Sparse Space-Time Deconvolution for Calcium Image Analysis

Authored by:

Fred A. Hamprecht
Ferran Diego Andilla

Abstract

We describe a unified formulation and algorithm to find an extremely sparse representation for Calcium image sequences in terms of cell locations, cell shapes, spike timings and impulse responses. Solution of a single optimization problem yields cell segmentations and activity estimates that are on par with the state of the art, without the need for heuristic pre- or postprocessing. Experiments on real and synthetic data demonstrate the viability of the proposed method.

1 Paper Body

A detailed understanding of brain function is a still-elusive grand challenge. Experimental evidence is collected mainly by electrophysiology and ?Calcium imaging?. In the former, multi-electrode array recordings allow the detailed study of hundreds neurons, while field potentials reveal the collective action of dozens or hundreds of neurons. The more recent Calcium imaging, on the other hand, is a fluorescent microscopy technique that allows the concurrent monitoring of the individual actions of thousands of neurons at the same time. While its temporal resolution is limited by the chemistry of the employed fluorescent markers, its great information content makes Calcium imaging an experimental technique of first importance in the study of neural processing, both in vitro [16, 6] and in vivo [5, 7]. However, the acquired image sequences are large, and in laboratory practice the analysis remains a semi-manual, tedious and subjective task. Calcium image sequences reveal the activity of neural tissue over time. Whenever a neuron fires, its fluorescence signal first increases and then decays in a characteristic time course. Evolutionary and energetic constraints on the brain guarantee that, in most cases, neural activity is sparse in both space (only a fraction of neurons fire at a given instant) and time (most neurons fire only intermittently). The problem setting can be formalized as follows: given an image sequence as input, the desired output is (i) a set of cells1 and (ii) a set of time points at which these cells were triggered. We here propose an unsuper-

vised learning formulation and algorithm that leverages the known structure of the data to produce the sparsest representations published to date, and allow for meaningful automated analysis. 1.1

Prior Art

Standard laboratory practice is to delineate each cell manually by a polygon, and then integrate their fluorescence response over the polygon, for each point in time. The result is a set of time series, one per cell. 1 Optical sectioning by techniques such as confocal or two-photon microscopy implies that we see only parts of a neuron, such as a slice through its cell body or a dendrite, in an image plane. For brevity, we simply refer to these as ?cells? in the following.

1

a) Matrix factorization [13, 15, 4, 3, 12]

b) Convolutional sparse coding [8, 25, 20, 17, 14]

Figure 1: Sketch of selected previous work. Left: Decomposition of an image sequence into a sum of K components. Each component is given by the Cartesian product of a spatial component or basis image Dk and its temporal evolution uk . In this article, we represent such Cartesian products by the convolution of multidimensional arrays. Right: Description of a single image in terms of a sum of latent feature maps Dk convolved with filters Hk Given that the fluorescence signal impulse response to a stimulus is stereotypic, these time series can then be deconvolved to obtain a sparse temporal representation for each cell using nonnegative sparse deconvolution [24, 5, 10]. The problem of automatically identifying the cells has received much less attention, possibly due to the following difficulties [16, 23]: i) low signal-to-noise ratio (SNR); ii) large variation in luminance and contrast; iii) heterogeneous background; iv) partial occlusion and v) pulsations due to heartbeat or breathing in live animals. Existing work either hard-codes prior knowledge on the appearance of specific cell types [16, 22] or uses supervised learning (pixel and object level classification, [23]) or unsupervised learning (convolutional sparse block coding, [14]). Closest in spirit to our work are attempts to simultaneously segment the cells and estimate their time courses. This is accomplished by matrix factorization techniques such as independent component analysis [13], nonnegative matrix factorization [12] and (hierarchical) dictionary learning [4, 3]. However, none of the above give results that are truly sparse in time; and all of the above have to go to some lengths to obtain reasonable cell segmentations: [13, 4] recur to heuristic post-processing, while [3] invokes structured sparsity inducing norms and [15] use an additional clustering step as initialization. All these extra steps are necessary to assure that each spatial component represents exactly one cell. In terms of mathematical modeling, we build on recent advances and experiments in convolutional sparse coding such as [8, 25, 20, 17, 14]. Ref. [21] already applies convolutional sparse coding to video, but achieves sparsity only in space and not in time (where only pairs of frames are used to learn latent representations). Refs. [19, 18] consider time series which they deconvolve, without however using (or indeed needing, given their data) a sparse spatial representation. 1.2

Contributions

Summarizing prior work, we see three strands: i) Fully automated methods

that require an extrinsic cell segmentation, but can find a truly2 sparse representation of the temporal activity. ii) Fully automated methods that can detect and segment cells, but do not estimate time courses in the same framework. iii) Techniques that both segment cells and estimate their time courses. Unfortunately, existing techniques either produce temporal representations that are not truly sparse [12, 4, 3] or do not offer a unified formulation of segmentation and activity detection that succeeds without extraneous clustering steps [15]. In response, we offer the first unified formulation in terms of a single optimization problem: its solution simultaneously yields all cells along with their actions over time. The representation of activity is truly sparse, ideally in terms of a single nonzero coefficient for each distinct action of a cell. This is accomplished by sparse space-time deconvolution. Given a motion-corrected sequence of Calcium images, it estimates i) locations of cells and ii) their activity along with iii) typical cell shapes and iv) typical impulse responses. Taken together, these ingredients afford the sparsest, and thus hopefully most interpretable, representation of the raw data. In addition, our joint formulation 2 We distinguish a sparse representation, in which the estimated time course of a cell has many zeros; and a ?truly sparse? representation in which a single action of a cell is ideally represented in terms of a single nonzero coefficient.

2

Figure 2: Summary of sparse space-time deconvolution. Top: Unified formulation in terms of a single optimization problem. Bottom: Illustration on tiny subset of data. Left: raw data. The fluorescence level to be estimated is heavily degraded by Poisson shot noise that is unavoidable at the requisite short exposure times. Middle: smoothed raw data. Right: approximation of the data in terms of a Cartesian product of estimated cell shapes and temporal activities. Each temporal activity is further decomposed as a convolution of estimated impulse responses and very few nonzero coefficients. allows to estimate a nonuniform and temporally variable background. Experiments on difficult artificial and real-world data show the viability of the proposed formulation. Notation Boldface symbols describe multidimensional arrays. We define A ? B as a convolution of multidimensional arrays A and mirror(B), with the result trimmed to the dimensions of A. Here, the ?mirror? operation flips a multidimensional array along every dimension. A B is the full convolution result of multidimensional arrays A and mirror(B). These definitions are analogous to the ?convn? command in matlab with shape arguments ?same? and ?full?, respectively. k ? k0 counts the number of non-zero coeficients, and k ? kF is the Frobenius norm.

2 2.1

Sparse space-time deconvolution (SSTD) No background subtraction

An illustration of the proposed formulation is given in Fig. 2 and our notation is summarized in Table. 1. We seek to explain image sequence X in terms of up to K cells and their activity over time. In so doing, all cells are assumed to have exactly one (Eq. 1.1) of J ¡¡ K possible appearances, and to reside at a unique location (Eq. 1.1). These cell locations are encoded in K latent binary feature maps. The activity of each cell is further decomposed in terms of a con-

volution of impulses (giving the precise onset of each burst) with exactly one of L ¡¡ K types of impulse responses. A single cell may ?use? different impulse responses at different times, but just one type at any one time ((Eq. 1.2). All of the above is achieved by solving the following optimization problem:

? ?

K J X X

? min X ? Dk,j ? Hj ? D,H,f ,s j=1 k=1 3

!2

sk,l ? fl

l=1

L X

F

(1)

Constraint P Pj kDk,j k0 ? 1, ?k such that l kst,k,l k0 ? 1, ?k, t kHj k2F ? 1, ?j kfl k22 ? 1, ?l

(1.1) (1.2) (1.3) (1.4)

Interpretation at most one location and appearance per component only one type of activation at each time per cell prevent cell appearance from becoming large prevent impulse filter from becoming large

Here, the optimization is with respect to the cell detection maps D, cell appearances H, activity patterns or impulse responses f as well as ?truly sparse? activity indicator vectors s. This optimization is subject to the two constraints mentioned earlier plus upper bounds on the norm of the learned filters. The user needs to select the following parameters: an upper bound K on the number of cells as well as the size in pixels H of their matched filters / convolution kernels H; upper bounds J on the number of different appearances and L on the number of different activity patterns that cells may have; as well as the number of coefficients F that the learned impulse responses may have. Considering that we propose a method for both cell detection and sparse time course estimation, the number of six user-adjustable parameters compares favourably to previous work. Methods that decouple these steps typically need more parameters altogether, and the heuristics that prior work on joint optimization uses also have a large number of (implicit) parameters. PK PK PJ While many other approximations such as k=1 Dk sk ? fk or k=1 j=1 Dk,j ? Hj sk,j ? fj are conceivable and may make sense in other applications areas, the proposed formulation is the most parsimonious of its kind. Indeed, it uses a small pool of J shapes and L firing patterns, which can be combined freely, to represent all cells and their activities. It is owing to this fact that we dub the method sparse spatio-temporal deconvolution (SSTD). 2.2

SSTD with background subtraction

In actual experiments, the observed fluorescence level is a sum of the signal of interest plus a nuisance background signal. This background is typically nonuniform in the spatial domain and, while it can be modeled as constant over time [15, 24], is often also observed to vary over time, prompting robust local normalization as a preprocessing step [7, 4]. Here, we generalize the formulation from (1) to correct for a background that is assumed to be spatially smooth and

4

time-varying. In more detail, we model the background in terms of the direct ?N ?1 1?1?T product Bs bt of a spatial component Bs ? RM and a time series bt ? R+ . Insights + into the physics and biology of Calcium imaging suggest that (except for saturation regimes characterized by high neuron firing rates), it is reasonable to assume that the normalized quantity (observed fluorescence minus background) divided by background, typically dubbed ?F/F0 , is linearly related to the intracellular Calcium concentration [24, 10]. In keeping with this notion, we now propose our final model, viz.

?
2 ? ? ? !
K J L X X X
s t? s t
? ? ? min X? Dk,j ? Hj sk,l ? fl ? B b B b D,H,f ,s,Bs ,bt
j=1 k=1 l=1 F
+ ?kBs kT V such that (1.1) ? (1.4), Bs ¿ 0, bt ¿ 0
(2)

with ?? denoting an elementwise division. Note that the optimization now also runs over the spatial and temporal components of the background, with the total variation (TV) regularization term3 enforcing spatial smoothness of the spatial background component [2]. In addition to the previously defined parameters, the user also needs to select parameter ? which determines the smoothness of the background estimate. 2.3

Optimization

The optimization problem in (2) is convex in either the spatial or the temporal filters H, f alone when keeping all other unknowns fixed; but it is nonconvex in general. In our experiments, we use a block coordinate descent strategy [1, Section 2.7] that iteratively optimizes one group of variables while 3

TV measures the sum of the absolute values of the spatial gradient.

4

Symbol ?N ?T X ? RM + K ? N+ J ? N+ Hj ? RH?H?1 + Dk,j ? {0, 1}M ?N ?1 L ? N+ fl ? R1?1?F + sk,l ? R1?1?T +

Definition image sequence of length T , each image is M ? N number of cells number of distinct cell appearances j th cell appearance / spatial filter / matched filter of size H ? H indicator matrix of the k th cell for the j th cell appearance number of distinct impulse responses / activity patterns lth impulse response of length F indicator vector of the k th spike train for the lth impulse response Table 1: Notation

fixing all others (see supplementary material for details). The nonconvex l0 -norm constraints require that cell centroids D and spike trains s are estimated by techniques such as convolutional matching pursuit [20]; while the spatio-temporal filters can be learned using simpler gradient descent [25], K-SVD [20] or simple algebraic expressions. All unknowns are initialized with standard Gaussian noise truncated to nonnegative values. The limiting number of cells K can be set to a generous upper bound of the expected true number because spatial components without activity are automatically set to zero during optimization.

3

Experimental Setup

This section describes the data and algorithms used for experiments and benchmarks. 3.1

Inferring Spike Trains

The following methods assume that cell segmentation has already been performed by some means, and that the fluorescence signal of individual pixels has been summed up for each cell and every time step. They can hence concentrate exclusively on the estimation of a ?truly sparse? representation of the respective activities in terms of a ?spike train?. Data We follow [24, 5] in generating 1100 sequences consisting of one-sided exponential decays with a constant amplitude of 1 and decay rate ? = 1/2s, sampled at 30f ps with firing rates ranging uniformly from 1 to 10Hz and different Gaussian noise levels ? ? [0.1, 0.6]. Fast non-negative deconvolution (FAST) [24] uses a one-sided exponential decay as parametric model for the impulse response by invoking a first-order autoregressive process. Given that our artificial data is free of a nuisance background signal, we disregard its ability to also model such background. The sole remaining parameter, the rate of the exponential decay, can be fit using maximum likelihood estimation or a method-of-moments approach [15]. Peeling [5] finds spikes by means of a greedy approach that iteratively removes one impulse response at a time from the residual fluorescence signal. Importantly, this stereotypical transient must be manually defined a priori. Sparse temporal deconvolution (STD) with a single impulse response is a special case of this work for given nonoverlapping cell segmentations and L = 1; and it is also a special case of [14]. The impulse response can be specified beforehand (amounting to sparse coding), or learned from the data (that is, performing dictionary learning on time-series data). 3.2

Segmenting Cells and Estimating Activities

Data Following the procedure described in [4, 12, 13], we have created 80 synthetic sequences with a duration of 15s each at a frame rate of 30f ps with image sizes M = N = 512 pixels. The cells are randomly selected from 36 cell shapes extracted from real data, and are randomly located in different locations with a maximum spatial overlap of 30%. Each cell fires according to a dependent Poisson process, and its activation pattern follows a one-sided exponential decay with 5

a scale selected uniform randomly between 500 and 800ms. The average number of active cells per frame varies from 1 to 10. Finally, the data has been distorted by additive white Gaussian noise with a relative amplitude (max. intensity ? mean intensity)/?noise ? {3, 5, 7, 10, 12, 15, 17, 20}. By construction, the identity, location and activity patterns of all cells are known. The supplemental material shows an example with its corresponding inferred neural activity. Real-world data comes from two-photon microscopy of mouse motor cortex recorded in vivo [7] which has been motion-corrected. These sequences allow us to conduct qualitative experiments. ADINA [4] relies on dictionary learning [11] to find both spatial components and their time courses. Both have many zero coefficients, but are not ?truly sparse? in the sense of this paper. The

method comes with a heuristic post-processing to separate coactivated cells into distinct spatial components. NMF+ADINA uses non-negative matrix factorization to infer both the spatial and temporal primitives of an image sequence as in [12, 15]. In contrast to [15] which uses a k-means clustering of highly confident spike vectors to provide a good initialization in the search for spatial components, we couple NMF with the postprocessing of ADINA. CSBC+SC combines convolutional sparse block coding [14] based on a single still image (obtained from the temporal mean or median image, or a maximum intensity projection across time) with temporal sparse coding. CSBC+STD combines convolutional sparse block coding [14] based on a single still image (obtained from the temporal mean or median image, or a maximum intensity projection across time) with the proposed sparse temporal deconvolution in Sect. 3.1. SSTD is the method described here. We used $J = L = 2$, $K = 200$, $F = 200$ and $H = 31, 15$ for the artificial and real data, respectively.

4 4.1

Results Inferring spike trains

To quantify the accuracy of activity detection, we first threshold the estimated activities and then compute, by summing over each step in every time series, the number of true and false negatives and positives. For a fair comparison, the thresholds were adjusted separately for each method to give optimal accuracy. Sensitivity, precision and accuracy computed from the above implicitly measure both the quality of the segmentation and the quality of the activity estimation. An additional measure, SPIKE distance [9], emphasizes any temporal deviations between the true and estimated spike location in a truly sparse representation. Fig. 3 shows that, unsurprisingly, best results are obtained when methods use the true impulse response rather than learning it from the data. This finding does not carry over to real data, where a ?true? impulse response is typically not known. Given the true impulse response, both FAST and STD fare better than Peeling, showing that a greedy algorithm is faster but gives somewhat worse results. Even when learning the impulse response, FAST and STD are no worse than Peeling. When learning the parameters, FAST has an advantage over STD on this artificial data because FAST already uses the correct parametric form of the impulse response that was used to generate the data and only needs to learn a single parameter; while STD learns a more general but nonparametric activity model with many degrees of freedom. The great spread of all quality measures results from the wide range of noise levels used, and the overall deficiencies in accuracy attest to the difficulty of these simulated data sets. 4.2

Segmenting Cells and Inferring spike trains

Fig. 4 shows that all the methods from Sect. 3.2 reach respectable and comparable performance in the task of identifying neural activity from nontrivial synthetic image sequences. CSBC+SC reaches the highest sensitivity while SSTD has the greatest precision. SSTD apparently achieves comparable performance to the other methods without the need for a heuristic pre- or postprocessing. Multiple random initializations lead to similar learned filters (results not shown), 6

7

FAST (fixed param.) FAST (learned param.) Peeling (fixed param.) STD (fixed param.) STD (learned param.) 00

20 20 40

60 40

8060

Sensitivity (%) Sensitivity (%)

100

80

1000

20

40

60

Precision (%)

100 0

80

20

40

60

Accuracy (%)

80

100 0

0.1

0.2

SPIKE distance

0.3

0.4

Figure 3: Sensitivity, precision, accuracy (higher is better) and SPIKE distance (lower is better) of different methods for spike train estimation. Methods that need to learn the activation pattern perform worse than those using the true (but generally unknown) activation pattern and its parameters. FAST is at an advantage here because it happens to use the very impulse response that was used in generating the data. so the optimization problem seems to be well-posed. The price to pay for the elegance of a unified formulation is a much higher computational cost of this more involved optimization. Again, the spread of sensitivities, precisions and accuracies results from the range of noise levels used in the simulations. The plots suggest that SSTD may have fewer ?catastrophic failure? cases, but an even larger set of sequences will be required to verify this tendency.

ADINA
NNMF+ADINA
CSBC+SC
CSBC+STD
SSTD
50 50
60 60
70

70 80
8090
Sensitivity (%) (%) Sensitivity
90 100
100 50
60
70
80
Precision (%)
90
100 50
60
70
80
Accuracy (%)
90
100

Figure 4: Quality of cell detection and and the estimation of their activities. SSTD does as well as the competing methods that rely on heuristic pre- or post-processing. Real Sequences: Qualitative results are shown in Fig. 5. SSTD is able to distinguish both cells with spatial overlap and with high temporal correlation. It compensates large variations in luminance and contrast, and can discriminate between different types of cells. Exploiting truly sparse but independent representations in both the spatial and the temporal domain allows to infer plausible neural activity and, at the same time, reduce the noise in the underlying Calcium image sequence.

5 Discussion

The proposed SSTD combines the decomposition of the data into low-rank components with the finding of a convolutional sparse representation for each of those components. The formalism allows exploiting sparseness and the repetitive motifs that are so characteristic of biological data. Users need to choose the number and size of filters that indirectly determine the number of cell types found and their activation patterns. As shown in Fig. 5, the approach gives credible interpretations of raw data in terms of an extremely sparse and hence parsimonious representation. The decomposition of a spacetime volume into a Cartesian product of spatial shapes and their time courses is only possible when cells do not move over time. This assumption holds for in vitro experiments, and can often be satisfied by good fixation in in vivo experiments, but is not universally valid. Correcting for motions in a generalized unified framework is an interesting direction for future work. The experiments in section 4.1 suggest that it may also be worthwhile to investigate the use of more parametric forms for the impulse response instead of the completely unbiased variant used here.

7

70 48
64 56

62
2
65
29
135
142
150 88
54
66 69
41
137
18
15
37 144
103 76
75
59 20 38
131 33 3
141 30
63
121
36
96 71
125 61 44 43 106
72
107
74 136 1
58
17
60
42
62 66
68
87
6 55
43
53
16
13
91
3
99
51
0.05
17 39
57

90
0 0
20
40
Frames
60
80
25 36 2
100
0.1
18 64
61
69
32
0.2 0.15
3510 89
27 95
148
94
26
32
79
47
12
100 65
5
81
63 47
0.05
16
31 98
23
76
96
93
0.25
75 38
28 1567 54
21
14 44
55
34
9
134
56
11

4 33
84 79
37
20
34 78 77
92
86
49 88
30
0.1
52
59
5
127
149
46 8
9 91
85 7
11683 14 67 132 23 87 78 6
7
28
40
1
109
112
77
92
84
0.15
12
147
73
80 45
24
0.3
50
71
83 48
39
13
51
81
8
24
146
73 42 152

97 58 29 72
22
filter 1 filter 2 filter 3
19
70
41
90
97
27 45
85
117
49 110
115 111
22
94
11
138
31
119
126
108113 102
40
99
35 100 105
133
50
52 80
60 128
74
104
26
82 46 10 129
4
filter 1 filter 2
145 143
53
151 114
124
101
93
122 140
120
0.2
118 130 86
19
68 25

89

57

123

95 21

139 98

0

82

0

20

40

Frames

60

80

100

1 1

11 21

11

31 21

41 31

61

Cell number

Cell number

51

71 81 91

41

51

61

101 71

111 121

81

131 91

141

100

151 0

50

100

150

Time (s)

200

250

1

0

50

100

0

50

100

150

200

250

150

200

250

Time (s)

1

11 11

21 31

21

41 31

61

Cell number

Cell number

51

71 81 91

41

51

61

101 71

111 121

81

131 91

141 151 152

100

0

50

100

150

Time (s)

200

250

Time (s)

Figure 5: Qualitative results on two real data sets. The data on the left column shows mostly cell bodies, while the data on the right shows both cell bodies (large) and dendrites (small). For each data set, the top left shows an average projection of the relative fluorescence change across time with cell centroids D (black dots) and contours of segmented cells, and the top right shows the learned impulse responses. In the middle, the fluorescence levels integrated over the segmented cells are shown in random colors. The bottom shows by means of small disks the location, type and strength of the impulses that summarize all the data shown in the middle. Together with the cell shapes, the impulses from part of the ?truly sparse? representation that we propose. When convolving these spikes with the impulse responses from the top right

15

insets, we obtain the time courses shown in random colors. Such advances will further help making Calcium imaging an enabling tool for the neurosciences.

8

2 References

[1] [2] [3] [4] [5] [6] [7]
[8] [9] [10] [11] [12] [13] [14] [15] [16]
[17] [18] [19] [20] [21] [22]
[23]
[24]
[25]

D. P. Bertsekas. Nonlinear Programming. Athena Scientific, 1999. A. Chambolle. An algorithm for total variation minimization and applications, 2004. F. Diego and F. A. Hamprecht. Learning multi-level sparse representations. In NIPS. 2013. F. Diego, S. Reichinnek, M. Both, and F. A. Hamprecht. Automated identification of neuronal activity from calcium imaging by sparse dictionary learning. ISBI 2013. Proceedings, pages 1058?1061, 2013. B. F. Grewe, D. Langer, H. Kasper, B. M. Kampa, and F. Helmchen. High-speed in vivo calcium imaging reveals neuronal network activity with near-millisecond precision. Nat Meth, 7(5):399?405, May 2010. C. Grienberger and A. Konnerth. Neuron, volume 73, chapter Imaging Calcium in Neurons, pages 862? 885. Cell Press,, Mar 2012. D. Huber, D. A. Gutnisky, S. Peron, D. H. O/?Connor, J. S. Wiegert, L. Tian, T. G. Oertner, L. L. Looger, and K. Svoboda. Multiple dynamic representations in the motor cortex during sensorimotor learning. Nature, 484(7395):473?478, Apr 2012. K. Kavukcuoglu, P. Sermanet, Y. Boureau, K. Gregor, M. Mathieu, and Y. LeCun. Learning convolutional feature hierachies for visual recognition. In NIPS, 2010. T. Kreuz, D. Chicharro, C. Houghton, R. G. Andrzejak, and F. Mormann. Monitoring spike train synchrony. Journal of Neurophysiology, 2012. H. Luetcke, F. Gerhard, F. Zenke, W. Gerstner, and F. Helmchen. Inference of neuronal network spike dynamics and topology from calcium imaging data. Frontiers in Neural Circuits, 7(201), 2013. J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online Learning for Matrix Factorization and Sparse Coding. Journal of Machine Learning Research, 2010. R. Maruyama, K. Maeda, H. Moroda, I. Kato, M. Inoue, H. Miyakawa, and T. Aonishi. Detecting cells using non-negative matrix factorization on calcium imaging data. Neural Networks, 55(0):11 ? 19, 2014. E. A. Mukamel, A. Nimmerjahn, and M. J. Schnitzer. Automated analysis of cellular signals from largescale calcium imaging data. Neuron, 2009. M. Pachitariu, A. M. Packer, N. Pettit, H. Dalgleish, M. Hausser, and M. Sahani. Extracting regions of interest from biological images with convolutional sparse block coding. In NIPS. 2013. E. A. Pnevmatikakis and L. Paninski. Sparse nonnegative deconvolution for compressive calcium imaging: algorithms and phase transitions. In NIPS. 2013. S. Reichinnek, A. von Kameke, A. M. Hagenston, E. Freitag, F. C. Roth, H. Bading, M. T. Hasan, A. Draguhn, and M. Both. Reliable optical detection

16

of coherent neuronal activity in fast oscillating networks in vitro. NeuroImage, 60(1), 2012. R. Rigamonti, A. Sironi, V. Lepetit, and P. Fua. Learning separable filters. In Conference on Computer Vision and Pattern Recognition, 2013. M. N. Schmidt and M. M?rup. Nonnegative matrix factor 2-D deconvolution for blind single channel source separation. In ICA, 2006. P. Smaragdis. Nonnegative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. In ICA, pages 494?499, 2004. A. Szlam, K. Kavukcuoglu, and Y. LeCun. Convolutional matching pursuit and dictionary training. Computer Research Repository (arXiv), 2010. G. W. Taylor, R. Fergus, Y. Lecun, and C. Bregler. Convolutional learning of spatio-temporal features, 2010. J. Tomek, O. Novak, and J. Syka. Two-photon processor and seneca: a freely available software package to process data from two-photon calcium imaging at speeds down to several milliseconds per frame. J Neurophysiol, 110, 2013. I. Valmianski, A. Y. Shih, J. D. Driscoll, D. W. Matthews, Y. Freund, and D. Kleinfeld. Automatic identification of fluorescently labeled brain cells for rapid functional imaging. Journal of Neurophysiology, 2010. J. T. Vogelstein, A. M. Packer, T. A. Machado, T. Sippy, B. Babadi, R. Yuste, and L. Paninski. Fast non-negative deconvolution for spike train inference from population calcium imaging. Journal of Neurophysiology, 2010. M. Zeiler, D. Krishnan, G. Taylor, and R. Fergus. Deconvolutional networks. In CVPR, 2010.

9