OPEN ACCESS Freely available online

PLoS one

Fast, Scalable, Bayesian Spike Identification for Multi-Electrode Arrays

Jason S. Prentice[1]*, Jan Homann[1], Kristina D. Simmons[2], Gašper Tkačik[1], Vijay Balasubramanian[1,2], Philip C. Nelson[1]

1 Department of Physics and Astronomy, University of Pennsylvania, Philadelphia, Pennsylvania, United States of America, **2** Department of Neuroscience, University of Pennsylvania, Philadelphia, Pennsylvania, United States of America

Abstract

We present an algorithm to identify individual neural spikes observed on high-density multi-electrode arrays (MEAs). Our method can distinguish large numbers of distinct neural units, even when spikes overlap, and accounts for intrinsic variability of spikes from each unit. As MEAs grow larger, it is important to find spike-identification methods that are *scalable*, that is, the computational cost of spike fitting should scale well with the number of units observed. Our algorithm accomplishes this goal, and is fast, because it exploits the spatial locality of each unit and the basic biophysics of extracellular signal propagation. Human interaction plays a key role in our method; but effort is minimized and streamlined via a graphical interface. We illustrate our method on data from guinea pig retinal ganglion cells and document its performance on simulated data consisting of spikes added to experimentally measured background noise. We present several tests demonstrating that the algorithm is highly accurate: it exhibits low error rates on fits to synthetic data, low refractory violation rates, good receptive field coverage, and consistency across users.

Citation: Prentice JS, Homann J, Simmons KD, Tkačik G, Balasubramanian V, et al. (2011) Fast, Scalable, Bayesian Spike Identification for Multi-Electrode Arrays. PLoS ONE 6(7): e19884. doi:10.1371/journal.pone.0019884

Editor: William Rowland Taylor, Oregon Health & Science University, United States of America

Received September 1, 2010; **Accepted** April 19, 2011; **Published** July 20, 2011

Copyright: © 2011 Prentice et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Funding: This work was supported by National Science Foundation (NSF) grants IBN-0344678, EF-0928048, National Institutes of Health (NIH) grant RO1 EY08124, NIH training grant T32-07035, and NIH training grant 5T90DA022763-04. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: jprentic@sas.upenn.edu

Introduction

The vertebrate retina is an important model system in neuroscience because it is amenable to detailed study despite having a complex structural and functional architecture [1]. Population coding and collective behavior in the retinal output is studied by use of multi-electrode arrays (MEAs) to record extracellularly from many retinal ganglion cells (RGCs) simultaneously [2,3]. Similar recordings can now also be made in other brain areas [4]. MEAs offer unprecedented possibilities to obtain both single neuron and single action potential resolution from large tissue samples. However, recordings obtained in this way are useful only if most spikes can be assigned, with sufficient accuracy, to the neurons that generated them. Even if each neuron spikes with a unique waveform signature, we must still determine all those "template" waveforms present in a dataset, separating them from each other and from noise. Moreover, in practice there can be wide variation in the spike waveforms from a given unit (for instance in amplitude), complicating the task of determining from data which units fired and when.

This "spike sorting problem" is therefore a bottleneck in the use of high density arrays with hundreds or thousands of electrodes. Methods that were manageable with tetrodes [5] do not generally scale up to the massive datasets that large arrays generate. For example, some standard methods cluster data by manually examining two-dimensional projections in a feature space of a few tens of dimensions. This approach is infeasible when the feature space contains thousands of dimensions.

Another challenge with large arrays is that the chance of seeing a single isolated spike becomes negligible, simply because there is so much activity. Thus we must find template waveforms corresponding to the activity of single neural units (as extracellular recording cannot unambiguously identify distinct single neurons, we will refer to our recovered putative neurons as "neural units") without ever seeing a pristine example of one, and we must be prepared to decompose temporally overlapping spikes in essentially every recorded event. Overlaps in both space and time are less frequent, but they nevertheless must be resolved if we wish to unravel the patterns of collective neural activity. Resolution methods that rely on exhaustively checking all possible combinations suffer a combinatorial explosion for large arrays. Further, any spike decomposition method must stop before every spike has been found, because there will be some units whose intrinsic amplitude is not larger than recording noise. We need a principled approach to terminating each fit and to deciding later which units' activities have been reliably captured.

Thus, to be most useful for large arrays, a spike identification algorithm should ideally both *scale well* and also *be able to decompose overlapping events*. This article outlines a method that accomplishes these goals (MATLAB code available upon request.) It first clusters a small subset of a larger dataset, using a partially automated, human-guided clustering technique. This manual step is efficient, and scalable, because (i) the clustering is based on an ordering that arranges event data by similarity along a single dimension, (ii) the ordered data display band-like features that are visually very salient for human operators, making cluster cutting unambiguous,

and (iii) the algorithm is robust to variations and outliers in the cluster-cutting procedure. The algorithm then fits the full dataset to the spike templates thus obtained, using a modified Bayesian approach. In our data (from guinea pig retina) most of the intrinsic variability of spikes from a given unit consists of amplitude variation only, whereas other variability can be summarized as a universal (spike-independent) noise process. By carefully modeling these circumstances we greatly reduce our computational burden.

After characterizing the spatiotemporal character of the noise, our algorithm identifies spikes iteratively in a matching-pursuit (or "greedy") scheme [6,7]. Fitting terminates when addition of another spike does not improve the posterior probability of a fit; a natural overfitting penalty is provided by prior probabilities of firing and of spike amplitude. No assumptions are made about spike time cross-correlations; in particular, we do not require *a priori* any refractory "hole" in the spike time autocorrelation functions. Nevertheless, all of the inferred spike trains corresponding to otherwise acceptable spike types do exhibit such a hole, which serves as a check on our results. Fitting is followed by several qualitative post-hoc validations of the templates. Where our method requires human judgment, we structure our techniques and develop tools to facilitate interaction and proofreading.

Our approach combines successful elements from previous techniques: the empirical characterization of the noise [8]; separation of clustering and fitting steps and the iterative subtraction scheme for handling overlaps [7]; and division of the clustering task by leader electrode address [9]. Novel features of our approach include systematic exploitation of the spatial organization of the signals, the use of an ordering algorithm to greatly simplify clustering, the observation that the noise temporal correlation is well represented by a simple function, an explicit model of spike amplitude variation, and the introduction of a principled Bayesian likelihood criterion for terminating spike fitting. Each of these innovations adds a critical element to the success of our spike sorting method. Although we focused on data taken on vertebrate retina, the methods should be equally applicable to other kinds of MEA data, for example in other brain areas [4].

Results

To illustrate our method, we tested our spike sorting algorithm on 120 minutes of recordings from guinea pig retinal ganglion cells (RGC), acquired with a 30-electrode, dense MEA covering about 0.018mm^2 of tissue (Fig. 1A). The analysis described in this paper identified 1,260,475 spikes in the dataset. A typical firing event took the form Fig. 1B, where each panel shows 3ms of the electrical potential recorded by each electrode (or "channel"). We identified spiking events as voltages crossing a threshold of -40 μV, taking into account the fact that simultaneous threshold crossings on neighboring channels represent the same spike event (see Methods for details). The duration of each spike event was taken to be 3.2ms centered on the event's peak.

In addition to identifiable spikes, each electrode had background activity with a standard deviation of $\sim 10\mu$V that we will collectively refer to as "noise." Potential sources for this activity include true (Johnson) noise in the electrode and electronics, electrical pickup from the environment, as well as a hash of background activity from weak or distant neurons [10]. A challenge for spike identification is that in general there is no way to separate these three classes of "noise" cleanly from each other, nor from the spikes of interest to us. Nevertheless, we will propose a technique for identifying spikes that is very accurate for firing events with intrinsic amplitude at least 4 times the standard deviation of the noise.

Fig. 1B illustrates that each single firing unit will be "heard" on multiple electrodes, and that those electrodes form a spatially localized group. Our method is scalable because it systematically exploits this simple observation: even on a large electrode array, most firing units will involve only a handful of electrodes. (Some of our signals were *not* local, and stretched over the entire electrode array in a line (e.g., Fig. 1B). We ignored such axonal signals, which were also distinguished by their low amplitude and triphasic shape.)

Preliminary visualization of our data

We first attempted a "geographical clustering": from each event we found the minimum of the potential on each channel and the channel containing the deepest minimum ("leader channel"). We then used the absolute values of the minima as weights in a weighted average of the locations of the 9 electrodes neighboring the leader channel. This weighted average gave a particularly salient two-dimensional feature, the event's *barycenter*:

$$\bar{\mathbf{x}} = \frac{\sum_{i=1}^{9} \mathbf{x_i} |\min V_i(t)|}{\sum_{i=1}^{9} |\min V_i(t)|}.$$

The sums run over the 9 electrodes neighboring the leader, and $\mathbf{x_i}$ is the location of the i^{th} neighbor electrode. We then augmented $\bar{\mathbf{x}}$

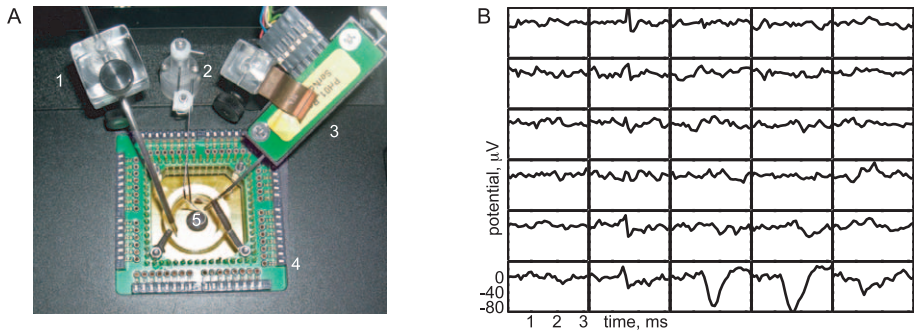

Figure 1. Recording chamber and typical data. (*A*) Typical MEA apparatus. A tissue sample was mounted in an inverted microscope, with images projected onto it via a small video monitor at the camera port (not visible). *Clockwise from left*, 1: suction; 2: tissue hold-down ring; 3: perfusion inflow, with temperature control; 4: preamplifier; 5: location of the multi-electrode array. (*B*) Example of a single-spike event. Each subpanel shows the time course of electrical potential (μV) on a particular electrode in the 5×6 array. The electrodes are separated by 30μm (similar to RGC spacing). A spike from one unit is visible in the lower right corner and an axonal spike can be seen running vertically in the second column of electrodes. Data were acquired at 10kHz. After baseline subtraction and high-pass filtering, a spatial whitening filter was applied (see Methods, Step 1).
doi:10.1371/journal.pone.0019884.g001

with a third feature, z, minus the absolute peak potential. These three features gave a scatterplot that clearly showed many well-separated clusters (Fig. 2A), without any need to deduce the "best" features by principal component analysis (Fig. 2B).

This extension of the "triangulation" method developed for tetrode recordings [5] already shows key aspects of the data: (a) many clusters are highly dispersed in amplitude, and (b) some cluster pairs appear at nearly the same spatial locations but are nevertheless well separated by amplitude. The first circumstance means that we must allow for variable amplitude when fitting spikes to templates representing the clusters. The second warns us that a simple least-squares fit to amplitude could confound two distinct units. For this reason our spike-fitting method employs a Bayesian prior for each cluster's amplitude variation, allowing us to make such discriminations.

Although the simple clustering based on spatial location in Fig. 2A looks promising, it can be misleading. Indeed, the restriction of the weighted average to the 9 electrodes around the leader can artificially separate clusters by biasing the barycenter to be located near a particular electrode. This problem could be alleviated by using a larger neighborhood, but on large arrays there will inevitably be temporal collisions of spikes from distinct units. The barycentric features in Fig. 2A will register such collisions as a haze of seemingly random spots. Thus, at a minimum the MEA voltage traces must be segmented by exploiting the spatial locality of recorded responses. Despite these shortcomings, Fig. 2A points out why the more sophisticated method developed in this article can succeed: the "geographic" information encoded by the MEA is a powerful intrinsic clue to each unit's identity.

Summary of our method

Our sorting method is outlined in Fig. 2C (details in Methods and Text S1 Sect. B). From a subset of the raw data, we made a preliminary classification of spike events in terms of the electrode on which they achieved their peak voltage. All events sharing a given leader channel were cropped to the 9 electrodes neighboring the leader, then ordered with the OPTICS algorithm [11] into a linear sequence. The OPTICS algorithm places similar events nearby in the sequence; distance was measured by a normalized Euclidean distance between event voltage traces (see Methods). The linear sequence of events was displayed to the user along with all the recorded voltage samples for each event (Fig. 2D), and

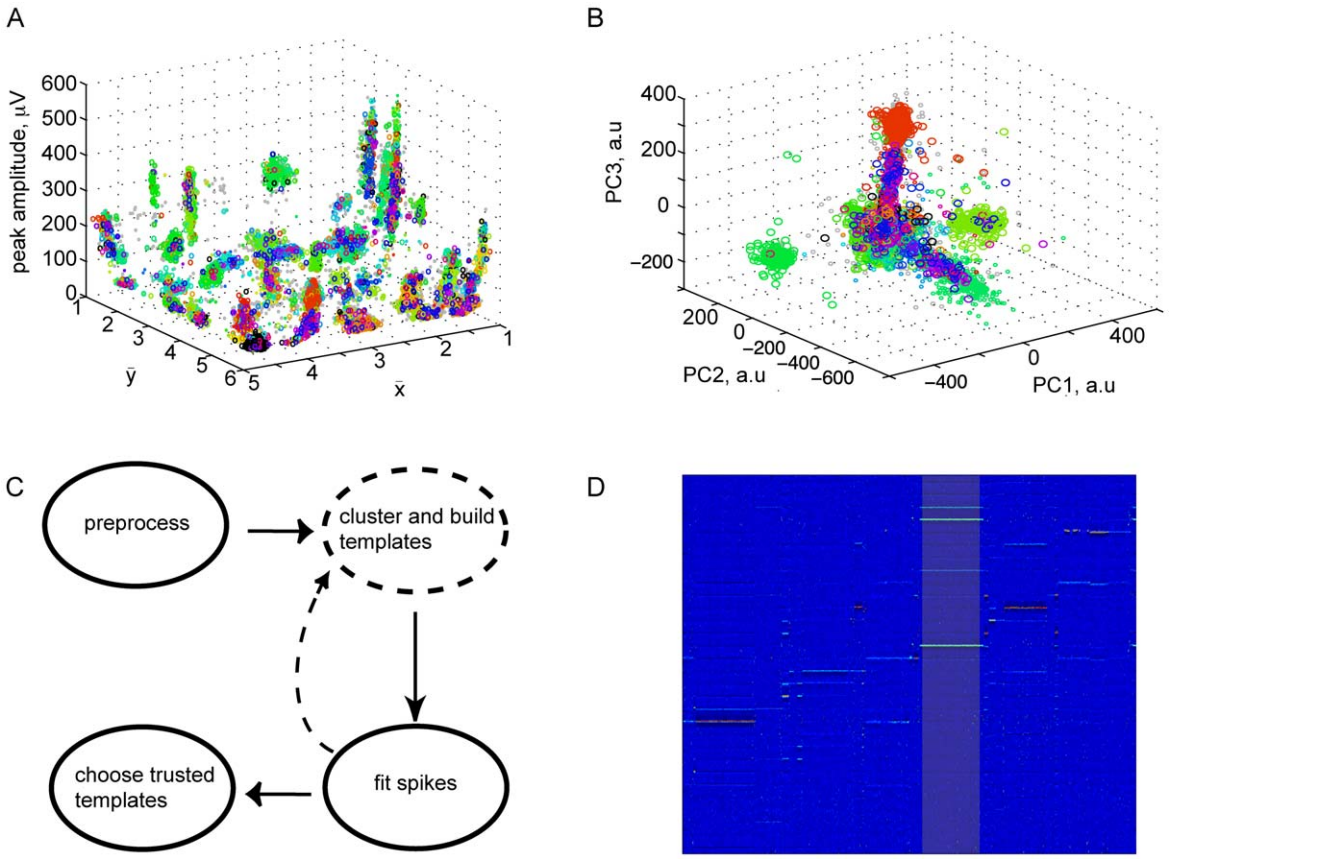

Figure 2. Visualization of data; flowchart of our method. (*A*) 22,234 firing events cluster well in terms of their *barycenter* (voltage weighted average spatial location) and absolute peak voltage (see text), despite wide amplitude dispersion in some groups; each combination of color and marker size corresponds to one spiking unit identified by the clustering method developed in the text. Grey points were unassigned to any cluster. A total of 107 clusters are marked. (*B*) Events cluster poorly when projected onto the three principal features uncovered by principal component analysis (PCA). Coloring as in (*A*). (*C*) Schematic of our spike sorting method. Dashed lines involve a small subset of a full dataset. The backwards arrow describes the introduction of new spike templates found after the first pass of fitting; a total of two passes are performed. (*D*) The optics algorithm orders all firing events into a linear sequence based on a distance measure (see text). Events are lined up in this order (x-axis), and represented in terms of the 960 voltage samples recorded by all the electrodes during a 3.2 ms firing event (y-axis; from top to bottom, 32 consecutive time samples from one channel, then 32 time samples from the second channel, and so on). The human operator highlights bands of events (typically very clear to an observer) that appear to constitute a single cluster; one such band is shown. Later automated diagnostics refine and check these assessments.
doi:10.1371/journal.pone.0019884.g002

manually clustered. Although the ordering was based on events cropped to 9 channels, the full waveforms were displayed to the user (Fig. 2D). Because the data are ordered in one dimension, and because precision is not required in view of later refinement, this manual step remains rapid. An automated method for cluster cutting could be implemented, but in view of the inevitable desire for human proofreading we preferred to simply carry out this step by hand. From each preliminary cluster, we estimated a template waveform representing the corresponding neural unit and then fit the templates to the remaining data. Fitting was accomplished by a Bayesian algorithm based on a probabilistic model capturing the dominant sources of variability we observed in our data: background noise, spike amplitude variation, and overlapping spikes from distinct units. After finding, for each event, the most probable template which accounts for the event, we subtracted it and then iterated. Finally, the fit results were used in a post-hoc validation of the initial clustering, and we repeated the procedure in a second pass if necessary. Details of each step are presented in Methods.

Tests of our method

OPTICS-based clustering of a subset of our dataset led to 107 potential templates for events from distinct neural units. Many of these templates had low amplitudes; such low-amplitude templates were sometimes mistakenly fit to noise by our algorithm. We therefore rejected units that were likely to contain substantial noise fits because they were of amplitude less than or comparable to the noise (details in Methods and Text S1 Sect. H). This left fifty potentially reliable units that were accepted in our dataset.

Comparison with geographical clusters. Our optics-based procedure identified 107 potential clusters of events in a subset of the data. To check that the procedure gave reasonable results we plotted each event in the barycentric coordinates of Fig. 2A, and colored the events according to the assigned cluster label. The clusters were spatially localized and separated in peak amplitude, as they should be if they were produced by distinct single neurons. Gray dots in Fig. 2A were not assigned to any cluster. Some of these events contained overlaps of spikes that were not resolved by the initial spatial segmentation of data during the preprocessing step. The subsequent spike fitting step in our algorithm resolved most such ambiguities.

Error rates on synthetic data. To validate our algorithm we tested its performance on synthetic data created by adding spikes to experimentally measured background noise clips, then fitting templates to each clip. We took noise clips to be 3.2ms segments of time during which no spikes were recorded on any channel; we identified 14,000 such clips. For each clip, we randomly chose a fixed number (1, 3, or 5) of templates from the initial set of 107, with uniform probability and without replacement. We then added these templates to the noise clip at random times, leaving a margin of 0.6ms on either side of the clip to prevent waveforms from being cut off. (Our typical template waveforms extended approximately 0.5ms to either side of the peak.) We gave each spike an amplitude drawn from a Gaussian distribution with mean equal to its template amplitude and standard deviation 10% of the template amplitude (this was similar to the observed distribution).

The template fitting algorithm was then run over this synthetic dataset and analyzed for false positive and false negative rates (Fig. 3A). We counted a false negative for a template every time that template was present in an event but not fit correctly to within 1ms; we counted a false positive every time a template was fit to the data without actually being present. The error rates increased with the number of template overlaps; thus, for the fifty templates with amplitudes that exceed the noise, we separately plotted error rate histograms for each degree of overlap. Error rates were robustly low – even within extremely complex events with 5 overlapping spikes (rare in the data), the majority of spike templates had an error rate of a few percent or less. To gain perspective on these values, we measured the number of templates fit to each event in our recorded data: 60% of events contained 1 spike, 94% 3 or fewer spikes, and 98% had 5 spikes or fewer. Most of the errors were made on lower amplitude templates for which amplitude variations can lead to confusion with noise.

Refractory violations. When sorting spikes recorded extracellularly, ground truth can be assessed if simultaneous intracellular recordings are available, e.g., [12]. Since we do not have such recordings, in order to validate our algorithm on real data we examined the rate of refractory violations – i.e., the fraction of interspike intervals of duration less that 1.5ms. Refractory violations can appear in our sorted data if spikes from distinct neural units are mis-assigned to the same unit, or if noise fluctuations are mistaken for spikes. Of the 107 templates constructed from the initial clustering 84% had less than 0.5% refractory violations and all had less than 2.5%, providing evidence that the templates produced by the initial clustering rarely merge distinct neural units (Fig. 3C). More significantly, all fifty templates describing units that rose reliably over the noise level had less than 0.5% refractory violations. Futhermore, 96% of these had less than 0.1% refractory violations (Fig. 3C). Note that the percentage of refractory violations only provides a lower bound on the number of misidentified spikes – nevertheless, the low refractory violation rates we observed provide strong evidence that our algorithm makes few fitting mistakes on the units otherwise identified as reliable.

Coverage. While the absence of refractory violations gives evidence that our algorithm does not merge different neural units together, it might still split spikes from the same unit into two distinct clusters if, e.g. there was substantial amplitude variation. To test for this, for each unit that was above the noise level we measured the linear receptive field by taking the spike triggered average (STA) of the flickering checkerboard stimulus (Text S1 Sect. B). We expect that such receptive fields will be connected regions of the visual field, roughly elliptical in shape, and that no two units will have identical receptive fields. 31 of the 50 reliably identified units had enough spikes to give reliable estimates of the spatial receptive field. Of these, examination of the temporal kernel showed that 19 were OFF cells (responding to light decrement) and 12 were ON cells (responding to light increment), consistent with the expected excess of OFF ganglion cells [13–18]. None of these receptive fields were identical, giving evidence that our algorithm did not split single units into multiple clusters. Further, all of the receptive fields were connected, suggesting that none of our clusters are mixtures of different RGC. In addition, essentially all of the recorded area was covered by at least one receptive field (coverage of OFF cells shown in Fig. 3D). The density of RGCs in guinea pig varies from 250mm^{-2} to 1500mm^{-2} [19]. We receive signals from a region slightly larger than the electrode array, roughly 0.065mm^2. Thus the expected number of RGC is 16–97, comparable to our total of 31 receptive fields, keeping in mind that many of the sluggish cell types would not have enough spikes to yield a good spike triggered average.

Different users. Because our method involves human intervention, there exists a risk that different users could obtain divergent results. One possible discrepancy is that different users could identify distinct sets of templates. This would not pose a serious problem in and of itself, as long as further analysis did not demand a complete population. Moreover, the completeness of the template set is ultimately limited only by the time invested by

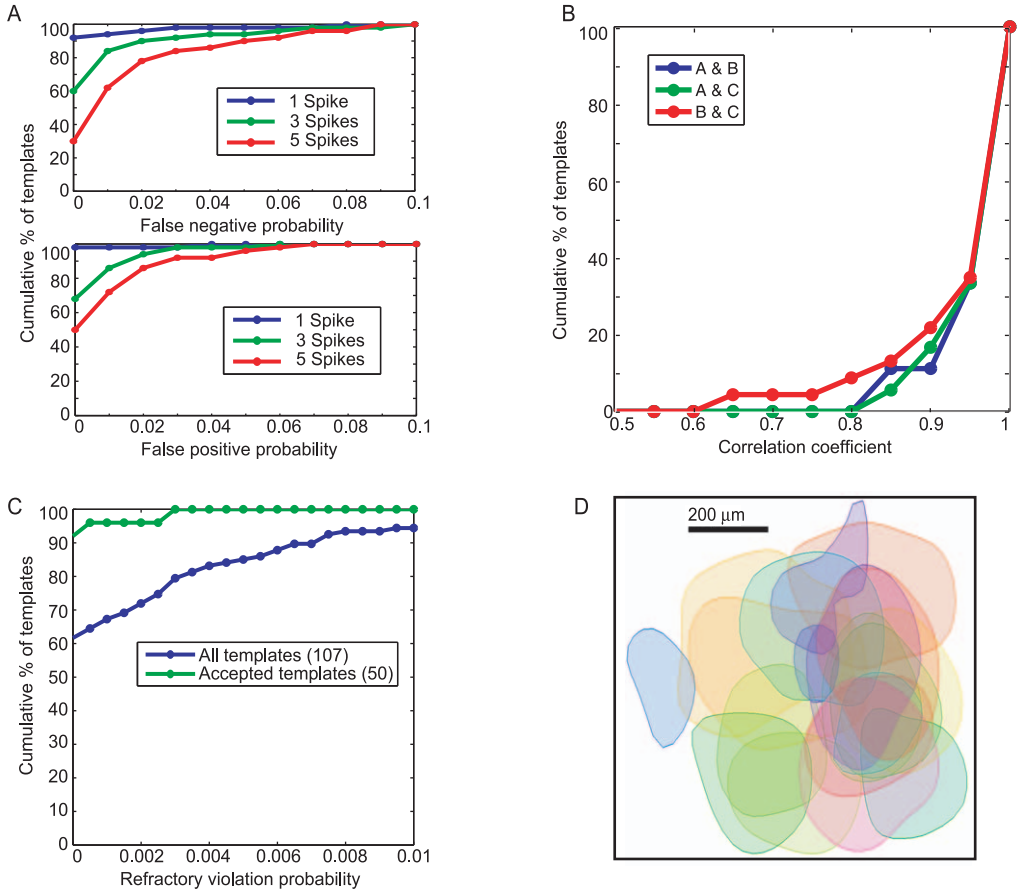

Figure 3. Tests of the method. (*A*) (*Top*) The cumulative percentage of templates having false negative probabilities less than the indicated values. Error rates were measured in fits to synthetic data as the fraction of times a fit was *not* reported for a template when it was actually present. (*Bottom*) As above, but showing false positive probabilities (fraction of times a fit was reported for a template when it was *not* actually present). Results reported separately for fits to events with different numbers of overlapping spikes (inset colors). (*B*) Correlation in spike trains across fits by three different users (A, B, and C). Each curve corresponds to one pair of users and gives the cumulative fraction of templates having lower correlation than indicated. See main text for further details. (*C*) Cumulative fraction of templates having fewer refractory violations than indicated. Refractory violations are rare (see text). (*D*) The centers of 19 OFF cell receptive fields recorded from a single piece of tissue. To map a neuron's receptive field center, we first find the peak (in space and time) of the spike-triggered average stimulus. Restricting to the peak time, we apply cubic spline interpolation in space and then draw contour lines at 75% of the peak value.
doi:10.1371/journal.pone.0019884.g003

the operator in clustering. However, it would be problematic if the spike trains fit to *the same* template by different users differed significantly. To assess the robustness of our method, we therefore had three different users (here referred to as A, B, and C) carry out our spike sorting procedure on the same data set and examined the correlation of the results. User A found 20 templates with large enough spike counts to assess cross-correlation, B found 25, and C found 28. While these numbers were smaller than the 50 we had identified previously, the difference consisted of units with extremely low firing rates. We compared the three sets of templates to identify those which were found by multiple users, by minimizing the euclidean distance between template pairs. 18 templates agreed between A and B, 18 between A and C, and 23 between B and C. A total of 17 templates were found by all three users. For each template that was found by at least two users, we computed the cross-correlation coefficient of the corresponding spike trains. Across all pairs of users, two-thirds of such templates had a spike train correlation higher than 0.95 (Fig. 3B).

Complex events. A major challenge for a spike sorting algorithm is dealing with variability in spikes produced by individual neural units. An even greater challenge arises from spatio-temporal overlaps between spikes from different neural units. Our low error rate in analysis of synthetic data containing both of these complexities (Fig. 3) provides evidence that our algorithm is effective at resolving overlaps and identifying variable spikes from given units. To test this further, we manually examined many events in the real data which a human observer could identify as representing overlaps or neural variability; and our algorithm typically did an excellent job of dealing with variable-amplitude bursts (Fig. 4B), as well as events that overlap in space and time (Fig. 4C).

Speed. Currently the main fitting code, written in MATLAB, requires about 5ms of real computer time per fit spike on a commercial 2.5 GHz computer, times 2 for the two passes. This is fast enough for our purposes; considerable further improvement is possible with existing software (Mex) and hardware (GPU) techniques.

Materials and Methods

Ethics statement

This study was carried out in accordance with recommendations from the National Institutes of Health and the guidelines of

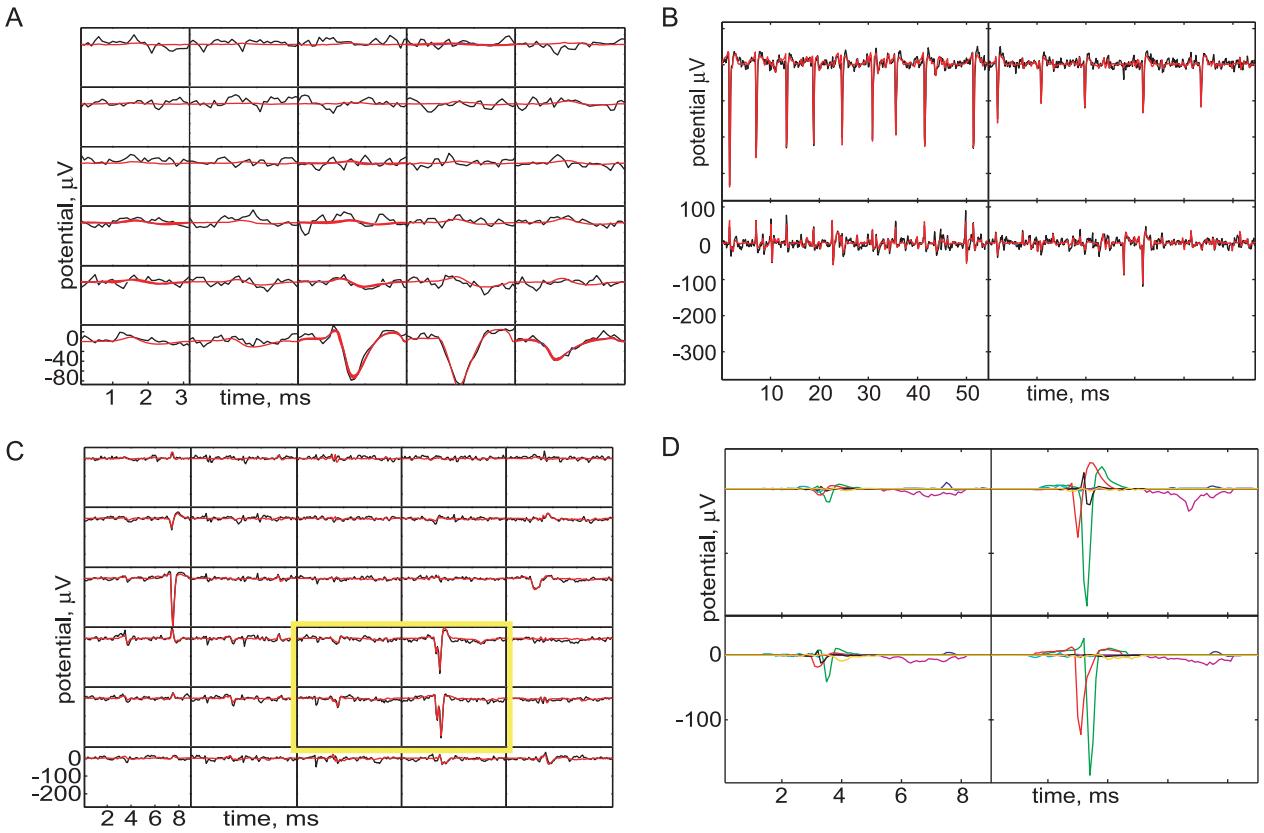

Figure 4. Template fitting to complex events. (*A*) Example of a single-spike event. Each subpanel shows the time course of electrical potential (in μV, *black curves*), on a particular electrode in the 5×6 array. After baseline subtraction and high-pass filtering, a spatial whitening filter was applied (see Methods). *Red curves* show the result of our fitting algorithm, in this case a single template waveform representing an individual neural unit. (*B*) Detail of a more complex event and its fit, in which a single unit fires a burst of 9 spikes of varying amplitudes (upper left channel), while a different unit fires 5 other spikes (upper right channel). Simultaneous data from four neighboring electrodes are shown. (*C*) Example of an overlap event and its fit, which now is a linear superposition of 7 templates. (*D*) Detail of (*C*), showing signals on four of the electrodes. This time individual fit spikes are displayed. The *red* and *green* traces show fit templates that, although similar, differ significantly in their overall strength, and in the relative strengths of their features. The *black* trace shows a fit to a low-amplitude template that was later classified as unusable, and hence was discarded, by the procedure in Methods, Step 4.
doi:10.1371/journal.pone.0019884.g004

the American Veterinary Medical Association. The protocol was approved by the Animal Care and Use Committee of the University of Pennsylvania (No. 803091). All surgery was performed under ketamine/xylazine and pentobarbital anesthesia, and all efforts were made to minimize suffering.

Experimental procedure

Our methods were developed and tested on retinal response data, from albino guinea pig, recorded with a dense 30-electrode array (30μm spacing, Multi Channel Systems MCS GmbH, Reutlingen, Germany). After anesthesia with ketamine/xylazine (100/20 mg/kg) and pentobarbital (100 mg/kg), the eyeball was enucleated and the animal was killed by pentobarbital overdose in keeping with the AVMA guidelines on euthanasia. The eyeball was hemisected and the retina was allowed to dark adapt. A small piece was cut out, separated from the pigment epithelium, mounted (ganglion cells up) onto a piece of filter paper, and placed ganglion cells down onto the MEA. A 15×15 flickering checkerboard consisting of binary noise, updated at 30 Hz, was projected onto the tissue (Text S1 Sect. C). We alternated between uncorrelated and exponentially correlated (space constant 50μm; time constant 33 ms) stimuli.

Our procedure for identifying spikes in the recorded data had four steps, each detailed below: (**1**) Preprocessing, where spatial locality was exploited to segment the data, (**2**) Clustering and template building, where a subset of the data was clustered to separate the responses of likely neural units, template waveforms for each neural unit were built, and their variability characterized, (**3**) Spike fitting, where every firing event was separated into a superposition of responses from different neural units, and (**4**) Validation of templates, where each template and the spikes identified with it were tested for reliability.

STEP 1: Data preparation and segmentation

The first step in our procedure (Fig. 2C) was to prepare the data for clustering of events from different neural units, by separating firing events from noise, and segmenting spatio-temporally distinct regions of spiking activity on the electrode array.

Data from the array were sampled at 10 KHz, high-pass filtered below 200 Hz with a finite impulse response filter to remove low frequency baseline fluctuations, and then packaged into 3.2ms clips: (a) "noise clips" in which the potential never fell below -30μV, and (b) "spike events" surrounding moments at which the potential crossed -40μV. The duration of the clip was chosen to include the full extent of most spike events with a small margin on either side. The threshold was set such that there were no discernible spikes among the subthreshold voltage deflections, although this low-amplitude noise likely contained contributions

from spikes of distant neurons (see Text S1 Sect. G2). Clips with potentials between -30μV and -40μV were neither used to characterize noise (since they might contain small spikes) nor used to identify spikes (since they were very noisy). The threshold for spikes was set to ~ 4 times the standard deviation of the potential in the noise clips. Each spike event thus consisted of $N = 3.2\text{ms} \times 10\text{kHz} \times 5 \times 6 = 960$ numbers, the potentials on a $32 \times 5 \times 6$ grid of space-time pixels ("stixels"). Spike events sometimes overlapped each other, for example if a burst of spikes lasted longer than 3.2ms. Cluster identification and spike template building (Step 2) used four 30-second segments sampled from different times, but subsequent spike fitting and sorting (Step 3) used all the data.

Electrodes can share signals because of instrumental cross-talk and because the activity of neurons spreads passively to nearby electrodes. We modeled both effects by a linear filter that spatially blurs signals. To find this filter empirically, we noted that it also applies to the noise. Accordingly, we measured the spatial covariance of noise clips, finding that it was spatially isotropic and had a roughly exponential falloff, with a correlation length of $\sim 30\mu$m. We applied the square root of the inverse of this covariance matrix to all data, and used the resulting "spatially whitened" data for all analysis. In some datasets this transformation sharpened the individual spikes spatially, improving our ability to distinguish them in the clustering stage. In other datasets the transformation had little effect. Our data also exhibited temporal correlations, but these have a different physical origin from the essentially instantaneous passive spatial spread. We found that *temporal* whitening prior to clustering [8] worsened our signal/noise ratio and impeded cluster determination. Thus we incorporated temporal correlations later, during the spike fitting.

Segmentation. Each spike event is a superposition of spikes from an unknown number of distinct neural units with stereotyped waveforms that we sought to identify. We first spatially segmented the data to isolate waveforms from individual units and their immediate neighbors. To this end, we identified all stixels at which the potential was more negative than the threshold of -40μV and divided this set into connected components (two stixels were considered connected if they were nearest neighbors in either time or space). Within each connected patch we identified the absolute peak electrode and time, then extracted a 3.2ms region centered temporally on the peak time and cropped spatially to a neighborhood of nine channels surrounding (and including) the leader electrode. Thus each spike event was segmented into one or more cropped events; each of which was then classified according to its leader electrode. A similar segmentation method has recently been applied to the spike identification problem by J. Schulman (unpublished); see http://caton.googlecode.com. In subsequent clustering, only those events having the same leader electrode were directly compared to each other [9].

Some cropped events might be composites of two spike types corresponding to neighboring, but distinct, neural units. However, this step at least decomposes composite events whose components are well separated in space or in time, and hence reduces the combinatorial burden inherent in large arrays; later steps handle composites missed at this stage. The method also ensures that, if spikes from two well-separated units frequently co-occur, the two units will nevertheless be correctly handled as separate.

STEP 2: Cluster identification and template building

The second step in our procedure (Fig. 2C) was to cluster spiking events in a subset of the data (four 30-second segments) into groups that had similar waveforms and thus probably came from the same neural unit. For each cluster, we produced a template waveform describing the typical spike, and determined the distribution of amplitude rescalings that best matched spikes to this template.

Cluster identification. In order to group events into clusters based on the similarity of their waveforms, some previous approaches have sought a low-dimensional set of discriminable waveform "features," and have assumed that variability between events in the same cluster arises only from additive noise. In practice, systematic variation in the shape of spikes from single units is often observed that is not well described by additive noise. Furthermore, identifying the correct set of salient waveform features that discriminate between units is challenging ([20]; see Fig. 2B). Thus, seeking a technique that did not require feature extraction, we adapted the optics algorithm [11]. Briefly, optics computes distances between all pairs of waveforms, then orders the waveforms such that similar ones are placed close together in a single linear sequence. optics makes no assumption that clusters have a Gaussian distribution in feature space, nor does it set any threshold density in that space to trigger cluster identification. The linear ordering allows for easy visualization and cutting of clusters.

We applied this algorithm to cropped and segmented spike events (obtained in Step 1) which were upsampled by a factor of 5 (using matlab's cubic spline interpolation) and then temporally aligned to place the absolute peak of the waveform at a common position before downsampling again. The interpolation was necessary to compensate for apparent variations in spike waveforms due to discrete sampling [21]. To reduce the fuzziness of the clusters, we masked spike events by setting voltage samples to zero if they were less negative than $-15\mu V$. As a distance metric between $\mathbf{V}$ and $\mathbf{V}'$, the masked potentials of spike events, we chose $d(\mathbf{V},\mathbf{V}') = \left(\sum_{i=1}^{N} [(V_i - V'_i)^2/(k\sqrt{|V_i| + |V'_i|})]\right)^{1/2}$ where i indexes the potentials at each channel and timepoint, and k is the total number of nonzero potentials after masking of either $\mathbf{V}$ or $\mathbf{V}'$. Division by k normalized for the effective dimensionality (given by the number of dimensions containing nonzero entries). We observed that higher voltage traces tended to have a higher variance; the factor $\sqrt{|V_i| + |V'_i|}$ partially compensated for this, leading to more homogeneous clusters.

We constructed a graphical user interface (gui) that allowed a human operator to visualize each spike event in the optics sequence as a vertical column of pixels color-coded by voltage (see Fig. 2D). Transitions between distinct spike types were usually obvious to the operator, who could quickly find and select bands corresponding to each spike type. (For the data in this paper, the operator found over 100 such clusters in about 30 minutes.) The software then wrote the corresponding cropped events to a set of data files. The ease of separation likely occurred because clusters could already be fairly well delineated with just the "geographical" features in Fig. 2A.

Up to this point, the events being clustered were still segregated into batches according to their leader channel x_0, y_0. Thus it was possible for a single unit to be multiply identified: If it stimulated two neighboring electrodes nearly equally, the unit could generate events in both of the corresponding batches. We tested for duplicates by manually examining pairs of clusters whose medians had a large cross-correlation and merged the clusters if necessary. There was also a possibility that the initial clustering would assign multiple units to one cluster. In these cases, visual examination of the superposed waveforms of the cluster often showed it to be a composite of multiple units. This was resolved by doing a principal components analysis on the waveforms in that cluster: if the cluster was composite, at least one of the first few principal component weights had a multimodal histogram. The cluster was split by thresholding at the valleys of the histogram; we then tested

whether any of the split components ought to be merged with an existing cluster. We developed a graphical user interface to assist the operator in performing these merging and splitting steps.

Generally it was clear to the human operator when a band in the gui output was clean enough and wide enough (contained enough events) to generate a good cluster; thus there was no need to specify *a priori* the desired number of clusters, an advantage over many automated clustering procedures. Marginally significant clusters were either eliminated during template building (see below), or else generated fits that were themselves discarded during spike fitting (Step 3) and evaluation of template reliability (Step 4). Any significant clusters *missed* at this stage, for example because of the small fraction of the data used in this step, were found and reincorporated later during spike fitting (Step 3).

Template building. Next we created a consensus waveform ("template") summarizing each cluster of cropped, upsampled events, and characterized meaningful deviations from that consensus. We created a draft template by finding the pointwise median over all events in a cluster, then aligned each event to the draft template by maximizing their cross-correlation over time shifts, which we found to be more accurate than aligning to each event's peak time. Finally, we found the pointwise median (to suppress the effects of outliers) of the aligned events; this waveform was our template (Fig. 5B).

A key step was to realize that, in our data, the most significant sources of variation of individual spikes from the template were (a) additive noise, and (b) overall multiplicative rescaling of the spike's amplitude (Fig. 5C). To quantify (b), we found the rescaling factor that optimized the overlap of each spike with its template, then stored the mean and variance of those factors in a lookup table for later use as a prior probability for amplitude variation. We also logged the number of events associated to each template, converted to an approximate firing rate, and saved those rates, again for later use as a prior.

STEP 3: Spike fitting

The third step in our procedure was to fit the spike templates constructed in Step 2 to each firing event in the data in order to determine which neural units were responsible for the activity. To this end, we constructed a simple generative model of firing events, and included distributions of firing rate and of amplitude variation for each template. The firing probabilities and amplitude distributions were inferred from the previously clustered data, and therefore served as priors in the template fitting procedure described below. The fitting procedure iteratively identified and subtracted the most likely templates in each firing event.

The cluster templates were produced using an upsampled 50kHz sample rate, but for fitting to data we downsampled back to the actual 10kHz, in each of 5 "reading frames"; that is, we created five versions of each template corresponding to subsample shifts. Let $F_{\mu;x,y}(t)$ be the potential of template μ, on the electrode with address x,y, at time t, with time measured in units of the sampling time $\delta t = 0.1$ms, and the template peak at the central point $t = 16$ within the 3.2ms template frame. We use the vector notation $\mathbf{F}_{\mu_i t_i}$ for the template μ_i shifted to time t_i, i.e. its x,y,t component is $F_{\mu_i,x,y}(t-t_i)$.

Generative model. The goal of spike fitting is to identify, for each spike event, all the units $\{\mu_i\}$ which contribute to the event and their firing times $\{t_i\}$ irrespective of their amplitudes $\{A_i\}$. Thus we assumed a probabilistic generative model of the data [8,21–23] and computed the posterior probability of $\{(\mu_i,t_i)\}$ given the observed data. We assumed that a spike event $\mathbf{V}$ could be explained by a linear combination of templates $\mathbf{F}_{\mu_i t_i}$ with variable amplitudes A_i and correlated, zero-mean Gaussian noise $\delta\mathbf{V}$:

$$\mathbf{V} = \sum_{i=1}^{K} A_i \mathbf{F}_{\mu_i t_i} + \delta\mathbf{V}. \quad (1)$$

Here K is the (a priori unknown) number of units contributing to the event. Given this model, to obtain the posterior probability that a firing event consists of a particular set of templates, we need to specify the prior probability of μ, t, and A. We chose a Gaussian prior for the amplitude A, a Poisson prior for μ, and a uniform prior for t. Although A is a strictly positive quantity, we modeled its distribution with a Gaussian for analytical tractability. In practice, the distribution of A was tightly concentrated around its mean of approximately 1 and the Gaussian approximation had negligible weight at negative values (Fig. 6D).

Our generative model assumes that spike waveforms from a given neural unit are stereotyped, apart from their amplitude. We did observe considerable variation in spike amplitude (Fig. 2A), in part due to bursting [10,12], and thus included it in the model as a distribution of amplitude rescaling factors. Allowing for the possibility of slight variations in spike width (Text S1 Sect. F) also slightly improved our results. But there was little additional

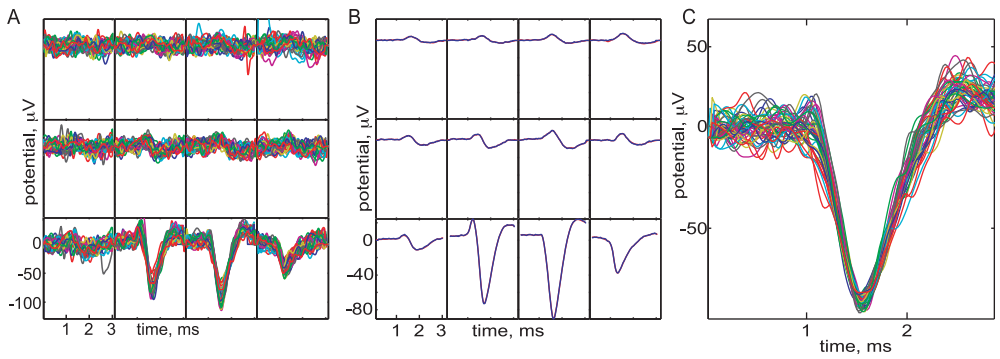

Figure 5. Template building. (*A*) Detail of 40 of the aligned events used to compute a template, upsampled and shifted into alignment as described in the text. Some outlier traces reflect events in which this unit fired together with some other unit; the unwanted peaks occur at random times relative to the one of interest, and thus do not affect the template. (*B*) *Blue*, detail of a template waveform, showing the potential on 12 neighboring electrodes. Time in ms runs horizontally; the vertical axis is potential in μV. *Red*, for comparison, the pointwise mean of the 430 waveforms used to find this template (nearly indistinguishable from the blue curve). (*C*) Detail of (*A*), showing only the leader channel. In addition, each trace has been rescaled by a constant to emphasize their similarity apart from variation in overall amplitude.
doi:10.1371/journal.pone.0019884.g005

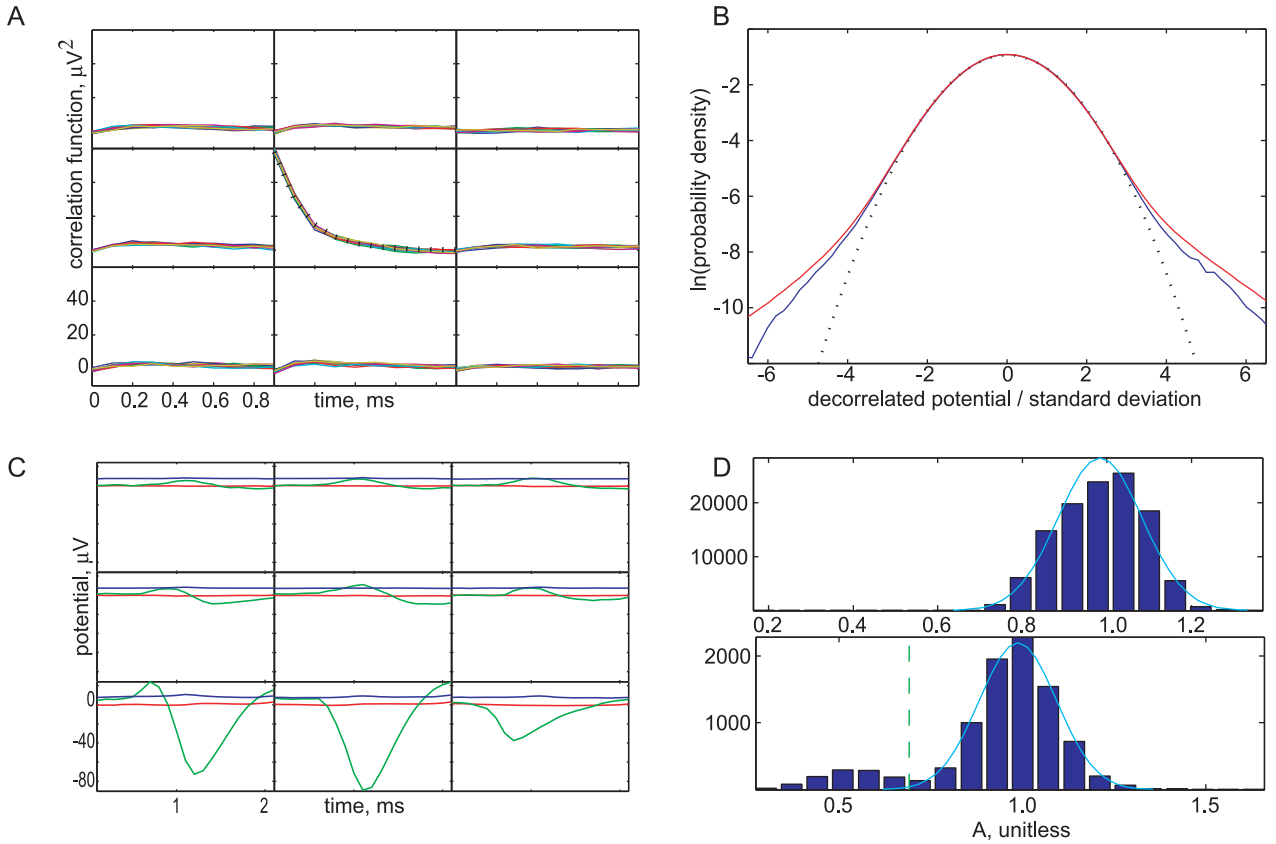

Figure 6. After fitting spikes, only noise remains. (*A*) Noise covariance after spatial whitening. Subpanels: spacetime covariance $C(x,y,t;x_*,y_*,t+\Delta t)$ between the central channel and its neighbors as a function of Δt, for various fixed t (colored curves). Central panel (dotted line): the function $(57\mu V^2)\exp(-\Delta t/(0.18\text{ms})$. (The various t lines and the dotted line are too similar to discriminate visually.) Horizontal axes: Δt in ms; Vertical axes: C in μV^2. (*B*) *Blue curve*, Semilog plot of the one point marginal probability density function of decorrelated noise samples. *Red curve*, same quantity, evaluated on residuals after spikes have been removed from spike events. *Dotted curve*, The Gaussian chosen to represent this distribution. (*C*) *Green*, detail of the same template waveform shown in Fig. 5. *Red*, pointwise mean of the residuals after the fit spike is subtracted from 4,906 one-spike events of this type is nearly flat. This validates our assumption that spikes vary only in overall amplitude, and that noise is independent of spiking. *Blue*, pointwise standard deviation of the residuals, again evidence that only noise remains after fitting and subtracting spikes. (*D*, *top*) Histogram of fit values of the scale factor A for a template with peak amplitude $-168\mu V$ (well above noise) obtained without a prior on A, superposed with a Gaussian of the same mean and variance. (*D*, *bottom*) Similar histogram for a low amplitude template. A secondary bump appears, due to noise-fits, but is well separated from the main peak; a cutoff is shown as a dashed green line. The superposed Gaussian has mean and variance computed from the part of the empirical distribution lying above the cutoff.
doi:10.1371/journal.pone.0019884.g006

variability to be modelled (Fig. 5C). Our model also assumes that signals from different units combine linearly, as does the noise. This is reasonable, because the biophysics of extracellular recording is governed by the equations of electrodynamics, Ohm's law, and other linear relations. A third assumption is that noise and the variability of spike amplitude are each well described by Gaussian distributions. Assuming Gaussianity (well-confirmed in some settings [8], but not others [24]) allows for a fast, partially analytic approach to fitting. We validate this assumption quantitatively below.

Our generative model has a Poisson prior probability for firing by each neural unit, i.e. a prior that is as unstructured as possible while being consistent with the mean firing rate. The prior probability could be made somewhat more accurate by including refractory periods, the likelihood of bursting, and correlations between neural units. But this would significantly increase the complexity of the model, and inferring the distribution would require much more data [25].

Finally, we assumed that all statistical distributions that enter into the model are stationary and independent of the stimulus. While our retinal preparation does not suffer electrode drift (as might implanted electrodes), there are occasionally shifts in spike amplitudes and firing rates over the course of a lengthy experiment. Although in principle our fixed priors could lead to biased estimates, these biases are small when spike identification is robust, i.e. when the likelihood function dominates the prior in the posterior probability of a neural unit [26].

Noise characterization. In the context of our generative model, in order to assess the probability that the residual after subtracting a putative spike is indeed noise, we first need to measure the distribution of noise. After applying the spatial whitening filter (described in Step 1), our noise clips are decorrelated in space, but not in time (Fig. 6A). Assuming that the noise has a correlated Gaussian distribution, we need the inverse of the noise covariance matrix, C^{-1}. One approach to finding C^{-1} is to invert the empirical covariance matrix C of a large set of noise clips. Besides being intractable for larger arrays, this approach has the disadvantage that a numerically stable evaluation requires a very large noise sample.

For these reasons we instead took a parametric approach. After evaluating the covariance $C(x,y,t;x',y',t')$ we noted that it was

approximately diagonal and translation-invariant in space (i.e. proportional to $\delta_{x,x'}\delta_{y,y'}$ and independent of x and y). It was also approximately stationary, i.e. invariant under time shifts $t\to t+\Delta t$, $t'\to t'+\Delta t$, and thus only depended on $t-t'$. Finally, we observed that the time dependence of $\mathbf{C}$ was roughly an exponential, $\approx\eta e^{-|t-t'|/\tau}$ (Fig. 6A). This gave:

$$\mathbf{C}^{-1}(x,y,t;x',y',t') = \eta^{-1}\times\delta_{x,x'}\delta_{y,y'}\begin{cases}(1+\xi^2)/(1-\xi^2), & \text{if } t=t'\\ -\xi/(1-\xi^2), & \text{if } t=t'\pm\delta t\\ 0, & \text{otherwise.}\end{cases} \quad (2)$$

Here $\delta t=0.1\text{ms}$ is the sample time, and $\delta_{x,x'}$ is the Kronecker symbol. η and $\xi=e^{-\delta t/\tau}$ are obtained from the noise covariance. The dataset used in Results yields noise strength $\eta\approx 57\mu\text{V}^2$ and $\xi\approx 0.58$.

By construction, our noise model reproduces the 2-point correlations in the noise clips. However, real noise may not be Gaussian distributed. One check on this is to construct the transformed quantities $\mathbf{U}=\mathbf{C}^{-1/2}\mathbf{V}$ empirically, find their full distribution as $\mathbf{V}$ ranges over noise clips (the "one-point marginal" distribution), and compare to a normal distribution. Fig. 6B shows this comparison, lumping together every element of $\mathbf{U}$. The empirical noise deviates from a Gaussian only in the far tails that contain very little weight.

Fitting algorithm for single spikes. Given the above characterization of the noise distribution, and our Gaussian prior for spike amplitude variation, the generative model Eqn. 1 defined the posterior probability $P(\{\mu_i,t_i,A_i\}|\mathbf{V})$ for templates $\{\mu_i\}$ to be present at times $\{t_i\}$ with amplitude scale factors $\{A_i\}$, given the recorded potentials $\mathbf{V}$. We ideally would have marginalized $P(\{\mu_i,t_i,A_i\}|\mathbf{V})$ over the nuisance parameters $\{A_i\}$ and then maximized with respect to $\{(\mu_i,t_i)\}$ to identify the most probable set of units and spike times. In practice, this maximization is computationally expensive to perform on many templates simultaneously. Instead, we used a greedy approximation which fit one template at a time.

We first assumed that the event contained exactly one spike and identified the spike's type μ and time of occurrence t. Bayes' formula gives for the posterior probability:

$$P(\mu,t,A|\mathbf{V})\,\mathrm{d}t\,\mathrm{d}A\propto P(\mathbf{V}|\mu,t,A)P(\mu,t,A)\,\mathrm{d}t\,\mathrm{d}A, \quad (3)$$

up to a constant independent of μ, t, and A. Here $P(\mu,t,A)$ is the prior probability of the template μ appearing at time t with an amplitude A:

$$P(\mu,t,A)\,\mathrm{d}t\,\mathrm{d}A = r_\mu\,\mathrm{d}t(2\pi\sigma_\mu{}^2)^{-1/2}\exp\left(-(A-\gamma_\mu)^2/2\sigma_\mu^2\right)\mathrm{d}A, \quad (4)$$

where γ_μ is the mean and σ_μ^2 the variance of the scale factor for cluster μ; r_μ is the estimated overall rate of firing for this cluster. The generative model gave the probability of the observed potential $\mathbf{V}$ given μ,t,A (the likelihood) as $P(\mathbf{V}|\mu,t,A)=P_{\text{noise}}(\mathbf{V}-A\mathbf{F}_{\mu,t})$, where P_{noise} is a Gaussian distribution with zero mean and covariance $\mathbf{C}$ (Eqn. 2). Combining the likelihood and prior, then integrating out A, gave the formula we ultimately used in our fitting algorithm:

$$\frac{P(\mu,t|\mathbf{V})}{P(\text{no spike}|\mathbf{V})} = \frac{r_\mu\,\mathrm{d}t}{\sqrt{1+\sigma_\mu^2\mathbf{F}_{\mu,t}^{\mathrm{t}}\mathbf{C}^{-1}\mathbf{F}_{\mu,t}}}\exp\left(\frac{1}{2\sigma_\mu^2}\frac{(\gamma_\mu+\sigma_\mu^2\mathbf{V}^{\mathrm{t}}\mathbf{C}^{-1}\mathbf{F}_{\mu,t})^2}{1+\sigma_\mu^2\mathbf{F}_{\mu,t}^{\mathrm{t}}\mathbf{C}^{-1}\mathbf{F}_{\mu,t}}-\frac{\gamma_\mu^2}{2\sigma_\mu^2}\right). \quad (5)$$

We have expressed Eqn. 5 as a ratio with the probability that no templates contributed to the event. Finally, we varied μ and t to find the template and time which maximized Eqn. 5. This gave the first fit template, μ_*, its firing time, t_*, and the probability ratio of the fit.

We improved scalability by a slight approximation. Starting from a spike event, we first identified the time and electrode address of its absolute peak and restricted the matrix products in expression Eqn. 5 to only sum over a spatiotemporal neighborhood surrounding this peak. The size of the neighborhood was chosen to match the typical spatial extent and temporal duration of the templates.

Multiple spikes. In principle, we could have extended the single template procedure described above to compare the probabilities of all possible combinations of two or more spikes. Such an exhaustive approach, however, would quickly have become impractical. We instead noted that, even if an event contains multiple spikes, the single-spike fit described above still identified that template whose removal would lead to the largest increase in the probability that the remaining waveform is noise. Thus we adopted an iterative (matching-pursuit or "greedy") approach: starting with a spike event, we found the absolute peak, fit it by the method described above, subtracted the fit, and then repeated the process [7].

The single-spike procedure found the most probable spike type μ_*; we then computed the scale factor A and spike time t' that would allow the fit spike to be subtracted as fully as possible. We thus held μ fixed to μ_* and minimized the ordinary norm $\|\mathbf{V}-A\mathbf{F}_{\mu_*,t'}\|^2$ over A and t'. The scaled and shifted template obtained in this way was subtracted before repeating the single-spike fitting procedure. The parameter t' was only used for template subtraction. The spike time which was actually reported by the algorithm was not t', but t_* described above. In practice, t' was always very close to t_*.

To determine when to stop fitting spikes, we adopted a likelihood ratio test. At each step of the fitting loop, we summed Eqn. 5 over t, obtaining the probability ratio that an additional spike of type μ is present relative to the probability that *no* additional spike was present. We can then say that fitting an additional spike is justified if the ratio exceeds unity for some μ_*. The fitting loop terminates when the significance test fails. Note that the factor $r_\mu\mathrm{d}t$ (the prior probability of a spike from unit μ) in Eqn. 5 is typically much smaller than one. In order for the fitting algorithm to accept a candidate spike, the remaining terms in Eqn. 5, which relate to goodness-of-fit, must be large enough to overcome this small factor. Furthermore, the amplitude prior is typically concentrated tightly around its mean. The marginalization over amplitudes thus suppresses templates that would have to stretch by a large factor in order to fit the data. Both of these effects counteract overfitting. Fig. 4C,D shows an example of the successful decomposition of a multiple-spike event using our method.

Second pass. The spike fitting algorithm might exit prematurely if a spike is present that does not appear in the list of templates initially extracted from the small subset of data. In this case, the fit will terminate, even though other identifiable spikes of

lower amplitude may remain. To check this, if the residual exceeds $V^*_{\text{trust}} = -44\mu\text{V}$ after termination, the code declares an "incomplete fit" and writes the residual to a file; the small set of resulting waveforms were then reintroduced into our clustering code and used for a second round of fitting. In this way we can be assured of finding even rare spike types, without having to perform clustering on the complete dataset. Using this method, only 0.02% of fits in the second pass were classified as incomplete. It can also happen that the small data sample used for clustering gives a poor estimate of some firing rates and amplitude distributions that enter our priors for spike fitting. Thus, before the second pass of fitting the priors are updated based on the outcome of the first pass.

STEP 4: Evaluation of template reliability

After spike identification, we performed a final evaluation to test whether templates and their sorted spike trains were trustworthy. The primary criteria were: (1) residuals after spike removal should resemble noise, (2) the histogram of amplitude scale factors should be unimodal, (3) the inter-spike interval (ISI) distributions should display "refractory holes"; and the cross-correlation functions should not. Additional criteria are described in Text S1 Sect. H. Reliable templates were taken to be those that passed all these tests. Most unreliable templates failed multiple tests.

Residuals. For single-spike events, the residual signal after subtracting the fit should resemble pure noise; in particular it should be stationary in time and translation-invariant in space. Fig. 6C shows that these expectations were met, validating our assumptions. For example, if the unit in Fig. 6C had significant variations other than amplitude rescaling, or if there had been an amplitude-dependent noise process, then we would expect significant non-stationarity in the residual curves [21]. To test that, after termination, the residual of a spike event consists only of noise, we computed the one-point marginal distribution of waveforms after all known spikes had been removed. Fig. 6B shows that this distribution closely resembled that of noise clips, indicating that our code indeed found the significant spikes. Of particular note, the standard deviation of the residuals matched that of the noise: For each template, we found the standard deviation of the residuals of events to which only that template was fit. This value ranged from $6.92\mu\text{V}$ to $8.63\mu\text{V}$, while the noise standard deviation was $7.57\mu\text{V}$.

Amplitude. For large amplitude templates, the distribution of amplitude scale factors (A) obtained during spike fitting was typically close to Gaussian (Fig. 6D). On the other hand, for low amplitude templates, accidental noise fits can sometimes lead to a histogram of A values with a secondary, low-amplitude peak well separated from the expected peak near $A \approx 1$ (Fig. 6D). Examining the A histograms allowed us to quickly set an individual threshold for each reliable template. Fit spikes with A value below this threshold were discarded. If two peaks were discernible but overlapped significantly, the entire cluster was deemed unreliable and its spikes were not used in further analysis. In addition, our trigger rejected any spike event that did not cross $-40\mu\text{V}$, thus any cluster whose A histogram extended closer to zero than this was probably missing some true spikes, and was not used.

ISI distribution and cross-correlation. Interspike interval distributions for single units are expected to have a refractory hole; our analysis of these distributions was described in Results. Two distinct neural units need not respect any mutual refractory period. Their spike-time cross-correlation function is therefore *not* expected to display any hole. We looked for such unexpected behavior and, when found, reexamined the corresponding templates. If the templates appeared to be duplicates, we merged the corresponding spike trains [7,9]. Another diagnostic for duplicate templates is a coincident receptive field.

Discussion

A review of early work on spike identification, prior to the widespread use of MEAs, can be found in [27]. Like some earlier work, our method separates spike identification into distinct steps of clustering and fitting. The clustering step uses all the waveform features, and makes no assumption about the cluster structure (e.g., that it is a mixture of Gaussian distributions). The fitting step acknowledges that each neural unit's signals are subject to intrinsic, multiplicative variation as well as additive noise, and uses a Bayesian approach to infer the identity of the most likely firing unit.

Our approach is intentionally not fully automatic, since human proof-reading of the results of automated clustering is generally essential. However, we have been careful to use human judgement only where it is indispensable. Further, both the human and machine steps are organized so as to scale well with array area (or number of units monitored). For example, cluster cutting was greatly simplified by representing spikes in an ordered one-dimensional array. This feature, along with systematic exploitation of the spatio-temporal locality of spikes (Fig. 2A), and the use of a simple but powerful generative model, make our method scalable to large arrays. Furthermore, we observed that our more ambiguous templates tended to be located on the boundary of the array due to recording of units located some distance from the electrodes. These "boundary effects" should become less important for larger arrays; we thus anticipate that the methods described in this paper will yield more accurate spike sorting for large, dense arrays.

Our method can be extended in many ways. For example, it would be straightforward to update the priors continually as fitting proceeds, allowing non-stationarity and stimulus dependence to be handled more gracefully. In some applications it may be preferable to report spike identification probabilistically, rather than just listing the most-likely spike events; our formulas already provide this information. The method can also be extended to non-planar arrangements of electrodes and neural tissues, for, e.g., cortical applications. Finally, the generative model in the present paper does not take into account correlations within and between spike trains, or the receptive field structure and stimulus dependence of responses. Performance could be further improved on complex overlapping spike events via a bootstrapping procedure. We could first use the simple independent, Poisson generative model of this article to produce an accurate preliminary assignment of spikes to units. From this assignment we could construct a more detailed model of correlated activity with pairwise interactions (e.g. [25] or the stimulus-dependent models [28,29]). This more complex generative model could then be used to further refine spike assignments for applications requiring a very high degree of accuracy.

Supporting Information

Text S1 Supplementary appendix describing additional methodological details.
(PDF)

Acknowledgments

Michael Berry and Olivier Marre have developed an algorithm similar to, but different from, ours (manuscript in preparation). We thank them for discussions of their work, and specifically thank Olivier Marre for

suggesting to us that the most complete subtraction of a spike can be obtained by refitting the spike without a prior.

Author Contributions

Conceived and designed the experiments: JSP JH KDS GT VB PCN. Performed the experiments: JSP KDS. Analyzed the data: JSP JH KDS PCN. Contributed reagents/materials/analysis tools: JSP JH PCN. Wrote the paper: JSP VB PCN.

References

1. Sterlng P (2004) How retinal circuits optimize the transfer of visual information. In: Chalupa L, Werner J, eds. The Visual Neurosciences.
2. Meister M, Pine J, Baylor DA (1994) Multi-neuronal signals from the retina: Acquisition and analysis. J Neurosci Methods 51: 95–106.
3. Devries S, Baylor D (1997) Mosaic arrangement of ganglion cell receptive fields in rabbit retina. J Neurophysiol 78: 2048.
4. Buzsáki G (2004) Large-scale recording of neuronal ensembles. Nature Neuroscience 7: 446–51.
5. Gray CM, Maldonado PE, Wilson M, McNaughton B (1995) Tetrodes markedly improve the reliability and yield of multiple single-unit isolation from multi-unit recordings in cat striate cortex. J Neurosci Methods 63: 43–54.
6. Mallat SG, Zhang ZF (1993) Matching pursuits with time-frequency dictionaries. IEEE Transactions on Signal Processing 41: 3397–3415.
7. Segev R, Goodhouse J, Puchalla J, Berry MJ (2004) Recording spikes from a large fraction of the ganglion cells in a retinal patch. Nature Neuroscience 7: 1154–61.
8. Pouzat C, Mazor O, Laurent G (2002) Using noise signature to optimize spike-sorting and to assess neuronal classification quality. J Neurosci Methods 122: 43–57.
9. Litke AM, Bezayiff N, Chichilnisky EJ, Cunningham W, Dabrowski W, et al. (2004) What does the eye tell the brain?: Development of a system for the large scale recording of retinal output activity. IEEE Trans Nucl Sci 51: 1434–1439.
10. Fee MS, Mitra P, Kleinfeld D (1996) Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-gaussian variability. J Neurosci Methods 69: 175–188.
11. Ankerst M, Breunig MM, Kriegel HP, Sander J (1999) OPTICS: Ordering points to identify the clustering structure. SIGMOD Rec 28: 49–60.
12. Harris KD, Hirase H, Leinekugel X, Henze DA, Buzsáki G (2001) Temporal interaction between single spikes and complex spike bursts in hippocampal pyramidal cells. Neuron 32: 141–9.
13. Ratliff C, Borghuis B, Kao Y, Sterling P, Balasubramanian V (2010) Retina is structured to process an excess of darkness in natural scenes. Proceedings of the National Academy of Sciences 107: 17368.
14. Chichilnisky E, Kalmar R (2002) Functional asymmetries in ON and OFF ganglion cells of primate retina. Journal of Neuroscience 22: 2737.
15. Peichl L, Buhl E, Boycott B (1987) Alpha ganglion cells in the rabbit retina. The Journal of Comparative Neurology 263: 25–41.
16. Peichl L (1989) Alpha and delta ganglion cells in the rat retina. The Journal of Comparative Neurology 286: 120–139.
17. Dacey D, Petersen M (1992) Dendritic field size and morphology of midget and parasol ganglion cells of the human retina. Proceedings of the National Academy of Sciences of the United States of America 89: 9666.
18. Tauchi M, Morigiwa K, Fukuda Y (1992) Morphological comparisons between outer and inner ramifying alpha cells of the albino rat retina. Experimental brain research 88: 67–77.
19. Do-Nascimento JLM, Do-Nascimento RSV, Damasceno BA, Silveira LCL (1991) The neurons of the retinal ganglion cell layer of the guinea pig: Quantitative analysis of their distribution and size. Brazilian J Med Biol Res 24: 199–214.
20. Quiroga RQ (2007) Spike sorting. Scholarpedia 2: 3583.
21. Lewicki MS (1994) Bayesian modeling and classification of neural signals. Neural computation 6: 1005–1030.
22. Atiya A (1992) Recognition of multiunit neural signals. IEEE Transactions on Biomedical Engineering 39: 723–729.
23. Sahani M (1999) Latent variable models for neural data analysis. Ph.D. thesis, California Institute of Technology, Pasadena CA.
24. Shoham S, Fellows MR, Normann RA (2003) Robust, automatic spike sorting using mixtures of multivariate t-distributions. J Neurosci Methods 127: 111–22.
25. Schneidman E, Berry MJ, Segev R, Bialek W (2006) Weak pairwise correlations imply strongly correlated network states in a neural population. Nature 440: 1007–12.
26. Ventura V (2009) Traditional waveform based spike sorting yields biased rate code estimates. Proc Natl Acad Sci USA 106: 6921–6.
27. Lewicki MS (1998) A review of methods for spike sorting: The detection and classification of neural action potentials. Network (Bristol, England) 9: R53–78.
28. Pillow JW, Shlens J, Paninski L, Sher A, Litke AM, et al. (2008) Spatio-temporal correlations and visual signalling in a complete neuronal population. Nature 454: 995–999.
29. Tkaficik G, Prentice J, Balasubramanian V, Schneidman E (2010) Optimal population coding by noisy spiking neurons. Proceedings of the National Academy of Sciences 107: 14419.