

Simultaneous Denoising, Deconvolution, and Demixing of Calcium Imaging Data

Highlights

- We present a new method for analyzing large-scale calcium imaging datasets
- The method identifies the cell locations and deconvolves their neural activity
- Applications to *in vivo* somatic and dendritic imaging are presented
- We make available MATLAB and Python implementations of our method

Authors

Eftychios A. Pnevmatikakis,
Daniel Soudry, Yuanjun Gao, ...,
Darcy S. Peterka, Rafael Yuste,
Liam Paninski

Correspondence

epnevmatikakis@simonsfoundation.org
(E.A.P.),
liam@stat.columbia.edu (L.P.)

In Brief

Advances in calcium imaging pose significant statistical analysis challenges. Pnevmatikakis et al. present a method for identifying and spatially demixing imaged neural components and deconvolving their activity from the indicator dynamics. The method is applied to a variety of datasets.

Simultaneous Denoising, Deconvolution, and Demixing of Calcium Imaging Data

Eftychios A. Pnevmatikakis,^{1,2,*} Daniel Soudry,² Yuanjun Gao,² Timothy A. Machado,^{2,3,4,5} Josh Merel,^{2,4} David Pfau,^{2,4} Thomas Reardon,^{3,4,5} Yu Mu,⁶ Clay Lacefield,⁴ Weijian Yang,⁷ Misha Ahrens,⁶ Randy Bruno,⁴ Thomas M. Jessell,^{3,4,5} Darcy S. Peterka,^{5,7} Rafael Yuste,^{4,7} and Liam Paninski^{2,4,5,7,*}

¹Center for Computational Biology, Simons Foundation, New York, NY 10010, USA

²Department of Statistics, Center for Theoretical Neuroscience, and Grossman Center for the Statistics of Mind, Columbia University, New York, NY 10027, USA

³Department of Biochemistry and Molecular Biophysics and Howard Hughes Medical Institute, Columbia University, New York, NY 10032, USA

⁴Department of Neuroscience and Kavli Institute of Brain Science, Columbia University, New York, NY 10032, USA

⁵Zuckerman Mind Brain Behavior Institute, Columbia University, New York, NY 10032, USA

⁶Howard Hughes Medical Institute, Janelia Research Campus, Ashburn, VA 20147, USA

⁷Neurotechnology Center, Department of Biological Sciences, Columbia University, New York, NY 10027, USA

*Correspondence: epnevmatikakis@simonsfoundation.org (E.A.P.), liam@stat.columbia.edu (L.P.)

<http://dx.doi.org/10.1016/j.neuron.2015.11.037>

SUMMARY

We present a modular approach for analyzing calcium imaging recordings of large neuronal ensembles. Our goal is to simultaneously identify the locations of the neurons, demix spatially overlapping components, and denoise and deconvolve the spiking activity from the slow dynamics of the calcium indicator. Our approach relies on a constrained nonnegative matrix factorization that expresses the spatiotemporal fluorescence activity as the product of a spatial matrix that encodes the spatial footprint of each neuron in the optical field and a temporal matrix that characterizes the calcium concentration of each neuron over time. This framework is combined with a novel constrained deconvolution approach that extracts estimates of neural activity from fluorescence traces, to create a spatiotemporal processing algorithm that requires minimal parameter tuning. We demonstrate the general applicability of our method by applying it to *in vitro* and *in vivo* multi-neuronal imaging data, whole-brain light-sheet imaging data, and dendritic imaging data.

INTRODUCTION

Calcium imaging is becoming a standard tool for monitoring large neuron populations. Recent technical advances have enabled complete imaging of small cortical volumes (e.g., Cotton et al., 2013), and whole-brain imaging of small animals (Ahrens et al., 2013; Prevedel et al., 2014) at reasonable imaging rates. In parallel, engineering of genetically encoded calcium indicators offers increasingly sensitive indicators that can reliably detect action potentials *in vivo* (Chen et al., 2013). From a statistical perspective, these developments pose significant challenges,

which can be condensed to three major problems: (1) identifying the spatial footprint of each neuron in the optical field, (2) demixing spatially overlapping neurons (where overlap is due either to the projection of a 3D volume onto a 2D imaging plane or to insufficient spatial resolution in 3D imaging methods), and (3) deconvolving the spiking activity of each neuron from the much slower dynamics of the calcium indicator. These tasks become harder due to the existence of measurement noise, unaccounted neural processes and/or neuropil activity, and limitations in imaging rate.

Traditionally, these three problems have been dealt with separately. Calcium deconvolution methods have focused largely on analyzing one-dimensional fluorescence time series data from one neuron at a time. Such methods include fast nonnegative deconvolution (Vogelstein et al., 2010), greedy algorithms (Grewe et al., 2010), finite rate of innovation methods (Oñativia et al., 2013), supervised learning (Theis et al., 2015), as well as particle filtering (Vogelstein et al., 2009) and Markov chain Monte Carlo (MCMC) methods (Pnevmatikakis et al., 2013). Although effective in the analysis of single fluorescence traces, these methods do not take full advantage of the spatiotemporal structure in the data, and in some cases require either data with available ground truth and/or significant parameter tuning.

Solutions to the spatial location identification problem are usually based on two observations: first, neurons are often spatially localized, yielding methods based on local correlations of neighboring pixels (Smith and Häusser, 2010), dictionary learning (Pachitariu et al., 2013), or graph-cut-related algorithms (Kaifosh et al., 2014). These methods typically aggregate the activity over time to produce a summary statistic (e.g., the mean, maximum, or correlation image or a weighted graph representation between the different imaged pixels) that is then processed (e.g., segmented) to identify the spatial components. While these methods can yield localized estimates, they do not account explicitly for the calcium dynamics, and more importantly their performance can deteriorate in the case of significant spatial overlap, since an aggregated summary image (e.g., the mean)

will not be able to distinguish between different overlapping neurons.

A second approach stems from the observation that spatio-temporal calcium activity can be approximated as a product of two matrices: a spatial matrix that encodes the location of each neuron in the optical field, and a temporal matrix that characterizes the calcium concentration evolution of each neuron. Based on this observation [Mukamel et al. \(2009\)](#) proposed an Independent Component Analysis (ICA) approach that seeks spatiotemporal components that have reduced dependence. While simple and widely used in practice, ICA is an inherently linear demixing method and can fail in practice when no linear demixing matrix is available to produce independent outputs, as is often the case when the neural components exhibit significant spatial overlap.

To overcome this problem, two nonlinear matrix factorization methods have been proposed: multilevel sparse matrix factorization ([Diego-Andilla and Hamprecht, 2013](#)) and nonnegative matrix factorization (NMF) ([Maruyama et al., 2014](#)). These methods can deal more effectively with overlapping neural sources but do not explicitly model the calcium indicator dynamics and do not always provide compact spatial footprint estimates. In this paper, we approach the factorization, deconvolution, and denoising problems simultaneously, by introducing a constrained matrix factorization method that decomposes the spatiotemporal activity into spatial components with local structure and temporal components that model the dynamics of the calcium. By accounting for overlapping neuronal spatial footprints, we can obtain improved temporal deconvolution results, and conversely by imposing a structured model of the dynamics of the calcium indicator, we can obtain improved identification of the spatial footprints of the observed neurons, especially in low signal-to-noise ratio (SNR) areas.

A key characteristic of our approach is that it requires no tuning of regularization parameters that control the tradeoff between the fidelity to the data and some desired signal structure (e.g., sparsity), as is necessary in related methods that have appeared recently in the literature ([Pnevmatikakis and Paninski 2013; Haeffele et al., 2014; Diego-Andilla and Hamprecht, 2014](#)). Instead, we take a constrained deconvolution (CD) approach, in which we seek the sparsest neural activity signal that can explain the observed fluorescence up to an estimated measurement noise level. We then embed this CD approach within a constrained nonnegative matrix factorization (CNMF) framework that enforces the dynamics of the calcium indicator and automatically sets individual sparsity constraints for both the spiking activity and the shape of each inferred neuron.

We first apply our deconvolution and denoising methods to time series calcium imaging datasets for which ground truth is available. Such datasets are important for quantitatively assessing the performance of deconvolution algorithms, but in practice are typically not available, emphasizing the need for unsupervised methods like those described here. Next, we apply our CNMF framework to analyze a variety of large scale *in vitro* and *in vivo* datasets. The results indicate that our framework can handle different types of imaging data in terms of imaged brain location, frame rate, imaging technology, and imaging scale, in a flexible and computationally efficient way that requires

minimal human intervention. A further example is provided in the companion paper ([Yang et al., 2015](#)), where our new analysis methods help enable a novel multi-plane imaging technique. We discuss the results of these various analyses next, deferring all technical methodological details to the [Experimental Procedures](#) and the [Supplemental Experimental Procedures](#).

RESULTS

Constrained Sparse Nonnegative Calcium Deconvolution

We first address the problem of deconvolving the neural activity from a time series trace of recorded fluorescence. In principle, this is achievable when the imaging rate is fast, i.e., the time between two consecutive measurements is small compared to the calcium indicator decay time constant. Modern resonant scanning ([Rochefort et al., 2009](#)), random access microscopy ([Dumanian Reddy et al., 2008](#)), and scanless imaging ([Nikolenko et al., 2008](#)) protocols can allow for this, by recording neural ensembles at high rates. Under this regime, we take a completely unsupervised approach for performing deconvolution that can be summarized as follows: first, we estimate a parametric model for the calcium concentration transient response that would be evoked by a single spike. Instead of fitting a parametric model to isolated calcium transients evoked by single spikes (often only available from dual electrophysiological recording and imaging experiments, as in e.g., [Grewe et al. \[2010\]](#)), we approximate the calcium transient as the impulse response of an autoregressive (AR) process of order p (with p small, just 1 or 2 in all the examples presented here), which models the rise and decay time constants; this AR(p) model is estimated by adapting standard AR estimation methods. After determining the shape of the calcium transient, we estimate the spiking signal by solving a non-negative, sparse constrained deconvolution (CD) problem: we seek the sparsest nonnegative neural activity signal that fits the data up to a desired noise level, which is estimated from the power spectral density of the observed fluorescence trace. The resulting optimization problem is convex, i.e., it has no local minima, and can be solved efficiently, with complexity that scales just linearly with the number of observed timesteps, and quadratically with the (modest) AR order p , and not with the length of the transient response; see [Experimental Procedures](#) and the [Supplemental Experimental Procedures](#) for full details.

While the AR parameter identification process is often very useful in estimating the transient response that would arise from a single spike, it is useful to refine these estimates given initial estimates of the spike times. We found that an extension of the Markov Chain Monte Carlo (MCMC) methods described in [Pnevmatikakis et al. \(2013\)](#) provided an effective strategy (see the [Supplemental Experimental Procedures](#) for full details).

We tested these methods using an *in vitro* dataset of $n=207$ spinal motor neurons obtained from seven sequentially acquired imaging fields in a single preparation ([Figure 1](#)). The neurons expressed the GCaMP6s indicator and were stimulated under an antidromic stimulation protocol that caused them to reliably fire in patterns that matched the stimulus pulses (similar to [Machado et al., 2015](#)); we treat the antidromic stimulus spike times

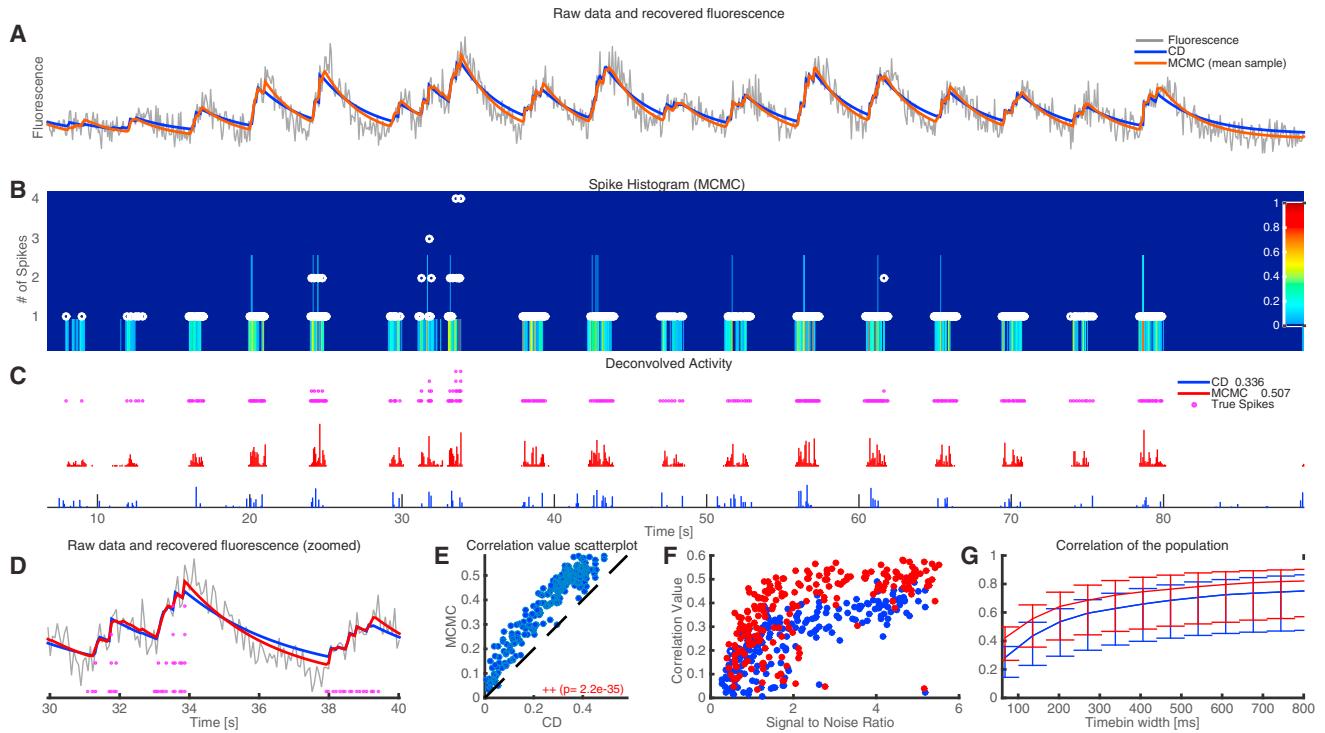


Figure 1. Application of the CD Method to Antidromically Driven In Vitro Spinal Cord Data

(A) Raw fluorescence data from an example neuron (gray) and reconstructed fluorescence trace with the proposed CD method (blue) and the mean sample obtained by the fully Bayesian MCMC method of Pnevmatikakis et al. (2013), with time constant updating (red). The CD method effectively denoises the observed fluorescence trace but overestimates the time constants slightly, while the more expensive MCMC method fine-tunes the time constants to match better the observed data.

(B) Color-coded depiction of the empirical posterior marginal histogram obtained with the MCMC method and true number of antidromic spikes during each timebin (white dots). The colormap displays the probability of a certain number of spikes within a given timebin. The MCMC method can quantify uncertainty and identify multiple spikes within a single timebin.

(C) Estimated neural activity (normalized) from the CD method (blue) and mean of the posterior marginal per timebin with the MCMC method (red). The legend also shows the spike correlation for each method at the imaged resolution. All methods detect accurately the bursting intervals of the neurons. The more expensive MCMC method gives a significant improvement in the spike deconvolution according to the spike correlation metric.

(D) Zoomed-in version of (A).

(E) MCMC outperforms CD for this dataset (Wilcoxon signed ranked test). Each circle corresponds to a single cell.

(F) Correlation values at the imaged resolution for all $n = 207$ cells as a function of the signal-to-noise ratio for the two methods. Performance increases with the SNR for all methods. Again, each circle corresponds to a single cell.

(G) Median correlation values for all $n = 207$ cells at various timebin widths. Error bars indicate the 0.25 and 0.75 quantiles, respectively.

as ground truth in this setting, though this neglects the effect of spontaneous (non-antidromically driven) activity on the observed fluorescence data. The imaging rate was 14.6 Hz and a first-order AR model ($p=1$) was found to be sufficient to model the calcium dynamics in this case. To quantify the performance, we computed the correlation between the true spiking signal (as is defined by the stimulus timing) and the inferred spiking signal, binned at the resolution defined by the imaging rate or coarser.

Figure 1A shows the reconstructed calcium trace for the CD algorithm (blue), and the mean calcium trace obtained with 500 samples from the MCMC algorithm (red) superimposed on the raw data (gray), and indicates that both methods track the observed fluorescence trace fairly well; however, the MCMC method can modify the inferred time constants to better fit the data (Figure 1A). Note that the MCMC method here was initialized with the results obtained with the CD approach. The

MCMC method produces samples of spike trains with continuous time resolution, and thus it can provide further insight into the number of spikes produced at every timebin and the uncertainty of these estimates due to noise and finite imaging rate. This is shown in panel B, where the marginal posterior of the number of spikes at each timebin is plotted and the true number of spikes (or stimulations) in this case is also shown (purple dots). This temporal uncertainty quantification is not available with the CD algorithm, which is based on a convex optimization framework and thus provides just a single estimate of the neural activity, binned at the imaging rate resolution. Panel C shows the inferred spiking signals for both methods and panel D displays the recovered traces and true antidromic stimulus spikes in more detail in a zoomed-in temporal interval. A comparison of the two methods for the whole population of the 207 neurons is shown in Figures 1E–1G. The computationally more intensive MCMC approach outperforms the CD method for almost every cell (Wilcoxon

signed rank test). The achieved correlation values increase, on average, with SNR (Figure 1F; SNR computed as the ratio between the standard deviation (SD) of the inferred calcium trace and the SD of the noise, as inferred from the MCMC method). Both inference methods improve their correlation coefficient at more coarse resolutions (Figure 1G). In the *Supplemental Information* (Figure S1), we apply these methods to a different publicly available dataset with ground truth, where the imaging rate is much higher (60 Hz) and AR(2) methods improve the deconvolution results significantly. We also illustrate the parameter identification and time constant updating methods in greater detail.

The proposed AR framework makes a number of simplifying assumptions on the fluorescence dynamics, with the benefit of increased computational tractability. The dynamics are assumed to be linear and time invariant, and no saturation level is assumed. It is possible to find clear violations of these assumptions in Figure 1A: bursts of spiking activity at the beginning of the trial, e.g., in the interval [10,20]s, have a weaker effect than bursts toward the end of the trial, and the strong bursts in the zoomed interval displayed in Figure 1D do not appear to add up linearly. It is natural to ask how much the performance of the algorithm might be limited by these unmodeled nonlinearities. Machado et al. (Machado et al., 2015, their Figure S1) addressed this question using data nearly identical to that used here (but with a different calcium indicator), by including an additional sigmoidal nonlinearity. Although this approach does not model the biophysical properties of the indicator binding, it can still be helpful in capturing weak activation from single spikes and saturation effects. The analysis in that paper showed that there was no statistically significant difference in the performance of the CD algorithm when this nonlinearity was included. Similar results were obtained here for the data shown in Figure 1 (Wilcoxon signed ranked test, p value 0.65, data not shown). Thus, in the following we use the simple linear AR model as the building block for our spatiotemporal framework but note that the choice of temporal deconvolution method is largely independent of the spatiotemporal demixing methods presented next; therefore, as discussed in the *Experimental Procedures*, other deconvolution methods that incorporate nonlinear effects (such as the particle filter methods discussed in Vogelstein et al., 2009) could easily be integrated within the framework presented here.

A Constrained Nonnegative Matrix Factorization Approach for the Analysis of Spatiotemporal Calcium Imaging Data

The problem of identifying the spatial footprint of each imaged neuron is also known in the literature as a problem of region of interest (ROI) selection or image segmentation. While these terms are intuitive, we prefer to use the phrase “spatial footprint identification,” which does not preclude significant overlap between the footprints of different neurons. In contrast, in (manual) ROI selection, a compact set of pixels/voxels is typically assigned to each neuron and the signal is then obtained by averaging the spatial ROI. Similarly, in image segmentation techniques, each pixel/voxel is generally assigned to a different object and overlap is not allowed. Here we argue that pre-

venting spatial overlap during spatial footprint identification can potentially lead to significant levels of “cross-talk” between spatially overlapping components, lower SNR, and therefore to misleading results; further examples are provided in the companion paper (Yang et al., 2015).

We approach this problem by employing a CNMF framework that decomposes the data into a spatial matrix that encodes the spatial footprint of each neuron and a temporal matrix that encodes the time-varying calcium activity of each neuron, together with components that model the background/neuropil activity. This procedure denoises the spatiotemporal fluorescence up to a desired voxel-dependent noise level and infers compact neuron shapes with fluorescence dynamics that obey the dynamics of the calcium indicator. We also include methods for merging spatial components that correspond to the same neuron, by combining components that overlap spatially and are significantly correlated temporally, and methods for removing components that do not correspond to neurons and/or have insignificant temporal contributions. We assume that all the data has been pre-processed to correct for motion artifacts during acquisition, although the proposed methods can tolerate small motion artifacts compared to the size of the spatial components. Moreover, our methods can handle missing data, e.g., due to motion in line-scanned methods. In the *Experimental Procedures* and the *Supplemental Experimental Procedures*, we present more details about the iterative algorithms we use to compute the matrix factorization solution (in which we first optimize with respect to the temporal components with the spatial footprints held fixed, then vice versa); the resulting solutions can be computed efficiently and in many cases are readily parallelizable.

The problem of optimally decomposing a matrix into a product of two other matrices is not convex, and alternating minimization methods such as those developed here converge only to a local optimum, which depends on the initial starting point. In some cases, good initializations are available: for example, through dual labeling of neurons so that the nucleus is labeled with a non-fluctuating red signal, while calcium imaging in the soma is performed in the green channel. Our core framework could be initialized by segmenting the red channel, or more broadly by any method that provides useful information about the locations of the observed neurons. When no initialization is already available, we designed two initialization algorithms effective in the case of somatic imaging. The first is an efficient greedy method that searches for a user-specified number of compact spatial regions that explain a significant percent of the variance observed in the data. The second is a convex approach that uses Gaussian spatial filtering to group correlated neighboring pixels together and initialize the spatial locations, while at the same time controlling for the number of formed groups via a “group lasso” (GL) penalty. We focus on applications of this procedure in the following and again defer to the *Experimental Procedures* and the *Supplemental Experimental Procedures* for a detailed presentation.

Resolving Overlapping Locations through CNMF

Before applying our CNMF framework to large-scale datasets, we illustrate through examples on artificial data its ability to demix overlapping neurons. We first constructed two largely

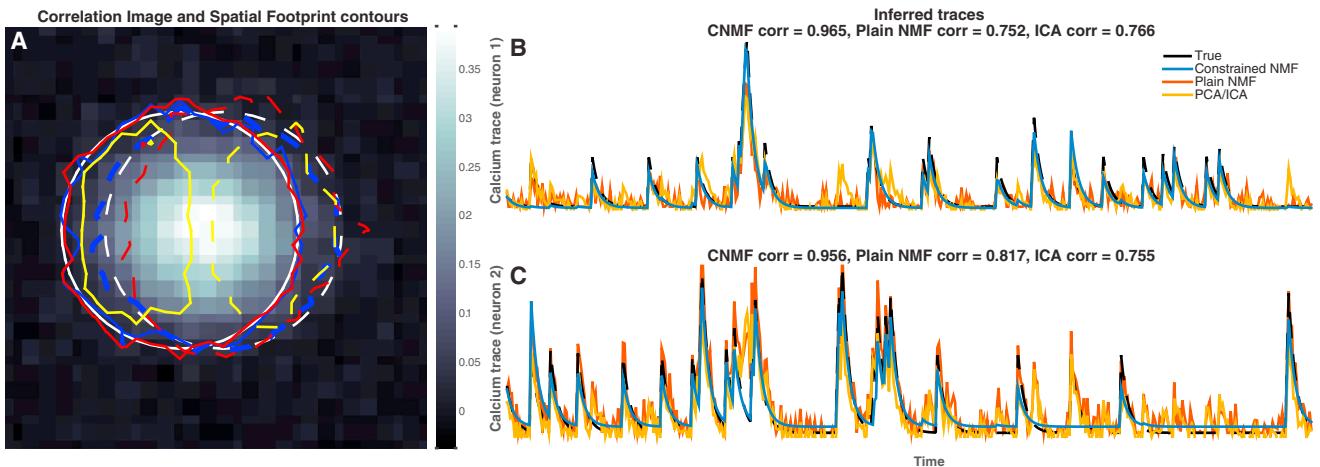


Figure 2. Resolving Overlapping Spatial Footprints in Simulated Data

Simulated fluorescence traces were generated from two significantly overlapping neuron spatial footprints in low SNR conditions. (Both spatial and temporal units in this example are arbitrary.)

(A) Correlation image generated from the raw data and superimposed contour plots (solid for neuron 1 and dashed for neuron 2) for the true spatial footprints (white) and the inferred spatial footprints with our proposed CNMF method (blue), the plain NMF method of Maruyama et al., (2014) (red), and the PCA/ICA method of Mukamel et al., (2009) (yellow). The correlation image cannot distinguish between the two different neurons. The spatial footprints inferred by PCA/ICA are significantly smaller than the true spatial footprints. NMF methods capture the full spatial extent of the spatial footprints.

(B and C) Inferred calcium traces with all the methods. PCA/ICA and plain NMF cannot satisfactorily demix the traces and attributes single neuron activity to both neurons. On the contrary, our CNMF approach eliminates most of the “cross-talk” between the overlapping neurons. Correlation values are computed on the estimated deconvolved neural activity.

overlapping neurons (contour plots shown in white in Figure 2A) that fired spikes according to independent Poisson processes. Due to the high degree of spatial overlap, time-aggregated measures of activity (e.g., the mean image) cannot display the existence of more than one neuron. This is illustrated again in Figure 2A, where the “correlation image” is shown. The correlation image for each pixel is computed by averaging the correlation coefficients (taken over time) of each pixel with its four immediate neighbors.

We compare our method against the popular PCA/ICA method of Mukamel et al. (2009) and the plain NMF method proposed in Maruyama et al. (2014). The PCA/ICA method searches for a matrix of linear spatial filters such that, when these filters are applied to the video data, the output time series are as independent as possible. However, in the case of significant spatial overlap, there is no such linear demixing matrix that can lead to independent outputs, and therefore in this example the PCA/ICA method tends to infer non-overlapping spatial filters (yellow contour plots in Figure 2A) by assigning pixels that should belong to both neurons to only one of the neurons and simply neglecting many pixels with strong contributions from both neurons. This solution has two shortcomings. First, since some overlapping pixels are uniquely assigned to one neuron, the inferred traces exhibit significant amounts of “cross-talk” (yellow traces in Figures 2B and 2C) that is potentially misleading for understanding the spiking activity of each neuron. (This cross-talk would corrupt any analysis that depends on the cross-correlation between these two neurons, for example.) Second, by excluding many overlapping high-SNR pixels, the temporal traces are computed over a smaller spatial region, resulting in decreased total output SNR. (See the companion paper, Yang et al. [2015], for further exam-

ples and analysis of this effect in real data.) The NMF method proposed in Maruyama et al. (2014) is a *nonlinear* demixing method and can therefore better handle spatially overlapping signals, especially in high-SNR conditions. However, this method (as well as PCA/ICA) does not model the temporal dynamics of the calcium indicator or impose any sparsity or locality constraints and thus can produce very noisy output traces (Figures 2B–2C, red traces), reducing the effective SNR of the output signals and again allowing for “cross-talk.”

In contrast, the CNMF method infers the spatial footprints of the two neurons very accurately (blue contour plots in Figure 2A) and demixes their activity over time with almost zero “cross-talk” between the two traces and minimal noise (blue traces in Figures 2B and 2C). More technical information about this example can be found in the [Supplemental Experimental Procedures](#).

We next examined the robustness of these methods as a function of the measurement noise level. We constructed a population of ten partially overlapping neurons placed randomly in a field of view (an example is shown in Figures 3A–3C), against a uniformly time-varying background. These neurons again fired spikes according to independent Poisson processes. We then simulated Gaussian white noise for each pixel, with SD proportional to the mean activity at this pixel, resulting in an approximately uniform SNR across the whole field of view. A range of different noise levels was considered, spanning over an order of magnitude of SNR. We repeated each simulation five times and considered two different neuron shapes that are encountered in practice: (1) “donut”- (as seen in panels A–C, corresponding to expression at the cytosolic regions of the cells and excluding the nucleus) and (2) “Gaussian”-shaped neurons

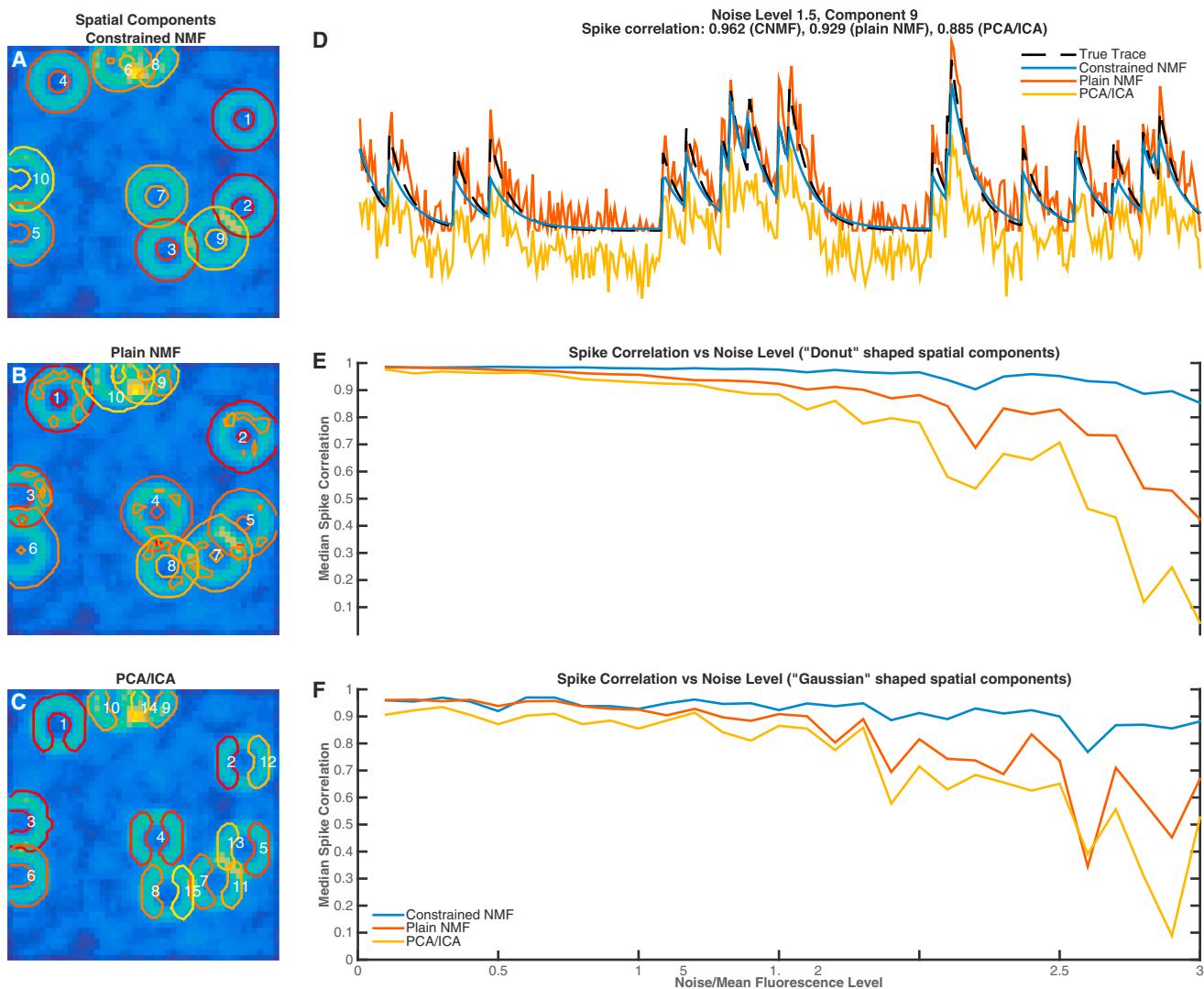


Figure 3. Performance of Proposed Spatiotemporal Method in Simulated Data

Simulated fluorescence traces were generated from a population of ten neurons placed randomly in the field of view, allowing significant spatial overlap. Two different neuronal shapes were considered across thirty different noise levels.

(A–C) Inferred spatial components with the proposed CNMF method (A), plain NMF as proposed in Maruyama et al. (2014) (B), and the PCA/ICA method of Mukamel et al. (2009) (C). Contour plots of the inferred spatial components are super-imposed on the image of mean activity. In this example, the noise level for every pixel was $1.5 \times$ the mean activity. The numbers in white are placed on the center of mass of each component. The proposed method identifies spatial components very accurately, compared to plain NMF that can group different components together. PCA/ICA tends to infer smaller non-overlapping components and can also split components into multiple parts in low SNR conditions.

(D) Inferred temporal traces for component 9 (as indicated in A). The proposed method infers a trace (blue) that matches the true trace (dashed black) much better than the plain NMF method (red) and PCA/ICA (yellow).

(E) Median spike correlation of the three methods over 30 different noise levels, with five iterations for each level, for donut-shaped neurons.

(F) Same, but for "Gaussian" shaped neurons. The proposed method is significantly more robust compared to other popular methods, especially for low-SNR conditions.

(more similar to the simulated neurons shown in Figure 2). Full simulation details can be found in the [Supplemental Experimental Procedures](#).

A specific instance of this simulation is shown in Figures 3A–3D, where the noise level was chosen to be $1.5 \times$ the mean activity over time. Results are similar to those shown in the previous figure; the PCA/ICA method tends to infer non-overlapping

spatial filters (e.g., components 11 and 5 in Figure 3C). Moreover, in low-SNR conditions, as the example shown in Figure 3, individual neurons are sometimes mistakenly split into two components, e.g., components 2 and 12 in Figure 3C.

The plain NMF method infers the spatial components better than PCA/ICA but does not penalize for the sparsity of the spatial footprints and therefore can infer components that are not

Table 1. Qualitative Differences between the Proposed CNMF Method, PCA/ICA, and Plain NMF

Method	PCA/ICA ^a	Plain NMF ^b	Constrained NMF ^c
Features			
Handles spatial overlap well	No	Yes	Yes
Eliminates “cross-talk”	No	Medium	Yes
Robustness in low SNR	Medium	Medium	Yes
Produces localized estimates	Yes	No	Yes
Unsupervised (after initial parameter setting)	Yes	No	Yes
Deconvolves neural activity	No	No	Yes
Models background activity	No	Yes	Yes
Enables merging	No	Manual	Automated

^aMukamel et al. (2009).^bMaruyama et al. (2014).^cThis paper.

localized or combine multiple neurons that do not necessarily overlap (Figure 3B). Moreover, this method is prone to local optima in low-SNR data and requires significant manual intervention (for component selection and merging) when applied to fields of view containing many neurons. Finally, as also shown in Figure 3D, the absence of a model for the temporal dynamics of the underlying calcium signal results in noisy traces that hinder activity deconvolution.

In contrast, by (1) penalizing the size of the spatial footprints, (2) estimating the noise level of the data for each pixel, (3) modeling the indicator dynamics, and (4) using appropriate initialization methods, the CNMF method infers the spatial footprints of all the neurons very accurately (contour plots in Figure 3A) and demixes their activity over time with almost zero “cross-talk” between neighboring traces and minimal noise (blue trace in Figure 3D), with no manual intervention. Our approach remains robust even in very low-SNR conditions, where the other two methods break down (Figures 3E and 3F). These qualitative differences between the three compared methods are summarized in Table 1.

Analysis of Large-Scale Imaging Data

We now turn to the analysis of large-scale spatiotemporal datasets. We begin by applying our methods to *in vivo* mouse V1 GCaMP6s spontaneous activity data (Figure 4). The algorithm was initialized with the greedy method (see **Experimental Procedures**) with 300 components. During the factorization iterations, 31 components were eliminated due to merging operations or negligible total contribution. The remaining components were ranked in decreasing order based on their size and maximum temporal value (see **Experimental Procedures**). The contours of the first 200 inferred spatial footprints are depicted in Figure 4A, superimposed on the correlation image of the raw data. Localized regions in the correlation image with high intensity correspond to strongly active cells, whereas localized regions with lower intensity correspond to neurons with lower intensity/SNR or other non-stationary processes. The algorithm efficiently iden-

tifies neurons with very few visually apparent false positives and denoises the calcium activity of each cell (Figure 4B). The inferred temporal traces match the time course and dynamics of the raw data. This is shown in Figure 4C, which zooms in to a point of local maximum activation for each trace (point marked in Figure 4B) and plots the traces in higher resolution superimposed with the spatial component weighted average of the raw data, after the removal of all the other components. The spatial footprints of the first 36 inferred components are shown in Figure 4D; in many cases detailed morphological structure is automatically extracted by the algorithm. The results are viewed best in Movie S1. Figure 4E displays an example of the merging procedure. The neuron depicted in the fourth panel is initially split across three components. Since the temporal activity of these components is highly correlated, they are merged into a single cell. Full implementation details are given in the **Supplemental Experimental Procedures**.

Segmentation of Whole-Brain Light Sheet Imaging Data

Next, we test our method on a larger scale. Specifically, we examine 3D imaging data from a whole zebrafish brain (Freeman et al., 2014). The imaging was done using light-sheet microscopy, with a GCaMP6s indicator, localized to the nucleus. In this dataset, the neurons are closely packed together and sometimes exhibit highly synchronous activity. We use the group lasso (GL) approach to initialize the estimates, followed by fine-tuning using CNMF. The length of the imaging timebin (~475 ms) is comparable to the decay time constant of the calcium indicator, and the temporal details of the spiking activity cannot be estimated at a fine temporal resolution here. However, we can still utilize a similar matrix factorization framework by dropping the temporal constraints due to the calcium dynamics and then inferring the spatial and temporal components.

The resulting spatial and temporal components are shown in Figures 5A and 5B. These generally match the neuronal shapes and the sparse nature of the activity visible in the Movies S2, S3, and S4. Specifically, high-ranking spatial components usually have smoother, more consistent spheroid shapes (consistent with the nuclear-localized calcium indicator) and the corresponding temporal components tend to be sparser. Low-ranking shapes seem to correspond to faint or partially obstructed neurons. In total, the algorithm detects a realistic distribution of components throughout the brain (Movie S8). Most visible neurons are detected, as can be seen in Figures 6A, 6C, and 6E, and more clearly in the Movies S2, S3, and S4.

In contrast, while the PCA/ICA method (Mukamel et al., 2009) usually performs well in locating “strong” neurons (i.e., high-SNR, well-separated neurons), this method misses many relatively “weak” neurons, as can be seen in Figures 6B, 6D, and 6F (see also Figure S2, and Movies S5, S6, and S7). PCA/ICA detects several thousand fewer high-quality components than the CNMF method with GL initialization in this dataset. Moreover, the components detected by CNMF usually have better quality than those detected by PCA/ICA. For example, the maximum DF/F traces of low-ranked components found by PCA/ICA are typically more noisy, attain smaller maximum values, and sometimes exhibit relatively strong negative values (Figure S2). In contrast, the CNMF method is able to denoise the activity and

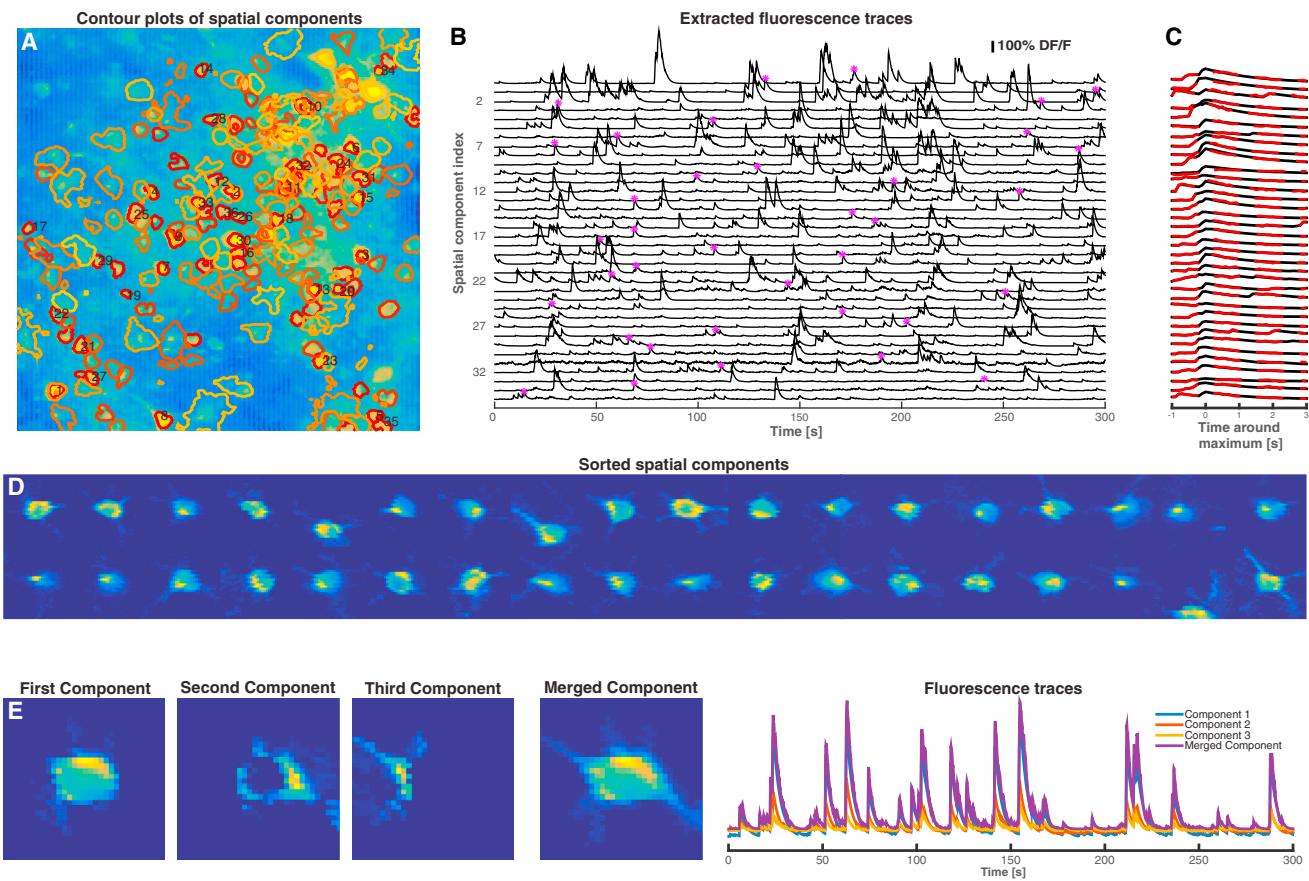


Figure 4. Application to Mouse In Vivo GCaMP6s Data

(A) Contour plots of inferred spatial components superimposed on the correlation image of the raw data. The components are sorted in decreasing order based on the maximum temporal value and their size. Contour plots of the first 200 identified components are shown, and the first 36 components are numbered.

(B) Extracted DF/F fluorescence traces for the first 36 components.

(C) Zoomed-in depiction of the fluorescence traces around a point of local maximum activity indicated by the star marker in (B), (black) super-imposed with the raw spatial component filtered data after removal of all the other components (red dashed).

(D) Spatial footprints of the first 36 components. The algorithm can in many cases pull out morphological details of the imaged processes.

(E) Depiction of the merging procedure: the first three panels in the left show three overlapping components with highly correlated temporal activity. These components are merged into a single component that is further refined in the algorithm. The temporal traces of the three initial components and the merged component are shown in the right panel (see also Movie S1).

also obtain a non-negative signal (Figure 5). These results are consistent with those obtained in simulated data, as illustrated in Figures 2 and 3. Note that the method used here is closer to the plain NMF of Maruyama et al. (2014), with three key differences: (1) the NMF iterations here are spatially constrained to keep inferred components localized; (2) the GL initialization method allows us to process large patches of data at once and bypass the high level of manual intervention required by plain NMF; and (3) this new initialization method helps the CNMF iterations converge to a better solution.

Application to Dendritic Imaging Data

A key advantage of the proposed CNMF framework is that we can apply similar methods to imaging data focusing on the dendrites of multiple neurons and not on the cell bodies. In this case, each spatial component corresponds to a set of dendritic branches from a given neuron, and the temporal component cor-

responds to the synchronous activity of these branches. These data exhibit certain qualitative differences compared to somatic imaging. Each spatial component is again sparse but is no longer spatially localized, since dendritic branches can stretch significantly along the observed imaging plane. As a result, the degree of overlap between the different branches is significantly higher, making even rough interpretation by eye a challenging task; the correlation and mean images in this setting provide very little useful segmentation information (Figures 7A and 7C). Moreover, the bound calcium dynamics no longer follow somatic calcium indicator dynamics. Dropping the temporal dynamics and spatial localization constraints from our problem, we obtain a simpler sparse CNMF problem that can still be solved efficiently using the methods described above.

We applied this approach to in vivo dendritic imaging data taken from the apical dendrites of layer 5 pyramidal neurons in the rodent barrel cortex (Lacefield and Bruno, 2013, SfN,

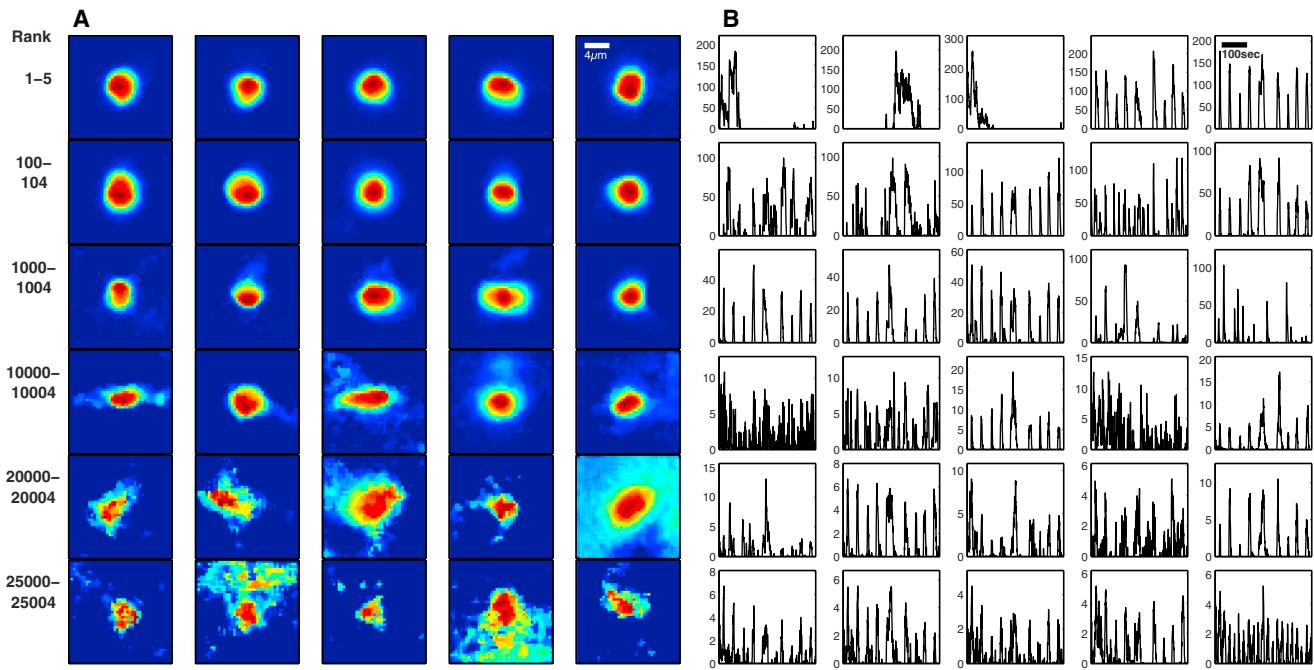


Figure 5. Components Detected in a Whole Zebrafish Brain

A sample of detected components (A, inferred neuronal shapes; B, inferred DF/F activity traces), ordered according to their rank. High-ranking components match expected nuclear-localized neuronal shapes and activity visible in the raw video data; low-ranking components tend to be more “noisy” in both shapes and activity. In all, the first 26,000 components largely correspond to reasonable neuronal signals (as determined by visual inspection of the video data; [Movies S2, S3, and S4](#)).

abstract) and show the results in [Figure 7](#). The raw data is typically dense when averaged over either time or space, as can be seen from the low intensity of the raw data correlation and mean image ([Figures 7A–7C](#)). We initialize randomly using a large number of components (in this case 50) and then order the inferred components as before. The top 23 of these components together with the background and the corresponding temporal traces in DF/F units are shown in [Figures 7D–7F](#). These components are sparse in both space and time, with the temporal components extracting temporally localized bursts visible in the raw video data, and the spatial components extracting segments of the apparent dendritic structures visible in the video data (see also [Movie S9](#)). The results show the effectiveness of the sparse CNMF procedure in obtaining separated spatiotemporal components given dendritic imaging data in which the degree of overlap is very high and the spatial components are not localized.

DISCUSSION

A Flexible and Efficient Model for Calcium Deconvolution

The major contribution of this work is to introduce a framework for analysis of calcium imaging data that is (1) applicable to a large variety of data, (2) computationally efficient and parallelizable, and (3) user friendly, in the sense that the required human intervention is limited to the setting of a few intuitive parameters (e.g., rough estimates of number and size of neurons). This framework is enabled by a number of novel tools (CD, CNMF,

efficient initialization, robust parameter estimation, and component sorting/merging) that can be combined in a modular fashion depending on the details of the imaging data (e.g., the spatio-temporal resolution). As a result, we were able to successfully apply our framework to many different imaging datasets without extensive specialized pre-processing.

Our CD approach builds upon and extends the fast nonnegative deconvolution (FOOPSI) method of [Vogelstein et al., \(2010\)](#) in several ways. Compared to FOOPSI, which permits only first-order autoregressive models with pre-specified time constant, our method allows for general AR (p) models and comes with a method for estimating the coefficients. More importantly, in the FOOPSI method, the loss function appears in the objective function, and a regularizer needs to be introduced and fine-tuned to control between the faithfulness to the observed data and the (unknown) sparsity of the underlying spiking signal. In our method, the tradeoff between sparsity and fidelity to observed data is set automatically by estimating the noise level a priori and using it as a hard constraint on the overall acceptable level of residual error, while retaining the same linear time computational complexity as FOOPSI.

Modeling the fluorescence trace with an AR process allows for a flexible and interpretable modeling framework that still retains computational tractability. When solving the CD problem, the complexity of the algorithm scales only with the order of the AR process and not with the length of the transient impulse response of the AR model, making this approach computationally more tractable than arbitrary template matching algorithms or blind

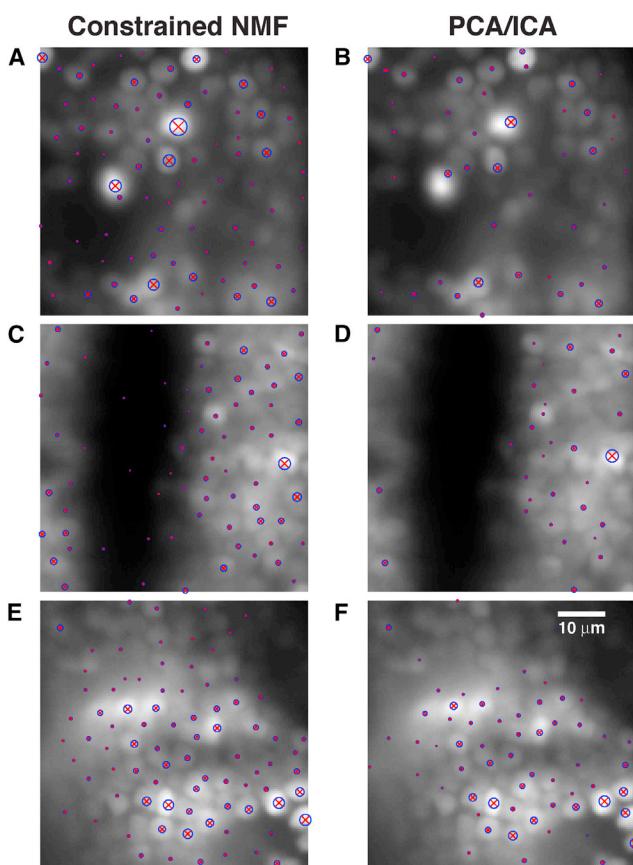


Figure 6. The CNMF Method Outperforms the PCA/ICA Method in Detecting Weak Neurons, on Patches of Zebrafish Data.

The patches are covered with the shape centers detected in that patch (blue circles around red "x" markers, with larger symbols indicating higher-ranked components), superimposed on the mean image for each patch.

(A, C, and E) Centers detected for the CNMF method, on patches containing the 1st, 1,000th, and 26,000th ranked neurons, respectively.

(B, D, and F) Centers detected for the PCA/ICA method on the same patches. Calcium video and detected components in these patches can be viewed in Movies S2, S3, S4, S5, S6, and S7. The detected components for the PCA/ICA method are shown in Figure S2. Movie S8 shows the components detected throughout the brain, using CNMF.

deconvolution approaches (Diego-Andilla and Hamprecht, 2014). Moreover, the AR coefficients can be directly estimated from the raw data in a completely unsupervised way, do not require the existence of isolated spikes for fine-tuning the various parameters, and can be further fine-tuned during the deconvolution algorithm. Although this AR framework makes certain simplifying assumptions about the linearity and dynamics of the calcium indicator, nonetheless it results in a highly computationally efficient estimator that achieves state of the art performance among unsupervised methods, as shown recently in Theis et al. (2015).

CNMF Provides a Flexible Framework for Analysis of Large-Scale Calcium Imaging Data

We applied our framework to a diverse set of *in vitro* and *in vivo* imaging data to illustrate the wide applicability of the matrix

factorization framework to large-scale imaging data. Matrix factorization inherently models the possible overlap between the different components as opposed to manual ROI selection or image segmentation-based techniques (Smith and Häusser, 2010; Kaifosh et al., 2014). Indeed, the matrix factorization approach often obtains highly overlapping spatial components in cases in which such components optimize the amount of variance explained in the data (up to the noise constraints). This contrasts with the PCA/ICA approach, which tends to reject overlapping areas in an attempt to seek maximally independent spatial components, at a risk of increased cross-talk and reduced SNR (Figures 2 and 3). As a result, our demixing method can be used in cases in which the spatial overlap is significant either naturally, because of the dense neural packing or deep imaging in scattering tissue, or artificially, due to the simultaneous observation of the projection of multiple *z* planes through spatial light modulation (Yang et al., 2015, companion paper), or more general projection patterns as suggested and studied in Pnevmatikakis and Paninski (2013); such approaches may lead to computational imaging schemes that allow us to simultaneously image larger neural populations than would be otherwise possible and are an important direction for future research.

The iterative factorization procedure is largely agnostic regarding the shape of the spatial components, inferring these shapes directly from the data with few prior constraints. Other approaches, especially for analysis of somatic imaging data (Pachitariu et al., 2013; Diego-Andilla and Hamprecht, 2014), use dictionary learning approaches to learn a compact basis for the representation of the spatial footprints. These methods can help reduce the dimensionality of the problem and can be helpful in more macroscopic images in which detailed structure is not available. In future work, it may be useful to combine these methods with the techniques developed here. However, these dictionary-learning methods do not appear to generalize to more microscopic imaging (e.g., dendritic imaging; Figure 7), where the structure of the spatial components can be rich and unpredictable.

Sparse and structured NMFs have been proposed in the recent literature (Pnevmatikakis and Paninski, 2013; Diego-Andilla and Hamprecht, 2013; Haeffele et al., 2014) in an attempt to constrain the factorization procedure and produce compact spatial footprints and denoised temporal traces. The structure there is promoted by the inclusion of regularizers that promote certain desired characteristics. For example, Pnevmatikakis and Paninski (2013) propose a nuclear norm penalty to identify the number and initialize the spatial components. Haeffele et al. (2014) propose the inclusion of a spatial total variation norm penalty to promote localized and compact spatial footprints. Diego-Andilla and Hamprecht (2014) propose a sparse space-time deconvolution for learning the neuron locations and extracting spike activity through dictionary learning and blind deconvolution. A typical drawback of these approaches is the need to tune regularization weights. Moreover, a single regularizer over the observed field of view implicitly assumes that the statistics are uniform across the whole observed field of view: i.e., the density and size of neurons is uniform, all the neurons fire at similar average rates, and the noise power across the observed field of view is similar. These assumptions can

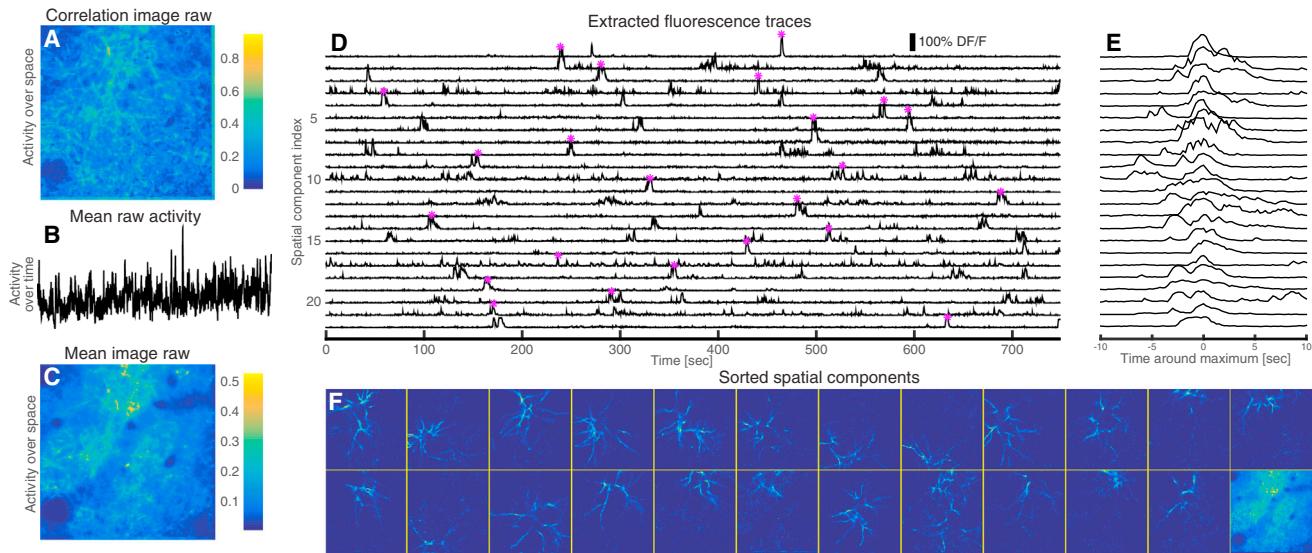


Figure 7. Application to In Vivo Dendritic Imaging Data from Rodent Barrel Cortex

- (A) Correlation image of the raw data. Due to the high degree of overlap, the correlation image cannot be used to segment the video.
- (B) Spatially averaged activity over time of the raw data.
- (C) Mean activity over time.
- (D) DF/F temporal traces of the top 23 components extracted from our algorithm.
- (E) Close up and centering around the point of maximum activation for each trace (indicated by the star marker in D).
- (F) Sorted spatial footprints and background (lower right corner). The proposed method can segment the dense dendritic imaging data and reveal a rich underlying sparse structure (see also Movie S9).

often lead to the omission of weaker components and require the processing of small patches at a time as noted by Maruyama et al. (2014). A distinctive feature of the CNMF framework presented here is that this regularization approach is replaced by a larger set of constraints on the noise power at each pixel. These noise levels are automatically estimated a priori and implicitly impose non-uniform statistics on the observed field of view.

A Modular Approach toward Automated Calcium Imaging Data Analysis

Our approach is highly modular and provides a flexible analysis framework. For example, depending on the imaging frame rate, the deconvolution step can be turned on or off, or the proposed CD framework can be replaced by other methods (e.g., particle filtering [Vogelstein et al., 2010], greedy optimization [Grewe et al., 2010], or MCMC [Pnevmatikakis et al., 2013]). Different neurons can have the same or different time constants that can also be re-estimated using the proposed AR approach. Our initialization algorithms target cell bodies whose images can be crudely modeled as Gaussians with specific sizes, although different shapes can be easily incorporated.

A specific objective in this work was to create a framework that requires minimal human intervention. In our models, the parameters that are user defined are specified during the initialization procedures and pertain to the number of components that are sought initially (greedy initialization, and sparse NMF in the case of dendritic imaging), and/or the size/scale of the spatial components that are sought (greedy, GL initialization).

The rest of the parameters (noise levels, calcium indicator dynamics, time-varying baseline concentration) are tuned automatically, though again, any prior knowledge about these quantities can be readily incorporated. While this level of human intervention is low, it remains an important open problem to automatically estimate the number or the size of the imaged components. To that end, we also introduced a measure that ranks the inferred components in terms of their significance, and correction moves that merge overlapping and correlated components, or remove insignificant components, and found that such moves can be very helpful in practice (Figures 4, 5, 6, and 7).

Calcium imaging as a neural recording method can provide rich and diverse datasets depending on many factors (imaging technique, experimental conditions, calcium indicators, etc.), and multineuronal data with spiking and/or spatial anatomical ground truth remains scarce and typically hard to obtain. Moreover, assessing the performance of different approaches remains an open issue, with no universally accepted performance measure. For example, the spike correlation metric used in this study to quantify the performance of activity deconvolution methods is empirically very sensitive to spike jitter. The situation is even more complicated in automatically quantifying the performance of spatial footprint identification methods. Simple correlation-based metrics, for example, can be insensitive to the detailed morphological structure, and even the comparison against ground truth data (e.g., through dual channel labeling) can be challenging due to potential spatial overlap, poor expression, or unmodeled background activity. As such, the goal of a

globally operating algorithm that requires no human intervention and has performance guarantees remains elusive. Nevertheless, we hope that the family of methods discussed here, under the unifying umbrella of CNMF, will provide a useful framework for future research.

EXPERIMENTAL PROCEDURES

Experimental Data

Full details on the experimental data are provided in the [Supplemental Experimental Procedures](#).

Data Analysis

All analysis was performed with custom-written MATLAB (RRID: nlx_153890) code. MATLAB and Python implementations of the CD and CNMF algorithms can be found in https://github.com/epnev/ca_source_extraction and linked repositories therein.

Autoregressive Model for Calcium Dynamics

We assume we observe the fluorescence signal for T timesteps and denote by $s(t)$ the number of spikes that the neuron fired at the t -th timestep, $t = 1, \dots, T$. We approximate the calcium concentration dynamics c using a stable autoregressive process of order p (AR(p)) where p is a small positive integer,

$$c(t) = \sum_{j=1}^p \gamma_j c(t-j) + s(t). \quad (\text{Equation 1})$$

If $p = 1$ then the calcium transient in response to a spike is modeled by an instantaneous-increase slowly decaying exponential function. This is recommended when the rise time constant is small compared to the length of the timebin. In case we want to explicitly model the rise time, we choose $p = 2$. More structured responses (e.g., multiple decay time constants) can also be modeled with higher values for the order p . The observed fluorescence is related to the calcium concentration as:

$$y(t) = \alpha(c(t) + b) + \varepsilon(t), \quad \varepsilon(t) \sim \mathcal{N}(0, \sigma^2), \quad (\text{Equation 2})$$

where α is a nonnegative scalar, b is the baseline concentration and the noise is assumed to be i.i.d. zero mean Gaussian with variance σ^2 . The goal of calcium deconvolution is to extract an estimate of the neural activity s from the vector of observations y .

Estimation of Model Parameters

For a given order p and under the assumption that the spiking signal s comes from a homogeneous Poisson process, the autocovariance function of y , C_y , satisfies the recursion:

$$C_y(\tau) = \begin{cases} \sum_{j=1}^p \gamma_j C_y(\tau-j) - \sigma^2 \gamma_\tau, & 1 \leq \tau \leq p \\ \sum_{j=1}^p \gamma_j C_y(\tau-j), & \tau > p. \end{cases} \quad (\text{Equation 3})$$

By plugging the sample autocovariance values into Equation 3, we can first estimate the AR coefficients $\gamma_1, \dots, \gamma_p$. While Equation 3 can also be used to estimate the noise variance σ^2 , a more robust estimate can be obtained by observing the power spectral density (PSD) of y . The uncorrelated additive noise has flat PSD, whereas the PSD due to the calcium signal decays with the frequency as $\sim (1/f^2)$. At high frequencies, and under sparse spiking, the PSD will be dominated by the noise, and therefore an estimate $\hat{\sigma}^2$ can be obtained by averaging the PSD over a range of high frequencies. We discuss these issues in more detail in the [Supplemental Information](#) ([Figure S1](#)).

Sparse, Nonnegative, Noise-Constrained Deconvolution

Solving for the spiking vector s in the domain of nonnegative integers is a computationally hard problem and also requires knowledge of the spike amplitude scaling constant α , which is in general unavailable. Instead, by following the approach in [Vogelstein et al. \(2010\)](#), we can absorb α into s , relax s to take arbitrary nonnegative values, and penalize the sum of the spike signal over

time (which can be seen as the l_1 -norm of a nonnegative signal), to promote spike sparsity and avoid overfitting. First note that Equation 1 can be expressed in matrix form as $Gc = s$, with

$$G = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ -\gamma_1 & 1 & 0 & \dots & 0 \\ -\gamma_2 & -\gamma_1 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & -\gamma_2 & -\gamma_1 & 1 \end{bmatrix}. \quad (\text{Equation 4})$$

(Note that we have ignored initial conditions for notational simplicity.)

Now based on the estimate of the noise variance, we can introduce a hard constraint on the residual energy; from Equation 2, we have that for each time $\mathbb{E}(y(t) - \alpha(c(t) + b))^2 = \sigma^2$. Summing over all timesteps, we find that the residual vector satisfies $\mathbb{E}\|y - \alpha(c + b\mathbf{1}_T)\|^2 = \sigma^2 T$ (with $\mathbf{1}_T$ a column vector of ones with length T); moreover, by the law of large numbers $\|y - \alpha(c + b\mathbf{1}_T)\|^2 \approx \sigma^2 T$ with high probability. We can use this estimate to derive the following convex program for estimating the calcium concentration up to a scaling constant:

$$\begin{aligned} \min_{\hat{c}, \hat{b}, \hat{s}} \quad & \mathbf{1}_T^\top \hat{s} \\ \text{subject to : } \quad & \hat{s} \geq 0, \hat{s} = G \hat{c}, \|y - \hat{c} - \hat{b}\mathbf{1}_T\| \leq \hat{\sigma}\sqrt{T}. \end{aligned} \quad (\text{P-CD})$$

Program (P-CD) also estimates the baseline concentration b , which for now is assumed to be constant; we relax this assumption in the spatiotemporal case. However, in many cases the baseline can be estimated a priori, e.g., by averaging the fluorescence over a large interval with no observed spikes.

Program (P-CD) is convex and a global optimum exists and is achievable with standard optimization methods that computationally scale only linearly with the number of observed timesteps; this enables the analysis of long time series or videos. A variety of methods can be used to solve (P-CD) in linear time, as detailed in the [Supplemental Experimental Procedures](#).

Note that the CD approach can estimate the \hat{s} (or \hat{c}), only up to a multiplicative constant, since the amplitude of a single spike in terms of fluorescence response (α in Equation 2) is in general unknown and in general cannot be readily estimated. If needed, estimates of the spike amplitude can be derived using the MCMC algorithm of [Pnevmatikakis et al. \(2013\)](#).

Updating the Time Constants

The time constants are updated in the continuous time domain, by using a simple MCMC scheme. Details are given in the [Supplemental Experimental Procedures](#).

Spatiotemporal Deconvolution and Component Demixing

Now we turn to the full spatiotemporal case. At every timestep, a field of view is observed for a total number of T timesteps. This field (either two- or three-dimensional) has a total number of d pixels/voxels and the observations at any point in time can be vectorized in a single column vector of length d . Thus, all the observations can be described by a $d \times T$ matrix Y . Now assume that the field contains a total number of (possibly overlapping) K neurons, where K is assumed known for now. For each neuron i the neural activity s_i and “calcium activity” c_i can be described again with the AR(p) dynamics of Equation 1. Now if $a_i \in \mathbb{R}_+^d$ denotes the (nonnegative) “spatial footprint” for neuron i (written in vector form), then we model the spatial calcium concentration profile at time t as

$$F(t) = \sum_{i=1}^K a_i c_i(t) + B(t), \quad (\text{Equation 5})$$

where $B(t) \in \mathbb{R}_+^d$ denotes the (time-varying) baseline vector all the pixels. Finally, at each timestep we observe $F(t)$ corrupted by additive Gaussian noise:

$$Y(t) = F(t) + \varepsilon(t), \quad \varepsilon(t) \sim \mathcal{N}(\mathbf{0}, \Sigma), \quad (\text{Equation 6})$$

where Σ is a diagonal matrix (indicating that the noise is spatially and temporally uncorrelated). These equations can be written in matrix form as

$$\begin{aligned} S &= CG^T \\ F &= AC + B \\ Y &= F + E, \end{aligned}$$

with $S = [s_1, \dots, s_K]^T$, $C = [c_1, \dots, c_K]^T$, $A = [a_1, \dots, a_K]$, $F = [F(1), F(2), \dots, F(T)]$, $E = [e(1), e(2), \dots, e(T)]$, $Y = [Y(1), Y(2), \dots, Y(T)]$, $B = [B(1), B(2), \dots, B(T)]$, and G is defined in Equation 4. In practice, we have found that the background activity matrix B can often be modeled as a rank 1 matrix, $B = bf^T$, where $b \in \mathbb{R}^d$, $f \in \mathbb{R}_+^T$ are nonnegative vectors encoding the background spatial structure (typically consisting of a sum of baseline activity from the neurons of interest and densely mixed neuropil structure below the observed spatial field) and global (time varying) intensity, respectively.

CNMF Framework

We approach this problem by employing a CNMF framework to infer the key parameters of interest (the spatial components A and temporal components C) along with b, f , and Σ . The factorization procedure itself is constrained in a number of ways. (1) The residual between the observed data and the reconstructed video is bound by similar noise power constraints as in the time series deconvolution case, where now we estimate a different noise level for each observed pixel/voxel, again from the temporal PSD. (2) Both the spatial and the temporal components are endowed with non-negativity constraints. (3) The spatial components are also constrained to be sparse, promoting more compact, regularized spatial footprints. Finally, (4) when the observed imaging rate is much higher than the decay rate of the indicator, we constrain the temporal traces to obey the indicator dynamics, similarly to the one-dimensional case. With this set of constraints, we can optimize the spatial and temporal components in an iterative alternating way, where a new estimate for A and b is obtained given the last estimate of C and f and vice versa, within the space defined by the constraints. At the end of each iteration, additional steps are introduced, where overlapping components that exhibit significant temporal correlations are merged into a single component, and existing components that do not contribute significant activity are removed and absorbed into the background. We also present two different methods for initializing the spatial components.

Our framework can be summarized into the following steps:

- (1) Determine the AR dynamics and noise power for each voxel.
- (2) Initialize A , b , C , f through one of the initialization procedures.
- (3) Update components A , b , C , f using constrained alternating matrix factorization.
- (4) Merge and/or remove existing components.
- (5) Repeat steps 3 and 4.

Iterative Matrix Updates

Assuming the number of neurons K and initial estimates of A , C and b , f as known, we can apply alternating matrix factorization methods to estimate the spatial components A , b given the temporal C , f and vice versa, from the fluorescence observations Y .

Estimating A , b

Since each column of A expresses the location and shape of a neuron, we want A to be sparse to promote localized spatial footprints. Again, this is done by minimizing the l_1 norm of A , which equals the sum of all its entries, since A is constrained to be non-negative. Given estimates of $C^{(k-1)}$ and $f^{(k-1)}$ from the previous iteration, the spatial matrix $A^{(k)}$ and background $b^{(k)}$ can be updated by solving the following convex program

$$\begin{aligned} & \min_{A, b} \|A\|_1, \\ & \text{subject to : } A, b \geq 0, \|Y(i,:) - A(i,:)\mathbf{C}^{(k-1)} - b(i)\mathbf{f}^{(k-1)T}\| \leq \sigma_i \sqrt{T}, \\ & \quad i=1, \dots, d, \end{aligned} \quad (\text{P-S})$$

where $A(i,:)$, $Y(i,:)$ denote the i -th rows of A and Y , respectively. Although the matrix A is of very large size, $d \times T$, the problem (P-S) can be readily parallelized into d programs for each pixel separately. Similarly to the CD problem (P-CD), each of these problems can be solved with a variety of methods (see the [Supplemental Experimental Procedures](#)).

When the fluorescence from each neuron is highly localized near the soma, the process of estimating A at the k -th iteration can be further sped up and regularized by restricting the candidate spatial support of cell j at the iteration k , to be a mildly dilated version of the support as is determined by the column j of $A^{(k-1)}$. When estimating the i -th row of $A^{(k)}$, we can restrict our search to the

neurons (columns of A) whose candidate support sets include the pixel i . This makes the dimensionality of each subproblem much smaller, leading to a highly efficient and parallelizable update.

Note that we have not yet incorporated any prior information about the detailed shape of the spatial components $A(:,i)$, which enabled the highly parallel approach described above. However, in many cases it is natural to assume that $A(:,i)$ is connected, or smooth in a suitable sense. Empirically, we have found it helpful to include a mild post-processing step at each iteration, using standard non-linear image filtering techniques, such as median filtering or morphological opening, which are effective in removing isolated pixels that appear as active. The removed pixels can then be absorbed by the background component.

Estimating C , f

For the temporal components, we want to introduce a sparsity penalty to the activity of each neuron to prevent overfitting. We can again use our estimates of the noise variance as hard constraints and derive a parameter-free convex program:

$$\begin{aligned} & \min_{c_1, c_2, \dots, c_K, f} \sum_{j=1}^K \mathbf{1}^T \mathbf{G} c_j \\ & \text{subject to : } \mathbf{G} c_j \geq 0, j=1, \dots, K, \|Y(i,:) - A(i,:)\mathbf{C}^{(k-1)} - b(i)\mathbf{f}^{(k-1)T}\| \leq \sigma_i \sqrt{T}, \\ & \quad i=1, \dots, d \end{aligned} \quad (\text{P-T})$$

Since the constraints $\mathbf{G} c_j \geq 0$ couple the entries within each row of C , and the residual constraints act within each column, the program (P-T) cannot be readily parallelized. Moreover, the large number of constraints and the potentially large number of neurons K make the direct solution of (P-T) computationally expensive. To overcome this, we employ a partially parallelizable block-coordinate descent approach, where we sequentially update the temporal component c_j of one neuron at a time; nonoverlapping components can be updated in parallel.

Incorporating Different Deconvolution Methods

The method for updating the temporal components C , f is a simple spatiotemporal extension of the CD method presented earlier. However, in principle, any time series deconvolution method can be used as part of our CNMF framework. To do so, we can construct for each component, a trace that represents the raw data, averaged over the corresponding spatial footprint, after the activity of all the other components has been removed. More specifically, if we want to compute the j -th temporal component at the k -th iteration $c_j^{(k)}$, provided we have computed $c_j^{(k)}, \dots, c_{j-1}^{(k)}, c_{j+1}^{(k)}, \dots, c_K^{(k)}$ we can form the quantity

$$\begin{aligned} \left(\mathbf{y}_j^{(k)}\right)^T &= \frac{\left(\mathbf{a}_j^{(k)}\right)^T}{\|\mathbf{a}_j^{(k)}\|^2} \left(Y - \left[\mathbf{a}_1^{(k)}, \dots, \mathbf{a}_{j-1}^{(k)}, \mathbf{a}_{j+1}^{(k)}, \dots, \mathbf{a}_K^{(k)}\right] \left[c_1^{(k)}, \dots, c_{j-1}^{(k)}, c_{j+1}^{(k)}, \dots, c_K^{(k)}\right]^T\right. \\ &\quad \left. - \mathbf{b}^{(k)} (\mathbf{f}^{(k-1)})^T\right), \end{aligned} \quad (\text{Equation 7})$$

and then apply our deconvolution method of choice to $\mathbf{y}_j^{(k)}$ to obtain $c_j^{(k)}$. This scheme adds flexibility and enables the different treatment of different components, in terms of time constants/dynamics, noise levels, etc.

Merging Existing Components

At the end of each iteration, we seek spatially overlapping components with temporal correlations above a certain threshold. These components are then merged into a single component. Details are given in the [Supplemental Experimental Procedures](#).

Ranking and Removing Components

Identifying the number of components a priori is a challenging problem. While the GL initialization approach (see below) offers an adaptive method to estimate this number, and the merge operations aim to correct for false estimates, it is usually difficult to automatically identify all the correct components without any false positives. As a strategy we have found that it is more effective to overestimate the number of components and then remove those that do not correspond to neural activity. To do this, we sort the components according to the product of the maximum values of the temporal and spatial components. Empirically, we find that the first components in the list have compact footprints and strong temporal activity, making it easy for the user to set a cutoff \bar{K} above which the obtained activity is retained, and below which the

components are discarded. The discarded components can then be absorbed into the background.

Greedy Initialization for Somatic Imaging Data

At every iteration, the residual of the spatiotemporal data matrix (or, in the first iteration, the data matrix itself) is spatially filtered with a Gaussian kernel of width similar to the size of a cell body. The algorithm finds the location where this filtering procedure explains the maximum variance and selects a square of size roughly twice the size of an average neuron. Within this square, a rank-1 NMF is performed to initialize the spatial and temporal components, and the product of these components is then subtracted from the observed data to form the new spatiotemporal residual. This procedure is repeated until a user-specified number of components is located; note that the different components obtained across all iterations are allowed to overlap. Then, the resulting residual signal is used to initialize the background component $B = bf^T$, again using rank-1 NMF (see [Supplemental Experimental Procedures](#) for implementation details).

Group Lasso Initialization for Somatic Imaging Data

The greedy initialization method can be problematic if neurons are packed too densely, especially if their activities are highly correlated. In this case, we use a different approach, where we initially assume that a potential neuron with a Gaussian shape of fixed size is centered at every observed voxel, and denote by X the $d \times T$ activity matrix of all these potential neurons. We then seek to minimize the number of active neurons that can explain the observed spatiotemporal data. We adopt a GL ([Yuan and Lin 2006](#)) approach, and form the following convex optimization problem

$$\min_{X \in \mathbb{R}_+^{d \times T}, b \in \mathbb{R}^d} \|Y - DX - b1_T^T\|_F^2 + \lambda \sum_i \sqrt{\sum_t X_{it}^2}, \quad (\text{Equation 8})$$

where again the data is given by $Y \in \mathbb{R}^{d \times T}$, λ is a regularization constant, and $D \in \mathbb{R}^{d \times d}$ is a matrix which performs convolution with a Gaussian kernel in the original spatial coordinates. We infer the underlying neuronal activity X and the background component $b1_T^T$. We use each column of D (which corresponds to a Gaussian kernel at each location) as a hypothetical neuron and select the significant locations by solving a GL problem Equation 8, where each group corresponds to the temporal activity of a hypothetical neuron. The GL penalty $\lambda \sum_i \sqrt{\sum_t X_{it}^2}$ forces locations with sufficiently low activity to have precisely zero activity ($\sum_t X_{it}^2 = 0$ for these locations i). The resulting “significant” locations i , in which $X_{it} > 0$, give an estimate of a neuron’s spatial footprint. However, the neuron centers cannot be identified simply by using all the nonzero locations, since one real neuron usually makes all the nearby locations have significant activity. Therefore, we identify regional maxima of $\sum_t X_{it}^2$ as the neuron centers. Note that by moderately sacrificing convexity, the procedure can be extended to include a time-varying baseline component f . The initialization to the matrix factorization approach is then given by the detected \hat{K} components, each with a Gaussian shape a_k centered around the identified neuron centers and activity c_k given by the averaging $X(i, :)$ over all non-zero locations i , which are closest to the center. Implementation details are given in the [Supplemental Experimental Procedures](#).

SUPPLEMENTAL INFORMATION

Supplemental Information includes Supplemental Experimental Procedures, two figures, and nine movies and can be found with this article online at <http://dx.doi.org/10.1016/j.neuron.2015.11.037>.

AUTHOR CONTRIBUTIONS

E.A.P. and L.P. devised the project. Methods were developed by E.A.P. (CD, CNMF), D.S. (PSD analysis), D.S. and D.P. (GL initialization), and Y.G. (greedy initialization), all with guidance from L.P. Data analysis was performed by E.A.P. (spinal motor neuron data, mouse V1 data, rodent barrel cortex data, publicly available ground truth data) and D.S., D.P., and Y.G. (zebrafish data). T.M., T.R., and T.J. provided the spinal motor neuron data. W.Y., D.S.P., and R.Y. provided the mouse V1 data. Y.M. and M.A. provided the zebra-

fish data. C.L. and R.B. provided the barrel cortex dendritic imaging data. Code was developed by E.A.P. with useful input from T.M. (Python implementation of CD), D.S. (MATLAB and Python implementation of GL initialization), Y.G. (greedy initialization), J.M. (time constant sampling), and W.Y. and D.S.P. (demo and documentation). E.A.P., D.S., and L.P. prepared the manuscript, with review and editing by D.S.P.

ACKNOWLEDGMENTS

We thank D. Chklovskii, J. Freeman, J. Friedrich, A. Giovannucci, L. Grosenick, P. Kaifosh, P. Mineault, A. Packer, and B. Shababo for useful discussions, and Y. Shin for technical assistance. This work was partially supported by the Gruss Lipper Charitable Foundation (D.S.), and grants ARO MURI W911NF-12-1-0594 (L.P. and R.Y.), DARPA W911NF-14-1-0269 (L.P. and R.Y.), NSF CAREER IOS-0641912 (L.P.), DARPA N66001-15-C-4032 (SIMPLEX) (L.P. and R.Y.), and NIDA 5R21DA034195 (R.Y.), NIMH 1R01MH101218 (R.Y.), NIMH R41MH100895 (R.Y.), NEI DP1EY024503 (R.Y.), NEI R01EY011787 (R.Y.) and NINDS R21NS081393 (R.Y.). Additional support was provided by the Gatsby Foundation.

Received: March 10, 2015

Revised: July 27, 2015

Accepted: November 23, 2015

Published: January 7, 2016

REFERENCES

- Ahrens, M.B., Orger, M.B., Robson, D.N., Li, J.M., and Keller, P.J. (2013). Whole-brain functional imaging at cellular resolution using light-sheet microscopy. *Nat. Methods* **10**, 413–420.
- Chen, T.W., Wardill, T.J., Sun, Y., Pulver, S.R., Renninger, S.L., Baohan, A., Schreiter, E.R., Kerr, R.A., Orger, M.B., Jayaraman, V., et al. (2013). Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature* **499**, 295–300.
- Cotton, R.J., Froudarakis, E., Storer, P., Saggau, P., and Tolias, A.S. (2013). Three-dimensional mapping of microcircuit correlation structure. *Front. Neural Circuits* **7**, 151.
- Diego-Andilla, F., and Hamprecht, F.A. (2013). Learning multi-level sparse representations. In *Advances in Neural Information Processing Systems 26 (NIPS 2013)*, C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, eds. (Curran Associates), pp. 818–826.
- Diego-Andilla, F., and Hamprecht, F.A. (2014). Sparse space-time deconvolution for calcium image analysis. In *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, eds. (Curran Associates), pp. 64–72.
- Duemani Reddy, G., Kelleher, K., Fink, R., and Saggau, P. (2008). Three-dimensional random access multiphoton microscopy for functional imaging of neuronal activity. *Nat. Neurosci.* **11**, 713–720.
- Freeman, J., Vladimirov, N., Kawashima, T., Mu, Y., Sofroniew, N.J., Bennett, D.V., Rosen, J., Yang, C.T., Looger, L.L., and Ahrens, M.B. (2014). Mapping brain activity at scale with cluster computing. *Nat. Methods* **11**, 941–950.
- Grewen, B.F., Langer, D., Kasper, H., Kampa, B.M., and Helmchen, F. (2010). High-speed *in vivo* calcium imaging reveals neuronal network activity with near-millisecond precision. *Nat. Methods* **7**, 399–405.
- Haeffele, B., Young, E., and Vidal, R. (2014). Structured low-rank matrix factorization: optimality, algorithm, and applications to image processing. In *Proceedings of the 31st International Conference on Machine Learning*, E.P. Xing and T. Jebara, eds., pp. 2007–2015. <http://jmlr.org/proceedings/papers/v32/haeffele14.pdf>.
- Kaifosh, P., Zaremba, J.D., Danielson, N.B., and Losonczy, A. (2014). SIMA: Python software for analysis of dynamic fluorescence imaging data. *Front. Neuroinform.* **8**, 80.
- Machado, T.A., Pnevmatikakis, E., Paninski, L., Jessell, T.M., and Miri, A. (2015). Primacy of Flexor Locomotor Pattern Revealed by Ancestral Reversion of Motor Neuron Identity. *Cell* **162**, 338–350.

- Maruyama, R., Maeda, K., Moroda, H., Kato, I., Inoue, M., Miyakawa, H., and Aonishi, T. (2014). Detecting cells using non-negative matrix factorization on calcium imaging data. *Neural Netw.* **55**, 11–19.
- Mukamel, E.A., Nimmerjahn, A., and Schnitzer, M.J. (2009). Automated analysis of cellular signals from large-scale calcium imaging data. *Neuron* **63**, 747–760.
- Nikolenko, V., Watson, B.O., Araya, R., Woodruff, A., Peterka, D.S., and Yuste, R. (2008). SLM Microscopy: Scanless Two-Photon Imaging and Photostimulation with Spatial Light Modulators. *Front. Neural Circuits* **2**, 5.
- Oñativia, J., Schultz, S.R., and Dragotti, P.L. (2013). A finite rate of innovation algorithm for fast and accurate spike detection from two-photon calcium imaging. *J. Neural Eng.* **10**, 046017.
- Pachitariu, M., Packer, A.M., Pettit, N., Dalgleish, H., Hausser, M., and Sahani, M. (2013). Extracting regions of interest from biological images with convolutional sparse block coding. In *Advances in Neural Information Processing Systems 26 (NIPS 2013)*, C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, eds. (Curran Associates), pp. 1745–1753.
- Pnevmatikakis, E.A., and Paninski, L. (2013). Sparse nonnegative deconvolution for compressive calcium imaging: algorithms and phase transitions. In *Advances in Neural Information Processing Systems 26 (NIPS 2013)*, C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, eds. (Curran Associates), pp. 1250–1258.
- Pnevmatikakis, E.A., Merel, J., Pakman, A., and Paninski, L. (2013). Bayesian spike inference from calcium imaging data. In *2013 Asilomar Conference on Signals, Systems and Computers*, pp. 349–353, <http://dx.doi.org/10.1109/ACSSC.2013.6810293>.
- Prevedel, R., Yoon, Y.-G., Hoffmann, M., Pak, N., Wetzstein, G., Kato, S., Schrödel, T., Raskar, R., Zimmer, M., Boyden, E.S., and Vaziri, A. (2014). Simultaneous whole-animal 3D imaging of neuronal activity using light-field microscopy. *Nat. Methods* **11**, 727–730.
- Rochefort, N.L., Garaschuk, O., Milos, R.I., Narushima, M., Marandi, N., Pichler, B., Kovalchuk, Y., and Konnerth, A. (2009). Sparsification of neuronal activity in the visual cortex at eye-opening. *Proc. Natl. Acad. Sci. USA* **106**, 15049–15054.
- Smith, S.L., and Häusser, M. (2010). Parallel processing of visual space by neighboring neurons in mouse visual cortex. *Nat. Neurosci.* **13**, 1144–1149.
- Theis, L., Berens, P., Froudarakis, E., Reimer, J., Rosón, M.R., Baden, T., Euler, T., Tolias, A., and Bethge, M. (2015). Supervised learning sets benchmark for robust spike detection from calcium imaging signals. arXiv, arXiv:1503.00135, <http://arxiv.org/abs/1503.00135>.
- Vogelstein, J.T., Watson, B.O., Packer, A.M., Yuste, R., Jedynak, B., and Paninski, L. (2009). Spike inference from calcium imaging using sequential Monte Carlo methods. *Biophys. J.* **97**, 636–655.
- Vogelstein, J.T., Packer, A.M., Machado, T.A., Sippy, T., Babadi, B., Yuste, R., and Paninski, L. (2010). Fast nonnegative deconvolution for spike train inference from population calcium imaging. *J. Neurophysiol.* **104**, 3691–3704.
- Yang, W., Miller, J.K., Carrillo-Reid, L., Pnevmatikakis, E.A., Paninski, L., Yuste, R., and Peterka, D.S. (2015). Simultaneous multi-plane imaging of neural circuits. *Neuron* **89**, this issue, 269–284.
- Yuan, M., and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Series B Stat. Methodol.* **68**, 49–67.

Neuron

Supplemental Information

Simultaneous Denoising, Deconvolution, and Demixing of Calcium Imaging Data

**Eftychios A. Pnevmatikakis, Daniel Soudry, Yuanjun Gao, Timothy A. Machado, Josh
Merel, David Pfau, Thomas Reardon, Yu Mu, Clay Lacefield, Weijian Yang, Misha
Ahrens, Randy Bruno, Thomas M. Jessell, Darcy S. Peterka, Rafael Yuste, and Liam
Paninski**

Contents

S1. Supplemental Data	2
S1.1. Supplemental Figures	2
S1.2. Captions for supplementary movies	3
S2. Supplemental Experimental Procedures	4
S2.1. Experimental data	4
S2.2. Algorithmic details	5
S2.2.1. Parameter estimation and AR modeling	5
S2.2.2. Algorithms for solving the one-dimensional constrained deconvolution problem	6
S2.2.3. Continuous time interpretation of AR models	7
S2.2.4. Updating the time constants	8
S2.2.5. Merging of existing components	8
S2.2.6. Description of the initialization procedures in somatic imaging	9
S2.2.7. Handling missing data	10
S2.2.8. Extraction of DF/F values	11
S2.2.9. Further algorithmic speedups	11
S2.3. Details of the data analysis	11
S2.3.1. Details of the application to spinal cord data of Fig. 1	11
S2.3.2. Details of the simulated experiment of Fig. 2	12
S2.3.3. Details of the simulated experiment of Fig. 3	12
S2.3.4. Details of the application to the large scale V1 data (Fig. 4)	12
S2.3.5. Details of the application to the light-sheet imaging data (Figs. 5-6)	13
S2.4. Application of constrained calcium deconvolution to datasets with available ground truth (Fig. S1) . .	13

S1 Supplemental Data

S1.1 Supplemental Figures

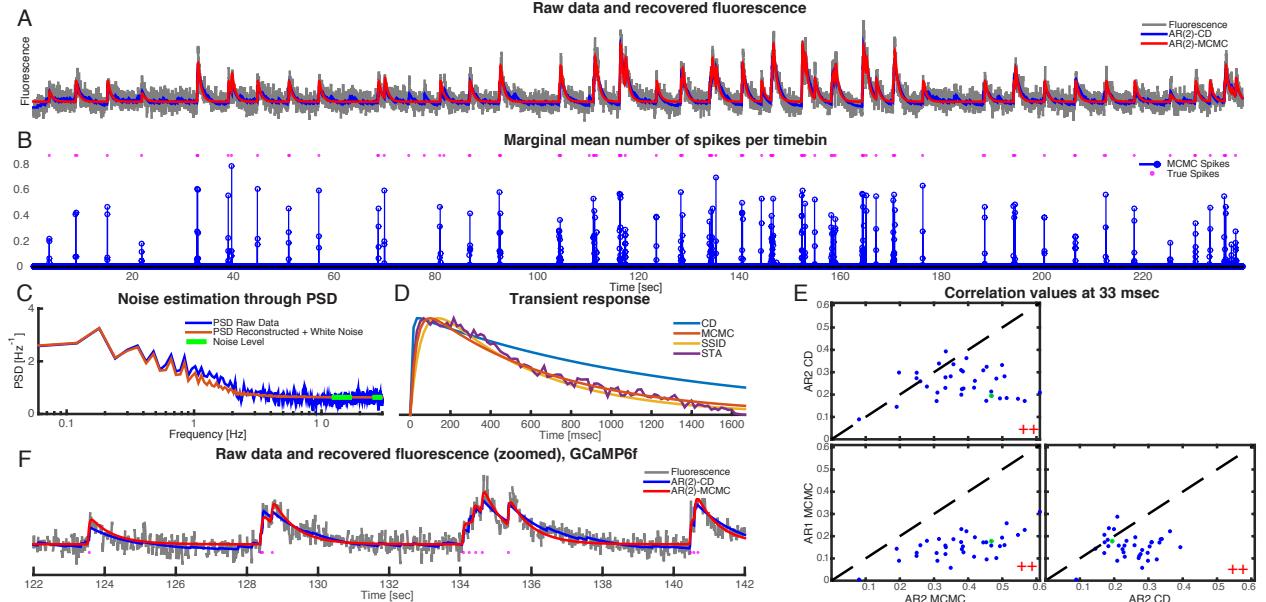


Figure S1, related to Figure 1. Application of the constrained deconvolution algorithm with available ground truth spiking data from the publicly available dataset (GENIE project, Janelia Farm Campus, HHMI; Karel Svoboda (contact), 2015), and illustration of the parameter identification process ($N = 37$ cells). A: Raw fluorescence data and reconstructed traces with constrained deconvolution (blue) and the MCMC (red) methods. B: Reconstructed spike train with MCMC (blue stem plot) and true spikes (purple dots). C: Noise level estimation. The noise level (green) is estimated from the PSD of the raw fluorescence (blue) at high frequencies. The PSD of the reconstructed trace with added white noise of the estimated level (red) matches the PSD of the raw trace. D: Estimation of the spike evoked transient response function with an AR(2) framework. Blue: estimated transient from the sample autocorrelation function (eq. (3) in the main paper). Adaptation of the time constants with the Metropolis-Hastings algorithm within MCMC (red trace). Optimal AR(2) estimate using the ground truth data and systems identification methods (yellow). Normalized spike triggered average (STA) response using the ground truth data (magenta). The sampling methods can approximate the optimal response function as is estimated from the ground truth data. F: Zoomed version of raw and reconstructed data and superimposed spikes. Re-estimation of the time constants improves the modeling of the fluorescence dynamics. E: Correlation scatter plot matrix at 33 msec resolution ($2 \times$ timebin width) for three different methods: AR(2)-MCMC performs better than AR(2)-CD. Both methods outperform all AR(1) methods (here shown AR(1)-MCMC), establishing that modeling the rise time can significantly improve the quality of deconvolution. Comparisons were done with the Wilcoxon signed-rank test at the 0.05 level.

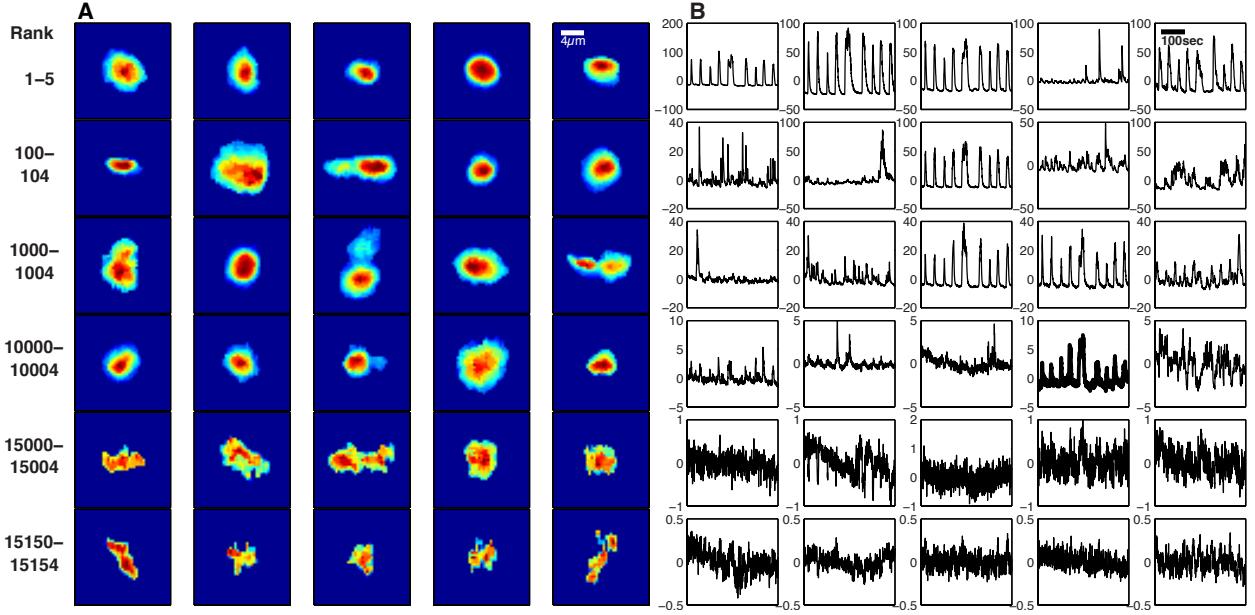


Figure S2, related to Figure 5. Inferred neuronal shapes A_k and DF/F temporal activity using the PCA/ICA algorithm (a subsample from a total of 19013 components detected in whole zebrafish brain). These samples were selected according to the ranking procedure described in the text.

S1.2 Captions for supplementary movies

Movie S1 (related to Figure 4): Application to GCaMP6s-expressing neurons in cortical layer 2/3 of adult mouse V1. Top row: Left: Raw data, Middle: Denoised data with synchronized background activity, Right: Residual signal at $2\times$ finer scale (the synchronized background activity is not included in the residual). Bottom left: Background synchronized activity. For the rest of the panels, 4 representative extracted spatiotemporal components (top) and the corresponding patches of the raw data. The contour plots indicate the locations of the representative components in the field of view. The algorithm successfully denoises the signal and demixes the overlapping neurons.

Movie S2 (related to Figure 6): Video of patch A in Fig. 6, with calcium signals from whole zebrafish recording, obtained with CNMF. Video contains: Raw data, background (**b** from Algorithm S3), denoised video containing only inferred neuronal activity, and the residual (data with Background and inferred neurons removed). Also, we zoom in on components ranked 1st, 10th, 20th, 30th, 40th and 50th in each patch (from left to right). For each neuron we show the inferred shape, the centralized data (with background **b** removed), and the raw data. For the first two, a smaller (occasionally saturated) color scale was used to emphasize very faint (and sometimes briefly firing) neurons, which are also detected by the algorithm.

Movie S3 (related to Figure 6): Video of patch C in Fig. 6, with calcium signals from whole zebrafish recording, obtained with CNMF. Panels similar as in Movie S2.

Movie S4 (related to Figure 6): Video of patch E in Fig. 6, with calcium signals from whole zebrafish recording, obtained with CNMF. Panels similar as in Movie S2.

Movie S5 (related to Figure 6): Video of patch B in Fig. 6, with calcium signals from whole zebrafish recording, obtained with PCA/ICA (Mukamel et al., 2009). Panels similar as in Movie S2.

Movie S6 (related to Figure 6): Video of patch D in Fig. 6, with calcium signals from whole zebrafish recording, obtained with PCA/ICA (Mukamel et al., 2009). Panels similar as in Movie S2. Note only 40 neurons were detected in this patch, so there is no 50th neuron.

Movie S7 (related to Figure 6): Video of patch B in Fig. 6, with calcium signals from whole zebrafish recording, obtained with PCA/ICA (Mukamel et al., 2009). Panels similar as in Movie S2.

Movie S8 (related to Figure 5): Centers of extracted neurons in the zebrafish brain obtained through CNMF. The distribution of visible neurons in the zebrafish whole-brain fluorescence corresponds well with the distribution of the components inferred using constrained NMF (cyan dots). Each frame in the movie is a 2D horizontal slice going upwards in the (dorsal) z direction, with $8\text{ }\mu\text{m}$ spacing. Each pixel in each frame is the 95% percentile of the absolute fluorescence across time.

Movie S9 (related to Figure 7): Application to calcium signals from apical dendrites of cortical Layer 5 pyramidal neurons. Top row: Left: Raw data, Middle: Denoised data with the background and noisy components removed. Right: Residual signal at $2\times$ finer scale. Bottom panels: 11 of the spatiotemporal extracted components plus the background synchronized activity (lower right panel). The video contains only the frames where at least one of the displayed components is significantly active. The algorithm extracts rich and structured spatiotemporal components that are not visible by plain observation of the raw data.

S2 Supplemental Experimental Procedures

S2.1 Experimental data

Motor spinal neuron data (Fig. 1) Spinal motor neurons expressing the calcium indicator GCaMP6s were imaged in an isolated C57BL/6 mouse spinal cord preparation (aged 4 days postnatal). Ventral roots were stimulated to antidromically evoke patterns of neuronal firing that matched timing of the stimulus pulses (Machado et al., 2015). GCaMP6s expression was achieved following direct spinal cord injections of RV strain SAD-B19 expressing GCaMP6s (Addgene 40753). This vector was constructed and packaged using standard rescue techniques (Osakada et al., 2011). Two-photon imaging (940 nm excitation wavelength, 525/50 emission filter) was conducted at 14.6 Hz using a 20x objective (1.0 N.A; Olympus).

Mouse V1 data (Fig. 4): In vivo calcium imaging data collected from GCaMP6s-expressing neurons in layer 2/3 of the primary visual cortex of an adult mouse. The field of view of the movie was $270\mu\text{m}\times270\mu\text{m}$ and the recording frame rate was 10 fps, and the neurons were imaged through an open skull with a glass cranial window. Expression was achieved via viral injection of AAV1-hsyn-GCaMP6s, four weeks prior to imaging, which was carried out on a home built two-photon laser scanning microscope (see the companion paper, Yang et al. 2015) using a 25X NA 1.05 objective.

Zebrafish light-sheet imaging data (Figs. 5-6): Data obtained as described in Freeman et al. (2014).

Dendritic imaging data (Fig. 7): Calcium signals from apical dendrites of cortical Layer 5 pyramidal neurons were obtained by injecting AAV2/9-hSyn-FLEX-GCaMP6f (UPENN vector core) into the barrel cortex of Rbp4:Cre BAC transgenic mice (GENSAT). Two-photon imaging was performed at 4Hz with a 16x, 0.8NA lens (Nikon) at 940nm while mice performed a whisker-based object detection task (Lacefield and Bruno, 2013, SfN abstract). Resulting TIF stacks were motion corrected with a dynamic programming algorithm implemented in Kaifosh et al. (2014).

S2.2 Algorithmic details

S2.2.1 Parameter estimation and AR modeling

Remember the autoregressive model for the calcium dynamics and the observation noise model

$$\begin{aligned} c(t) &= \sum_{k=1}^p \gamma_k c(t-k) + s(t) \\ y(t) &= \alpha(c(t) + b) + \varepsilon(t), \quad \varepsilon(t) \sim \mathcal{N}(0, \sigma^2). \end{aligned} \quad (\text{S1})$$

Under the assumption of a homogeneous Poisson spiking process with $\mathbb{E}[s(t)] = \lambda$, and that the AR process is stationary (it is necessary that $\gamma_1 + \dots + \gamma_p < 1$), we have

$$\mu \triangleq \mathbb{E}[c(t)] = \frac{\lambda}{1 - \sum_{k=1}^p \gamma_k}. \quad (\text{S2})$$

For the auto covariance function C_c we have

$$\begin{aligned} C_c(\tau) &= \mathbb{E}[c(t+\tau)c(t)] - \mu^2 \\ &= \mathbb{E}\left[\left(\sum_{k=1}^p \gamma_k c(t+\tau-k) + s(t+\tau)\right) c(t)\right] - \mu^2 \\ &= \sum_{k=1}^p \gamma_k \mathbb{E}[c(t+\tau-k)c(t)] + \lambda\mu - \mu^2 \\ &= \sum_{k=1}^p \gamma_k (C_c(\tau-k) + \mu^2) + \left(1 - \sum_{k=1}^p \gamma_k\right) \mu^2 - \mu^2 \\ &= \sum_{k=1}^p \gamma_k C_c(\tau-k). \end{aligned} \quad (\text{S3})$$

Since the noise is white we have

$$C_y(\tau) = \alpha^2 C_c(\tau) + \sigma^2 \delta(\tau), \quad (\text{S4})$$

and combining (S3) with (S4) we derive

$$C_y(\tau) = \begin{cases} \sum_{k=1}^p \gamma_k C_y(\tau-j) - \sigma^2 \gamma_\tau, & 1 \leq \tau \leq p \\ \sum_{k=1}^p \gamma_k C_y(\tau-j), & \tau > p. \end{cases} \quad (\text{S5})$$

The AR coefficients and noise variance can be estimated from the above equations using the sample auto-covariance. Although this method works well for estimating the noise variance in model data, in practice a more robust way to estimate σ^2 is independently by using the power spectral density (PSD) of y . This method is somewhat less dependent on parametric model assumptions about the data. Due to the slow decay dynamics of the calcium indicator, the AR process acts typically as a low pass filter on the incoming spikes, and therefore the noiseless calcium trace has very low power in the high frequency range. Since the noise is assumed to be white, its PSD is flat among all frequencies. To estimate σ^2 we compute the PSD of y and then average its value at the range of high frequencies (e.g. in the range $[F_s/4, F_s/2]$), where F_s is the imaging rate. This point is illustrated in more detail in Fig. S1C. A trace from the GCaMP6f dataset presented above is used to illustrate the parameter identification process. The noise level was estimated by averaging the PSD of the raw trace (computed using Welch's method) over a range of high frequencies $[F_s/4, F_s/2]$ (Fig. S1C). The order of the AR process p is typically low ($p=1, 2$), and is determined by the imaging rate. Alternatively, given a noise level, p can be determined as the minimum order such that the deconvolution problem accepts a feasible solution.

S2.2.2 Algorithms for solving the one-dimensional constrained deconvolution problem

We briefly discuss the four different approaches that can be used for the constrained deconvolution problem, which we repeat here for completeness:

$$\begin{aligned} & \underset{\mathbf{c}}{\text{minimize}} && \mathbf{1}_T^T G(\mathbf{c} - \mathbf{c}_{\text{in}}), \\ & \text{subject to:} && G(\mathbf{c} - \mathbf{c}_{\text{in}}) \geq 0, \quad c_1 \geq 0 \\ & && \|\mathbf{y} - \mathbf{c} - b\mathbf{1}_T\| \leq \sigma\sqrt{T}. \end{aligned} \tag{P-CD}$$

Here \mathbf{c}_{in} models the initial calcium concentration, a parameter that was omitted in the main text for simplicity. The initial concentration at time $t = 1$, is modeled as c_1 , and \mathbf{c}_{in} is defined as the vector $\mathbf{c}_{\text{in}} = c_1[1, \gamma, \dots, \gamma^T]^\top$ that models the effect of the initial concentration at the observed time points. The time constant γ models the decay rate of calcium transients initiated before the start of the experiment, as we choose it to be equal to longest decay time constant of the calcium indicator. Using the AR framework, γ is equal to the largest root of the characteristic polynomial $\lambda^p - \gamma_1\lambda^{p-1} - \dots - \gamma_p = 0$.

For simplicity of the presentation, we assume that the baseline b and initial concentration c_1 are known, although all methods can be trivially modified to include estimation of these parameters. MATLAB implementations for all methods can be found in <https://github.com/epnev/constrained-foopsi>.

Dual ascent methods: We introduce Lagrange multipliers for the constraints and define as \mathbf{c}^λ as the solution to the following program

$$\begin{aligned} & \underset{\mathbf{c}}{\text{minimize}} && \mathcal{L}(\mathbf{c}, \lambda) = \mathbf{1}^T G(\mathbf{c} - \mathbf{c}_{\text{in}}) + \lambda(\|\mathbf{y} - \mathbf{c} - b\mathbf{1}_T\|^2 - \sigma^2 T), \\ & \text{subject to:} && G(\mathbf{c} - \mathbf{c}_{\text{in}}) \geq 0. \end{aligned} \tag{S6}$$

The problem (S6) can be readily solved in $O(T)$ time with the interior point method of Vogelstein et al. (2010). After solving (S6), the Lagrange multiplier can be updated as

$$\lambda_k = \lambda_{k-1} - a_k \nabla_\lambda \mathcal{L}(\mathbf{c}^{\lambda_{k-1}}, \lambda) = \lambda_{k-1} - a_k(\|\mathbf{y} - \mathbf{c} - b\mathbf{1}_T\|^2 - \sigma^2 T), \tag{S7}$$

where a_k is an appropriate step size, determined e.g. by line search.

Conic programming: The program can also be solved with standard interior point methods for conic programming (Boyd and Vandenberghe, 2004). Due to the simplicity of the residual and non-negativity constraints the solution can be efficiently computed in $O(T)$ time using standard computational methods, e.g. the CVX computational package (Grant et al., 2008).

Nonnegative LARS: The problem can also be solved directly in the spike domain using a nonnegative LARS algorithm (Efron et al., 2004). More specifically we consider the modified problem in the spike domain as follows

$$\begin{aligned} & \underset{\mathbf{s}}{\text{minimize}} && \frac{1}{2\sigma^2} \|\mathbf{y} - G^{-1}\mathbf{s} - b\mathbf{1}_T - \mathbf{c}_{\text{in}}\|^2 + \lambda \mathbf{1}^T \mathbf{s}, \\ & \text{subject to:} && \mathbf{s} \geq 0. \end{aligned} \tag{S8}$$

Instead of putting a hard noise constraint as before the LARS algorithm computes the solution path of (S8) for all λ and stops when the noise constraint is satisfied. This process is efficient since the solution is piecewise linear in λ . We start from $\lambda_0 = \infty$ where $\mathbf{s} = \mathbf{0}_T$. As λ decreases, more spikes are added in the solution, reducing the l_2 -norm of the residual signal. Given the solution at the $(k-1)$ -th step, at the k -th step, the algorithm includes an additional spike (or removes an existing one), and then optimizes over the spike heights. The path algorithm is stopped when the produced solution satisfies the residual constraint with equality, with a total computational cost $O(TN^2 + N^3)$, where N is the total number of steps. Note that the total number of nonzero spikes is at most k at the k -th step, and therefore $\leq N$ upon termination. Thus this method is particularly efficient and preferred when the spiking signal is expected to be highly sparse.

Spectral projected gradient methods: This method relies on the observation that the curve that characterizes the trade-off between the l_2 norm of the residual ($\|\mathbf{y} - \mathbf{c} - b\mathbf{1}_T^\top \mathbf{s}\|$), and the sum of the spiking signal ($\mathbf{1}_T^\top \mathbf{s}$), is convex, and uses an interior point method to explore this curve until the desired noise constraint is satisfied. More information can be found in van den Berg and Friedlander (2008). For our case, this method is highly efficient due to our ability to perform fast matrix vector operations with the banded and Toeplitz matrix G , because of the autoregressive model for the calcium dynamics.

S2.2.3 Continuous time interpretation of AR models

We discuss briefly the continuous time interpretation of our autoregressive framework to connect the AR coefficients with some biophysical properties of the calcium indicators. The following discussion is fairly standard, and a more thorough exposition can be found in any standard linear systems textbook (e.g. Oppenheim and Willsky, 1997). An autoregressive model of order p can be written as a discrete time linear dynamical system as follows

$$\underbrace{\begin{bmatrix} c[n] \\ c[n-1] \\ \vdots \\ c[n-p+1] \end{bmatrix}}_{\mathbf{c}_d[n]} = \underbrace{\begin{bmatrix} \gamma_1 & \gamma_2 & \dots & \gamma_p \\ 1 & 0 & \dots & 0 \\ \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 1 & 0 \end{bmatrix}}_{A_d} \underbrace{\begin{bmatrix} c[n-1] \\ c[n-2] \\ \vdots \\ c[n-p] \end{bmatrix}}_{\mathbf{c}_d[n-1]} + \underbrace{\begin{bmatrix} s[n] \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{\mathbf{s}[n]}. \quad (\text{S9})$$

This can be mapped to the following continuous time dynamical system

$$\dot{\mathbf{c}}_c(t) = A_c \mathbf{c}_c(t) + \frac{1}{h(\Delta)} \mathbf{s}(t), \quad (\text{S10})$$

with

$$\begin{aligned} A_d &= \exp(A_c \Delta), \\ s(t) &= \frac{1}{h(\Delta)} \sum_n s[n] \delta(t - n\Delta), \end{aligned} \quad (\text{S11})$$

where Δ denotes the timebin width, $\delta(\cdot)$ the Dirac delta function, and $h(\cdot)$ is the Green's function of (S10). The eigenvalues of A_d are given by the roots of the characteristic polynomial $\lambda^p - \gamma_1 \lambda^{p-1} - \dots - \gamma_p = 0$ and the process is stable if all the eigenvalues are within the unit circle. We can apply this framework to model calcium transients with finite rise-time of the form

$$h(t) = \begin{cases} e^{-t/\tau_d} - e^{-t/\tau_r}, & t > 0 \\ 0, & t \leq 0 \end{cases}, \quad (\text{S12})$$

with an AR(2) process. The function $h(t)$ is the solution of the continuous time differential equation

$$\underbrace{\begin{bmatrix} \ddot{\mathbf{c}}(t) \\ \dot{\mathbf{c}}(t) \end{bmatrix}}_{\dot{\mathbf{c}}_c(t)} = \underbrace{\begin{bmatrix} -\left(\frac{1}{\tau_d} + \frac{1}{\tau_r}\right) & -\frac{1}{\tau_d \tau_r} \\ 1 & 0 \end{bmatrix}}_{A_c} \underbrace{\begin{bmatrix} \dot{\mathbf{c}}(t) \\ \mathbf{c}(t) \end{bmatrix}}_{\mathbf{c}_c(t)} + \begin{bmatrix} \delta(t) \\ 0 \end{bmatrix}. \quad (\text{S13})$$

If λ_1^c, λ_2^c are the eigenvalues of A_c , and $\lambda_1^d = \exp(\lambda_1^c \Delta), \lambda_2^d = \exp(\lambda_2^c \Delta)$, then $\gamma_1 = \lambda_1^d + \lambda_2^d$ and $\gamma_2 = -\lambda_1^d \lambda_2^d$. The conditions $\tau_d, \tau_r > 0$ imply that $\lambda_1^c, \lambda_2^c < 0 \Rightarrow 0 < \lambda_1^d, \lambda_2^d < 1$ which implies that $0 < \gamma_1 + \gamma_2 < 1$ and $-1 < \gamma_2 < 0$. λ_1^d, λ_2^d are real numbers when $\gamma_1^2 + 4\gamma_2 > 0$, which is the over-damping condition that prohibits oscillatory behavior in the calcium transient response. With this relationship between γ_1, γ_2 and τ_d, τ_r we have an exact mapping between the continuous and discrete time representations in the sense that

$$\mathbf{c}_c(n\Delta) = \mathbf{c}_d[n]. \quad (\text{S14})$$

and we also have $h(\Delta) = \sqrt{\gamma_1^2 + 4\gamma_2}$.

S2.2.4 Updating the time constants

As discussed in the main text, it is useful to refine the time constants of the AR process for each neuron. To do this we augmented the MCMC approach of Pnevmatikakis et al. (2013) to also include sampling of the time constants with a Metropolis-Hastings algorithm (Gelman et al., 2003). This method can fine tune the time constants of the indicator response, at a multiplicative computational cost, since the MCMC method by default draws a number of samples to approximate the full posterior distribution of the spike times and other model parameters.

We briefly describe the process for an AR(2) model of the indicator dynamics. For given time constants τ_d, τ_r in the continuous domain, we sample new values from Gaussian distribution $\tau_d^{\text{new}} \sim \mathcal{N}(\tau_d, \sigma_d^2)$, $\tau_r^{\text{new}} \sim \mathcal{N}(\tau_r, \sigma_r^2)$. We also impose a minimum value of τ_{\min} for the rise time, and a maximum value τ_{\max} for the decay time and require that $\tau_r \leq \tau_d$. The proposed time constant values $\tau_d^{\text{new}}, \tau_r^{\text{new}}$ give rise to a new calcium trace \mathbf{c}^{new} , and are accepted with probability

$$\mathbb{P}(\text{accept } \tau_d^{\text{new}}, \tau_r^{\text{new}}) = \min \left(1, \frac{\exp(-\|\mathbf{y} - \mathbf{c}^{\text{new}} - \mathbf{b}\|^2/2\sigma^2)}{\exp(-\|\mathbf{y} - \mathbf{c} - \mathbf{b}\|^2/2\sigma^2)} \right).$$

An algorithmic description of this procedure is depicted in Alg. S1. In the simpler AR(1) case, then $\tau_r = 0$, and we only need to sample τ_d . The method is also easily extendable to higher order models.

Algorithm S1 Neural activity deconvolution with time constant updating

Require: Data $\mathbf{y} \in \mathbb{R}^T$, number of samples N_{samples} for MCMC, timebin width Δ , lower and upper bound of time constants τ_{\min}, τ_{\max} , variance of proposal kernel σ_r^2, σ_d^2 .
Estimate γ_1, γ_2 from (S5), and σ^2 through PSD.
Convert γ_1, γ_2 in continuous time τ_r, τ_d

for $i = 1:N_{\text{samples}}$ **do**
 Draw new samples for $\mathbf{s}, \mathbf{c}, c_1, b, \sigma^2$ using the MCMC algorithm of Pnevmatikakis et al. (2013).
 Update time constants using **UPDATETIMECONSTANTS**($\mathbf{y}, \mathbf{c}, \mathbf{s}, b, c_1, \tau_r, \tau_d, \sigma^2, \sigma_r^2, \sigma_d^2, \tau_{\min}, \tau_{\max}$)
end for
return all samples

procedure **UPDATETIMECONSTANTS**($\mathbf{y}, \mathbf{c}, \mathbf{s}, b, c_1, \tau_r, \tau_d, \sigma^2, \sigma_r^2, \sigma_d^2, \tau_{\min}, \tau_{\max}$)
 Draw new time constants

$$\begin{bmatrix} \tau_r^{\text{new}} \\ \tau_d^{\text{new}} \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \tau_r \\ \tau_d \end{bmatrix}, \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_d^2 \end{bmatrix} \right), \quad \text{with } \tau_{\min} \leq \tau_r^{\text{new}} \leq \tau_d^{\text{new}} \leq \tau_{\max}.$$

 Construct proposed calcium trace \mathbf{c}^{new} .
 Draw $r \sim U([0, 1])$
 if $r < \min \left(1, \frac{\exp(-\|\mathbf{y} - \mathbf{c}^{\text{new}} - \mathbf{b}\|^2/2\sigma^2)}{\exp(-\|\mathbf{y} - \mathbf{c} - \mathbf{b}\|^2/2\sigma^2)} \right)$ **then**
 $\tau_r \leftarrow \tau_r^{\text{new}}, \quad \tau_d \leftarrow \tau_d^{\text{new}}$
 end if
return τ_r, τ_d
end procedure

S2.2.5 Merging of existing components

Depending on the initialization procedure, a neuron can sometimes be initially split into two or more different spatial components that subsequently need to be merged. To detect such components, we construct a graph where each vertex corresponds to a neuron and two neurons are connected with an edge if their spatial components overlap. For this graph we detect all the maximal cliques, i.e., the cliques of the graph that are not part of larger cliques. (Fast approximate

methods effective for large sparse graphs exist for this problem (Eppstein et al., 2010).) Now for each of these maximal cliques we compute the correlation matrix of the temporal components of the corresponding nodes. We find the largest principal submatrix where all the correlation coefficients are above a certain threshold, and merge the corresponding spatial components.

S2.2.6 Description of the initialization procedures in somatic imaging

Algorithm S2 Greedy neuron identification

Require: Data $Y \in \mathbb{R}^{d \times T}$; number of neurons needed K ; standard deviation of the 2-d Gaussian kernel used for filtering $\tau = (\tau_x, \tau_y)$; window size $w = (w_x, w_y)$.

procedure GREEDYNEURONID(Y, K, τ, w)

$$R = Y;$$

Define Gaussian blur matrix $D \in \mathbb{R}^{d \times d}$, where column i is a (vectorized) truncated 2-d Gaussian kernel centered at pixel i with variance (τ_x^2, τ_y^2) , supported in a $w_x \times w_y$ window centered at i ($1 \leq i \leq d$);

for $i = 1 : d$ **do**

Subtract and store median value for each pixel, $m(i) = \text{Median}(Y(i, :))$.

end for

for $k = 1 : K$ **do**

Calculate variance explained by each kernel, $\rho = D^T R$, $v_i = \sum_{t=1}^T \rho_{it}^2$;

Identify the center of neuron k , $i_k = \arg \max_i v_i$

Define S_k to be the set of all pixels lie in the $w_x \times w_y$ window centered at i_k ; solve by alternating least squares

$$\begin{aligned} & \underset{\mathbf{a}_k \in \mathbb{R}^d, \mathbf{c}_k \in \mathbb{R}^T}{\text{minimize}} \quad \|R - \mathbf{a}_k \mathbf{c}_k^T\|^2 \\ & \text{subject to:} \quad a_k(i) \geq 0, i \in S_k \\ & \quad a_k(i) = 0, i \notin S_k. \end{aligned} \tag{S15}$$

$R(S_k) \leftarrow R(S_k) - \mathbf{a}_k \mathbf{c}_k^T$;

end for

$R \leftarrow R + \mathbf{m1}_T^T$. Add median values back to the residual and solve by alternating least squares

$$\begin{aligned} & \underset{\mathbf{b} \in \mathbb{R}^d, \mathbf{f} \in \mathbb{R}^T}{\text{minimize}} \quad \|R - \mathbf{b} \mathbf{f}^T\|^2 \\ & \text{subject to} \quad b(i) \geq 0, i = 1, \dots, d \\ & \quad f(t) \geq 0, t = 1, \dots, T. \end{aligned} \tag{S16}$$

return $A = [\mathbf{a}_1, \dots, \mathbf{a}_K]$, $C = [\mathbf{c}_1, \dots, \mathbf{c}_K]^T$, \mathbf{b}, \mathbf{f} .

end procedure

Algorithmic description for the greedy initialization procedure: Algorithm S2 provides full details of the greedy initialization procedure, which was described at a high level in the methods section. Several details are worth noting here. To begin we center the data at each pixel around zero, by subtracting the median over time. Also, since the spatial component S_k is localized, at each step only a small portion of the residual is updated and therefore only a small portion of the explained variance needs updating. Also note that the nonnegative matrix factorization step (equation (S15) in algorithm S2) can be efficiently done by alternating between optimizing \mathbf{a}_k and \mathbf{c}_k , since the Gaussian kernel scan gives a reasonable initialization. Usually ~ 5 iterations are enough for convergence. Since the solution of (S15) is identifiable up to a multiplicative scalar, we constrain the spatial components to have unit norm.

Algorithmic description for the Group Lasso initialization: In order to solve the convex optimization problem of eq. (8), we use the FISTA algorithm (Algorithm S3, derived from Eq. 4.1-4.3 on Beck and Teboulle (2009)), where $R = Y - \mathbf{b}\mathbf{1}_T^\top$ is the residual from the data after we subtract the background component. As explained in Beck and Teboulle (2009) a key parameter of the algorithm is the "Lipschitz constant," which in the setting considered here is twice the maximal eigenvalue of $D^\top D$. In our case, we use a filter matrix D whose sum is normalized to 1, and whose Fourier spectrum is strictly decreasing as a function of the magnitude of the frequency, so the maximal eigenvalue is 1. Therefore the Lipschitz constant is $L = 2$.

Next, we list a few implementation details. (1) It usually takes 30-100 iterations of the algorithm to converge. (2) All Gaussian dictionary elements have equal standard deviation τ and window size $w = 4\tau$, chosen to match the typical size of neurons in the image. (3) Convolution with the Gaussian Kernel is implemented efficiently using the Fast Fourier Transform. This is especially fast if graphical processing units (GPUs) are used. (4) The regularization constant λ can be automatically adjusted by requiring that the portion of significant locations (i.e. i in which $\sum_t F_{it}^2 \geq 0$) in the image is equal to the estimated neuronal density in the image area in Algorithm S3). This can be done efficiently using exponential search (which is an efficient generalization of binary search for unbounded lists). (5) If the neuronal shape detected by the matrix factorization approach has a center which is far (e.g., 2τ) from the original shape detected by the group lasso initialization, then we discard that shape.

Algorithm S3 FISTA for Group Lasso

Require: Residual $R \in \mathbb{R}^{d \times T}$; Initial points standard deviation of the 2-D Gaussian kernel $\tau = (\tau_x, \tau_y)$; Regularization constant λ ;

1: **procedure** GROUPLASSOFISTA(R, τ, α, λ)

2: Define Gaussian blur matrix $D \in \mathbb{R}^{d \times d}$, where column p is a (vectorized) truncated 2-D Gaussian kernel centered at p with variance (τ_x^2, τ_y^2) , and the operator

$$(\mathcal{T}_\mu(a))_{qt} \triangleq \max \left[a_{qt} \left(1 - \frac{\mu}{\sqrt{\sum_t a_{qt}^2}} \right), 0 \right].$$

3: Initialize $\mu = \lambda/L$, $L = 2$, $F_{(0)} = 0^{d \times T}$, $W_{(1)} = F_{(0)}$, $t_{(1)} = 1$, $M = \frac{2}{L}D^\top D$, $v = \frac{2}{L}D^\top R$.

4: **Repeat** until convergence for $k \geq 1$:

5: FISTA equations:

$$\begin{aligned} F_{(k)} &= \mathcal{T}_\mu [W_{(k)} - MW_{(k)} + v] \\ t_{(k+1)} &= (1 + \sqrt{1 + 4t_{(k)}^2}) / 2 \\ W_{(k+1)} &= F_{(k)} + \left(\frac{t_{(k)} - 1}{t_{(k+1)}} \right) (F_{(k)} - F_{(k-1)}) \end{aligned}$$

return F, b .

6: **end procedure**

S2.2.7 Handling missing data

Our method can easily handle the case of missing data that can arise in practice, e.g., due to brain movement during line scanning (Dombeck et al., 2007). Our framework remains the same, with the only difference that entries where observations are missing are omitted from the noise constraints, but are included in the constraints for the temporal dynamics. To compute the autocovariance function or PSD (for estimating the AR parameters and noise level) for a pixel with missing data, we interpolate the missing values using nearest neighbor interpolation

	Imaging rate	AR order	Initialization method	Include noise constraints
Somatic or axonal imaging	$f \geq 15\text{Hz}$	$p = 2$	Greedy/GL	Yes
	$5\text{Hz} \leq f \leq 15\text{Hz}$, slow indicator	$p = 1, 2$	Greedy/GL	Yes
	$5\text{Hz} \leq f \leq 15\text{Hz}$, fast indicator	$p = 1, 2$	Greedy/GL	Yes
	$f \leq 5\text{Hz}$	$p = 0$ (no deconvolution)	Greedy/GL	No
Dendritic Imaging	$f \geq 10\text{Hz}$	$p = 0$ (no deconvolution)	Sparse NMF	Yes
	$f \leq 5\text{Hz}$	$p = 0$ (no deconvolution)	Sparse NMF	No

Table S1, related to the Experimental Procedures. Recommended algorithm setting depending on imaged modality, imaging rate and calcium indicator.

S2.2.8 Extraction of DF/F values

Our approach enables us to express the temporal trace of each component in the DF/F domain, independently of the relative amplitude between the spatial and temporal components. To do so, we can assign a temporal background signal to each component, by averaging the spatiotemporal background over the spatial component:

$$\mathbf{f}_j^0 = (\mathbf{a}_j^\top (Y - AC))^\top,$$

and then express the DF/F values as $\mathbf{c}_j/m(\mathbf{f}_j^0)$, where $m(\cdot)$ is an appropriate summary statistic (e.g., mean, median, running average).

S2.2.9 Further algorithmic speedups

Further algorithmic speedups can be obtained during the initialization phase, by spatially downsampling the raw to obtain faster initial estimates and using specific block coordinate descent strategies during the CNMF iterations. Such approaches can lead to up to an order of magnitude speedups as highlighted in Friedrich et al. (2015) on zebrafish light-sheet imaging data.

S2.3 Details of the data analysis

Our proposed method is modular and comes with multiple different variants that can be used during the initialization and/or the alternating minimization process. Before we present the details for all the datasets and examples presented in the main paper, Table S1, related to the Experimental Procedures presents some general guidelines on what variant to use depending on the imaged modality, imaging rate and/or calcium indicator used.

S2.3.1 Details of the application to spinal cord data of Fig. 1

The constrained deconvolution problem (P-CD) was solved with conic programming using the CVX computational package. For updating the time constants, the standard deviations for the proposal density were chosen as $\sigma_d = 10\text{msec}$, $\sigma_r = 2\text{msec}$. Furthermore, the bounds were set as $\tau_{\min} = 0$, $\tau_{\max} = 2000\text{ms}$. For the MCMC algorithm, $N_{\text{samples}} = 500$ samples were drawn as described in Pnevmatikakis et al. (2013).

S2.3.2 Details of the simulated experiment of Fig. 2

The field of view was 50×50 pixels large. The two neurons had a 2-d spherical Gaussian shape with standard deviation 5 pixels and were centered on the same horizontal axis, 3 pixels apart, giving a high overlap between the two spatial footprints (correlation value = 0.9). Spikes were simulated from a Bernoulli process with probability of spiking per timebin 0.05. 2000 timebins were simulated, and the calcium was generated from a first order process with $\gamma = 0.8$, corresponding to a decay time constant 4.48 times the width of the timebin. We used a simple clustering procedure to initialize the estimates: after an iteration of the sparse constrained NMF, we clustered the extracted spiking signals according to a simple max-assignment

$$s_1^{\text{new}}(t) = \begin{cases} s_1^{\text{old}}(t) + s_2^{\text{old}}(t), & s_1^{\text{old}}(t) > s_2^{\text{old}}(t) \\ 0, & \text{otherwise} \end{cases},$$

and similarly for $s_2^{\text{new}}(t)$. These clustered temporal components were then used to warm-start the constrained NMF.

S2.3.3 Details of the simulated experiment of Fig. 3

The field of view was 50×50 pixels large. Simulations were performed with neurons that had two different shapes: (i) spherical 2-d Gaussians with standard deviation 5 pixels and (ii) "donut" shaped centered at $[x_0, y_0]$ with shape given by

$$\alpha(x, y) = \exp\left(-\frac{(\sqrt{(x - x_0)^2 + (y - y_0)^2} - r_0)^2}{2\sigma_r^2}\right),$$

with $r_0 = 4$, $\sigma_r = 1$. In each simulation 10 neurons were placed in the field with centers drawn from a spatial Poisson process with intensity function at each pixel chosen from a uniform distribution. This setup allowed for arbitrary spatial patterns and degree of overlap. Spikes were simulated from a Bernoulli process with probability of spiking per timebin 0.05. 2000 timebins were simulated, and the calcium was generated from a first order process with $\gamma = 0.9$, corresponding to a decay time constant 9.49 times the width of the timebin. White Gaussian noise was simulated to corrupt the data. The standard deviation for each pixel was equal to $\xi \times$ the mean activity, and thirty different noise levels were considered, $\xi = 0.1, 0.2, \dots, 3$. For each combination of noise level and neuron shape 5 simulations were performed. The plain NMF of Maruyama et al. (2014) searched for 10 neurons, with the temporal background fixed as described in Maruyama et al. (2014). For our constrained NMF framework, we used the greedy method to initialize the 10 components, and estimated the noise level and time constant from the data. To estimate the spiking signal for the PCA/ICA and plain NMF methods, the estimated temporal components were deconvolved from the true indicator dynamics (giving these methods a bit of a relative advantage). Fig. 3 reports the median spike correlation values among all 5 trials and 10 estimated neurons.

S2.3.4 Details of the application to the large scale V1 data (Fig. 4)

The greedy initialization algorithm was used to initialize 300 spatio-temporal components with size of filtering kernel 4×4 and window 10×10 . This number was chosen after visual inspection of the raw data. Then the CNMF framework was applied and the constrained deconvolution algorithm was used for estimating the temporal components. Since the imaging rate was relatively low (10Hz), an AR(1) process was used to model the temporal traces, and each component had a separate time constant that was estimated from the CD algorithm. At the end of each iteration, the merging procedure was applied with a merging threshold of 0.8. In total 31 components were merged. The remaining 269 components had time constants that are within a factor of two of the values reported in the literature for GCaMP6s (Chen et al., 2013), with a median value of 1180ms. The temporal traces were then transformed into the DF/F domain, and ordered according to the criterion explained in the Experimental Procedures.

S2.3.5 Details of the application to the light-sheet imaging data (Figs. 5-6)

The data is segmented to 5600 patches, and the algorithm is then run in parallel on all patches. In order to determine the value of the regularization parameter λ used for the group lasso initialization approach (8), we sample a few patches out of the whole data set, and find a single value of λ which gives reasonable results for all these patches. We then use this single value of λ to provide initial group lasso estimates for the whole brain. To prevent the detection of partial shape components we ignore any shape detected near the edge of the patch, and use overlapping patches to compensate. To do so, we remove any shape whose activity is highly correlated (above 0.95) to some other higher-ranked overlapping shape component.

Since the spatial components are restricted to lie within their corresponding spatial patches (which are not large compared to the size of the cell body), the inferred spatial components are localized (and therefore sparse) by construction. Thus there is no need to impose a spatial sparsity constraint. Similarly, because of the low temporal resolution of these recordings, the inferred neural activity vectors are not expected to be particularly sparse, and therefore we do not impose sparsity in the temporal domain either. This leads to a somewhat simplified optimization problem:

$$\begin{aligned} & \underset{A,C,\mathbf{b},\mathbf{f}}{\text{minimize}} \quad \|Y - \mathbf{b}\mathbf{f}^\top - AC\|^2 \\ & \text{subject to} \quad A_{xk} \geq 0, \quad \forall x \in S_k \\ & \quad A_{xk} = 0, \quad \forall x \notin S_k \\ & \quad C, \mathbf{b}, \mathbf{f} \geq 0 \end{aligned} \tag{S17}$$

where S_k denotes the k -th fixed spatial patch. We solve this problem by block-coordinate descent; upon convergence, we lightly smooth the inferred shape using median filtering. Because the resulting spatial components here were not quite as sparse as in the other examples (because we did not sparsen A within the patches S_k), we found it useful when ranking the obtained components to multiply the maximum value of the temporal components by the squared l_4 norm of the corresponding spatial footprints, to penalize overly broad and/or noisy spatial shapes.

S2.4 Application of constrained calcium deconvolution to datasets with available ground truth (Fig. S1)

We applied the constrained deconvolution algorithm to two publicly available datasets (GENIE project, Janelia Farm Campus, HHMI; Karel Svoboda (contact), 2015) with available ground truth. Prior to deconvolution the data was high pass filtered with a quantile filter that subtracted the 10th percentile value over a moving window with width 8.3sec (500 timebins, imaging rate 60Hz) to remove slowly variable baseline due to background/neuropil activity. Due to the high imaging rate, an AR(2) process was used to model both the rise and the decay of the calcium indicator spike triggered response function. The results are shown in Fig. S1. The benefits that can be derived from this time constant updating scheme are shown in Fig. S1D. The AR estimation method (eq. S5, blue trace) estimates $\tau_r = 8$ msec and $\tau_d = 1277$ msec, a much faster rise and slower decay than trace recovered if we compute the spike triggered average response (magenta trace), and does not fit data well (Fig. S1F). These estimates are corrected by the MCMC (red) approach, $\tau_r = 32$ msec, $\tau_d = 645$ msec, respectively. These traces match better the STA response. For reference applying the (supervised) n4sid systems identification method (Verhaegen and Verdult, 2007) to estimate a second order model gives $\tau_r = 62$ msec, $\tau_d = 518$ msec (yellow).

References

- Beck A., and Teboulle M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences 2, 183–202.
- Boyd S., and Vandenberghe L. (2004). Convex Optimization (Oxford University Press).
- Chen T.W., Wardill T.J., Sun Y., Pulver S.R., Renninger S.L., Baohan A., Schreiter E.R., Kerr R.A., Orger M.B., Jayaraman V., et al. (2013). Ultrasensitive fluorescent proteins for imaging neuronal activity. Nature 499, 295–300.

- Dombeck D.A., Khabbaz A.N., Collman F., Adelman T.L., and Tank D.W. (2007). Imaging large-scale neural activity with cellular resolution in awake, mobile mice. *Neuron* *56*, 43–57.
- Efron B., Hastie T., Johnstone I., Tibshirani R., et al. (2004). Least angle regression. *Ann Statist* *32*, 407–499.
- Eppstein D., Löffler M., and Strash D. (2010). Listing all maximal cliques in sparse graphs in near-optimal time. In *Algorithms and Computation* (Springer), vol. 6506 of *Lecture Notes in Computer Science*, pp. 403–414.
- Freeman J., Vladimirov N., Kawashima T., Mu Y., Sofroniew N.J., Bennett D.V., Rosen J., Yang C.T., Looger L.L., and Ahrens M.B. (2014). Mapping brain activity at scale with cluster computing. *Nat Methods* *11*, 941–950.
- Friedrich J., Soudry D., Mu Y., Freeman J., Ahres M., and Paninski L. (2015). Fast constrained non-negative matrix factorization for whole-brain calcium imaging data. In *NIPS workshop on statistical methods for understanding neural systems*.
- Gelman A., Carlin J., Stern H., and Rubin D. (2003). *Bayesian Data Analysis* (CRC Press).
- GENIE project, Janelia Farm Campus, HHMI; Karel Svoboda (contact) (2015). Genie project, janelia farm campus, hhmi. <http://dx.doi.org/10.6080/K02R3PMN>.
- Grant M., Boyd S., and Ye Y. (2008). CVX: Matlab software for disciplined convex programming.
- Kaifosh P., Zaremba J.D., Danielson N.B., and Losonczy A. (2014). SIMA: Python software for analysis of dynamic fluorescence imaging data. *Front Neuroinform* *8*.
- Machado T., Pnevmatikakis E., Paninski L., Jessell T., and Miri A. (2015). Primacy of flexor locomotor pattern revealed by ancestral reversion of motor neuron identity. *Cell* *162*, 338–350.
- Maruyama R., Maeda K., Moroda H., Kato I., Inoue M., Miyakawa H., and Aonishi T. (2014). Detecting cells using non-negative matrix factorization on calcium imaging data. *Neural Netw* *55*, 11–19.
- Mukamel E.A., Nimmerjahn A., and Schnitzer M.J. (2009). Automated analysis of cellular signals from large-scale calcium imaging data. *Neuron* *63*, 747–760.
- Oppenheim A.V., and Willsky A.S. (1997). *Signals and systems* (Prentice-Hall).
- Osakada F., Mori T., Cetin A.H., Marshel J.H., Virgen B., and Callaway E.M. (2011). New rabies virus variants for monitoring and manipulating activity and gene expression in defined neural circuits. *Neuron* *71*, 617–631.
- Pnevmatikakis E., Merel J., Pakman A., and Paninski L. (2013). Bayesian spike inference from calcium imaging data. In *Asilomar Conference on Signals, Systems & Computers*. pp. 349–353.
- van den Berg E., and Friedlander M.P. (2008). Probing the pareto frontier for basis pursuit solutions. *SIAM Journal on Scientific Computing* *31*, 890–912.
- Verhaegen M., and Verdult V. (2007). *Filtering and system identification: a least squares approach* (Cambridge Univ Pr).
- Vogelstein J., Packer A., Machado T., Sippy T., Babadi B., Yuste R., and Paninski L. (2010). Fast non-negative deconvolution for spike train inference from population calcium imaging. *J Neurophysiol* *104*, 3691–3704.