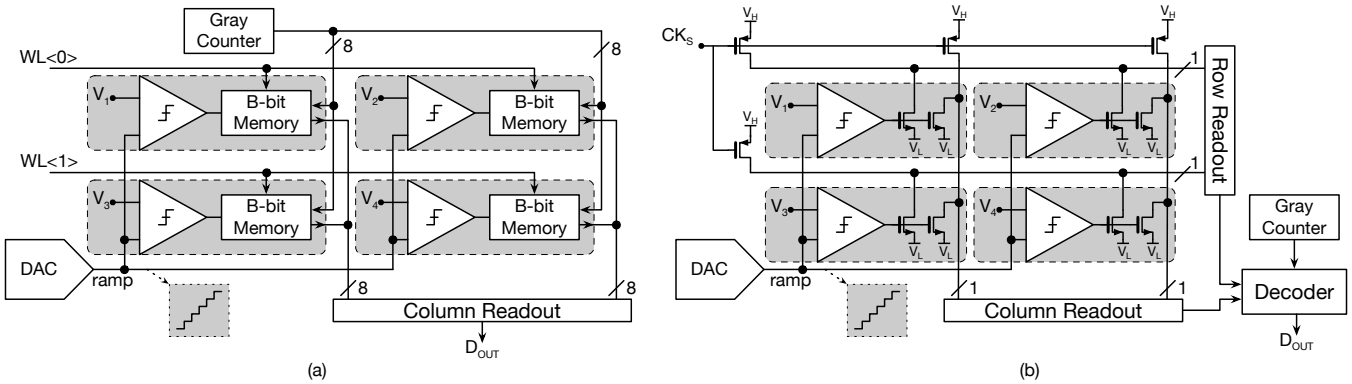

Fig. 2 (a) CMOS image sensor readout concept of [14] (2x2 example). (b) Modified architecture with wired-OR readout. Input signal conditioning is omitted for simplicity.

grates the comparator of a single-slope ADC within each active pixel. The compression occurs through a digital wired-OR competition between comparators in the same rows/columns and avoids the analog noise summation issue of [13]. Section III describes simulation results using *ex-vivo* experimental data. We observe compression rates up to ∼73x while maintaining ≥90% reconstruction coverage for parasol cells in the primate retina.

II. Readout Architecture

Conceptually, a multi-channel neural readout is similar to a digital CMOS image sensor (CIS), which digitizes each of its pixels within a single frame. However, while the pixel count of a CIS can be very large (typically millions), the frame rate is usually much lower than the required sample rate for neural signals (tens of frames/s vs. ∼20 kS/s). Our design was inspired by an exceptionally fast CIS that operates at 10,000 frames/s [14] (see Fig. 2(a)). In this architecture, the digitization is based on the single-slope conversion principle. The required ramp signal is globally distributed, and each pixel contains a voltage comparator that latches a B-bit counter signal to perform a voltage-to-time conversion. At the end of each ramp cycle, the latch data is read one row at a time.

While this architecture can achieve fast and relatively efficient array digitization, its power dissipation is dominated by the cost of distributing the counter state and reading out the digital bits from each location (the aggregate data rate will be on the order of Gb/s). In the proposed architecture, we eliminate this bottleneck by combining the comparator outputs through wired-OR connections, see Fig. 2(b). The horizontal and vertical wire states are sensed by peripheral readout circuits at each ramp value and are passed to a decoder.

Fig. 3(a) illustrates an idealized (and typically rare) event in this readout scheme. If only one comparator within the array triggers for a given ramp value, then the location of this comparator and thus the associated A/D conversion value can be uniquely determined. However, if multiple comparators trigger simultaneously, there exists no unique decoding solution and we refer to this situation as a collision. In addition to the collision-free case, we distinguish between a small collision where a limited number of channels are activated together (Fig. 3(b)) and a massive collision (Fig. 3(c)).

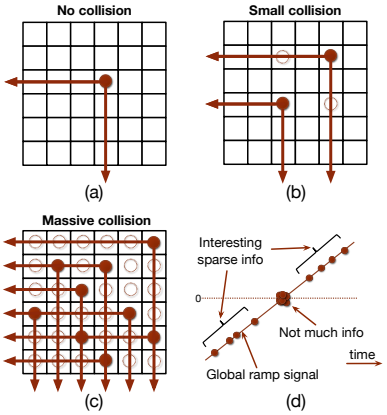

Fig. 3 Wired-OR signal scenarios. (a) Collision-free case. (b) Small collision caused by a few comparators triggering at the same time. (c) Massive collision. (d) Distribution of information during ramp cycle.

The probability and severity of a collision depends on the signal distribution. From Fig. 1, we know that the neural signals spend most of their time near the baseline. Thus, a massive collision likely occurs for ramp values around zero (middle region in Fig. 3(d)), which does not correspond to useful information (no spike activity). Note that we assume here that the offset between channels is small with respect to the ramp quantization step. This imposes an important, but manageable constraint for the circuit implementation. Conversely, most of the interesting information (spiking activity) leads to collision-free or small-collision behavior. The simulations described in the next section study how well these recoverable samples represent the desired spike information.

III. Simulations

A. Methods

To evaluate the proposed scheme, we use 512-channel data recorded from *ex-vivo* experiments with a primate retina [15], [16]. Each channel is digitized at 20 kS/s with 10 bits of resolution (post offset removal) and an input-referred noise of ∼7.5 μVrms. Fig. 4(a) shows the experimental setup and the data processing pipeline. A live retina is placed on top of the 16x32 electrode array, and a movie displayed on a computer screen is focused on the photoreceptors. The spikes generated in the ganglion cells are recorded and digitally processed after

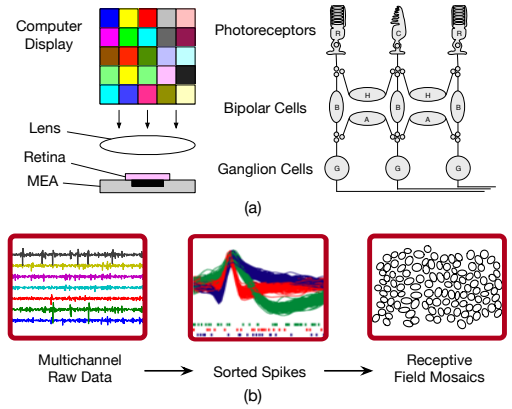

Fig. 4 *Ex-vivo* recording setup and data processing.

offset removal. In addition to spike sorting (the process that attributes action potentials to putative individual neurons), the optical receptive field of each cell is calculated by correlating the visual images focused on the retina with the neuron's spiking activity. Neurons can be further classified into different cell types based on their temporal response properties and their receptive fields. Ultimately, the receptive fields of a single cell type form a mosaic that covers the entire scene (see Fig. 4(b)).

For the purpose of our study, we emulate the array digitization process described in the previous section (with ramp resolutions of 6-10 bits) by re-processing the recorded data in software. Since the ultimate goal of the neural interface is to infer the structure of the underlying biology, we compute the mosaic completeness of ON and OFF parasol cells as our main metric of interest. Additionally, we report the normalized mean square error (NMSE) for sample magnitudes larger than $4\sigma_n$ (see Fig. 1) as a measure of signal fidelity.

B. Encoding and Decoding Strategies

1) Naïve Decoder: The most basic decoder takes only collision-free samples and discards all other data from the array. Fig. 5(a) illustrates an example that shows the kept points using blue markers. Missing samples due to collisions are initially set to zero and subsequently reconstructed using a 3-tap non-causal finite impulse response (FIR) filter (orange curve) with coefficients $b_{-1} = 0.5, b_0 = 0, b_{+1} = 0.5$. Figs. 5(a)-(b) show a comparison between the original and the reconstructed data for 10 and 8 bits. As expected, reduced resolution leads to an increased number of collisions and additional waveform distortion. Compression is achieved by outputting only the address of the collision-free channels and reconstructing the data off-chip. The required data rate depends on the rate of collision-free channels per sample, α_{cf}.

2) Multi-Wire Encoder: Beyond their distribution, another important metric of the neural signals is their correlation. Fig. 6(a) depicts the correlation matrix for channel (6, 19), showing significant correlation with adjacent electrodes. As a result, collisions are likely to happen due to nearby comparators triggering at the same time. Encoding the comparator outputs onto different row and column wires for adjacent channels rejects this correlation and reduces the number of collisions (Fig. 6(b)). The incurred cost of extra digital wires

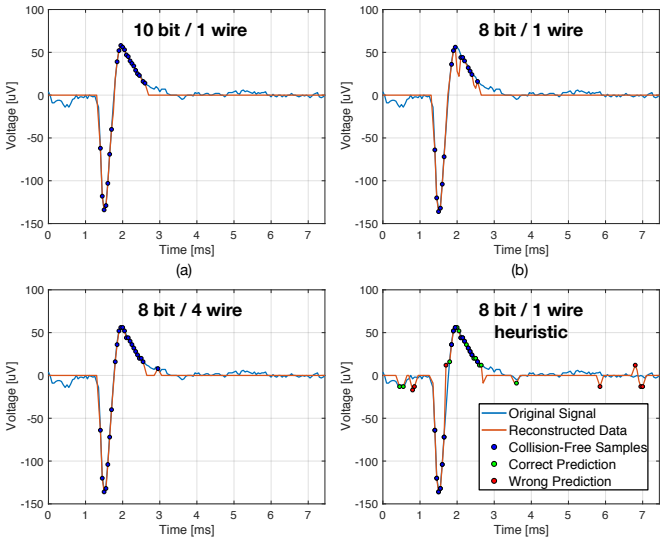

Fig. 5 Examples of reconstructed spikes for (a) naïve 10 bits/1 wire, (b) naïve 8 bits/1 wire, (c) naïve 8 bits/4 wires, (d) heuristic 8 bits/1 wire.

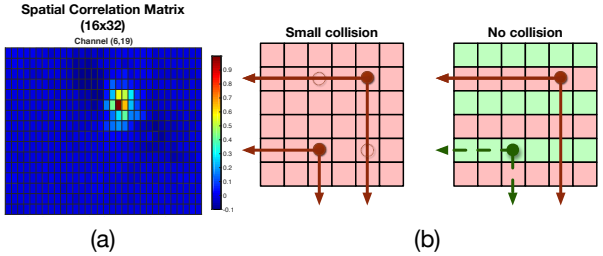

Fig. 6 (a) Correlation matrix for channel (6, 19) and correlation-induced collision. (b) Example of collision avoidance with 2-wire encoding.

is minimal for scaled technologies, as long as the wire count is kept significantly smaller than the number of electrodes. Here, we limit our analysis to 2, 4 and 8 wires per row/column. Figs. 5(b)-(c) show waveforms for naïvely decoded data with single-wire and four-wire encoding at 8-bit resolution. Four-wire encoding reduces the number of collisions and the waveform distortion. The data rate is

$$R = \lceil log_2(N_{row}/W) + log_2(N_{col}) \rceil \alpha_{cf,W} f_s \qquad (1)$$

where N_{row} and N_{col} are the number of rows and columns in the array (16x32 in this paper), W is the number of wires per row/column, $\alpha_{cf,W} > \alpha_{cf}$ (for $W > 1$) is the total rate of collision-free channels per sample and f_s is the sampling frequency. The channels are connected to the additional wires such that rows are interleaved in the encoding strategy. Hence, rows can be addressed with only $\lceil log_2(N_{row}/W) \rceil$ bits (see Fig. 6(b)).

3) Probabilistic/Heuristic Decoder: The above-discussed schemes operate only with collision-free samples, but there should exist reconstruction algorithms that can utilize collision data to improve the reconstruction at the expense of lower compression. In this section, we investigate a heuristic decoder that exploits the spatial and amplitude continuity of the neural signals to resolve collisions.

The decoder maintains a prior probability matrix $P \in \mathbb{R}^{N_{\text{row}} \times N_{\text{col}}}$ over the whole array at each time sample. If

TABLE I Mosaic coverage for parasol cells as a function of bit resolution and wiring scheme.

	10 bit		8 bit		6 bit	
	ON	OFF	ON	OFF	ON	OFF
1w	99%	95%	98%	90%	83%	81%
2w	99%	97%	100%	93%	90%	85%
4w	99%	99%	99%	97%	93%	88%
8w	100%	99%	100%	99%	97%	84%

TABLE II Average compression rate as a function of bit resolution and wiring scheme.

	10 bit	8 bit	6 bit
1 wire	35x	73x	151x
1 wire - heuristic*	22x	48x	102x
2 wire	20x	41x	85x
4 wire	11x	24x	49x
8 wire	7x	14x	28x

*$P_{\text{inc}} = 1.5$, $P_{\text{dec}} = 0.9$, $c = 2$ and $r_{\text{firing}} = 1$

a spike is received at a particular electrode, its surrounding electrodes will present a similar amplitude with high probability (see Fig. 6(a)), leading to a potential collision. The heuristic algorithm captures this effect by increasing the prior matrix P by a hyper-parameter P_{inc} for all the collision-free channels and the ones adjacent to it. The prior P for all the other channels will be decreased by another hyper-parameter P_{dec} to signify receding activity from these channels. If a small collision occurs at a particular ramp-step the algorithm chooses the r_{firing} (another hyper-parameter) channels with highest prior value P and thus resolves the collision by choosing the most-likely channels to carry that signal value. A collision is defined as small if the number of rows (and columns) activated is less than a hyper-parameter c. To help the heuristic algorithm resolve collisions effectively, the decoder processes the signal from the lowest ramp value to zero and then from the highest ramp value to zero, so that the prior matrix is first setup with collision-free cases before it must resolve collision cases.

Fig. 5(d) shows an example waveform produced by the heuristic algorithm with $P_{\text{inc}} = 1.5$, $P_{\text{dec}} = 0.9$, $c = 10$ and $r_{\text{firing}} = 5$. Samples that were not recovered for the case of Fig. 5(b) are now correctly predicted (green markers). Extra errors are added in cases where the algorithm makes a wrong prediction (red markers).

C. Results

Table I summarizes the mosaic coverage metric for our target application (with a ground truth of 150 OFF parasol and 116 ON parasol cells) and Table II shows the achieved average compression rate (CR). The CR is calculated relative to a conventional architecture for each bit resolution, $CR = N_{row}N_{col}Bf_s/R$, where R is defined in (1). As bit resolution and wire complexity increase, the mosaic coverage improves and approaches the reference case for 10 bits, while the compression rate decreases. For reference, Fig. 7 shows a reconstructed OFF parasol mosaic with 90% coverage, which is an acceptable outcome for our application. Table III also reports the achieved NMSE for our experiments with the naïve and heuristic decoders, and the inherent NMSE due to re-quantization given in the first row. Overall, we found that the heuristic decoder does not outperform the naïve approach in detecting cells, even when it achieves better NMSE. Also, comparing Tables I and III, it is interesting to note that the 8 bit/1 wire configuration has a worse NMSE than the 6 bit/8 wire configuration, but shows superior cell reconstruction.

TABLE III Normalized mean squared error as a function of bit resolution and wiring scheme.

	10 bit	8 bit	6 bit
Conventional	$-\infty$ dB	-30.1 dB	-17.4 dB
1 wire	-14.7 dB	-7.3 dB	-2.1 dB
1 wire - Heuristic*	-16.4 dB	-8.3 dB	-2.6 dB
2 wire	-19.4 dB	-11.0 dB	-4.3 dB
4 wire	-24.1 dB	-15.2 dB	-6.9 dB
8 wire	-29.7 dB	-19.9 dB	-9.9 dB

*$P_{\text{inc}} = 1.5$, $P_{\text{dec}} = 0.9$, $c = 2$ and $r_{\text{firing}} = 1$

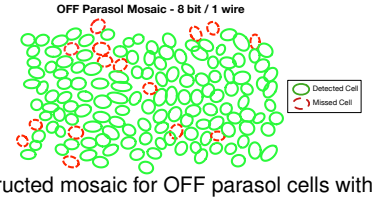

Fig. 7 Reconstructed mosaic for OFF parasol cells with 90% coverage.

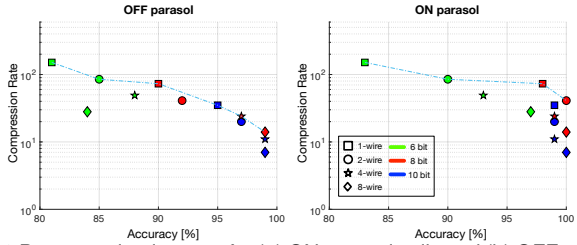

Fig. 8 Pareto-optimal curves for (a) ON parasol cells and (b) OFF parasol cells for naïve decoding and multi-wire encoding. Data points are from Table I.

In summary, we find that a cell coverage $\geq$95% can be achieved with compression rates up to 35x (see Fig. 8). If lower coverage can be tolerated, the data can be compressed by more than two orders of magnitude. As the number of wires per row/column approaches the total number of wires (16 in this analysis), collisions become impossible and the coverage approaches 100%.

IV. Discussion

We have presented a data-compressive readout strategy for the digitization of neural action potentials. The compression occurs through a wired-OR competition between single-slope ADCs and maintains sufficient performance for a representative application in retina cell mapping. Although our experiments are based on a 512-electrode array, the scheme will extend to larger arrays through sub-partitioning (an extension of the discussed multi-wire encoding). In addition, the approach may be useful for other types of sensor arrays with sparse signal activity.

Our future work will focus on alternative decoding and reconstruction strategies, as well as the design of a prototype IC. An important aspect of the hardware implementation is to maintain small offsets between the baselines of each channel, which is required to align the PMFs of the competing signals.

REFERENCES

[1] M.W. Slutzky, "Brain-Machine Interfaces: Powerful Tools for Clinical Treatment and Neuroscientific Investigations," *Neuroscientist*, May 2018.

[2] M. A. Lebedev, and M. A. L. Nicolelis, "Brain-Machine Interfaces: From Basic Science to Neuroprostheses and Neurorehabilitation," *Physiological Rev.*, vol. 97, no. 2, pp. 767-837, Mar. 2017.

[3] A. Marblestone, B. Zamft, Y. Maguire, M. Shapiro, T. Cybulski, J. Glaser, D. Amodei, P. B. Stranges, R. Kalhor, D. Dalrymple, D. Seo, E. Alon, M. Maharbiz, J. Carmena, J. Rabaey, E. Boyden, G. Church, and K. Kording, "Physical Principles for Scalable Neural Recording", *Frontiers in Computational Neuroscience*, vol. 7, no. 137, pp. 1-34, Oct. 2013.

[4] J. Dragas, V. Viswam, A. Shadmani, Y. Chen, R. Bounik, A. Stettler, M. Radivojevic, S. Geissler, M. E. J. Obien, J. Mller, and A. Hierlemann, "In Vitro Multi-Functional Microelectrode Array Featuring 59 760 Electrodes, 2048 Electrophysiology Channels, Stimulation, Impedance Measurement, and Neurotransmitter Detection Channels," *IEEE J. Solid-State Circuits*, vol. 52, no. 6, pp. 1576-1590, Jun. 2017.

[5] A. M. Sodagar, K. D. Wise, and K. Najafi, "A Fully Integrated Mixed-Signal Neural Processor for Implantable Multichannel Cortical Recording", *IEEE Trans. on Biomedical Engineering*, vol. 54, no. 6, pp. 1075-1088, Jun. 2007.

[6] R. R. Harrison, R. J. Kier, C. A. Chestek, V. Gilja, P. Nuyujukian, S. Ryu, B. Greger, F. Solzbacher, and K. V. Shenoy, "Wireless Neural Recording with Single Low-Power Integrated Circuit", *IEEE Trans. Neural Systems and Rehabilitation Engineering*, vol. 17, no. 4, pp. 322-329, Aug. 2009.

[7] Z. T. Irwin, D. E. Thompson, K. E. Schroeder, D. M. Tat, A. Hassani, A. J. Bullard, S. L. Woo, M. G. Urbanchek, A. J. Sachs, P. S. Cederna, W. C. Stacey, P. G. Patil, and C. A. Chestek, "Enabling Low-Power, Multi-Modal Neural Interfaces through a Common, Low-Bandwidth Feature Space", *IEEE Trans. Neural Systems and Rehabilitation Engineering*, vol. 24, no. 5, pp. 521-531, May 2016.

[8] V. Karkare, S. Gibson and D. Markovic, "A 75-μW, 16-Channel Neural Spike-Sorting Processor With Unsupervised Clustering," *IEEE J. Solid-State Circuits*, vol. 48, no. 9, pp. 2230-2238, Sep. 2013.

[9] M. Pagin and M. Ortmanns, "A Neural Data Lossless Compression Scheme Based on Spatial and Temporal Prediction," *IEEE Biomedical Circuits and Systems Conf.*, Oct. 2017.

[10] V. Majidzadeh, A. Schmid and Y. Leblebici, "A 16-Channel, 359 μW, Parallel Neural Recording System Using Walsh-Hadamard Coding," *IEEE Custom Integrated Circuits Conf.*, Nov. 2013.

[11] D. Tsai, R. Yuste and K. L. Shepard, "Statistically Reconstructed Multiplexing for Very Dense, High-Channel-Count Acquisition Systems," *IEEE Trans. Biomedical Circuits and Systems*, vol. 12, no. 1, pp. 13-23, Feb. 2018.

[12] M. Sharma, A. T. Gardner, H. J. Strathman, D. J. Warren, J. Silver and R. M. Walker, "Acquisition of Neural Action Potentials Using Rapid Multiplexing Directly at the Electrodes", *Micromachines*, vol. 9, no. 477, pp. 1-22, Sep. 2018.

[13] J. D. Rieseler, M. Kuhl, "A Superposition-Based Analog Data Compression Scheme for Massively-Parallel Neural Recordings," *IEEE Biomedical Circuits and Systems Conf.*, Oct. 2017.

[14] S. Kleinfelder, S. Lim, X. Liu, and A. El Gamal, "A 10000 Frames/s CMOS Digital Pixel Sensor," *IEEE J. Solid-State Circuits*, vol. 36, no. 12, pp. 2049-2059, Dec. 2001.

[15] A. M. Litke, N. Bezayiff, E. J. Chichilnisky, W. Cunningham, W. Dabrowski, A. A. Grillo, M. Grivich, P. Grybos, P. Hottowy, S. Kachiguine, R. S. Kalmar, K. Mathieson, D. Petrusca, M. Rahman, and A. Sher, "What Does the Eye Tell the Brain?: Development of a System for the Large-Scale Recording of Retinal Output Activity," *IEEE Trans. Nuclear Science*, vol. 51, no. 4, pp. 1434-1440, Aug. 2004.

[16] E. S. Frechette, A. Sher, M. I. Grivich, D. Petrusca, A. M. Litke, and E. J. Chichilnisky, "Fidelity of the Ensemble Code for Visual Motion in Primate Retina," *J. Neurophysiology*, vol. 94, no. 1, pp. 119-135, Jul. 2005.