Journal of Neural Engineering

PAPER

A closed-loop compressive-sensing-based neural recording system

To cite this article: Jie Zhang *et al* 2015 *J. Neural Eng.* **12** 036005

IOP Publishing Journal of Neural Engineering
J. Neural Eng. 12 (2015) 036005 (17pp) doi:10.1088/1741-2560/12/3/036005

A closed-loop compressive-sensing-based neural recording system

Jie Zhang[1], Srinjoy Mitra[2], Yuanming Suo[1], Andrew Cheng[3], Tao Xiong[1], Frederic Michon[2], Marleen Welkenhuysen[2], Fabian Kloosterman[2], Peter S Chin[4], Steven Hsiao[3], Trac D Tran[1], Firat Yazicioglu[2] and Ralph Etienne-Cummings[1]

[1] Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, USA
[2] IMEC, Leuven, Belgium
[3] Department of Biomedical Engineering, Johns Hopkins University, Baltimore, USA
[4] Draper Laboratory, Boston, USA

E-mail: jzhang41@jhu.edu

Received 3 October 2014, revised 26 January 2015
Accepted for publication 16 February 2015
Published 15 April 2015

Abstract

Objective. This paper describes a low power closed-loop compressive sensing (CS) based neural recording system. This system provides an efficient method to reduce data transmission bandwidth for implantable neural recording devices. By doing so, this technique reduces a majority of system power consumption which is dissipated at data readout interface. The design of the system is scalable and is a viable option for large scale integration of electrodes or recording sites onto a single device. *Approach*. The entire system consists of an application-specific integrated circuit (ASIC) with 4 recording readout channels with CS circuits, a real time off-chip CS recovery block and a recovery quality evaluation block that provides a closed feedback to adaptively adjust compression rate. Since CS performance is strongly signal dependent, the ASIC has been tested *in vivo* and with standard public neural databases. *Main results*. Implemented using efficient digital circuit, this system is able to achieve >10 times data compression on the entire neural spike band (500–6KHz) while consuming only 0.83uW (0.53 V voltage supply) additional digital power per electrode. When only the spikes are desired, the system is able to further compress the detected spikes by around 16 times. Unlike other similar systems, the characteristic spikes and inter-spike data can both be recovered which guarantes a >95% spike classification success rate. The compression circuit occupied 0.11mm^2/electrode in a 180nm CMOS process. The complete signal processing circuit consumes <16uW/electrode. *Significance*. Power and area efficiency demonstrated by the system make it an ideal candidate for integration into large recording arrays containing thousands of electrode. Closed-loop recording and reconstruction performance evaluation further improves the robustness of the compression method, thus making the system more practical for long term recording.

Keywords: compressed sensing, neural recording, compression, silicon probe, integrated circuit

1. Introduction

1.1. The need for efficient compression in neural recording systems

Neural recording microsystems are essential tools for neuroscientists to study the activity of the brain. These devices, consisting of one or more recording sites or electrodes, can be deployed within the cortex to collect neural action potentials (a.k.a 'spikes') generated by individual neurons. Studying these neural signals allows neuroscientists to analyze the function and connectivity of brain circuits and their role in cognition and behavior (Mitra *et al* 2013), (Lopez *et al* 2013). Clinically, the neural recordings collected by the device can

also be utilized to diagnose neuropsychological illnesses such as epilepsy, depression and traumatic brain injuries (Staba *et al* 2002), (Aziz *et al* 2009).

The development of neural recording microsystems has continued to evolve in the past few decades. The number of electrodes integrated into one device has increased from one (Hubel and Wiesel 1959) to arrays that contain up to hundreds of electrodes (Shahrokhi *et al* 2010). However, given a cortical density of 100 000 neurons per mm^3 volume, the neural recording devices must be able to integrate an even higher number of recording sites into a small volume to fully access the brain circuits (Braitenberg and Schüz 1991). To prompt the next generation of neural recording devices, the latest NeuroSeeker project, funded by the European Commission, aims to develop a neural probe with more than 10 000 electrodes (NeuroSeeker 2013).

A major challenge that impedes massive electrode integration is the amount of data acquired by large numbers of electrodes in a device. Assuming each electrode is sampled at a Nyquist rate of 20 kHz with at least 10 bits of resolution, the data collected by the system will exceed 200 Mbps as the number of integrated electrodes increases above 1000. This enormous amount of data poses significant challenges for the design of digital data readout interfaces. This is mostly due to the available power budget that can be dissipated close to the brain that does not result in a temperature rise that exceeds the safe limit of 1 °C (Kim *et al* 2006). The challenge is even greater when neural probes with wireless transmissions are considered. The limited weight of head-mounted devices that can be carried by small laboratory animals, such as mice, rats, etc (~10% of body weight), puts a severe restriction on the battery life of these devices.

The design challenges can be summarized below:

- *Readout Interface*: Most of the power in the recording circuits is needed to drive the output pads. Even with a wired connection, there are no standard cables that can carry this large amount of data and yet be lightweight and flexible for the animal to move in an unrestricted way.
- *Wireless transmission*: The same high data rate challenge occurs whenever wireless transmission of the data is desired. Furthermore, the current state-of-the-art wireless neural recording chips are limited in capacity and consequently rarely allow more than a few Mbps to be transmitted (Abdelhalim *et al* 2013).

Both of the aforementioned issues are difficult to resolve without applying some kind of signal compression techniques prior to data readout or wireless transmission.

1.2. Prior works on neural signal compression

Many prior multi-electrode array designs rely on spike detection and windowing techniques to reduce transmission bandwidth (Mitra *et al* 2013), (Gosselin and Sawan 2009a), (Chae *et al* 2008), (Gosselin *et al* 2009b). After a spike is identified through a threshold crossing detector, a small 1–2 ms long window around each spike is retained. This event-based compression method achieves a decent CR for electrodes with sparse neuronal firing rates. When the aggregate firing rate of all detectable neurons is high (e.g. >150 Hz), however, the CR is greatly reduced (Chen 2013).

To further reduce the transmission bandwidth, wavelet transform-based techniques have been proposed to provide compression for detected spikes (Oweiss *et al* 2007), (Kamboh *et al* 2008). While a high CR can be achieved, the wavelet transform method requires a significant amount of additional hardware. Its implementation consists of digital filters with additional memories that operate at a speed several times faster than the Nyquist rate of neural signals (~20 kHz). The complexity of the processing unit increases circuit area and on-chip power consumption and hence hinders the utilization of this technique to arrays of recording electrodes or silicon probes with a large number of recording sites.

CS is a technique that gained popularity for compression of bio-signals due to its simple and power-efficient mathematical operations using only addition and subtraction (Chen 2013), (Mamaghanian *et al* 2011), (Dixon *et al* 2012), (Charbiwala 2013), (Gangopadhyay *et al* 2014). Different from wavelet transform-based techniques, data compression can be implemented using a few digital accumulators. Despite this advantage, previously implemented CS approaches only achieve limited CR before signal recovery quality degrades below an acceptable level for data analysis (Baluch *et al* 2012). The recovery quality of a CS-based system heavily depends on the choice of sparsifying transforms (a.k.a. a dictionary) through which the signal can be compactly represented. The limitation of previous systems is largely due to a less optimal choice of the dictionary.

Additionally, all previous compression systems are unable to measure the recovery quality of the signal since they have no knowledge of the original signal. Hence, the user has no understanding of how well the recovered signal resembles the real neural signal. Without such an evaluation, these compression systems operate in open-loop and provide no feedback to adjust the CR to balance the tradeoff between compression and recovery quality.

1.3. A closed-loop compressive-sensing-based compression system

In previous works, we have demonstrated that by leveraging the unique shape of each neuron's spike, a signal-dependent dictionary can be constructed and utilized to increase the CR while maintaining high recovery quality in the CS framework (Zhang *et al* 2013), (Suo *et al* 2013). It is often the case that spike trains recorded on a single electrode contain spikes from several nearby neurons. Each neuron's spike has a characteristic shape and amplitude, depending on its morphology and proximity to the recording electrode. Given that spike waveforms are generally stable over time, they can be used to learn a signal-dependent dictionary to sparsely represent similar spikes recorded at the same electrode. We have also demonstrated that this method allows CS to achieve a comparable CR and recovery quality as the wavelet transform-based method while using extremely efficient circuitry.

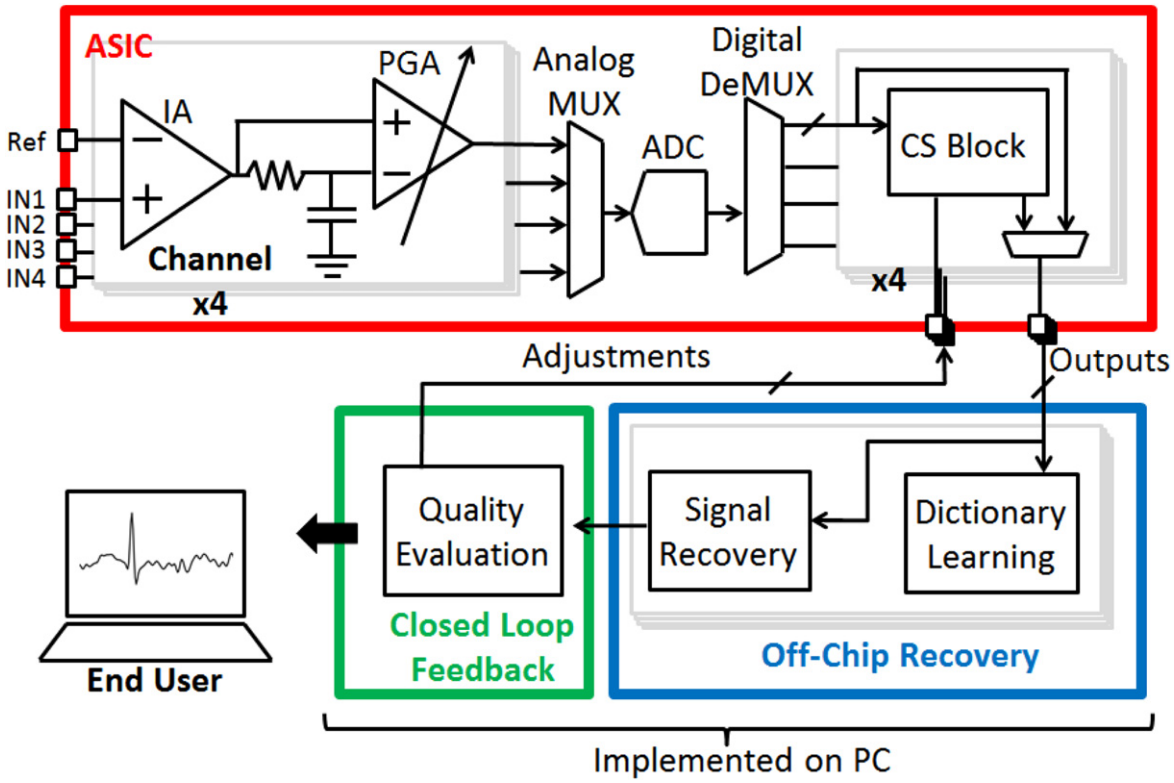

Figure 1. Overview of the neural recording ASIC, the off-chip recovery system and the closed-loop feedback linking two sub-systems.

However, a signal-dependent dictionary in the CS framework needs to be adaptable to accommodate changes in the neural signals that may occur during the recording. Without adaptation, the recovery quality would degrade over time because the learned dictionary can no longer represent spikes sparsely. To address this issue, we introduce a closed-loop CS neural recording system in this paper. The system includes application specific integrated circuits (ASIC) with four recording electrodes and compression circuits, an off-chip recovery algorithm that recovers the signal in real-time and, most importantly, a recovery QE method that provides adaptive closed-loop feedback to the ASIC for optimal tradeoff between CR and recovery quality.

In the main sections of the paper, we first introduce a relevant background of CS and dictionary learning (DL). We then describe the design of each component of the system and finally present a validation of the system using simulations and experimental data.

2. Background

We first introduce the basics of CS and the framework of DL.

2.1. Compressive sensing

CS originated as a theoretical framework regarding encoding and recovery of an S-sparse signal, x, of length N (Candes *et al* 2006), (Donoho 2006). A signal is $S-$sparse if it can be well approximated by its largest S coefficients in a certain transform domain (or a dictionary) where $S \ll N$. The S-sparse signal, x, can be encoded by a small measurement vector, y, of length M, such that

$$y = Ax, \tag{1}$$

where $S < M \ll N$, and A is a sensing matrix of dimension $M \times N$. The CR achieved in this case is N/M. However, recovering x, given y and A is not trivial because this system of linear equations contains more unknown variables than equations. Fortunately, considering that matrix A satisfies the restricted isometry property (RIP) and that x is S-sparse, this underdetermined problem can be solved, and x can be recovered exactly with extremely high probability from y using optimization methods (Candes *et al* 2006).

RIP is the key factor to determine the optimal choices of sensing matrices. RIP describes how well the distance of the S-sparse signal can be preserved after the projection using sensing matrix A. Many matrices, such as the random Gaussian, random Bernoulli and Partial Fourier matrices all satisfy the RIP universally with a small number of M. The choices of the sensing matrix can be determined based on specific applications and desired performance tradeoffs.

2.2. Dictionary learning

The number of samples, M, required to successfully recover x is proportional to the sparsity, S, of the signal represented using a dictionary. Therefore, a desired dictionary should be able to represent x using as few coefficients as possible to improve CR. Various DL methods can be used for this purpose (Lewicki and Sejnowski 2000), (Aharon *et al* 2006), (Engan *et al* 2000). Given L training signals $X = \left\{x_{ll=1}^{L}\right\}$, the

DL algorithms find a dictionary $\boldsymbol{D}$ that can represent the training signals using $S-$sparse signal $V = \left\{ v_{ll=1}^{L} \right\}$. In other words, it solves the optimization problem

$$\text{argmin} \sum_{l=1}^{L} x_l - \boldsymbol{D} v_{l2}^2 \text{ such that } v_{l0} \leqslant s,\ 1 \leqslant l \leqslant L, \qquad (2)$$

where S is the bound on the l_0 $-$norm of S-sparse signal v_l. It sets the bound on the number of non-zero coefficients for every v_l.

3. Methods

Figure 1 presents the blocks of the entire system. We also describe the system design in detail in this section. First, we describe the ASIC, which consists of analog preprocessing blocks, analog to digital converters (ADCs) and the CS compression block. Next, we present our off-chip DL and CS recovery algorithms. Finally, we conclude the section describing our adaptive mechanism of the closed-loop feedback between on-chip and off-chip blocks.

3.1. ASIC

As shown in figure 1, the ASIC contains four neural recording channels with corresponding CS circuits. The signal is first conditioned by the analog front end and sampled by a shared 10 bit successive approximation register (SAR) ADC. The ASIC can be configured to operate in either DL mode or compression mode (CM). During the DL mode, the CS circuit is bypassed, and the raw waveforms are transmitted to allow the off-chip DL algorithm to construct a dictionary. Then, the chip is switched back to CM in which the raw data is condensed by the CS block. For large arrays of electrodes, DL can be performed per group of electrodes to avoid large data transmission during a small period of time.

3.1.1. Analog front end and ADC. The analog front end (AFE) consists of two gain stages. In the first stage, a capacitive coupled instrumentation amplifier (IA) is used. The output of the IA passes through a band-pass filter to extract neural spiking signals (500 Hz – 6 kHz). A programmable gain amplifier (PGA) is used at the second stage to provide additional gain before the signal is sampled by the ADC. The AFE has an integrated noise of 3.1 uVrms (500 – 6 kHz band) and a common mode rejection ratio (CMRR) of 75 dB while providing a configurable gain of 230 – 6 K. The SAR ADC, operating at 80 KHz, is used to sample the conditioned analog signals from all four recording electrodes. Operating at a VDD of 1.8 V, the AFE and ADC together consume 15 μW per electrode.

3.1.2. Compresive sensing block. In CM mode, The CS block can be further configured into two sub-modes of operation: either the entire band-passed neural signal or only the spikes are compressed. When configured to compress the entire neural signal, the CS block preserves the fidelity of the spikes as well as the inter-spike signals. If only spikes are desired, the compressed output is only produced when a spike is detected by a threshold crossing detector applied to the absolute magnitude of the signal. Sixty-four samples of the detected spikes are kept for compression.

The CS block implements the linear operation of equation (1): $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$. This equation can also be written as a system of linear equations

$$\begin{aligned} y_1 &= A_{1,1}x_1 + A_{1,2}x_2 + \ldots + A_{1,N}x_N \\ y_2 &= A_{2,1}x_1 + A_{2,2}x_2 + \ldots + A_{2,N}x_N \\ &\vdots\ \vdots\ \vdots \\ y_M &= A_{M,1}x_1 + A_{M,2}x_2 + \ldots + A_{M,N}x_N, \end{aligned} \qquad (3)$$

where $x_1 \ldots x_N$ are the digitized neural signal from the ADC at discrete time 1 to N, $y_1 \ldots y_M$ are entries of compressed sample y of length M ($M \leqslant N$) and $\boldsymbol{A}$ is the sensing matrix of size $M \times N$. In our design, matrix $\boldsymbol{A}$ is a random Bernoulli matrix, which can be configured by the user. Among matrices that satisfy the RIP requirement, random Bernoulli matrices are the most optimal for hardware implementation as their entries are either +1 or −1. Therefore, the system of equation (3) can be implemented using M digital accumulators. Depending on the corresponding value of $\boldsymbol{A}$, the accumulators either add or subtract digitized signal x_i from the value of the accumulator to generate y_i. Other matrices, such the random Gaussian and optimized sensing matrices, all contain fractional entries (Elad 2007), (Duarte-Carvajalino *et al* 2009). Thus, implementing equation (3) with these choices requires the use of digital or analog multipliers in addition to accumulators. These additional components consume a large amount of chip area. For example, an implementation of Gaussian matrices using M-DAC occupies around 0.6 mm^2 (Gangopadhyay *et al* 2014), whereas our digital implementation of a Bernoulli sensing operation only occupies 0.11 mm^2.

The CS block uses arrays of accumulator shift-registers (ASRs) to implement the matrix multiplication of equation (1), shown in figure 2. The accumulations are clocked at a signal Nyquist rate of 20 kHz (C20 K). The matrix block, shared across all the channels, contains registers to hold one row of a random Bernoulli matrix. Their values are updated off-chip at every Nyquist period. Depending on the value of a particular matrix entry (either 1 or 0), the corresponding ASR either adds or subtracts the current digitized signal from the accumulated value. Each ASR can be disabled by applying clock gating to control the CR (CR = N/M). To avoid implementation of extra registers to buffer the data for transmission, a 4 MHz (C4M) clock is used to shift the data in the ASRs to the output pin near the end of each accumulation cycle. Vector $\boldsymbol{y}$ is generated every N clock cycles. N can be configured to be either 128 or 64, depending on the operation mode, corresponding to a signal length of 6.4 ms and 3.2 ms.

To conserve power, when the ASIC is configured to compress only the spikes, clock gating is applied to the ASRs and the matrix block so that they remain inactive until the spike detection block registers a threshold crossing event. A digital FIFO is used to buffer a variable length of pre-trigger samples (up to 15 samples) before a threshold crossing event.

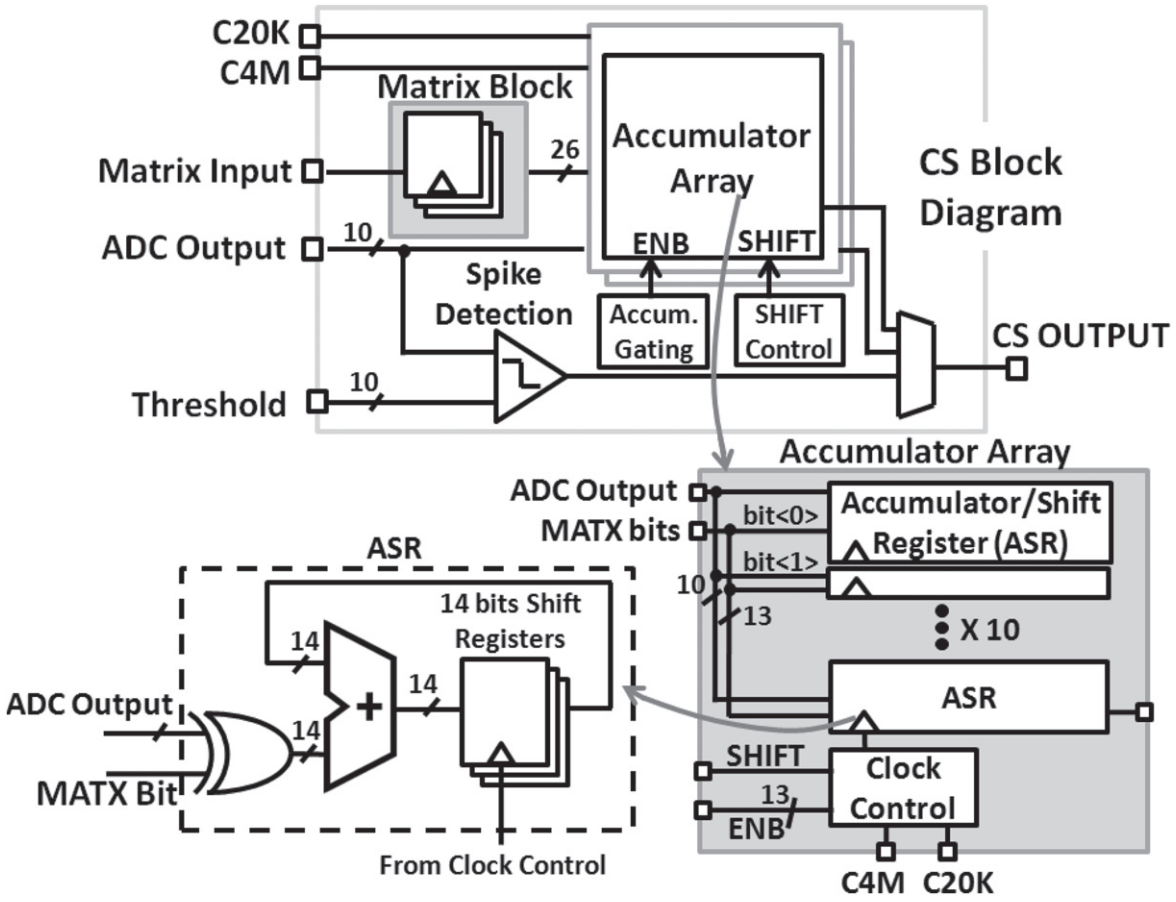

Figure 2. Architecture of the CS Block.

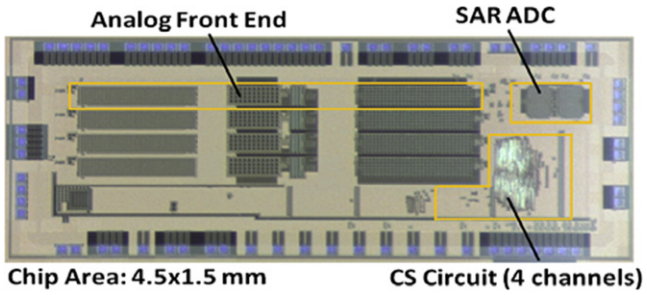

Figure 3. Micrograph of the ASIC.

After the compressed vector y is sent off-chip, the CS block becomes inactive again until the next threshold crossing event. The spike detection information is also used when the ASIC is configured to compress the entire band-passed neural signal. It informs the off-chip recovery block how many spikes have occurred and their peak locations within the signal segment (Zhang *et al* 2013). A Micrograph of the ASIC is shown in figure 3.

3.2. Off-chip dictionary learning and signal recovery

The off-chip recovery algorithms consist of two blocks: the DL block and the signal recovery block. Both of these blocks are implemented using MATLAB.

3.2.1. Dictionary learning. Operating in the DL mode, the ASIC bypasses the CS block and outputs the uncompressed neural signal. The raw signals form a training signal set that is used to learn a dictionary. The most straightforward method to construct a dictionary is to use detected neural signals to form bases in this dictionary (Suo *et al* 2013). Alternatively, DL methods, such as K-SVD (Aharon *et al* 2006), can be utilized to train a dictionary given a set of training spike waveforms (Zhang *et al* 2013). As described in section 2, like the other dictionary learning method, the K-SVD algorithm finds a dictionary, D, through iterations to minimize errors between the training data and its corresponding sparse representations using D. Analysis has shown that dictionaries created with raw spikes result in a slightly better reconstruction performance when tested using a synthetic neural database (with various amounts of additive noise) (Suo *et al* 2013), (Quiroga 2004). On the other hand, the K-SVD-trained dictionary does well when evaluated using an *in vivo* recording database (Henze *et al* 2000), (Suo *et al* 2013), (Suo *et al* 2014).

Here, we implement the K-SVD algorithm to learn the dictionary, due to its fast computation speed and superior performance over dictionaries created with raw spikes. For example, implemented using MATLAB on a PC with Intel Core i7 and 16 Gbytes of RAM, the K-SVD algorithm takes approximately 0.01 s to compute a dictionary of size 64 by

100, using around 300 observations of different spikes. The size of the dictionary and training data size could vary, depending on the user's preferences.

3.2.2. Compressive sensing recovery. After a dictionary is trained, the ASIC switches back to the CM mode and outputs the compressed vector y. From the compressed measurement, y, the signal can be reconstructed by solving a L1-minimization problem

$$\text{argmin}_x \| x \|_1 \quad \text{such that } y = Ax. \tag{5}$$

We solve (5) using matching pursuit methods due to their efficient computation time. A detailed discussion on the recovery method and signal model is provided here (Zhang *et al* 2013). The average computational time for recovery is around 1.3 ms if only spikes are reconstructed and 2 ms if the entire neural signals are reconstructed. This suggests that the system can recover around 700 spikes per second for real-time applications. For a large array of electrodes, multiple systems could be used to handle the recovery, or FPGA implementation of the recovery algorithm could be developed to speed up the recovery.

We measure the recovery performance of a spike train using the signal to noise and distortion ratio (*SNDR*). Here, we add a subscript x to derive a notation $SNDR_x$ to represent *SNDR* measured between the original signal and the recovered signal

$$SNDR_x = \frac{1}{T}\sum_{i=1}^{T} 20 \log \frac{\|x_i\|_2}{\|x_i - \hat{x}_i\|_2}, \tag{6}$$

where x_i is ith spike belonging to a spike train having T spikes, and $\hat{x}_i$ is the reconstructed spikes from compressed measurements. $SNDR_x$ is a purely theoretical estimate. It has been used by other authors to verify the validity of their compression and for recovery approaches using known signals (Chen 2013).

3.3. Closed-loop feedback

An obvious disadvantage of $SNDR_x$ is that it requires the knowledge of the original signal, which is not available when the ASIC is generating the compressed measurements. The failure to address this problem makes previously reported neural signal CS systems impractical for real recording applications. Without a quality QE block, the users have no means to quantify the performance and adjust the CR for optimal tradeoff between recovery quality and compression. Furthermore, the QE block is essential in our system where a learned dictionary is used. A QE block can detect the case when the existing dictionary can no longer represent the neural spike trains and then switch the ASIC back to DL mode where a new dictionary will be learned. No data is lost while the system is in the DL mode, since raw data is transmitted in this mode. As we shall demonstrate in an *in vivo* experiment, the system only needs to switch to DL mode once for around 2 min during a two hour recording session.

As shown in figure 1, a QE block examines the quality of the recovered signal and provides feedback to the ASIC to adjust the CR or to switch between the DL mode and CM mode. A QE block is also implemented using MATLAB.

Due to the inability to calculate $SNDR_x$, the QE block calculates signal to noise and distortion ratio measured in compressed domain ($SNDR_y$) as the metric for recovery performance. $SNDR_y$ is defined as

$$SNDR_y = \frac{1}{T}\sum_{i=1}^{T} 20 \log_{10} \frac{\|y_i\|}{\|y_i - \hat{y}_i\|} \tag{7}$$
$$\hat{y}_i = A\hat{x}_i,$$

where y_i is the CS measurement of the ith spike within a spike train containing a total of T spikes, and $\hat{y}_i$ is the CS measurements estimated from the recovered spike $\hat{x}_i$. When a signal is not well reconstructed, the reconstruction error can also be reflected in the CS measurements after a linear mapping using the sensing matrix, A. In the Experiments section, we shall demonstrate the correlation between $SNDR_x$ and $SNDR_y$.

The QE block calculates the moving average and the standard deviation (s.t.d.) of the $SNDR_y$ across many time intervals. It can initiate feedback to the ASIC to increase the CR or to learn a new dictionary if the measured $SNDR_y$ decreases below a tolerable threshold set at a few s.t.d from the moving average.

4. Experiments and results

In this section, we describe the experiments we conducted to validate each part of the system. First, we used a dataset from a fifteen week long multi-electrode recording experiment to characterize the recovery performance. This data is essential to validate the efficacy of the system under different noise conditions and over an extended period of time. Here, we also show the correlation between performance metrics $SNDR_x$ and $SNDR_y$. Second, we used a dataset from a two hour tetrode experiment to evaluate the closed-loop feedback system and demonstrate the dynamic CR evaluation and dictionary updates. We further tested the ASIC's functionality by deploying it during a recording experiment conducted on an awake Rhesus Macaque. Finally, we also characterized the system performance using a standard database in order to compare our system with previous published works.

4.1. Off-chip recovery performance characterization

4.1.1. Experiment set-up. Data from a fifteen week long multi-electrode recording experiment is used to characterize the off-chip recovery performance. The close relationship between performance metrics $SNDR_x$ and $SNDR_y$ is also validated using this experiment. Data from this experiment is ideal for recovery performance evaluation since they contain recordings collected at many different electrodes with different signal to noise ratios (SNRs). Additionally, a few electrodes also detect multi-neuron activities. Both signal

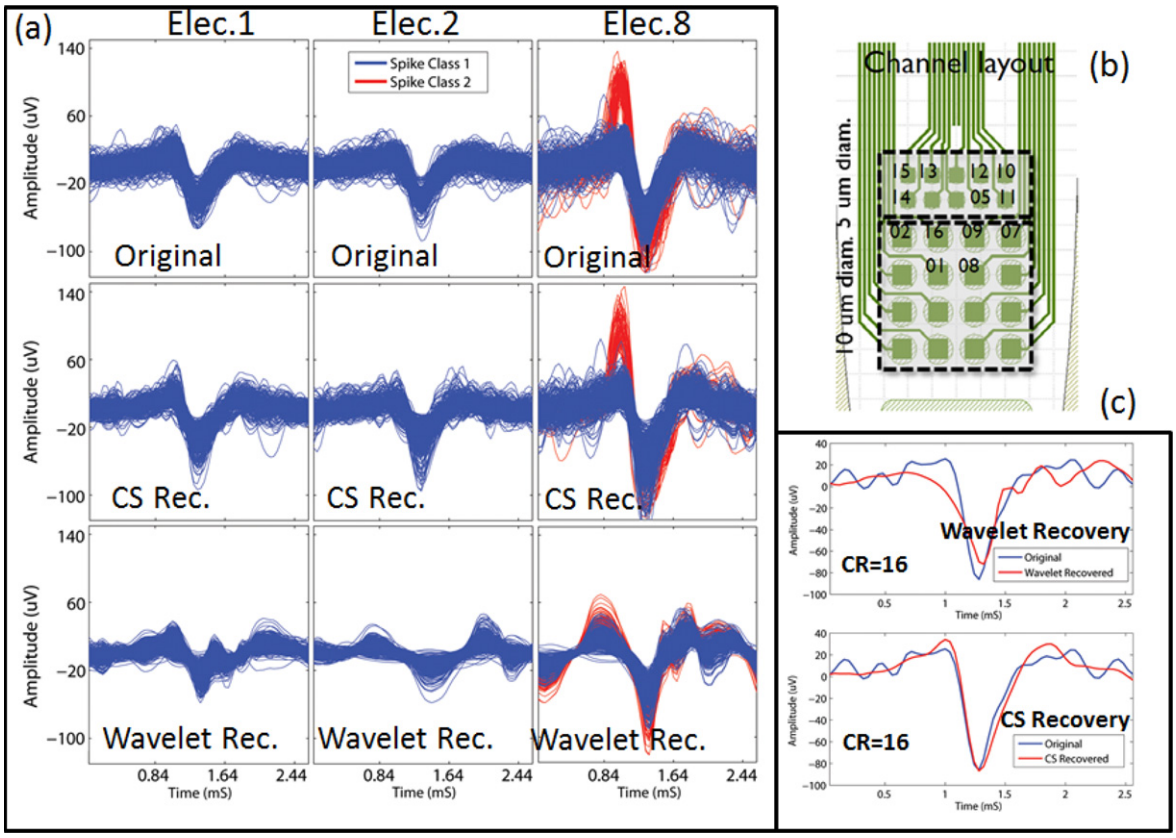

Figure 4. (a) Example of the original recorded neural waveforms and recovered waveforms using CS and the wavelet transform method (week 0). (b) Layout of the electrode on the silicon probe. (c) Zoomed-in views of CS and the wavelet recovery.

compression and recovery are carried out using an offline algorithm implemented using a PC. Since the compression block on the ASIC is implemented using digital circuits, it does not introduce additional noise to the recording. Therefore, its performance can be exactly modeled by an offline algorithm. The performance of the ASIC is characterized through an *in vivo* experiment discussed later in the paper.

In this fifteen week long recording experiment, a high-density recording array containing 32 electrodes on a single shank (70 μm wide) is implanted in the thalamus of a rat's brain when it is under anesthesia. The scientific aim of the experiment is to examine the long-term recording SNR of an implanted silicon probe. We acquired one minute of raw data from the implanted electrodes every week for fifteen weeks. The recorded signals are digitized at 25 KHz and filtered at 500 – 5 KHz. During week 0, spikes are observed at 13 out of 32 electrodes. The relative position of the electrodes are shown in figure 4(b). Electrodes 1, 2, 7, 8, 9 and 16 have a contact diameter of 10 μm, while electrodes 5 and 10 to 15 have a contact diameter of 5 μm. During 15 weeks of recording, the electrode average SNR is found to be stable and does not suffer major loss over time.

For each recording electrode, we used 20% of the extracted spikes (around 50–120 spikes) to train a representation dictionary with the K-SVD method, while using the remaining 80% as test signals to evaluate the CS recovery performance. The raw recording first goes through an offline spike detection block, which extracts the spikes after their amplitude exceeds a pre-set threshold at around four s.t.d.s above and below the average signal amplitude of the dictionary training data. For each detected spike, 64 discrete samples are retained around a spike corresponding to 2.6 ms temporal duration. In week 0 s recording, electrode 8 records the spikes from two neurons, while the rest of the electrode only has one distinguishable spike cluster.

We compress each spike by multiplying it with a random Bernoulli sensing matrix of size $M \times 64$, where M is the number of compressed samples. For comparison, we also present recovery results using wavelet transform-based recovery. In this method, the extracted spike first undergoes a wavelet transform. Wavelet components at the M biggest locations determined using the training data are retained and used to reconstruct the spike. The wavelet used is the Daubachies-8 wavelet, which is a standard wavelet choice for compression (Bulach *et al* 2012). Figure 4 illustrates the electrode layout on the silicon probe, the original signal and the signals recovered using CS and the wavelet at three of the 13 electrodes at week 0.

As performance metrics, we first compute the $SNDR_x$ of the spike train at every electrode. In addition, for each recovered spike, we also compute the difference of its amplitude at its main trough compared to the original spike. To account for the variation of the CS recovery over the choice of sensing matrix, we compress and recover each spike train using 20 different randomly generated Bernoulli

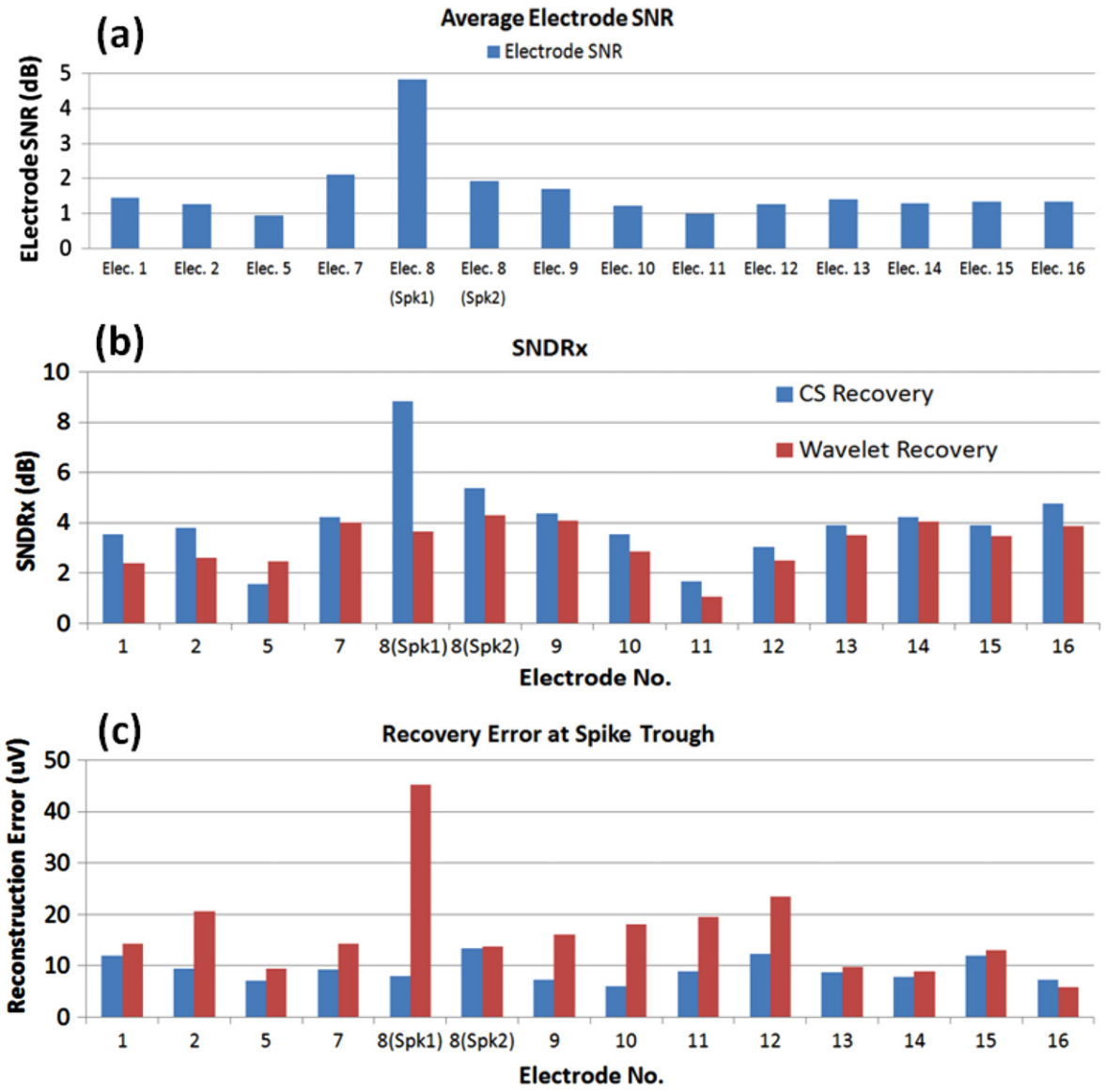

Figure 5. (a) SNR for every type of spike seen at electrodes at week 0. (b) $SNDR_x$ of the recovered spikes. (c) Recovery error at the main trough of the spikes.

matrices. We then average $SNDR_x$ over the entire 20 trials to acquire a single measurement of $SNDR_x$ for the spike trains collected at that particular electrode.

To analyze the spike recovery quality under different noise levels, we calculate the signal to noise ratio (SNR) for each type of spike. The SNR is computed as

$$SNR = \frac{\text{Signal Amplitude}}{\text{Peak to Peak Amplitude of Noise Floor}}. \quad (8)$$

Peak to peak amplitude of the noise floor is taken as six times the s.t.d. of the recorded signal after spikes are removed, spanning ~99.7% of the normally distributed noise data (Ludwig *et al* 2009). Spike clusters with a SNR of 1.1 or greater are considered to be discriminable units (Ludwig *et al* 2009).

For electrode No. 8 in which distinguishable multi-unit activities are seen, we examined the spike clusters' distance in the principal component subspace after they are reconstructed by CS and the wavelet method. The larger the cluster distance, the easier it is to cluster the spikes. From the original spikes, we observed that there are two types of spikes at electrode No. 8: class C1 and C2. They contain 608 and 86 spikes, respectively. In this analysis, we first use the same training spikes to learn a dictionary for CS reconstruction. The spikes are then compressed and reconstructed using CS and the wavelet method for various CRs. We perform principal component analysis (PCA) on the reconstructed spikes. Finally, we calculate the cluster mean of reconstructed C1 and C2 in the subspace spanned by their first two principal components. Figure 6(b) shows the original spikes cluster C1, C2 and their means.

4.1.2. Results on CS recovery quality versus wavelet recovery quality. Figure 5(a) shows the SNR for spikes at each electrode. Figures 5(b) and (c) illustrate the recovery quality

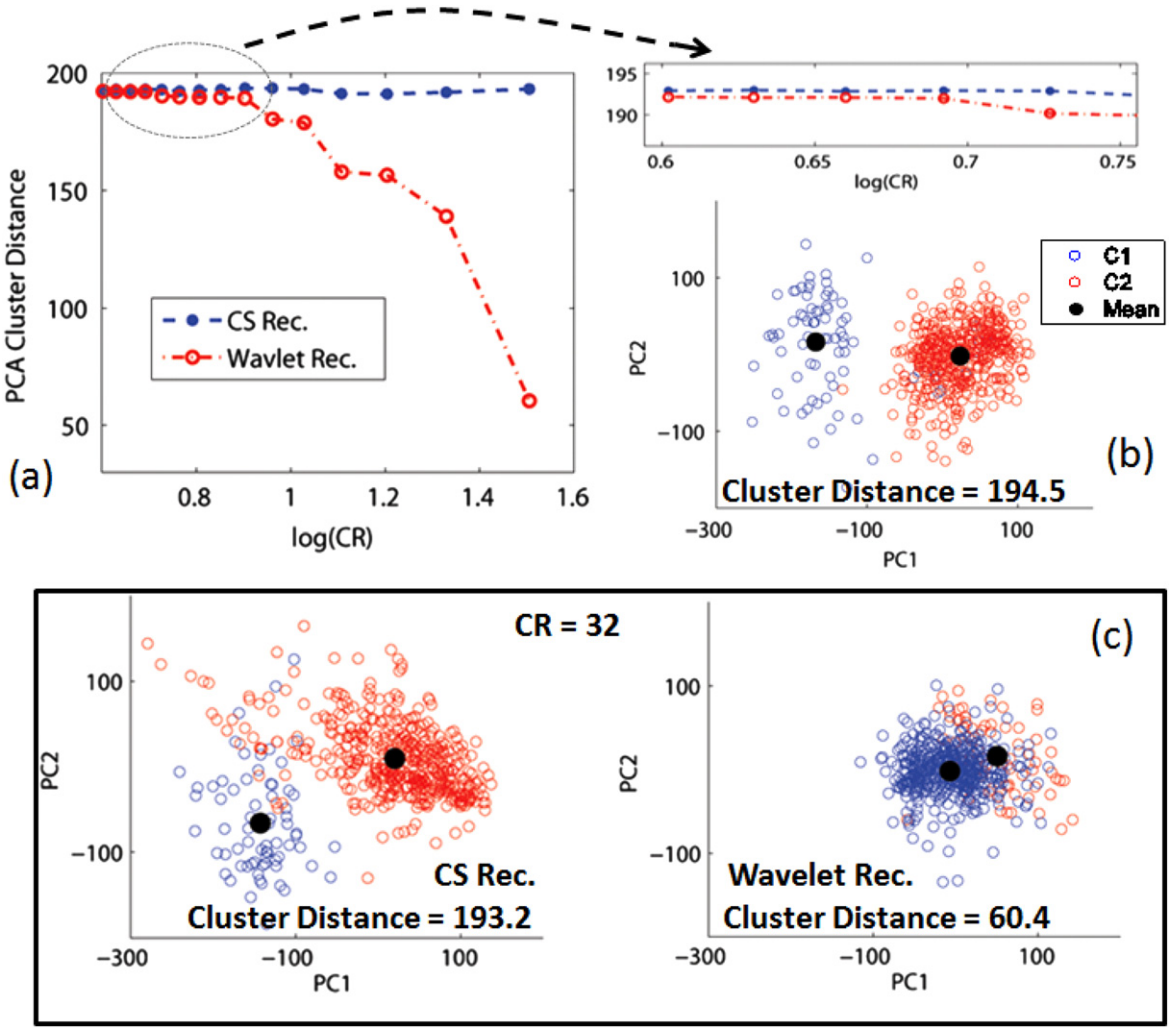

Figure 6. (a) PCA cluster distance with respect to the CR. (b) Original spikes' clusters: C1 and C2. (c) C1 cluster and C2 cluster when the spikes are reconstructed using CS and the wavelet method at CR = 32.

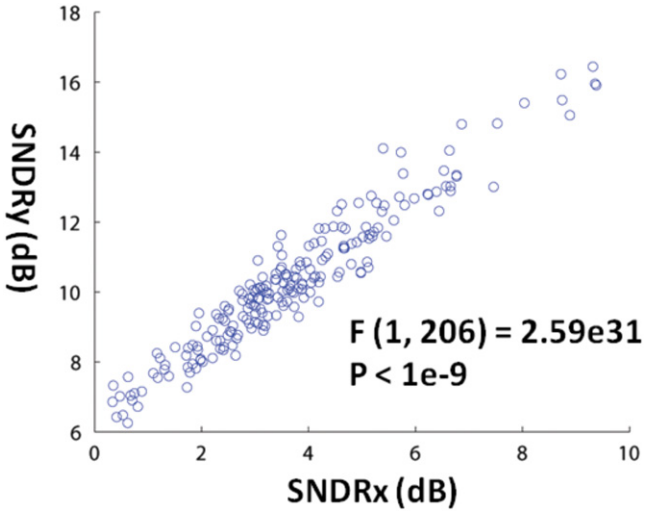

Figure 7. $SNDR_x$ and $SNDR_y$ values for all the recording electrodes over 15 weeks of recordings.

of CS and wavelet method for week 0 s data. In this case, the length of compressed samples (M) is set to be 4, corresponding to a CR of 16. In terms of both $SNDR_x$ and the recovery error at the main trough of the spike, CS performs better or comparable to the wavelet recovery method across all electrodes with different SNRs.

Figure 6(a) presents the PCA cluster distance for C1 and C2 spike clusters when spikes are reconstructed across different CRs by CS and the wavelet method. The CS reconstructed clusters maintain very high separation even at a CR of 32, when only 2 CS samples are retained to reconstruct the spike. This is because the CS method only uses the sparse dictionary atom from either C1 or C2 to reconstruct a spike. As long as it can choose the correct atom during reconstruction, high cluster distance is guaranteed. On the other hand, wavelet reconstructed cluster distance starts to decrease when the CR increases above 9. This is because as more wavelet coefficients are removed, the clusters lose their discriminative features. Figure 1(b) show the scatter cluster for the original

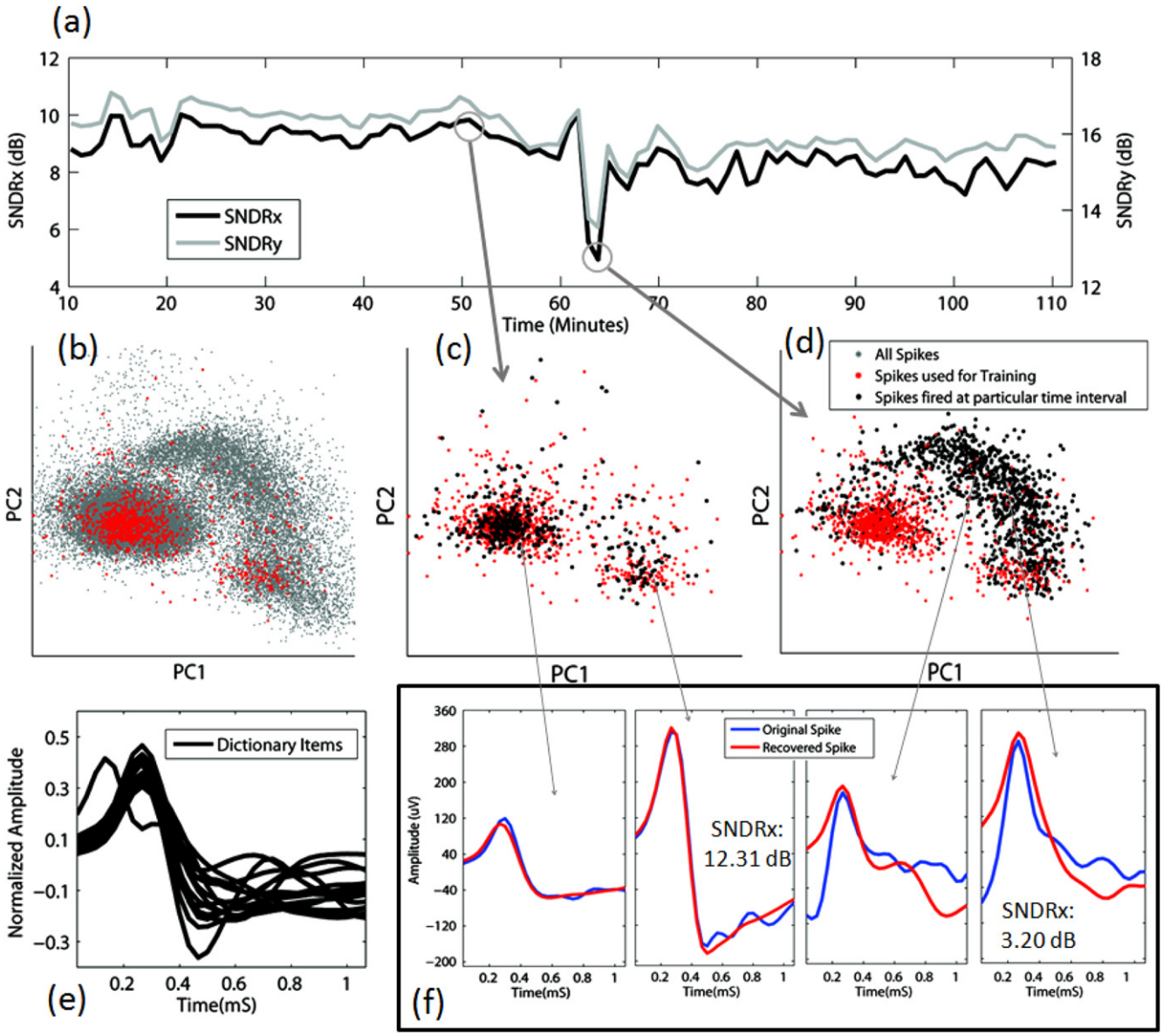

Figure 8. (a) $SNDR_x$ and $SNDR_y$ of the recovery over a 2 h experiment. (b) Gray dots: PCA results of all the spikes collected in the 2 h experiment. Red dots: spikes used for DL. (c) Black dots: spikes recorded at around a 50–51 min interval. They are well recovered since their shapes do not vary too much from the dictionary. (d) Black dots: spikes recorded at around a 62–63 min interval. A new type of spike starts to appear. As its shape varies significantly from the dictionary, the recovery results deteriorate, shown by a decrease of $SNDR_x$ and $SNDR_y$. (e) The trained dictionary. (f) Temporal view of original spikes and recovered spikes at different time intervals.

spikes, while figure 1(c) shows the cluster of CS and wavelet reconstructed spikes at CR = 32.

The recovery results from this dataset suggest that the CS performance is comparable to that of the wavelet transform-based method. For clustering, the advantage of CS is more apparent at a high CR (>9). The result from the clustering experiment is consistent with a similar spike classification experiment described in our previous publication (Zhang *et al* 2013). In terms of hardware power and area efficiency, a 6 level wavelet transform would require around 32 708 transistors (Oweiss *et al* 2007). On the other hand, to achieve CR = 16, the CS method only needs 4 digital accumulators of 13 bits, assuming each digitized data has 10 bits resolution. This corresponds to only 2496 transistors if the D-Flip Flop has 20 transistors and the Full-Adder has 28 transistors (Zhang *et al* 2013). Hence, the power consumption and the area required to implement the CS system is much more efficient than the wavelet method to achieve comparable performance. A quantitative comparison on hardware efficiency is elaborated in our previous work (Zhang *et al* 2013).

4.1.3. Correlation between $SNDR_x$ with $SNDR_y$. To verify that $SNDR_y$ can effectively be used as a metric to measure recovery quality, we must show that these two metrics are highly correlated. We calculate the $SNDR_x$ and $SNDR_y$ for the spike trains recording at every electrode from week 0 to week 15, shown in figure 7. On average, a one minute recording

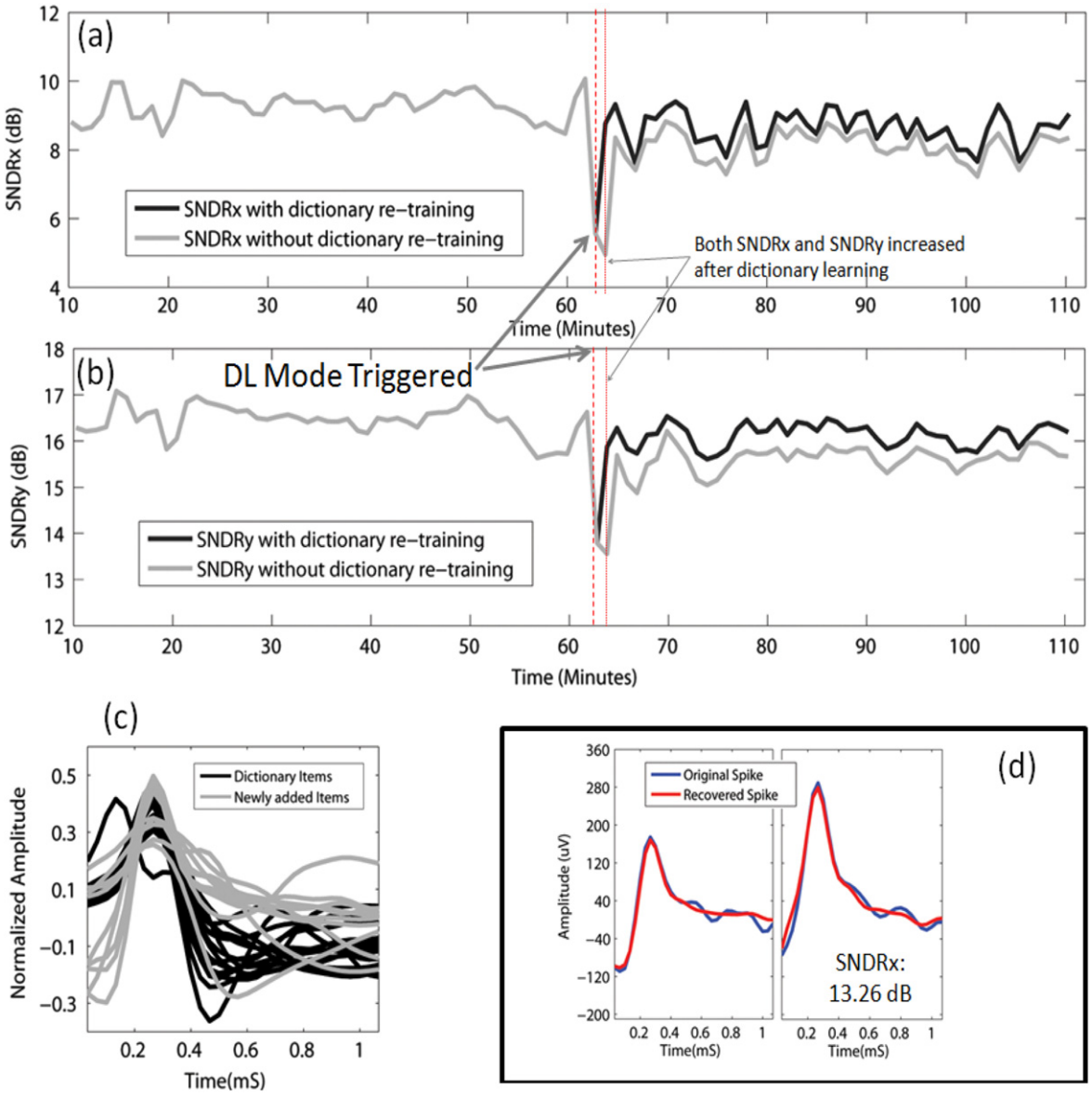

Figure 9. (a) $SNDR_x$ of the 2 h experiment with and without dictionary retaining. (b) $SNDR_y$ of the 2 h experiment with and without dictionary retaining. (c) The dictionary after dictionary re-training. (d) Temporal view of original spikes and recovered spikes at a 62–63 min interval.

from each electrode, every week, contains around 500 spikes ($T = 500$). A regression analysis between $SNDR_x$ and $SNDR_y$ results in $F(1206) = 2.59 \times 10^{31}$ and $P < 10^{-9}$, suggesting a strong linear relationship between $SNDR_x$ and $SNDR_y$. Therefore, we could use $SNDR_y$ as an alternative metric to evaluate signal recovery quality.

4.2. Quality evaluation block and the closed-loop feedback characterization

4.2.1. Experiment set-up. We use the data from a two hour tetrode recording experiment, as well as a synthetically generated spike train, to characterize the performance of the closed-loop feedback block. The tetrode recording was acquired with digital Lynx (from Neuralynx). In this experiment, a micro-drive array carrying tetrodes was chronically implanted in a rat. The recording is from the CA1 of the hippocampus. For the first hour of the experiment, the rat is sleeping inside of a box. Then, it is placed onto a treadmill to perform running tasks. To detect spikes, a threshold is set at 100 uV, and 32 samples around the spikes are retained after a threshold crossing event. Similar to the previous experiment, the compression and recovery are all completed offline without using the ASIC, as the offline model is an exact replication of the ASIC functionality. In this continuous two hour experiment, we observed activities of different neurons at different time intervals. Hence, we can evaluate the performance of the QE block when new types of spikes are detected that were not included in the dictionary.

For each tetrode, the spikes collected during the first two minutes are used to train a dictionary. Then, we compress the spikes using a random Bernoulli matrix of size 4×32,

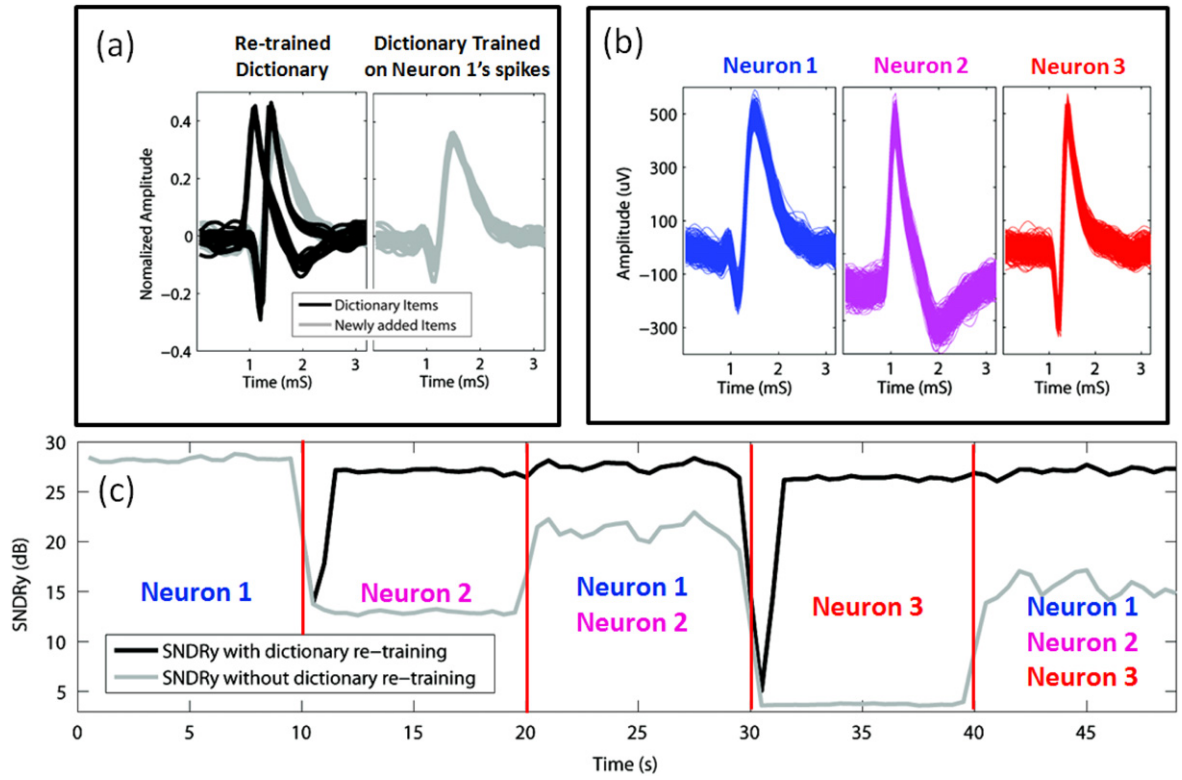

Figure 10. (a) Dictionary trained on Neuron 1's spikes and the re-trained dictionary after Neuron 3 fires. (b) Three types of spikes from different neurons. (d) $SNDR_y$ of the experiment with and without dictionary re-training.

corresponding to a CR of 8. Each spike is recovered using the same recovery method mentioned in the previous section. $SNDR_y$ is calculated by the QE block at a per minute interval. We also computed $SNDR_x$, which is a truth recovery quality metric. The moving average and s.t.d. of $SNDR_y$ is also calculated.

We computed two trials of compression and recovery: In the first trial, we compress and recover all the spikes collected on one of the tetrodes using the initially learned dictionary. In the second trial, QE blocks trigger the DL mode when $SNDR_y$ decreases by more than four s.t.d.s compared to its moving average. The recovery system then recovers the subsequent spikes in CM mode using the newly learned dictionary, together with the initially trained dictionary.

In addition to the tetrode data, we have also created a synthetic spike train to further demonstrate improvement in recovery quality after closed-loop feedback and dictionary retraining. These spikes, taken from the Leicester neural database (Quiroga 2004), originates from three different neurons. Their shapes are shown in figure 10(b). Five thousand spikes are drawn randomly from three spike clusters and placed 10 ms apart to form the synthetic spike train of 50 s in duration. In the first 10 s, only Neuron 1 fires, followed by the firing of Neuron 2 between 10 s to 20 s. Then, Neurons 1 and 2 both fire between 20 s to 30 s before Neuron 3 fires between 30 s to 40 s. Finally, all three neurons fire between 40 s to 50 s. The CR is set to 16, compressing 64 samples down to four samples. In the recovery experiment, we first recover the signal using the DL from only spike 1. We then repeat the experiment to allow dictionary re-training after observing a decrease of more than four s.t.d.s in the $SNDR_y$.

4.2.2. Closed-loop feedback experiment results. Figure 8(a) shows the $SNDR_x$ and $SNDR_y$ measured at every minute throughout the duration of the experiment. In figure 8(b), the grey dots represent the first and second principal components of all the spikes collected during the experiment after a PCA. The red dots are the spikes used for DL. They are collected in the first 2 min of the experiments. The training data covers only a portion of the PCA space where the spikes occurred. The learned dictionary using the training data is shown in figure 8(e). As we could expect, if a spike falls into the PCA space overlapped by the training data, it can be well recovered. On the other hand, if spikes fall outside of this region, most likely they cannot be recovered with great accuracy since their shapes are different than the dictionary.

In figure 8(a), the recovery quality measured by $SNDR_x$ and $SNDR_y$ stays constant above 8 dB and 15 dB for the time intervals before 60 min. This is because the spikes detected during this time can be well represented by the learned dictionary. Examples of these spikes' PCA plot are shown in figure 8(c), where the black dots represent the spikes detected between 50 and 51 min. Most of the black dots fall onto the PCA space covered by the training data and therefore have a similar shape to the dictionary. However, at around 61 to 62 min, when the rat is first placed on the treadmill, a lot more spikes are detected that do not fall into the PCA space covered by the training data (figure 8(d)). These spikes have different shapes compared to the training data and the learned dictionary, as shown in figure 8(f). Therefore, they cannot be recovered accurately using the learned dictionary. There exists a significant amount of mismatch between the original

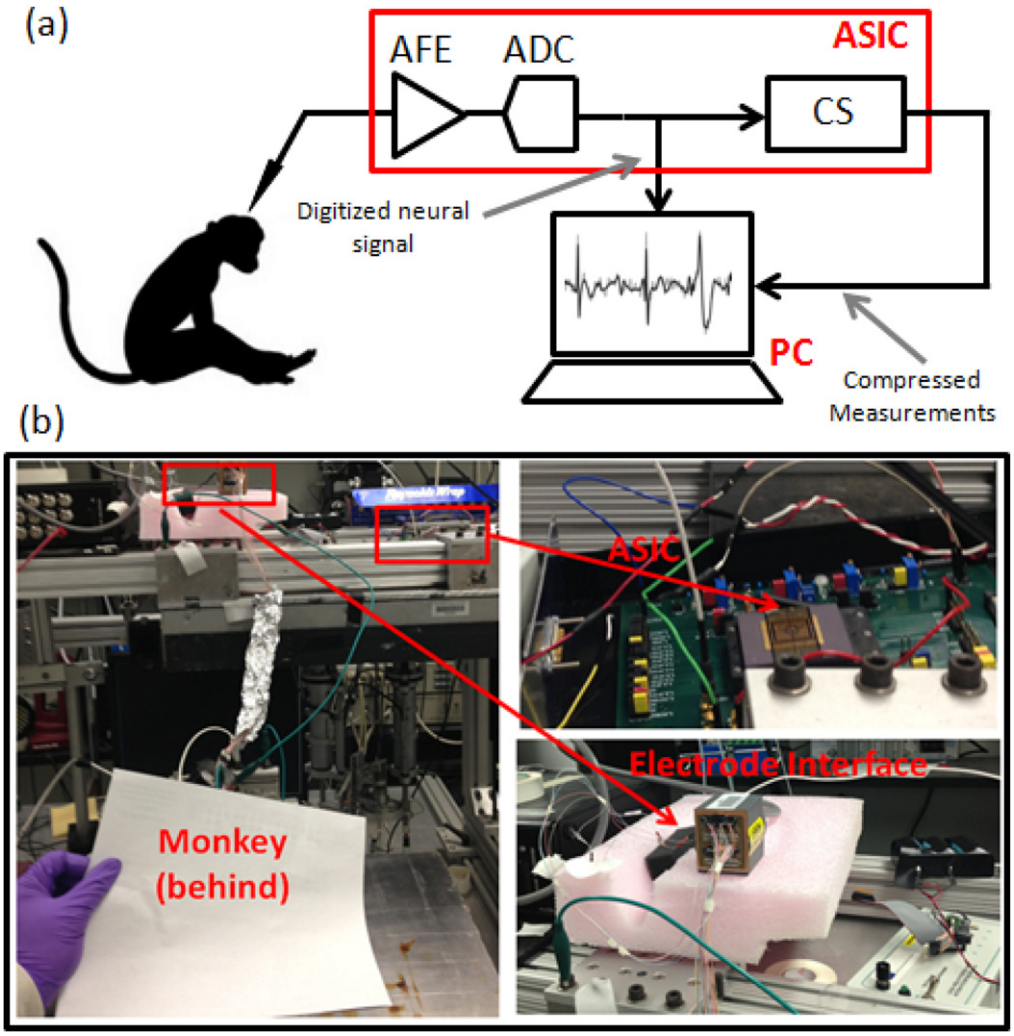

Figure 11. (a) *In vivo* recording experiment using the ASIC. (b) The actually *in vivo* experiment.

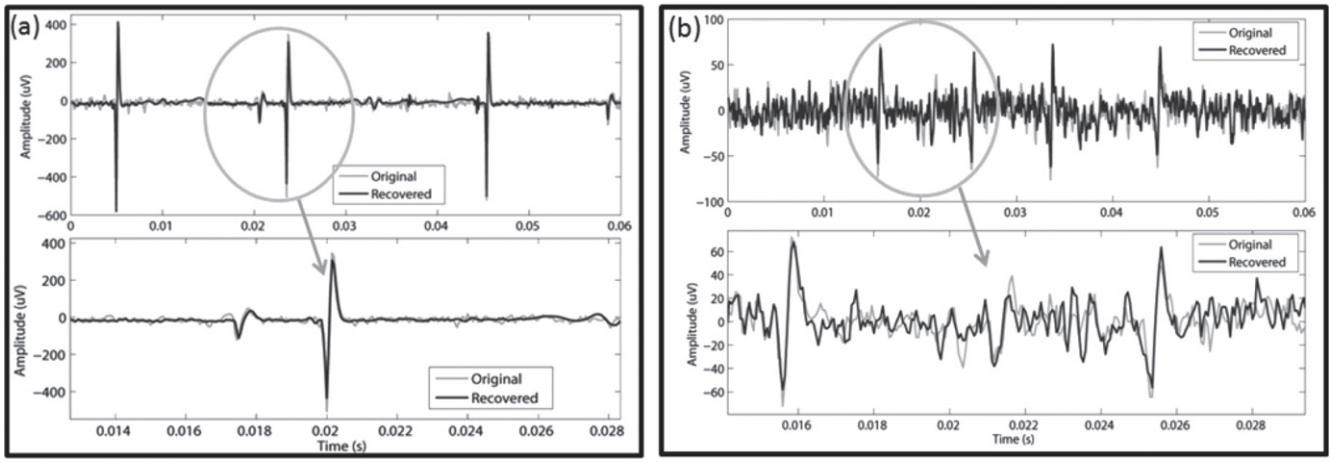

Figure 12. (a) *In vivo* test data acquired by the system in CES mode. The electrode's SNR = 6.21. (a) *In vivo* test data acquired by the system in CES mode. The electrode's SNR = 2.14.

signal and the recovered signal. Both $SNDR_x$ and $SNDR_y$ experience a decrease of 4.5 dB and 3.5 dB, more than 10 and 8 s.t.d.s from their corresponding running averages.

Figure 9 demonstrates the scenario when the DL mode is trigged at a 61–62 min interval when the measured $SNDR_y$ decreases by more than four s.t.d.s from its running average. Spikes from this time interval are used to learn a new dictionary. The new dictionary items are added to form a dictionary that is used to recover a signal after 62 min, as shown in figure 9(c). When the system switches from DL

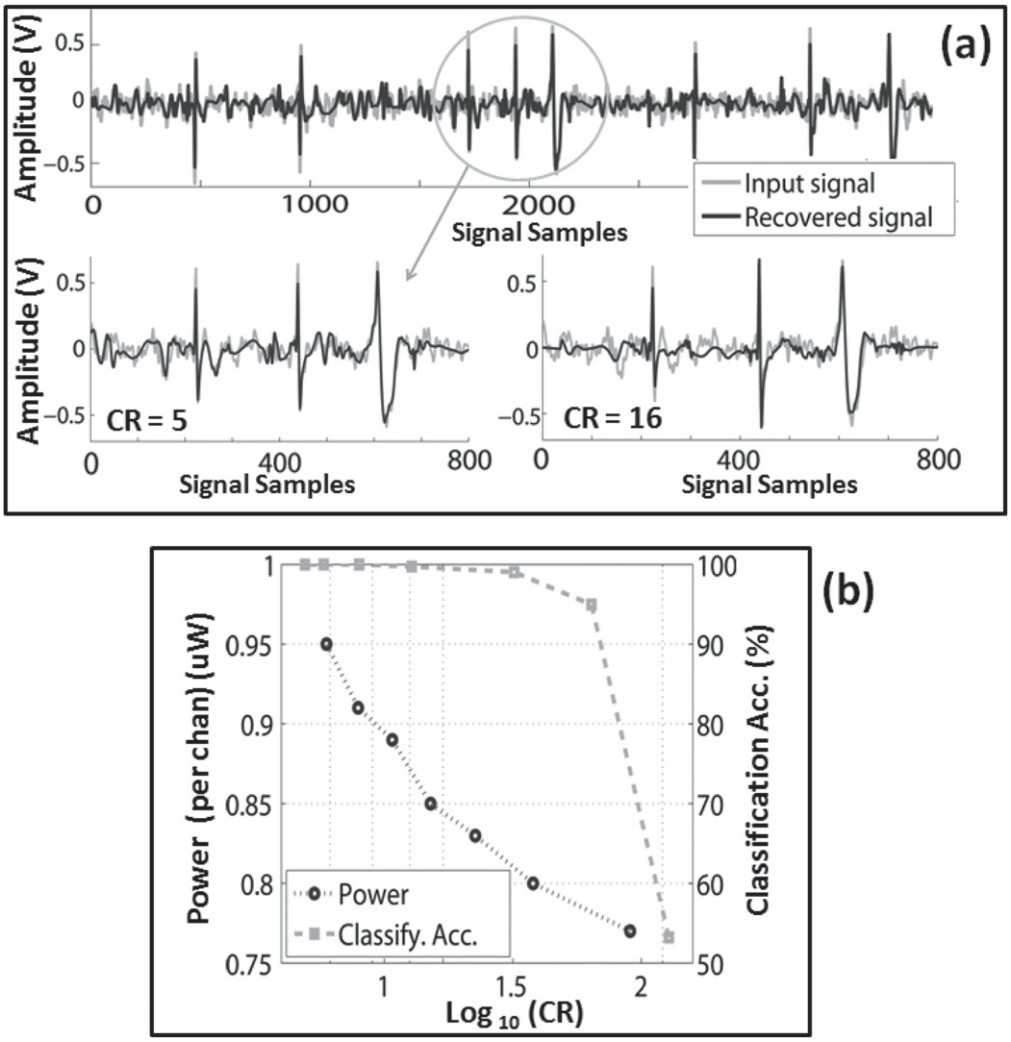

Figure 13. (a) Recovery quality at CR = 5 and 16. (b) ASIC power consumption and spike classification accuracy with respect to CR. Classification is evaluated using the Easy1 database with noise s.t.d of 0.05 from (Quiroga 2004) using a wavelet classification method similar to (Quiroga 2004).

mode back to CM mode, we see an increase of both $SNDR_x$ and $SNDR_y$ at 62 to 63 min compared to the $SNDR_x$ and $SNDR_y$ measurement in figure 8. Figure 9(d) shows that the spikes that appeared at the 62 to 63 min intervals can now be well recovered using the newly learned dictionary.

The results of the experiment using the synthetic spike train is plotted in figure 10(c). Without dictionary re-training, the $SNDR_y$ decreases to around 15 to 25 dB whenever a new type of spike appears. On the other hand, after re-training the dictionary at around 11 s and 31 s, the $SNDR_y$ remains constant at around 28 dB when new types of spikes appear. The recovery dictionaries are shown in figure 10(a).

4.3. In vivo experiment and system characterization using a standard neural database

4.3.1. Experiment set-up. We recorded neural data using the ASIC from a chronic microelectrode array positioned over the premotor cortex of the right hemisphere of an awake Rhesus Macaque. The chronic array features 18 independently movable electrodes, with each electrode positioned by a screw mechanism at a resolution of 50 microns. The electrodes themselves were epoxylite-coated tungsten electrodes with impedances ranging from 4–6 MOhms (FHC Inc.). Contact can be made from each of the 18 electrodes to any of the four electrodes on the ASIC. Each electrode was driven into the cortex until neural activity was found. From this point, electrodes would be maintained at this position for days or weeks. In the experiment, we acquire the digitized data directly after the ADC, as well as the compressed data. A detailed experiment set-up is shown in figure 11. The monkey is not shown due to regulation and ethics reasons.

To compare the performance of the system with previous works, we also characterized the system using a standard neural database (Quiroga 2004). We utilized a 12-bit DAC, followed by a 40 dB attenuator, to play back the recorded neural waveforms to be recorded by the ASIC. The ASIC then transmits the compressed samples to a PC in which the signal recovery block and QE block are implemented.

	CS-MEA	Kamboh '09	Chen '13	Charbiwala '11	Gao '12
Compression Technique	CS + Dictionary Learning	DWT (lifting scheme)	CS	CS	NONE
Implementation	**ASIC**	ASIC	ASIC	PCB	ASIC
ADC resolution	**10bits**	N/A	6-8bits	12 bits	10 bits
CR @ 95% Class. Accuracy	**10.6**	Not Provided	2.05 **(a)**	2.00 **(b)**	No Compression
Layout Area per electrode (mm^2)	0.55 (AFE+ADC) 0.11 (digital compression)	0.17	0.09 (comp. circuit)	N/A	0.26 (AFE+ADC)
Power Consumption per electrode	**15uW (AFE+ADC) 0.83uW (digital compression)**	95uW (digital compression)	1.9uW (digital compression)	20uA current consumed	41uW (AFE + ADC)
process	0.18um	0.5um	0.09um	N/A	0.13um

(a) The authors of [Chen '13] did not benchmark recovery quality using the standard Univ. Leicester neural database . However, their method, is evaluated on this database by the authors of [Bulach '12].

(b) This system only compresses the detect spikes, instead of the raw neural waveform

Figure 14. Comparison of our work with other state-of-the-art approaches.

4.3.2. Raw and recorded waveforms from the in vivo experiment. We recorded neural signals from three of the 18 electrodes placed within the premotor cortex of a Rhesus Macaque, where the electrodes SNR > 1.1. To calculate the SNR, the method described in section 4.1 is used. Around one minute of data is collected prior to compression to train a dictionary using the method outlined in section 3.1c. Around one minute of the data is collected for each electrode. During recording the ASIC is configured to compress the entire neural signal instead of just the spikes. Figure 12(a) shows the recovery results from an electrode with SNR = 6.21. For this electrode, the CR is set to 12.8. Two types of spikes are seen at this electrode. The off-chip recovery block could recover both with great precision, achieving a $SNDR_x$ of 9.52 dB. Figure 12(b) shows the recovery quality of an electrode with a lower SNR (2.14), where only one spike cluster is seen. The CR is set to 4.3. The off-chip recovery block could recover spikes and the inter-spike signal at great accuracy, achieving spike $SNDR_x$ of 4.14 dB. This result validates the functionality of the complete system and demonstrates that its performance agrees with the offline analysis presented in previous sections.

4.3.3. System characterization using a standard neural database. In order to compare our system's performance with previous published works, we characterized the system's performance using a standard neural data base (Quiroga 2004). The dataset used is named 'Easy1' in the database. All the spikes in this dataset have a normalized amplitude of 1, with various amounts of noise added as the zero mean Gaussian noise with a s.t.d. from 0.05 to 4.0. Figure 13(a) shows a temporal view of the recovery at CR = 5 and 16 for Easy 1, with noise s.t.d. = 0.05. Figure 13(b) demonstrates the recovery quality and power consumption with respect to the CR. The classification accuracy decreases as we increase the CR. But as the CR increases, more accumulators are turned off; therefore, the power consumption of the ASIC decreases. For this particular dataset, spike classification accuracy reaches >95% when the CR decreases below 21.3.

Figure 14 shows a comparison of our system to prior works. All these systems intend to perform on-chip compression of neural signals to achieve reduction in data bandwidth for wireless (or wireline) communication. Like previous works, we have evaluated our approach on the Easy1 dataset from (Quiroga 2004) across all the noise s.t.d.s (from 0.05 to 4.0). The lowest CR needed to achieve >95% spike classification accuracy for all the datasets is used here as a performance metric (CR @ 95%).The CS circuit in this work can function with a VDD of 0.53 V without performance degradation and hence consumes only 0.83 uW (per electrode) for the compression architecture. The CS block itself uses only 0.11 mm^2 area per electrode. Even with the lowest power consumption and comparable area, this implementation achieves a better CR (CR = 10.6) that is over five times better than the state-of-the-art CR. The total power consumption per electrode (<15.83 μW), including AFE and ADC, is comparable to published state-of-the-art systems.

5. Conclusion

We have demonstrated a CS neural recording system. Using a learned dictionary, this system is capable of achieving high rate of compression for both raw neural signals (CR > 10.6) as well as detected spikes (CR > 16). This system is extremely low power (<0.83 uW per electrode) and consumes a very small area (<0.11 mm^2). Thus, this system can be scaled and

integrated into large recording arrays containing thousands of electrodes.

Because the demonstrated compressed sensing technique relies on reconstructing the spike using DL using a small duration of raw recording, an open-loop recording cannot adapt to changes in spike shape. By introducing close-looped recording with a reconstruction performance evaluation, the system can detect and adapt to this change, thus making the system more practical for long-term recording.

While showing superior performance, the proposed system also has a few limitations: The experiments demonstrate that the proposed CS method is able to achieve an extremely high CR for recording channels with a high SNR. The CR degrades as the electrode's SNR decreases, as noise affects the performance of the DL algorithm. Furthermore, without a proper dictionary, the system might not be able to reconstruct the waveform of a sparsely firing neuron. We will address these limitations in our future work.

References

Abdelhalim K, Jafari H M, Kokarovtseva L, Perez V J L and Genov R 2013 64-channel UWB wireless neural vector analyzer SOC with a closed-loop phase synchrony-triggered neurostimulator *IEEE J. Solid-State Circuits* **48** 2494–510

Aharon M, Elad M and Bruckstein A 2006 K-svd: an algorithm for designing overcomplete dictionaries for sparse representation *IEEE Trans. Signal Process.* **54** 4311–22

Aziz J N, Abdelhalim K, Shulyzki R, Genov R, Bardakjian B L, Derchansky M, Derchansky D, Serletis D and Carlen P L 2009 256-channel neural recording and delta compression microsystem with 3D electrodes *IEEE J. Solid-State Circuits* **44** 995–1005

Baraniuk R G, Cevher V, Duarte M F and Hegde C 2010 Model-based compressive sensing *IEEE Trans. Information Theory* **56** 1982–2001

Braitenberg V and Schüz A 1991 *Anatomy of the Cortex: Statistics and Geometry* (Berlin: Springer)

Bulach C *et al* 2012 Evaluation study of compressed sensing for neural spike recordings *IEEE EMBC* 3507–10

Candes E, Romberg J and Tao T 2006 Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information *IEEE Trans. Information Theory* **52** 489–509

Chae M, Liu W, Yang Z, Chen T, Kim J, Sivaprakasam M and Yuce M 2008 A 128-channel 6 mw wireless neural recording ic with on-thefly spike sorting and UWB tansmitter *Solid-State Circuits Conf.* pp 146–603

Charbiwala Z, Karkare V, Gibson S, Markovic D and Srivastava M B 2011 Compressive sensing of neural action potentials using a learned union of supports *Proc. of the IEEE Int. Conf. on Body Sensor Networks (BSN) (Dallas, TX, 23–25 May 2011)* pp 53–8

Charbiwala Z *et al* 2012 CapMux: a scalable analog front end for low power compressed sensing *Proc. of the IEEE Int. Green Computing Conf. (IGCC) (San Jose, CA, 4–8 June 2012)* pp 1–10

Dixon A M R, Allstot E G, Gangopadhyay D and Allstot D J 2012 Compressed sensing system considerations for ECG and EMG wireless biosensors *IEEE Trans. Biomedical Circuits and Systems* **6** 156–66

Donoho D L 2006 Compressed sensing *IEEE Trans. Information Theory* **52** 1289–306

Duarte-Carvajalino J M and Sapiro G 2009 Learning to sense sparse signals: simultaneous sensing matrix and sparsifying dictionary optimization *IEEE Trans. Image Process.* **18** 1395–408

Elad N 2007 Optimized projections for compressed sensing *IEEE Trans. Signal Process.* **55** 5695–702

Engan K, Aase S O and Husøy J H 2000 Multi-frame compression: theory and design *Signal Process.* **80** 2121–40

Gangopadhyay D, Allstot E G, Dixon A M, Natarajan K, Gupta S and Allstot D J 2014 Compressed sensing analog front-end for bio-sensor applications *IEEE J. Solid-State Circuits* **49** 426–38

Gao H *et al* 2012 HermesE: a 96-channel full data rate direct neural interface in 0.13 um CMOS *IEEE JSSC* **47** 1043–55

Gosselin B and Sawan M 2009a An ultra low-power CMOS automatic action potential detector *IEEE Trans. Neural Systems and Rehabilitation Engineering* **17** 346–53

Gosselin B, Ayoub A E, Roy J F, Sawan M, Lepore F, Chaudhuri A and Guitton D 2009b A mixed-signal multichip neural recording interface with bandwidth reduction *IEEE Trans. Biomedical Circuits and Systems* **3** 129–41

Henze D A, Borhegyi Z, Csicsvari J, Mamiya A, Harris K D and Buzsáki G 2000 Intracellular features predicted by extracellular recordings in the hippocampus *in vivo* *J. Neurophysiol.* **84** 390–400

Hubel D H and Wiesel T N 1959 Receptive fields of single neurones in the cat's striate cortex *J. Physiol.* **148** 574–91

Kamboh A M, Mason A and Oweiss K G 2008 Analysis of lifting and B-spline DWT implementations for implantable neuroprosthetics *J. Signal Process. Syst.* **52** 249–61

Kamboh A M, Oweiss K G and Mason A J 2009 Resource constrained VLSI architecture for implantable neural data compression systems *Proc. of the IEEE Int. Symp. on Circuits and Systems (ISCAS) (Taipei, 24–27 May 2009)* pp 1481–4

Kim S, Normann R A, Harrison R and Solzbacher F 2006 Preliminary study of the thermal impact of a microelectrode array implanted in the brain *Proc. 28th IEEE Ann. Int. Conf. of the Engineering in Medicine and Biology Society (EMBS) (New York, NY, 30 August-6 September 2006)* pp 2986–9

Lewicki M and Sejnowski T 2000 Learning overcomplete representations *Neural Comput.* **12** 337–65

Lopez C M, Andrei A, Mitra S, Welkenhuysen M, Eberle W, Bartic C, Puers P, Yazicioglu R F and Gielen G 2013 An implantable 455-active-electrode 52-channel CMOS neural probe *IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers (San Francisco, CA, 17–21 February 2013)* pp 288–9

Ludwig K A, Miriani R M, Langhals N B, Joseph M D, Anderson D J and Kipke D R 2009 Using a common average reference to improve cortical neuron recordings from microelectrode arrays *J. Neurophysiol.* **101** 1679

Mamaghanian H, Khaled N, Atienza D and Vandergheynst P 2011 Compressed sensing for real-time energy-efficient ECG compression on wireless body sensor nodes *IEEE Trans. Biomed. Eng.* **58** 2456–66

Mitra S *et al* 2013 24-channel dual-band wireless neural recorder with activity-dependent power consumption *IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers (San Francisco, CA, 17–21 February 2013)* pp 292–3

Needell D and Tropp J A 2009 CoSaMP: iterative signal recovery from incomplete and inaccurate samples *Appl. Comput. Harmon. Anal.* **26** 301–21

Oweiss K G, Mason A, Suhail Y, Kamboh A M and Thomson K E 2007 A scalable wavelet transform VLSI architecture for real-time signal processing in high-density intra-cortical implants *IEEE Trans. Circuits Syst. I* **54** 1266–78

Quian Q R, Nadasdy Z and Ben-Shaul Y 2004 Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering *Neural Comput.* **16** 1661–87

Quiroga R, Nadasdy Z and Ben-Shaul Y 2004 Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering *Neural Comput.* **16** 1661–87

Shahrokhi F *et al* 2010 The 128-channel fully differential digital integrated neural recording and stimulation interface *IEEE Trans. Biomed. Circuits Syst.* **4** 149–61

Staba R J, Wilson C L, Bragin A, Fried I and Engel J 2002 Sleep states differentiate single neuron activity recorded from human epileptic hippocampus, entorhinal cortex, and subiculum *J. Neurosci.* **22** 5694–704

Suo Y, Zhang J, Etienne-Cummings R, Tran T D and Chin S 2013 Energy-efficient two-stage compressed sensing method for implantable neural recordings *Proc. of the IEEE Biomedical Circuits and Systems Conf. (BioCAS) (Rotterdam, 31 October-2 November 2013)* pp 150–3

Suo Y *et al* 2014 *Structured Dictionary Learning for Classification* (arXiv:1406.1943)

Tropp J A and Gilbert A C 2007 Signal recovery from random measurements via orthogonal matching pursuit *IEEE Trans. Information Theory* **53** 4655–66

Neuro Seeker 2013 www.neuroseeker.eu/

Zhang J *et al* 2013 An efficient and compact compressed sensing microsystem for implantable neural recordings *IEEE Trans. BioCAS* **8** 455–96