Hierarchical Coupled-Geometry Analysis for Neuronal Structure and Activity Pattern Discovery

Gal Mishne, Ronen Talmon, Ron Meir, Jackie Schiller, Maria Lavzin, Uri Dubin, and Ronald R. Coifman

***Abstract*—In the wake of recent advances in experimental methods in neuroscience, the ability to record in-vivo neuronal activity from awake animals has become feasible. The availability of such rich and detailed physiological measurements calls for the development of advanced data analysis tools, as commonly used techniques do not suffice to capture the spatio-temporal network complexity. In this paper, we propose a new hierarchical coupled-geometry analysis that implicitly takes into account the connectivity structures between neurons and the dynamic patterns at multiple time scales. Our approach gives rise to the joint organization of neurons and dynamic patterns in data-driven hierarchical data structures. These structures provide local to global data representations, from local partitioning of the data in flexible trees through a new multiscale metric to a global manifold embedding. The application of our techniques to in-vivo neuronal recordings demonstrate the capability of extracting neuronal activity patterns and identifying temporal trends, associated with particular behavioral events and manipulations introduced in the experiments.**

***Index Terms*—Dimensionality reduction, diffusion maps, geometric analysis, manifold learning, neuronal data analysis.**

I. Introduction

A FUNDAMENTAL goal in neuroscience is to understand how information is represented, stored and modified in cortical networks. New experimental methods in neuroscience not only enable chronic, minimally invasive, recordings of large populations of neurons with cellular level resolution, but also allow recordings from identified neuronal subtypes [1]. The ability to acquire complex large-scale detailed behavioral and neuronal datasets calls for the development of advanced data analysis tools, as commonly used techniques do not suffice to capture the spatio-temporal network complexity. Such a framework should deal effectively with the challenging characteristics of neuronal and behavioral data, namely connectivity structures between neurons and dynamic patterns at multiple time-scales.

Manuscript received November 1, 2015; revised April 13, 2016; accepted August 12, 2016. Date of publication August 24, 2016; date of current version September 23, 2016. The work of R. Talmon was supported in part by the European Union's Seventh Framework Programme under Marie Curie Grant 630657 and in part by the Horev Fellowship. The guest editor coordinating the review of this paper and approving it for publication was Dr. Viktor Jirsa.

G. Mishne, R. Talmon, and R. Meir are with the Viterbi Faculty of Electrical Engineering, Technion—Israel Institute of Technology, Haifa 32000, Israel (e-mail: galga@tx.technion.ac.il; ronen@ee.technion.ac.il; meir@ee.technion.ac.il).

J. Schiller, M. Lavzin, and U. Dubin are with the Department of Physiology, Technion Medical School, Haifa 31096, Israel (e-mail: jackie@tx.technion.ac.il; maria.lavzin@gmail.com; uri.dubin@gmail.com).

R. R. Coifman is with the Department of Mathematics, Yale University, New Haven, CT 06520 USA (e-mail: ronald.coifman@math.yale.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTSP.2016.2602061

Due to natural and physical constraints, the accessible high-dimensional data often exhibit geometric structures and lie on a low-dimensional manifold. Manifold learning is a class of data driven methods; these methods aim to find meaningful geometry-based non-linear representations that parametrize the manifold underlying the data [2]–[6]. Only very recently have we begun to witness seeds of its applicability to real biological data, and, in particular, to neuroscience (e.g., [7], [8]). Yet, most existing manifold learning methods are unable to deal with the complex datasets arising in neuroscience, since they do not account for several fundamental characteristics of the structures and patterns underlying such data. First, current methods are sensitive to noise and interferences. Second, to a large extent, they do not accommodate the dynamical patterns underlying the neuronal activity. Third, manifold learning does not take into account co-dependencies between neuronal connectivity structures and dynamical patterns.

Previous work has addressed analysis of data exhibiting such co-dependencies. To exemplify the generality of such a problem, consider the Netflix prize [9], where it is desired to provide systematic suggestions and recommendations to viewers. A co-organization enables to both group together viewers based on their similar tastes and, at the same time, group together movies based on their similar ratings across viewers. This clustering of viewers or of movies can be highly dependent on the particular viewer, and on the particular movie; two viewers may be similar under one metric, since they both like similar adventure movies, but at the same time, quite different since they do not like the same comedies. Thus, the suggestion system needs different metrics for recommending different types of movies to different viewers.

Data arising in such settings can be viewed as a 2D matrix, where in the Netflix Prize the first dimension is the viewers (observations) and the second is the movies (variables). The need for matrix co-organization arises when observations are not independent and identically distributed, i.e., correlations exist among both observations and variables of the data matrix. Similar settings also arise in analysis of documents, psychological questionnaires, gene expression data, etc., where there is no particular reason to prefer treating one dimension as independent, while the other is treated as dependent [10]–[13]. To address problems of this sort, Gavish and Coifman [14], [15] proposed a methodology for matrix organization relying on the construction of a tree-based bi-organization of the data.

The analysis of natural data poses an even greater challenge, since such data may also depend on a massive number of marginally relevant variables, including distortions and unrelated measurements, requiring metrics that are not sensitive

to such variability, and that are capable of suppressing noise or irrelevant factors. In particular, it is insufficient to represent neuronal activity recordings, which were acquired in repetitive trials, as a 2D matrix by concatenating the trials. The trial-based data is inherently three dimensional, measured from multiple neurons for a fixed number of time frames and acquired over the course of many trials. Thus, we propose to analyze the data as a 3D database whose dimensions are the neurons, the time frames and the trial indices.

In this paper, to accommodate the three-dimensional nature of this data, we extend [14], [15] to a triple-geometry analysis obtaining a nonparametric model for data tensors. We propose a completely data-driven analysis of a given rank-3 tensor that provides a co-organization of the data, i.e., we obtain a reordering (permutation) of each of the dimensions so that the data in each dimension vary smoothly along the other two dimensions. Specifically, we focus on trial-based neuronal data, however, our approach is general and can be used to analyze other types of n-dimensional data-sets.

In addition to the challenge of organizing the data, applying manifold learning methods requires a "good" metric between samples, which conveys local similarity, as in the Netflix example. Regular metrics do not perform well due to the high dimensionality and hierarchical structure of the trial-based data, as well as their inability to encompass the multi-dimensional nature of the data. For example, using the Euclidean distance or cosine similarity between two neurons, treats the neuronal recordings as a 1D vector, and does not take into account the separation of the data into trials. We therefore want a metric that takes into account the multi-dimensional nature of a sample as a 2D matrix, and does not perform just naïve element-wise differences. The metric should respect the coupling between the dimensions, as in the Netflix example, and should take advantage of clustering in the data that occur in two dimensions. This coupling is exhibited, for example, in groups of neurons that are active together during the same time frames in the experiments, due to an external trigger.

Using more sophisticated metrics for manifold learning, such as the Mahalanobis or PCA-based distances proposed in [16]–[21], requires a notion of locality, which is non-trivial in the given application, as the data do not necessarily follow a regular Euclidean 3D grid. For example, the physical location of the neurons in the brain does not indicate their similarity in response or that they are connected. Thus, spatial locality should not be used to define local neighborhoods among the neurons. Instead, we want a notion of locality that relies on similarity in the response space. Therefore, we also address the problem of defining a new multiscale metric, that incorporates both the coupling between the dimensions and the hierarchical structure of the data in each dimension.

Broadly, our focus is on finding a good description of the data; our analysis enables us to build intrinsic data-driven multiscale hierarchical structures. In particular, our analysis builds three types of data structures, conveying a local to global representation, from hierarchical clustering of the data to a multiscale metric to a global embedding. These three structures are constructed in an iterative refinement procedure for each dimension, based on the other two dimensions. Thus, we exploit the coupling between the dimensions.

At the micro-scale, we learn a multiscale organization of the data, so that each sample is organized in a bottom-up hierarchical structure using a partition tree. Thus, each sample belongs to a set of nested folders, where each folder defines a coarser and coarser notion of locality/neighborhood.

At the intermediate scale, the hierarchical organization of the data is then used to define a novel 2D multiscale metric for the comparison of samples (2D matrices) in each dimension. This metric enables to organize each dimension based on a coarse-to-fine decomposition of the other two dimensions. Thus, the metric respects the hierarchy and compares samples not only based on the raw measurements, but also based on their values across scales. It is based on a mathematical foundation, stemming from the approximation of the earth-mover's distance (EMD) proposed by Leeb [22]. We show that this metric is equivalent to the l_1 distance between samples after applying a data-adaptive filter-bank. We extend the tree-based metric to a bi-tree multiscale metric and corresponding 2D filter bank.

At the macro scale, the local organization of the data and the multiscale metric enable the calculation of a global manifold representation of the data, via the construction of an affinity kernel and its eigendecomposition [6]. In each dimension, the data are embedded in a low-dimensional Euclidean space that preserves local structures in the data. Thus, the samples are now represented by new coordinates that can be used to provide a single smooth organization of each dimension. The data can also be clustered based on this representation into meaningful groups.

Our tri-geometry approach is applied to neuronal recordings and is used for exploratory analysis, interpretability and visualization of the data. This organization is needed to identify latent variables that control the activity in the brain and to develop the automated infrastructure necessary to recover complex structures, with less external information and without expert guidance. Our experimental results on neuronal recordings of head-fixed mice demonstrate the capability of isolating and filtering regional activities and relating them to specific stimuli and physical actions, and of automatically extracting pathological dysfunction. Specifically neuronal groupings, temporal activity groupings and experimental condition scenarios are simultaneously extracted from the database, in a data-driven, model-free and network-oriented manner.

The remainder of the paper is organized as follows. Section II briefly reviews related work regarding state-of-the-art methods in neuroscience data analysis. In Section III, we formulate the problem. In Section IV, the proposed methodology for tri-organization of trial-based data is presented, detailing the three components of our approach. Section V presents analysis of experimental neuronal data, in a motor forepaw reach task in mice.

II. Related Work

Current network analysis approaches in neuroscience can be divided into two main classes [23], [24]. The first class comprises methods, which aim to determine functional connectivity, defined in terms of statistical dependencies between measured

elements (e.g., neurons or networks), by constructing direct statistical models from the data (e.g., Granger causality, transfer entropy, point process modeling and graph based methods [24]–[27]. The second class of methods is often based on Latent Dynamical Systems (LDS), which accommodates effective connectivity characterizing the causal relations between elements through an underlying hidden dynamical system [23], [28], [29]. Non-linear and non-Gaussian extensions of the Kalman filter, contemporary sequential Monte Carlo methods and particle filters, have also been introduced in neuroscience [30], [31].

These methods share significant drawbacks, as they are mostly heuristic, providing only an approximation of a largely unknown system, and their quality is often hard to assess [8]. More importantly, they are all prone to the "curse of dimensionality". On the one hand, designing a parametric generative model for truly complex high-dimensional data, such as neuronal/behavioral recordings, requires considerable flexibility, resulting in a model with a large number of tunable parameters. On the other hand, estimating a large number of parameters, and fitting a predefined class of dynamical models to high-dimensional data, is practically infeasible, thereby leading to poor data representations. Our approach is better designed to deal with dynamical systems and aims to alleviate the shortcomings present in current analysis methods. The proposed framework deviates from common methods recently used in neuroscience as it makes only very general smoothness assumptions, rather than postulating a-priori specific structural models. In addition, we show that it takes into consideration the high dimensional spatio-temporal neuronal network structure.

This work is related to the analysis and decomposition of higher-order tensors [32]. For example, the Tucker decomposition [33] and PARAFAC decomposition [34] propose a generalization of matrix singular value decomposition (SVD) to tensors, while multilinear principal component analysis (MPCA) [35] is a multilinear extension of principal component analysis (PCA). Our work differs in two respects. First, we do not propose a tensor decomposition into a set of lower-rank tensors; instead, we aim at organizing the tensor into smooth multiscale local neighborhoods. Decomposition of the tensor via a data-adaptive wavelet basis will be explored in future work. Second, we obtain a lower-dimensional representation of the data via a *non-linear embedding*, instead of the linear projections proposed by MPCA.

III. Problem Formulation

In the sequel we denote the three axes of the 3D data with a trial-based experiment in mind. However, our methodology can be applied to general 3D coordinates. Consider data acquired in a set of fixed-length trials, composed of measurements from a large number of sensors (specifically neurons). Mathematically, we have a rank-3 tensor of neuronal measurements denoted by $\mathbf{X} = f(r, t, T)$ that is a function of three variables: neurons r, time frames t and trials T. This tensor is a discretization of the continuous neuronal activity $f(r, t, T)$ of the subject, such that a tensor element $\mathbf{X}[i, j, k] = f(r_i, t_j, T_k)$ is the neuronal measurement of neuron r_i, at time frame t_j in trial T_k. We collect at each neuron or identified region of interest (ROI) r_i a time series of the neuronal activity (e.g., fluorescence intensity levels in identified ROIs along time). In general trial-based data, this dimension corresponds to the multiple sensors that acquire the data, such as in EEG [36]. The time frames t_j within a given trial can be viewed as a dynamic window profiling the neuron. The time series are acquired over the course of many repetitive trials T_k, which should be organized according to global trends, such as learning a skill, or long lasting external stimuli introduced in the experiments. This tensor can be separately organized into a triple of geometries involving each variable, r, t, and T. However, the *joint* organization of all three variables leads to an organization of dynamic neuronal activity regimes, using a global representation via the diffusion maps embedding [6]. One result of our analysis is a permutation of the indices of each of the three variables, such that applying the permutations to the indices in all three dimensions results in a smooth tensor.

Let $\mathbf{X} \in \mathbb{R}^{n_r \times n_t \times n_T}$ be a rank-3 tensor, where n_r is the number of neurons, n_t is the number of time frames in an individual trial and n_T is the number of trials. Let $\mathbf{X}_{r_i \cdot \cdot} = \{\mathbf{X}[i, j, k] \mid 1 \leq j \leq n_t, 1 \leq k \leq n_T\} \in \mathbb{R}^{n_t \times n_T}$ denote the two-dimensional matrix (slice) of all measurements for a fixed neuron r_i throughout all trials and time-frames. In similar fashion, $\mathbf{X}_{\cdot t_j \cdot} \in \mathbb{R}^{n_r \times n_T}$ is the 2D matrix of all measurements of all neurons, for a fixed time t_j in all trials. Finally, $\mathbf{X}_{\cdot \cdot T_k} \in \mathbb{R}^{n_r \times n_t}$ is the 2D matrix of all measurements of all neurons for all time frames in a single fixed trial T_k. A visualization of a 3D dataset and examples of 2D slices in each dimension is presented in Fig. 1.

Considering trial-based data, we assume the ordering of the time frames t_j is smooth, i.e. the order of the indices $\{j\}$ indicates sequential time frames, and all trials are of the same length n_t. It is easy to define neighbors in this dimension, as it is associated with a regular fixed-length grid. We assume the trials follow a repetitive protocol, controlled by the experimenters, yet the trials T_k are not necessarily contiguous, i.e., trials can occur on different dates, with non-uniform intervals between trials. In addition, the measurements of a given trial relate to behavioral events, which can vary greatly even among sequential trials. Thus, the ordering of the trial index k does not imply that two consecutive trials in the data are similar. In the experimental results in Section V we show that trials are grouped logically based on behavioral similarity and not based on consecutive experiments. A global trend in the organization of the trials is evident only when introducing a pathological inhibitor, which has a long term effect on the test subject. Finally, we assume that the indexing i of the neurons r_i are randomly assigned and therefore do not impose any smoothness or structure, and two consecutive indices do not imply any similarity between neurons.

Thus, although the trial-based measurements are organized as a 3D database so they are supposedly associated with a regular Euclidean grid, in practice the data suffers from non-uniform sampling, and consecutive indices do not indicate actual proximity as in time-series data (temporal smoothness) or a 2D image (spatial smoothness). Thus, conventional analysis methods, such as multiscale representations via wavelets, are not

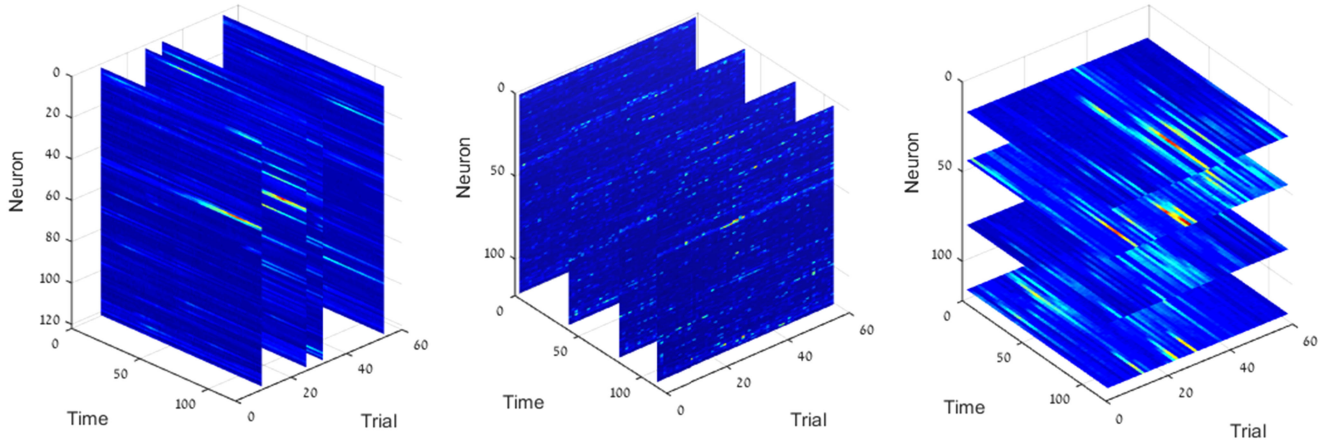

Fig. 1. Visualization of 3D database. The data is visualized here as 2D slices from multiple viewpoints: for several trials $\mathbf{X}_{\cdot\cdot T}$ (left), time frames $\mathbf{X}_{\cdot t\cdot}$ (center), and neurons $\mathbf{X}_{r\cdot\cdot}$(right). The neuronal activity is represented by the intensity level of the image (blue - no activity, red - high activity).

straightforward in the given application. In order to define a multiscale analysis of the data, it is necessary to be able to define neighborhoods and a sense of locality between samples.

The notations in this paper follow these conventions: matrices and tensors are denoted by bold uppercase and sets are denoted by uppercase calligraphic.

IV. Tri-Geometry Analysis

Our analysis is based on the assumption that an underlying true "good" organization of the data exists, such that under a permutation of the indices in each dimension of the data, the resulting tensor is smooth in the three dimensions. This assumption can be formulated as having the data fulfill a tri-Hölder condition, which is a straight-forward extension of the bi-Hölder condition in [22]. Our aim is to recover this organization of the data through a local to global processing of the data. Note that we do not impose smoothness as an explicit constraint; instead it manifests itself implicitly in the data-driven result attained by our approach.

A three-phase organization of each dimension is carried out in an iterative procedure, where each dimension is organized in turn based on the other two. We begin with learning the hierarchal structure of the data in each dimension via partition trees, which convey local clustering of the data. We then construct a new multiscale bi-tree metric for one dimension based on the coupled geometry between the other two dimensions. Finally, the tree-based metric enables us to define an affinity between samples from which we derive a global representation via manifold learning. Thus, our approach does not treat each dimension separately, but introduces a strong coupling between the dimensions. The organization process can be iterated several times.

An advantage of our approach is that it is based on modular components. We describe three methods fulfilling the motivation of each stage, but these methods can be replaced with others. For example, we propose flexible trees for the partition tree construction, but binary trees can be used instead. We expand on the three components of our approach in detail in the following subsections.

A. Partition Trees and Flexible Trees

Following the assumption that under a proper organization the dataset is smooth, we aim to build a fine-to-coarse set of neighborhoods for each element in the tensor, by constructing partition trees in each dimension. In the tri-geometric organization, the neighborhoods are 3D cubes. Permuting the indices in each dimension based on the constructed partition trees will recover the smooth structure respecting the coupling between the neurons, time frames and trials.

Given a set of high-dimensional samples, we construct a hierarchical partitioning of the samples, defined by a tree. In our setting, for each dimension the samples are the 2D slices of the data in that dimension (see Fig. 1). Without loss of generality, we will define the partition trees in this section with respect to partitioning the neurons, but this process is performed in the remaining two dimensions as well. Also note that the algorithm can be initialized by constructing a partition tree for any of the three dimensions, and the choice to start with the neurons is arbitrary.

Let $\mathcal{X}_r = \{\mathbf{X}_{r_i\cdot\cdot}\}_{i=1}^{n_r}$ be the set of all 2D neuron slices. We define a finite partition tree $\mathcal{T}_r$ on $\mathcal{X}_r$ as follows. The partition tree is composed of $L+1$ levels, where a partition of the samples $\mathcal{P}_l$ is defined for each level $0 \le l \le L$. The partition $\mathcal{P}_l$ at level l consists of $n(l)$ mutually disjoint non-empty subsets of indices in $\{1,\ldots,n_r\}$, termed folders and denoted by $\mathcal{I}_{l,i}$, $i \in \{1,\ldots,n(l)\}$:

$$\mathcal{P}_l = \{\mathcal{I}_{l,1}, \mathcal{I}_{l,2}, \ldots, \mathcal{I}_{l,n(l)}\}. \tag{1}$$

Note that we define the folders on the indices of the samples and not on the samples themselves.

The partition tree $\mathcal{T}_r$ has the following properties:

1) The finest partition ($l=0$) is composed of $n(0)=n_r$ singleton folders, termed the "leaves", where $\mathcal{I}_{0,i}=\{i\}$.

2) The coarsest partition ($l = L$) is composed of a single folder, $\mathcal{P}_L = \mathcal{I}_{L,1} = \{1, \ldots, n_r\}$, termed the "root" of the tree.
3) The partitions are nested such that if $\mathcal{I} \in \mathcal{P}_l$, then $\mathcal{I} \subseteq \mathcal{J}$ for some $\mathcal{J} \in \mathcal{P}_{l+1}$, i.e., each folder at level $l-1$ is a subset of a folder from level l.

The partition tree is the set of all folders at all levels $\mathcal{T} = \{\mathcal{I}_{l,i} \mid 0 \leq l \leq L,\ 1 \leq i \leq n(l)\}$, and the number of all folders in the tree is denoted by $|\mathcal{T}|$. The size, or cardinality, of a folder $\mathcal{I}$, i.e. the number of samples in that folder, is denoted by $|\mathcal{I}|$.

Given a dataset, there are many methods to construct a partition tree, including deterministic, random, agglomerative and divisive [13], [37], [38]. Partition trees can be constructed in a top-down or bottom-up approach. In a top-down approach, the data are divided into few folders, then each of these folders is divided into sub-folders, and so on until all folders at the bottom level consist of only one sample. In a bottom-up approach, we begin with the lowest level of the tree, clustering the samples into small folders. Then these folders are merged into larger folders at higher levels of the tree, until all folders are merged at the root of the tree.

A simple approach to bottom-up construction is a k-means based construction. The leaves of the tree are clustered via k-means into k folders. Each folder is then assigned a centroid, and these centroids are then clustered again using k-means, with smaller k. This process is repeated until all samples are merged at the root folder with $k = 1$.

More sophisticated approaches take into account the geometric structure and multiscale nature of the data by incorporating affinity matrices on the data, and manifold embeddings. Gavish *et al.* [37] propose constructing a partition tree via bottom-up hierarchical clustering, given a symmetric affinity matrix $\mathbf{W}$ describing a weighted graph on the dataset. Ankenman [39] proposed "flexible trees", whose construction requires an affinity matrix on the data, and is based on a low-dimensional diffusion embedding of the data, and not on the high-dimensional samples. The advantage of this construction, which uses the embedding rather than the high-dimensional space is that distances between samples in the diffusion space are meaningful and robust to noise, as opposed to high-dimensional Euclidean distances. This tree construction enables us to integrate both the multiscale metric and the resulting global embedding. Since our approach is based on an iterative procedure of all three components, the tree construction is refined from iteration to iteration.

Another important advantage of flexible trees is that there are relatively few levels and the level at which folders are joined is meaningful across the entire dataset. Thus, the tree structure is logically multiscale and follows the structure of the data. This also reduces the computational complexity of the metric calculation. The construction is controlled by a constant ϵ which affects the number of levels in the tree. Higher values of ϵ result in "tall" trees, while small values lead to "flatter" trees.

We briefly describe the flexible trees algorithm, given the set $\mathcal{X}_r$ and an affinity matrix on the neurons denoted $\mathbf{W}_r$. For a detailed description see [39].

1) Input: The set of samples $\mathcal{X}_r$, an affinity matrix $\mathbf{W}_r \in \mathbb{R}^{n_r \times n_r}$, and a constant ϵ.
2) Init: Set partition $\mathcal{I}_{0,i} = \{i\}\ \forall\ 1 \leq i \leq n_r$, set $l = 1$.
3) Given an affinity on the data, we construct a low-dimensional embedding on the data [6].
4) Calculate the level-dependent pairwise distances $d^{(l)}(i,j)\ \forall\ 1 \leq i,j \leq n_r$ in the embedding space.
5) Set a threshold $\frac{p}{\epsilon}$, where $p = \text{median}\left(d^{(l)}(i,j)\right)$.
6) For each index i which has not yet been added to a folder, find its minimal distance $d^{\min}(i) = \min_j\{d^{(l)}(i,j)\}$.
   a) If $d^{\min}(i) < \frac{p}{\epsilon}$, i and j form a new folder if j also does not belong to a folder. If j is already part of a folder $\mathcal{I}$, then i is added to that folder if $d^{\min}(i) < \frac{p}{\epsilon}2^{-|I|+1}$. Thus, the threshold on the distance for adding an element to an existing folder is divided by 2 for each added element.
   b) If $d^{\min}(i) > \frac{p}{\epsilon}$, i remains as a singleton folder.
7) The partition $\mathcal{P}_l$ is set to be all the formed folders.
8) For $l > 1$ and while not all samples have been merged together in a single folder, steps 4)-7) are repeated. Instead of iterating over samples, we iterate over all the folders $\mathcal{I}_{l-1,i} \in \mathcal{P}_{l-1}$. The distances between folders depend on the level l, and on the samples in each of the folders.

In the proposed hierarchical representation of the data via partition trees, the nodes are grouped into disjoint sets. Thus, a limitation of using partition trees is that a node can only be connected to a single "parent", i.e. grouped in a single folder in the level above. However, it can be beneficial to enable a node to participate in several folders, such that there is an overlap between folders, as in [40]. Since our approach is modular, each component can be replaced by a different algorithm. Specifically, to enable nodes participating in several folders, the partition tree construction can be replaced by a directional bi-partite graph such that each node can participate in more than one folder in the above level. This will enable identifying overlapping clusters of nodes.

The trees yield a hierarchical multiscale organization of the data, which then enables us to apply signal processing methods. For example, we can apply non-local means to each sample based on its neighborhood, to denoise the data, or multiscale analysis via tree based wavelets [37], [41], [42]. However, we aim at a global analysis of the data. To this end, we define a bi-tree multiscale metric, which compares two samples, based on their decomposition via the trees.

B. Data-Adaptive Bi-Tree Multiscale Metric

Applying manifold learning requires an appropriate metric between samples. As we cannot associate a sense of locality based on the indexing of the dimensions, we treat the data as vertices in a graph and develop a metric that is based on the multiscale neighborhoods constructed in the partition tree. Given the partition trees in two of the dimensions, our aim is to define a distance d between two 2D slices in the remaining dimension. This distance should incorporate the multiscale nature of the data.

For a two-dimensional matrix, Leeb [22] defines a tree-based metric between two samples in one dimension based on a partition tree in the other dimension. We will present this metric in

our context: Consider a single 2D slice of the trial data $\mathbf{X}_{\cdot\cdot T_k}$ for fixed T_k, and the partition tree on the neurons $\mathcal{T}_r$. Considering a single time frame t_j, $\mathbf{X}_{\cdot t_j T_k}$ is a vector of length n_r, consisting of all the neuronal measurements for the time frame t_j during the given trial T. The tree metric $d_{\mathcal{T}_r}(\mathbf{X}_{\cdot t_i T_k}, \mathbf{X}_{\cdot t_j T_k})$ between two time frames t_i and t_j within this trial, given the tree $\mathcal{T}_r$ is defined as

$$d_{\mathcal{T}_r}(\mathbf{X}_{\cdot t_i T_k}, \mathbf{X}_{\cdot t_j T_k}) = \sum_{\mathcal{I} \in \mathcal{T}_r} |m(\mathbf{X}_{\cdot t_i T_k} - \mathbf{X}_{\cdot t_j T_k}, \mathcal{I})| \omega(\mathcal{I}), \tag{2}$$

where $\omega(I) > 0$ is a weight function, depending on the folder I. The value $m(\mathbf{X}_{\cdot t_j T_k}, \mathcal{I})$ is the mean of vector $\mathbf{X}_{\cdot t_j T_k}$ in $\mathcal{I}$:

$$m(\mathbf{X}_{\cdot t_j T_k}, \mathcal{I}) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \mathbf{X}[i, j, k]. \tag{3}$$

The metric encompasses the ability to weight the data based on its multiscale decomposition since each folder is assigned a weight via ω. The weights can incorporate prior smoothness assumptions on the data, and also enable enhancing either coarse or fine structures in the similarity between samples.

Following Leeb [22], in our setting of a 3D dataset, we propose a tree-based metric between two samples (2D matrices) in one dimension that incorporates the coupling of the other two dimensions, given their partition trees. We define this distance for the trial dimension, given trees on the time and neuron dimensions, but the same applies in the other dimensions as well. Given a partition tree $\mathcal{T}_r$ on the neurons and a partition tree $\mathcal{T}_t$ on the time frames, the distance between two trials T_k and T_n is defined as

$$d_{\mathcal{T}_r, \mathcal{T}_t}(\mathbf{X}_{\cdot\cdot T_k}, \mathbf{X}_{\cdot\cdot T_n}) = \sum_{\substack{\mathcal{I} \in \mathcal{T}_r \\ \mathcal{J} \in \mathcal{T}_t}} |m(\mathbf{X}_{\cdot\cdot T_k} - \mathbf{X}_{\cdot\cdot T_n}, \mathcal{I} \times \mathcal{J})| \times \omega(\mathcal{I}, \mathcal{J}), \tag{4}$$

where $\omega(\mathcal{I}, \mathcal{J}) > 0$ is a weight function depending on folders $\mathcal{I} \in \mathcal{T}_r$ and $\mathcal{J} \in \mathcal{T}_t$. We term this distance a bi-tree metric. The value $m(\mathbf{X}_{\cdot\cdot T_k}, \mathcal{I} \times \mathcal{J})$ is the mean value of a matrix $\mathbf{X}_{\cdot\cdot T_k}$ on the bi-folder $\mathcal{I} \times \mathcal{J} = \{(i, j) | i \in \mathcal{I}, j \in \mathcal{J}\}$:

$$m(\mathbf{X}_{\cdot\cdot T_k}, \mathcal{I} \times \mathcal{J}) = \frac{1}{|\mathcal{I}||\mathcal{J}|} \sum_{i \in \mathcal{I}, j \in \mathcal{J}} \mathbf{X}[i, j, k], \tag{5}$$

i.e., for a given trial T_k, we are averaging the sub-matrix of the 2D slice $\mathbf{X}_{\cdot\cdot T_k}$ defined by the subset of neurons in $\mathcal{I}$ and the subset of time frames in $\mathcal{J}$.

We present a new interpretation of the tree-based metrics (2) and (4). These metrics are equivalent to the l_1 distance between samples, after applying a multiscale transform to the data, where the tree metric (2) corresponds to a 1D transform and the bi-tree metric (4) corresponds to a 2D transform. For the sake of simplicity we begin with describing the 1D transform in the case of a single 2D slice of the trial data $\mathbf{X}_{\cdot\cdot T_k}$, and then generalize to the 2D transform.

The partition tree $\mathcal{T}_r$ can be seen as inducing a multiscale decomposition on the data, via the construction of a data-adaptive filter bank. Define the filter $f_{\mathcal{I}} \in \mathbb{R}^{n_r}$ as

$$f_{\mathcal{I}} = \frac{\omega(\mathcal{I})}{|\mathcal{I}|} \mathbb{1}_{\mathcal{I}}, \tag{6}$$

such that $\mathbb{1}_{\mathcal{I}}$ is the indicator function on the neurons $i \in \{1, \ldots, n_r\}$ belonging to folder $\mathcal{I} \in \mathcal{T}_r$. For each filter we calculate the inner product between the filter $f_{\mathcal{I}}$ induced by folder $\mathcal{I}$ and the measurement vector $\mathbf{X}_{\cdot t_j T_k} \in \mathbb{R}^{n_r}$, yielding a scalar coefficient $g_{\mathcal{I}}$:

$$\begin{aligned} g_{\mathcal{I}}(\mathbf{X}_{\cdot t_j T_k}) &= \langle f_{\mathcal{I}}, \mathbf{X}_{\cdot t_j T_k} \rangle \\ &= \frac{\omega(\mathcal{I})}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \mathbf{X}[i, j, k] = m(\mathbf{X}_{\cdot t_j T_k}, \mathcal{I}) \omega(\mathcal{I}). \end{aligned} \tag{7}$$

The tree $\mathcal{T}_r$ defines a multiscale transform by applying filter bank $f_{\mathcal{T}_r} = \{f_{\mathcal{I}}\}_{\mathcal{I} \in \mathcal{T}_r}$ to the measurements vector $\mathbf{X}_{\cdot t_j T_k}$, resulting in the set of coefficients $g_{\mathcal{T}_r} = \{g_{\mathcal{I}}\}_{\mathcal{I} \in \mathcal{T}_r}$. The filters of each level l of the tree output $n(l)$ coefficients, such that $g_{\mathcal{T}_r} : x \mapsto \mathbb{R}^{|\mathcal{T}_r|}$. This is demonstrated in Fig. 2. In the middle, a 2D slice $\mathbf{X}_{\cdot tT}$ is viewed as a 2D matrix and on the left is a partition tree $\mathcal{T}$ defined on the rows of the matrix. We assume that the rows of the matrix have been permuted so they correspond with leaves of the tree (level 0). In applying the transform $g_{\mathcal{T}}$, each folder $\mathcal{I}$ defines an element in the new vector $g_{\mathcal{T}}(X_{\cdot i})$ (right), proportional to the average of the entries in the original vector $(X_{\cdot i})$ on the support defined by the folder $\mathcal{I}$. The new entries in the vector are colored according to the corresponding folders in the tree.

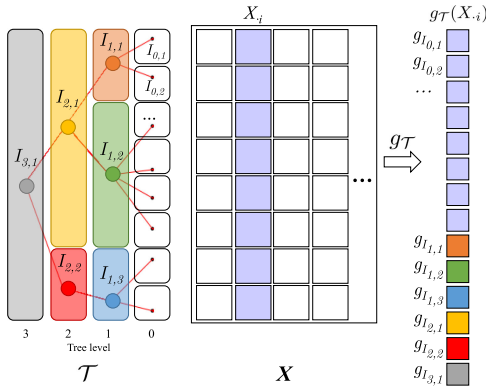

Fig. 2. Multiscale 1D tree-transform applied to a 2D slice from Fig. 1, viewed here as a 2D matrix (middle). On the left is a given partition tree $\mathcal{T}$ on the rows of the 2D matrix, and we assume the rows have been permuted so the leaves of the tree correspond to the rows. The partition tree $\mathcal{T}$ defines a multiscale transform on the columns of the matrix $X_{\cdot i}$, resulting in new vectors $g_{\mathcal{T}}(X_{\cdot i})$. In applying the transform $g_{\mathcal{T}}$, the entries in X_i corresponding to each folder in the tree, are averaged and weighted according to (7). This yields new scalar coefficients which form the output vector $g_{\mathcal{T}}(X_{\cdot i})$ (right). For visualization, each new entry g_I is colored by the corresponding folder I in the tree.

Theorem 4.1: Given a partition tree on the neurons $\mathcal{T}_r$, the tree metric (2) between two time frames t_i and t_j for a given trial T_k is equivalent to the l_1 distance between the multiscale transform defined by the tree and applied to the two vectors:

$$d_{\mathcal{T}_r}(\mathbf{X}_{\cdot t_i T_k}, \mathbf{X}_{\cdot t_j T_k}) = \|g_{\mathcal{T}_r}(\mathbf{X}_{\cdot t_i T_k}) - g_{\mathcal{T}}(\mathbf{X}_{\cdot t_j T_k})\|_1. \tag{8}$$

Proof:

$$d_{\mathcal{T}_r}(\mathbf{X}_{\cdot t_i T_k}, \mathbf{X}_{\cdot t_j T_k}) = \sum_{\mathcal{I}\in\mathcal{T}_r} |m(\mathbf{X}_{\cdot t_i T_k} - \mathbf{X}_{\cdot t_j T_k}, \mathcal{I})|\omega(\mathcal{I})$$

$$= \sum_{\mathcal{I}\in\mathcal{T}_r} |m(\mathbf{X}_{\cdot t_i T_k}, \mathcal{I})\omega(\mathcal{I}) - m(\mathbf{X}_{\cdot t_j T_k}, \mathcal{I})\omega(\mathcal{I})|$$

$$= \sum_{\mathcal{I}\in\mathcal{T}_r} |g_{\mathcal{I}}(\mathbf{X}_{\cdot t_i T_k}) - g_{\mathcal{I}}(\mathbf{X}_{\cdot t_j T_k})|$$

$$= \sum_{n=1}^{|\mathcal{T}_r|} |g_{\mathcal{T}_r}(\mathbf{X}_{\cdot t_i T_k})[n] - g_{\mathcal{T}_r}(\mathbf{X}_{\cdot t_j T_k})[n]|$$

$$= \|g_{\mathcal{T}_r}(\mathbf{X}_{\cdot t_i T_k}) - g_{\mathcal{T}_r}(\mathbf{X}_{\cdot t_j T_k})\|_1. \quad (9)$$

■

This result can be generalized to a multiscale 2D transform applied to 2D matrices as in our setting. Define the 2D filter $f_{\mathcal{I}\times\mathcal{J}}$ by:

$$f_{\mathcal{I}\times\mathcal{J}} = \frac{\omega(\mathcal{I},\mathcal{J})}{|\mathcal{I}||\mathcal{J}|}\mathbb{1}_{\mathcal{I}} \otimes \mathbb{1}_{\mathcal{J}}^T, \quad (10)$$

where $\otimes$ denotes the Kronecker product between the two indicator vectors. Then the elements of the 2D matrix $g_{\mathcal{T}_r,\mathcal{T}_t} \in \mathbb{R}^{|\mathcal{T}_r|\times|\mathcal{T}_t|}$ are the coefficients obtained from applying the 2D filter bank $f_{\mathcal{T}_r,\mathcal{T}_t} = \{f_{\mathcal{I}\times\mathcal{J}}\}_{\mathcal{I}\in\mathcal{T}_r,\mathcal{J}\in\mathcal{T}_t}$ defined by the bi-tree $\mathcal{T}_r \times \mathcal{T}_t$.

Corollary 4.2: The bi-tree metric (4) between two matrices given a partition tree $\mathcal{T}_r$ on the neurons and a partition tree $\mathcal{T}_t$ on the time frames is equivalent to the l_1 distance between the 2D multiscale transform of the two matrices:

$$d_{\mathcal{T}_r,\mathcal{T}_t}(\mathbf{X}_{\cdot\cdot T_k}, \mathbf{X}_{\cdot\cdot T_n}) = \|g_{\mathcal{T}_r,\mathcal{T}_t}(\mathbf{X}_{\cdot\cdot T_k}) - g_{\mathcal{T}_r,\mathcal{T}_t}(\mathbf{X}_{\cdot\cdot T_n})\|_1. \quad (11)$$

This interpretation of the metric as the l_1 distance between multiscale transforms has two computational advantages. First, given large datatsets, it is inefficient to calculate full affinity matrices on the samples, and instead sparse matrices are used by finding k-nearest neighbors of each sample. Thus, we can apply the multiscale transform to our data, yielding a new feature vector for each sample, and then apply approximate nearest-neighbor search for the l_1 distance to the new vectors [43], [44]. Second, we can relax the l_1 norm to other norms such as l_2 or l_∞. In future work, we intend to establish the properties of this transform and its application to other tasks.

Note that we claimed that regular metrics are inappropriate in processing the data due to its high-dimensionality in each dimension of the 3D dataset, i.e., each 2D slice of the data contain a large number of elements. This interpretation of the metric via the transform yields that the proposed metric is equivalent to the l_1 distance between vectors/matrices of even higher-dimensionality, supposedly contradicting our aim for a good metric. However, due to encompassing weights on the folders, the effective size of the new vectors is smaller than the original dimensionality, as the weights are chosen such that they rapidly decrease to zero based on the folder size.

We note that by using full binary trees in each of the two dimensions, the output of applying the multiscale transform is similar to that of applying the Gaussian pyramid representation, popular in image processing [45], to each 2D matrix $\mathbf{X}_{\cdot\cdot T_k}$, $1 \le k \le n_T$. Instead of applying the 5×5 Gaussian filter proposed by Burt and Adelson, our transform applies a 2×2 averaging filter, weighted by $\omega(\mathcal{I},\mathcal{J})$, and the resolution at each level will be reduced by 2 as in the Gaussian pyramid. Also, unlike the 2D Gaussian pyramid, our transform includes combinations of all fine and coarse scales in both dimensions.

Relationship to EMD: The Earth mover's distance (EMD) is a metric used to compare probability distributions or discrete histograms, and is popular in computer vision [46]. It is fairly insensitive to perturbations since it does not suffer from the fixed binning problems of most distances between histograms. EMD quantifies the difference between the two histograms as the amount of mass one needs to move (flow) between histograms, with respect to a ground distance, so they coincide. In its discrete form, the EMD between two normalized histograms h_1 and h_2 is defined as the minimal total ground distance "traveled" weighted by the flow:

$$\text{EMD}(h_1, h_2) = \min \sum_{i,j} g_{\text{ij}} d_{\text{ij}}$$

$$\text{s.t.} \quad \sum_i g_{\text{ik}} - \sum_j g_{\text{kj}} = h_1(k) - h_2(k),$$

where $d_{\text{ij}} \ge 0$ is the ground distance, and g_{ij} is the flow from bin i to bin j.

It was shown [47] that a proper choice of the weight $\omega(\mathcal{I})$ makes the tree metric (2) equivalent to EMD, i.e., the ratio of EMD to the tree-based metric is always between two constants. The proof follows the Kantorovich-Rubinstein theorem regarding the dual representation of the EMD problem. The weight $\omega(\mathcal{I})$ in (2) is chosen to depend on the tree structure:

$$\omega(\mathcal{I}) = \left(\frac{|\mathcal{I}|}{M}\right)^{\beta+1}, \quad (12)$$

where β weights the folder by its relative size. Positive values of β correspond to higher weights on coarser scales of the data, whereas negative values emphasize differences in fine structures in the data. For trees with varying-sized folders, unlike a balanced binary tree, β helps to normalize the weights on folders. For $\beta = 0$, the filter $f_{\mathcal{I}}$ defined in (6) is a uniform averaging filter whose support is determined by $\mathcal{I}$. In EMD the histograms are associated with a fixed grid and bins quantizing this grid. In our setting, where the data does not follow a fixed grid, the folders take the place of the bins, and by incorporating their multiscale structure via the weights, they can be seen as bins of varying sizes on the data.

Shirdhonkar and Jacobs [48] proposed a wavelet-based metric (wavelet EMD) shown to be equivalent to EMD. The wavelet EMD is calculated as the weighted l_1 distance between the wavelet coefficients of the difference between the two histogram. Following [48], Leeb [47] proposed a second metric based on the l_1 distance between the coefficients of the difference of distributions expanded in the tree-based Haar-like basis [37], which was also shown to be equivalent to EMD. Our interpretation of the metric (2) as an l_1 distance between a

multiscale filter bank applied to the data, simplifies the calculation even more as it does not require calculating the Haar-like basis defined by the tree, and instead requires only low-pass averaging filters on the support of each folder. This generalizes the wavelet EMD [48], to high-dimensional data that is not restricted to a Euclidean grid.

For the bi-tree metric (4), the weight on a bi-folder $\mathcal{I} \times \mathcal{J}$ can be chosen in an equivalent manner to (12) as

$$\omega(\mathcal{I},\mathcal{J}) = \left(\frac{|\mathcal{I}|}{n_r}\right)^{\beta_r+1}\left(\frac{|\mathcal{J}|}{n_t}\right)^{\beta_t+1}, \tag{13}$$

where β_r weights the bi-folder $\mathcal{I} \times \mathcal{J}$ based on the relative size of folder $\mathcal{I} \in \mathcal{T}_r$ and β_t weights the bi-folder based on the relative size of $\mathcal{J} \in \mathcal{T}_t$. The values should be set according to the smoothness of the dimension and whether we intend to enhance coarse or fine structures in the data.

C. Global Embedding

The intrinsic global representation of the data is attained by an integration process of local affinities, often termed "diffusion geometry". Specifically, the encoding of local variability and structure from different locations (e.g., cortical regions, or trials) is aggregated into a single comprehensive representation through the eigendecomposition of an affinity kernel [6]. This global embedding preserves local structures in the data, thus enabling us to exploit the fine spatio-temporal variations and inter-trial variability typical of biological data, in contrast to other methods based on averaging and smoothing the data [49].

Given the bi-tree multiscale distance between two samples (4), we can construct an affinity on the data along each dimension. We choose an exponential function, but other kernels can be considered, dependent on the application. Without loss of generality, we describe the embedding calculation with respect to the dimension of the neurons, but this procedure is applied to the time and trials as well, within our iterative framework. Given the multiscale distance $d_{\mathcal{T}_t,\mathcal{T}_T}(\mathbf{X}_{r_i\cdot\cdot}, \mathbf{X}_{r_j\cdot\cdot})$ between two neurons r_i and r_j, the affinity is defined as:

$$a(r_i,r_j) = \exp\{-d_{\mathcal{T}_t,\mathcal{T}_T}(\mathbf{X}_{r_i\cdot\cdot}, \mathbf{X}_{r_j\cdot\cdot})/\sigma_r\}, \tag{14}$$

where σ_r is a scale parameter, and depends on the current considered dimension of the 3D data, i.e., each dimension uses a different scale in its affinity. Typically, σ_r is chosen to be the mean of distances within the data. The exponential function enhances locality, as samples with distance larger than σ_r have negligible affinity.

The affinity is used to calculate a low-dimensional embedding of the data, using manifold learning techniques, specifically diffusion maps [6]. Defining an affinity matrix $\mathbf{A}[i,j] = a(r_i,r_j)$, $\mathbf{A} \in \mathbb{R}^{n_r \times n_r}$, we derive a corresponding row-stochastic matrix by normalizing its rows:

$$\mathbf{P} = \mathbf{D}^{-1}\mathbf{A}, \tag{15}$$

where $\mathbf{D}$ is a diagonal matrix whose elements are given by $\mathbf{D}[i,i] = \sum_j \mathbf{A}[i,j]$. The eigendecomposition of $\mathbf{P}$ yields a sequence of positive decreasing eigenvalues: $1 = \lambda_0 \geq \lambda_1 \geq \ldots$, and right eigenvectors $\{\psi_\ell\}_\ell$. Retaining only the first d eigenvalues and eigenvectors, the mapping Ψ_r embeds the data set X into the Euclidean space $\mathbb{R}^d$:

$$\Psi_r : \mathbf{X}_{r_i\cdot\cdot} \rightarrow \left(\lambda_1\psi_1(i), \lambda_2\psi_2(i), \ldots, \lambda_d\psi_d(i)\right)^T. \tag{16}$$

Note that for simplicity of notation we omit denoting the eigenvalues and eigenvectors by the relevant dimension r, t or T. The embedding provides a global low-dimensional representation of the data, which preserves local structures. The Euclidean distance between samples embedded in this space, termed the *diffusion distance*, is more meaningful than in the original high-dimensional space, as it has been shown to be robust to noise. The distance calculations $d^{(l)}(i,j)$ in the flexible tree construction are based on the embedding for these reasons. Finally, the embedding integrates the local connections found in the data into a global representation, which enables visualization of the data, reveals overlying temporal trends, organizes the data into meaningful clusters, and identifies outliers and singular samples. For more details on diffusion maps, see [6].

Algorithm 1: Hierarchical tri-geometric analysis.

Initialization

Input 3D data matrix $\mathbf{X}$

Starting with the neuron dimension r

  Calculate initial affinity matrix $\mathbf{A}_r^{(0)}$

  Calculate initial neuron embedding $\Psi_r^{(0)}$.

  Calculate initial flexible tree $\mathcal{T}_r^{(0)}$.

For time dimension t repeat steps 2-4 and obtain $\mathcal{T}_t^{(0)}$.

Iterative 3D analysis

Input Flexible trees $\mathcal{T}_r^{(0)}$ and $\mathcal{T}_t^{(0)}$

for $n \geq 1$ **do**

  Calculate multiscale bi-tree distance between two trials $d(T_i,T_j) = d_{\mathcal{T}_r^{(n-1)},\mathcal{T}_t^{(n-1)}}(\mathbf{X}_{\cdot\cdot T_i}, \mathbf{X}_{\cdot\cdot T_j})$

  Calculate trial affinity matrix $\mathbf{A}_T^{(n)}[i,j] = \exp\{-d(T_i,T_j)/\sigma_T\}$

  Calculate trial embedding $\Psi_T^{(n)}$

  Calculate flexible tree on the trials $\mathcal{T}_T^{(n)}$.

  For the neuron dimension r, repeat steps 7-10, given the trees on the time and trials, $\mathcal{T}_t^{(n-1)}$ and $\mathcal{T}_T^{(n)}$ respectively, and obtain $\mathcal{T}_r^{(n)}$.

  For the time dimension t, repeat steps 7-10, given the trees on the trials and neurons, $\mathcal{T}_T^{(n)}$ and $\mathcal{T}_r^{(n)}$ respectively, and obtain $\mathcal{T}_t^{(n)}$.

end for

D. Algorithm

Our iterative analysis algorithm composing all three components (tree construction, metric construction, embedding) is summarized in Algorithm 1. Each dimension is processed in turn, relying on the previous iteration of the other two dimensions. Specifically, calculation of the bi-tree metric for one dimension requires that partition trees be calculated on the other two dimensions. Therefore, an initialization is required.

To initialize the algorithm, one option is to calculate an initial affinity matrix based on a general distance such as the Euclidean distance or cosine similarity. We use the cosine similarity:

$$a^{\cos}(r_i, r_n) = \frac{\sum_{j,k} \mathbf{X}[i,j,k]\mathbf{X}[n,j,k]}{\sqrt{\sum_{j,k} \left(\mathbf{X}[i,j,k]\right)^2}\sqrt{\sum_{j,k} \left(\mathbf{X}[n,j,k]\right)^2}}. \tag{17}$$

Note that although the affinity is supposedly between two matrices, effectively it is equivalent to reshaping the matrices as 1D vectors and calculating the affinity using 1D distances. In other words, a general affinity does not take into account the 2D structure of the slices of the 3D data, in contrast to our bi-tree metric. In addition, these distances are uninformative, as the data are extremely high-dimensional. For example, in each dimension of the dataset in the experimental results in Section V, the dimension of the measurements is of order 10^4.

Given the initial affinity, an embedding and flexible tree are calculated for the neuron dimension r (steps 3-4). This is then repeated for the time dimension (step 1). A second option is to initialize the partition tree for the time dimension to be a binary tree, since the intra-trial time t is a smooth variable.

Given the trees in two of the dimensions, we can calculate the multiscale metric (4) in the trial dimension T (step 7). A corresponding embedding and flexible tree are then calculated (steps 9-10). We now have a partition tree in each dimension, so we continue in an iterative fashion, going over each dimension and calculating the multiscale metric, diffusion embedding and flexible tree in each iteration, based on the other two dimensions. The resulting output of the algorithm can be used to analyze the data both in terms of its hierarchical structure and through visualization of the embedding. Furthermore, each dimension can be organized by calculating a smooth trajectory in its embedding space. This yields a permutation on the indices of the given dimension. Permuting all three dimensions recovers the smooth structure of the data, respecting the coupling between the neurons and the time dimensions of the data. Python code implementing Algorithm 1 is available at [50].

Note that the order in which the dimensions are processed is arbitrary, and can affect the final results. The initialization of each dimension does not rely on the other two dimensions. However, the iterative phase relies on choosing in which order to process the three dimensions. This can affect the number of iterations required for the algorithm to achieve a meaningful representation of the data, i.e. more iterations may be required depending on the dimension we start with. In the case of trial-based data, we recommend to begin the iterations with either the neurons or the trials; since the time dimension is inherently smooth, its initial decomposition is usually good and relatively stable throughout the iterations. Thus, it is only necessary to initialize a second dimension and then begin the iterative organization procedure with the third.

V. Results

A. Experimental Setup

Our experimental data consists of repeated trials of a complex motor forepaw reach task in awake mice. The animals were trained to reach for a food pellet upon hearing an auditory cue [51]. This complex and versatile task exploits the capability of rodents to use their forepaw very similarly to distal hand movements in primates [51]. The hand reach task is typically learnt by mice over a period of few weeks to become "experts" (success rate of $\sim 70\% - 80\%$ after training over 2-3 weeks).

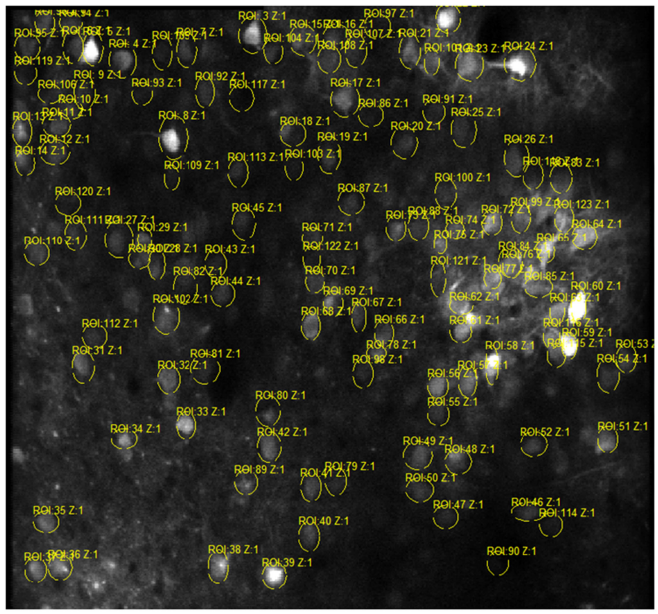

Fig. 3. Two-photon imaging in the primary motor cortex (M1). The neuronal measurements are gathered into regions of interest (ROIs), consisting of ellipses, and preprocessed as in (18)-(19).

Neuronal activity in the motor cortex during task performance was measured using two photon in-vivo calcium imaging with the recently developed genetically encoded indicators (GECIs) [52]. In addition, the network was silenced using DREADDS [53], which was activated using intraperitoneal (IP) injection of the inert agonist (clozapine-N-oxide - CNO). The analyzed neuronal measurements are of optical calcium fluorescent activity collected from a large population of identified neurons from cortical regions of interest, acquired using two photon microscopy imaging (see Fig. 3). In conjunction, high-resolution behavioral recordings of the subject are acquired using a camera (400 Hz). This serves to label the time frames and to determine whether the subject performed the task successfully during the trial.

The fluorescent measurements are manually grouped into elliptical regions of interest (ROIs) (see Fig. 3), and preprocessing is applied as follows. The spatial average fluorescence of each ROI k per time frame t in a single trial is

$$F_k(t) = \frac{1}{|\mathrm{ROI}_k|} \sum_{i,j \in \mathrm{ROI}} I[i,j,t], \tag{18}$$

where I is the florescence image, i and j are the pixel row and column indices in the image, respectively, and $|\mathrm{ROI}_k|$ is the area of the k-th ROI. The baseline florescence for ROI k in a single trial T is calculated using a subset of time frames S_k corresponding to the florescent averages $F_k(t)$ with the 10% lowest

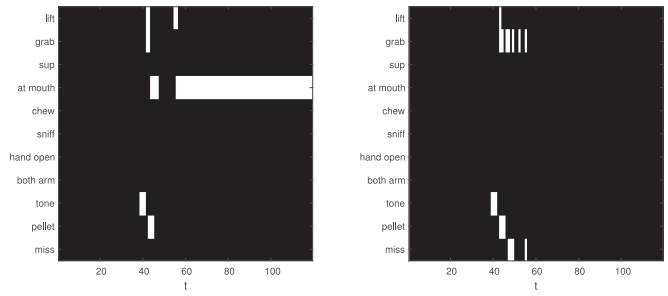

Fig. 4. Binary event labels for two trials. (left) Successful trial in which the subject grabs and eats the food pellet. (right) Failure in which the subject makes several failed attempts to grab the food.

values $\bar{F}_k = \sum_{t \in S_k} F_k(t)$. Finally, the neuron measurement at each time frame $\mathbf{X}[k,t,T]$ is set using $\frac{\Delta F}{F}$:

$$\mathbf{X}[k,t,T] = \frac{F_k(t) - \bar{F}_k}{\bar{F}_k}. \tag{19}$$

For simplicity, we refer to the ROIs as neurons in our analysis.

B. Data

We focus on neuronal measurements from the primary motor cortex region (M1), taken from a specific mouse in a single day of experimental training sessions. The data is composed of 59 consecutive trials, where the first 19 trials are considered "control" followed by 40 trials in which the activity of the somatosensory region was silenced by injection of CNO, thus activating DREADDS. Each trial lasts 12 seconds, during which activity in 121 neurons is measured for 119 time frames. Thus, the data can be seen as 3-dimensional, measuring a vector of neurons at each time frame within each trial. The data is visualized as 2D slices for several neurons, time frames and trials in Fig. 1.

Along with neuron measurements, we also have binary data labeling an event for each time and trial (see Fig. 4). The labeling is performed using a modified version of the machine learning based JAABA software, annotating discrete behavioral events [54]. There are 11 labeled events that provide additional prior information helpful in verifying our analysis. An auditory cue ("tone" event) is activated after 4 seconds (frames 40-42) and the food "pellet" comes to position at 4.4 seconds (frames 44-46). The "tone" event is typically followed by either a successful "grab" event and "at mouth" event, which lasts until the end of the trial, or by a several failed "grab" events and then labeled as a "miss" event, i.e., the subject failed to grab the food pellet and bring it to its mouth.

The control data consists of 19 trials, 11 of which were successful, i.e., the mouse managed to grab and eat the food pellet. After 19 trials, CNO was injected IP to silence the sensory cortex (S1), which sends feedback information to M1. The next 40 trials, referred to as "silencing trials" included 10 successful trials. During these trials, the behavior of the mouse changes, demonstrated by a decrease in "at mouth" (chewing) events and an increase in "miss" events (in which the mouse does not manage to grab the food). Note that not all silencing trials are "miss" and not all control trials are successful.

C. Tri-Geometric Analysis

The activity of the neurons is such that they are correlated at certain times, but completely unrelated at others, and certain neurons are sensitive to the auditory trigger, whereas others completely disregard it. The goal is to automatically extract coactive communities of neurons, as they relate to the activity of the mouse. We first analyze all 59 trials together, using Algorithm 1. For the weights (13) used in the mutilscale metric (4), we choose $\beta_r = 1, \beta_t = 1, \beta_T = 0$. We describe in the following how the analysis is used to derive meaningful results for each dimension.

Fig. 5 presents the 3D embedding of the time frames, where each 3D point is colored by the time frame $t \in \{1, \ldots, 119\}$ (a). The embedding clearly organizes the time frames through the various repetitive experiments into two dominant clusters: "pre-tone" and "post-tone" frames (see Fig. 5(b)), where the tone signifies the cue for the animal to begin the hand reach movement. We emphasize that this prior information was not used in the data-driven analysis. The embedding in effect isolates the time where the auditory tone is activated for the subject to reach for food.

Fig. 5(c) presents the first eleven non-trivial eigenvectors $\{\psi_d(t)\}_{d=1}^{11}$ obtained by the decomposition of the affinity matrix on the time dimension. Some eigenvectors correspond to harmonic functions over the entire interval $t \in [1, 119]$. However, some are localized either on the pre-tone region (e.g., $\psi_{t,9}$) or on the post-tone region (e.g., $\psi_{t,8}$ and $\psi_{t,11}$). In addition, each eigenvector captures the time at varying scales. This result demonstrates the power of our analysis; it shows that in a completely data-driven manner, a Fourier-like (harmonic) basis is attained. However, in contrast to the "generic" Fourier-basis, which is fixed, the obtained basis is data adaptive and captures and characterized true hidden phenomena related to external stimuli (the tone) and to different patterns of behavior (before and after the tone).

Thus, the embedding provides a verification of the knowledge we have regarding the time dimension in terms of regions of interest, and enables to pinpoint specific times of interest, essentially capturing the "script" of the trial. We do not present the local decomposition of the time frames via the flexible tree since it is not of interest, as this dimension is smooth, and therefore is just decomposed into local temporal neighborhoods.

We next examine the analysis of the trial dimension. In Fig. 6, we compare the embedding of the trials, obtained from the initial cosine affinity (a), and from the bi-tree multiscale metric (b). The points are colored by the trial index where blue corresponds to control trials (1-19), green-orange trials corresponds to the first silencing trials (19-44), and red corresponds to last silencing trials (45-59). Our tri-geometry analysis yields an embedding (see Fig. 6(b)) in which the blue and red points, corresponding to the first and last trials, respectively, are grouped together. This clearly indicates the temporal effect of silencing the somatosensory cortex on the activity of motor cortex. This is a promising result since solely from the neuronal activity, the data is self-organized functionally according to the brain activity manipulation we performed without the need to provide this

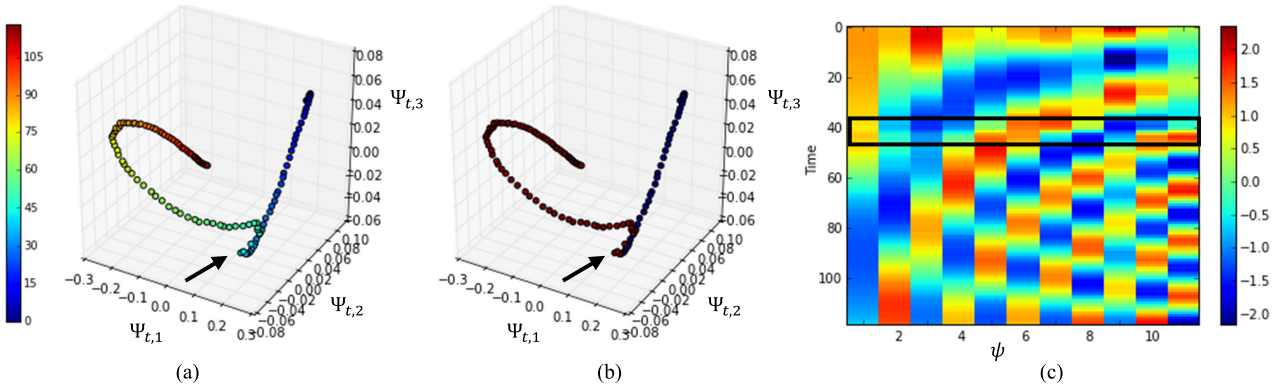

Fig. 5. Embedding of time frames. (a-b) 3-dimensional embedding of all the 2D time frame slices (as in Fig. 1(center), constructed by our tri-geometry analysis, where each time sample ($t \in \{1,\ldots,119\}$) is a 3D point. In (a) the points are colored by the time frame index, and in (b) they are colored according to pre-tone frames (blue) and post-tone frames (red). The tone, played at sample t=42 (marked by an arrow), is distinctively recovered from the data. (c) First 11 eigenvectors of time embedding. Each column is an eigenvector $\psi_{t,\ell} \in \mathbb{R}^{119}$ $\ell \in \{1,\ldots,11\}$. In general, the eigenvectors take the form of harmonic functions at different scales. Time $t = 42$ (the tone) is apparent (marked by the box). Some eigenvectors correspond to harmonic functions over the entire trial (e.g., $\psi_{t,1}$), while some are localized in the pre-tone region (e.g., $\psi_{t,9}$), and some in the post-tone region (e.g., $\psi_{t,11}$).

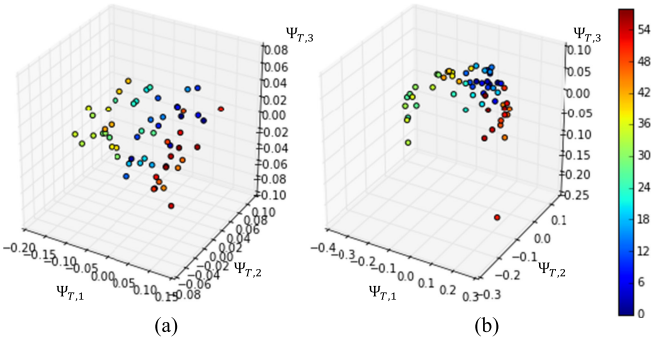

Fig. 6. The 3D embedding of the 2D trial slices (Fig. 1(left)) of all the trials $T \in \{1,\ldots,59\}$. Each trial slice is represented by a single 3D point, colored by the trial index (here as well, the trial index was not taken into account in the analysis). (a) Initial trial embedding based on cosine affinity. (b) Trial embedding derived from bi-tree multiscale metric. Trials are clustered in three main groups, where red and blue clusters are closer together.

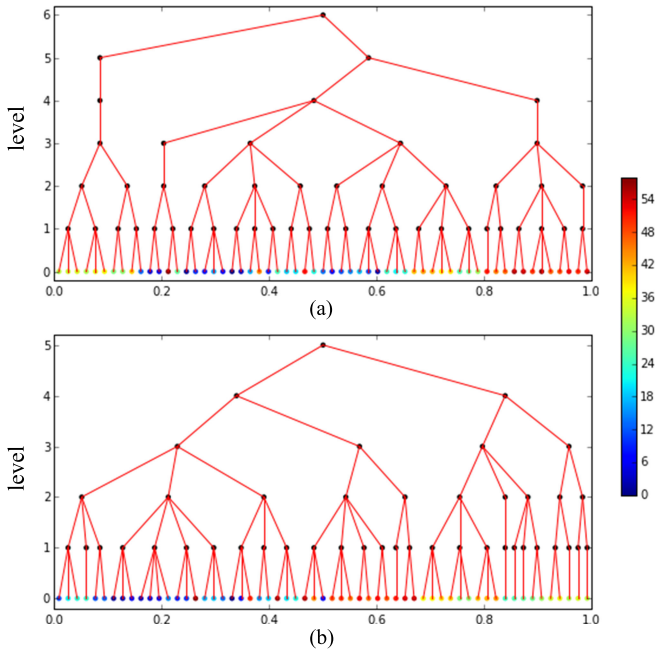

Fig. 7. Flexible tree of trials ($T \in \{1,\ldots,59\}$). The leaves are colored by trial index. (a) Tree corresponding to initial trial embedding in Fig. 6(a). (b) Tree corresponding to bi-tree multiscale metric embedding in Fig. 6(b). This tree better captures the nature of the trials, separating the pathological dysfunction caused by the silencing from the normal trials.

information during the analysis. This result leads us to hypothesize that our silencing manipulation has a lag, and also that it expires over the duration of the experiment. Our analysis recovers hidden biological cues and enables accurate indication of pathological dysfunction driven by neuronal activity evidence.

To highlight the contribution of our approach in the analysis of such data, we compare our embedding to the 3D diffusion maps obtained by the initial cosine affinity (see Fig. 6(a)), which does not exhibit any particular organization. Thus, the refinement via iterative application of the algorithm is essential. The multiscale local organization via the trees and coupling of the dimensions via the metric contribute to deriving a meaningful global embedding.

The improved clustering of the trials achieved by the bi-tree multiscale metric is also apparent when examining the flexible trees obtained from the two embeddings (see Fig. 7). The leaves are colored by the trial index as in the embedding. The tree obtained from the new embedding better separates the trials in which the pathological dysfunction caused by the silencing is evident from the normal trials. Remembering that flexible trees are constructed bottom-up using the embedding coordinates, this validates the claim that proximity in the embedding space captures the global temporal trend in the data.

To analyze the neurons, we split the data into two parts and analyze each separately, as this enables us to discover both behavioral patterns and pathological dysfunction. First, we examine the 40 trials composing the silencing trials. The neurons were preprocessed by subtracting the mean of each neuron over all trials, and normalizing it by its standard deviation across all trials. This enables us to examine the increase and decrease of

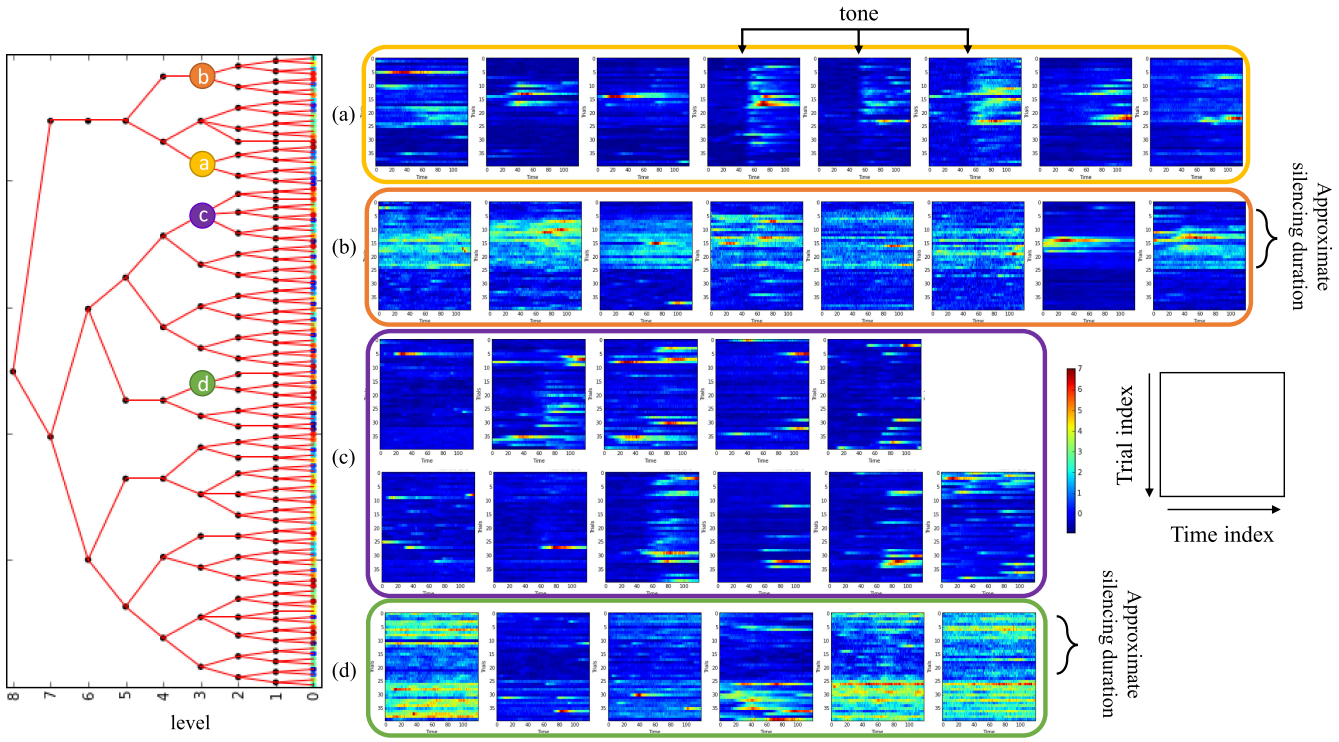

Fig. 8. Neuron tree for the silencing trials for iteration $n = 2$. To demonstrate the organization obtained by the tree, we highlight several interesting tree folders from level $l = 3$, marked with different colors and letters. Neurons belonging to the highlighted folders are grouped together, with a colored border corresponding to the folder color. Each neuron has been reorganized as a 2D matrix of $n_T \times n_t$ (40×119). The neurons are grouped together according to similar properties. (a) Yellow folder: 8 neurons that are active only at or after the tone (vertical separation), and mostly in trials under the effect of the silencing (horizontal separation). First three are associated with the tone itself, 5 right are associated with post-tone activity. (b) Orange folder: 8 neurons that were dominant mostly in trials under the effect of the silencing (horizontal separation), but are not sensitive to tone. (This node is joined with the yellow node at level $l = 5$). (c) Purple folder: 11 neurons that were mostly active during trials not under the effect of the silencing, 8 of which are active after the tone (vertical separation). (d) Green folder: 5 neurons that were silenced by the manipulation (horizontal separation).

activity in the neuron without being sensitive to the intensity of the measurements.

Fig. 8 presents the multiscale hierarchical organization of the 2D slices of all the neurons in a flexible tree $\mathcal{T}_r^{(2)}$, obtained after two iterations of our analysis, highlighting several interesting tree folders from level $l = 3$. Neurons composing four folders from this level are presented. The folders are marked in different colors on the tree and the neurons belonging to each folder are grouped together, with a border in corresponding color to the folder. Each neuron has been reorganized as a 2D matrix of size $n_T \times n_t$ (40×119). The neurons are grouped together according to similar properties and the displayed folders clearly relate to pathological dysfunction. For example, the orange folder consists of neurons that are active only during trials under the effect of somatosensory silencing (horizontal separation). The yellow folder consists of neurons that are active only at or after the tone (vertical separation), and mostly in trials under the effect of the silencing (horizontal separation). In contract, the purple folder consists of neurons, which are active after the tone but during trials without the silencing effect. Finally, the green folder consists of neurons, which were silenced by the manipulation. This leads us to hypothesize, as with the trial analysis, that the effect of the somatosensory silencing has a slight delay, and in addition that it wears off after a certain number of trials, since the experiment was very long. Our analysis groups neurons demonstrating the same activity patterns together in an automatic data-driven manner without manual intervention.

The silencing trials enable us to analyze the neurons in terms of how they are affected by the introduced virus. We now treat the 19 control trials, which allows us to analyze the behavioral aspect of the neurons without external intervention. In Fig. 9, we display the neuron tree, $\mathcal{T}_r^{(1)}$, obtained after one iteration of our analysis, and examine folders for levels $l = 2, 3, 4$. Neurons composing five folders from this level are presented. The folders are marked in different colors on the tree and the neurons belonging to each folder are grouped together, with a border in corresponding color to the folder. Each neuron has been reorganized as a 2D matrix of size $n_T \times n_t$ (19×119). We use the labeled "at mouth" event and the prior information on the time of the auditory tone to analyze the results. The binary labels indicating "at mouth" activity has also been reordered as a 2D matrix of size $n_T \times n_t$ (19×119), and is displayed in within the black border.

The results indicate that neurons are grouped by similarity, clearly related to the behavioral data. The upper two folders (red and orange) show increased activity before and during the auditory tone. The next three folders show increased activity

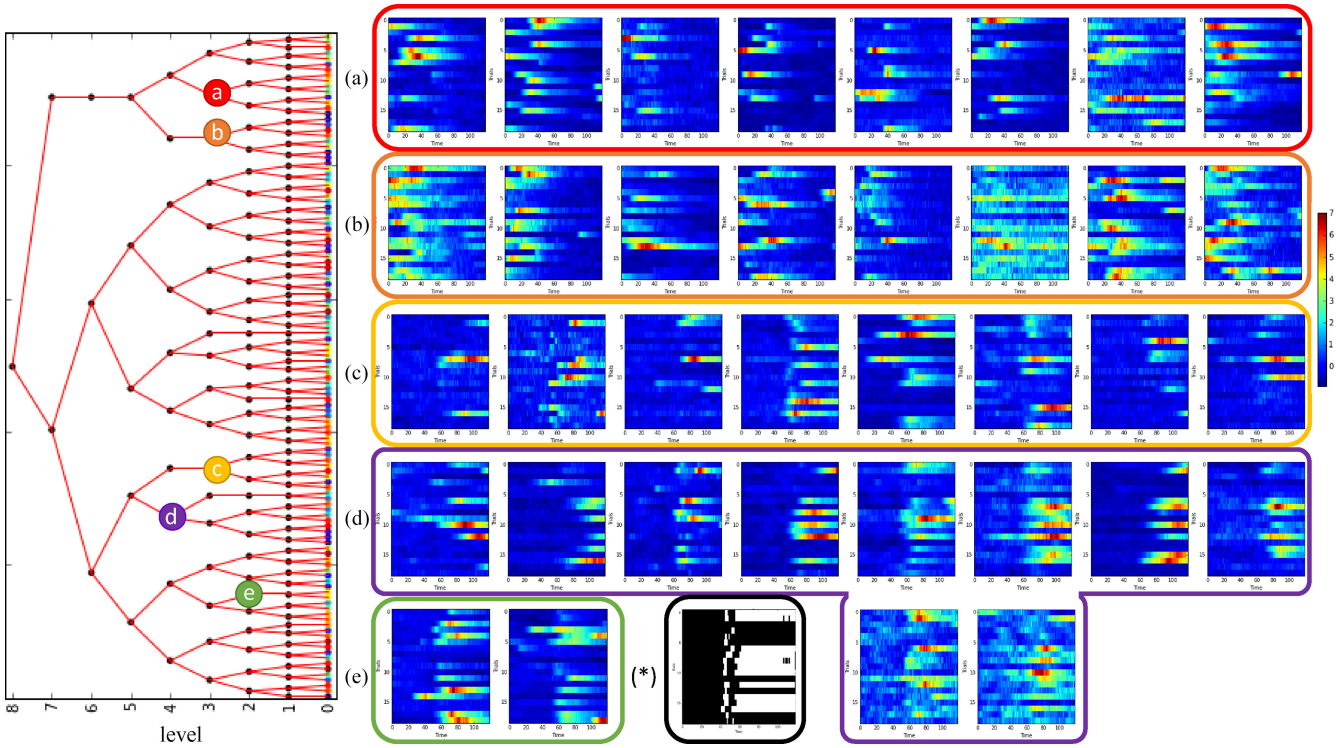

Fig. 9. Neuron tree for control trials for iteration $n = 1$. We highlight several interesting tree folders from level $l = 2 - 4$, marked with different colors and letters. Each neuron has been reorganized as a 2D matrix of $n_T \times n_t$ (19×119). Neurons belonging to the highlighted folders are grouped together, with a colored border corresponding to the folder color. (a-b) Red folder (8 neurons) and orange folder (8 neurons) in level $l = 3$ are active before the tone. (Note these nodes are joined at level $l = 5$). (c) Yellow folder: 8 neurons that are active post-tone. (d) Purple folder ($l = 4$): 10 neurons that are active post-tone only during trials which were labeled as "at mouth". (e) Green folder: 2 neurons that with significant activity post-tone only during trials which were labeled as "miss". (*) Black border contain binary labeling of at mouth event, ordered as $T \times t$ matrix.

after the tone. The yellow folder composes of neurons that are activity during different trials, regardless of "at mouth" activity. The purple folder, on the other hand, contains neurons that are active post-tone, almost entirely during which were successful, i.e., the subject managed to eat the food pellet, indicated by continuous "at mouth" labeling till the end of the trial. Finally, the green folder is composed of two neurons with the opposite activity. They are most active post-tone during trials in which the subject failed to eat the food pellet. Note that this analysis is data-driven, i.e., no prior information on the event labels is used in grouping the neurons.

In analysis of the neurons, the main contribution is the produced partition tree. The global embedding did not yield meaningful results, and the examination of local folders in the tree was most informative. Note that we are looking at a limited set of "sensors" since the neurons were manually grouped together into ROIs. In future work we intend to analyze a larger group of sensors, by examining all pixels acquired from the 2-photon imaging video separately. We know from previous work that increasing the number of sensors is typically beneficial to the iterative analysis. This will remove any introduced biases yielded by the pre-processing and enable to identify spatial structures not limited to ellipses.

Our experimental results demonstrate that our approach identifies for the first time (to the best of our knowledge), solely from observations and in a purely data-driven manner: (i) functional subsets of neurons, (ii) activity patterns associated with particular behaviors, and (iii) the dynamics and variability in these subsets and patterns as they are related to context and to different time scales (from the variability within a trial, to a global trend in trials, induced by the silencing method). In analyzing the time dimension, we pinpoint the time of the auditory trigger, and separate the time frames into multiscale local regions, before and after the trigger. Finally, in organizing the trials, we are able to both separate the trials to "success" and "failure" cases, and to determine a global trend that relates to an introduced external intervention. Thus, these methods lay a solid foundation for modeling the sensory-motor system by providing sufficiently fine structures and accurate view of the data to test our hypotheses, within an integrated computational theory of sensory-motor perception and action.

We note that conventional manifold learning tools did not yield any intelligent data organization for this case. Thus, organizing the neurons or the time samples separately by a 1D geometry using conventional manifold learning methods is inappropriate for this complex data. The fact, demonstrated here, that the neuronal activity of different types of neurons is correlated only during specific times, and might be random otherwise, verifies the need for coupled organization analysis which simultaneously organizes time, trials and neurons into tri-geometries.

VI. Conclusion

In this paper we presented a new data-driven methodology for the analysis of trial-based data, specifically trials of

neuronal measurements. Our approach relies on an iterative local to global refinement procedure, which organizes the data in coupled hierarchical structures and yields a global embedding in each dimension. Our analysis enabled extracting hidden biological cues and accurate indication of pathological dysfunction extracted solely from the measurements. We identified neuronal activity patterns and variability in these patterns related to external triggers and behavioral events, at different time scales, from recovering the local "script" of the trial, to a global trend across trials. In this paper we focused on neuronal measurements, but our approach is general and can be applied to other types of trial-based experimental data, and even to general high-dimensional datasets such as video, temporal hyperspectral measurements, and more.

In future work we intend to address theoretical extensions of our methodology and application-dependent aspects that are beyond the scope of the current paper. First, our approach relies on a symmetric affinity measure between samples. However, in processing time-series it can be of interest to reveal causality between measurements, for example as estimated by Granger causality [25]. Yet this requires introducing an asymmetric affinity as causal relationships imply a directional weight between nodes in the graph. An extension of our approach to include asymmetric affinities is non-trivial and will be explored in future work. Second, the algorithm does not introduce a stopping criterion for the iterative procedure. In the case of 2D data organization, Gavish introduced a "coherency" criterion to determine when the organization could be stopped [15]. This can be extended to our setting and relies on decomposing the data into a 3D Haar wavelet basis, which is beyond the scope of this paper.

An inherent aspect of neuroscience applications is the lack of ground-truth regarding the "true" connections between neurons as this is essentially unknown, and indeed is the goal of developing such data-driven analysis tools. Thus, regarding the analysis of the neuronal measurements, we have relied on the organization of the time and trial dimensions, which validates our prior knowledge on the experimental setting, as well as visual inspection of the nodes in the neurons tree, to assess the performance of our approach. In future work we intend to develop a quality measure to evaluate the output of the analysis in terms of smoothness. This will enable comparing different runs of the algorithm, for example using different initializations or different tree construction algorithms.

In addition, the current implementation performs a preprocessing of the two-photon imaging data by clustering them into ROIs. In future work, we intend to analyze the raw neuronal imaging measurements (all the pixels in the image). This significantly increases the number of "sensors" and should enable to learn complex spatial structures in the cortex. Finally, our analysis can be extended to higher dimensions, e.g., incorporating behavioral data as a fourth dimension in the neuronal measurements.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their constructive comments and useful suggestions.

REFERENCES

[1] G. M. Shepherd, "Corticostriatal connectivity and its role in disease," *Nature Rev. Neurosci.*, vol. 14, no. 4, pp. 278–291, 2013.
[2] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.
[3] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, 2000.
[4] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.
[5] D. L. Donoho and C. Grimes, "Hessian eigenmaps: New locally linear embedding techniques for high-dimensional data," *Proc. Nat. Acad. Sci. USA*, vol. 100, pp. 5591–5596, 2003.
[6] R. R. Coifman and S. Lafon, "Diffusion maps," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 5–30, Jul. 2006.
[7] J. T. Vogelstein *et al.*, "Discovery of brainwide neural-behavioral maps via multiscale unsupervised structure learning," *Science*, vol. 344, no. 6182, pp. 386–392, 2014.
[8] J. P. Cunningham and B. M. Yu, "Dimensionality reduction for large-scale neural recordings," *Nature Neurosci.*, vol. 17, pp. 1500–1509, 2014.
[9] J. Bennett and S. Lanning, "The Netflix prize," in *Proc. KDD Cup Workshop*, 2007, vol. 2007, pp. 3–6.
[10] Y. Cheng and G. M. Church, "Biclustering of expression data," in *Proc. Intell. Syst. Mol. Biol.*, 2000, vol. 8, pp. 93–103.
[11] C. Tang, L. Zhang, A. Zhang, and M. Ramanathan, "Interrelated two-way clustering: An unsupervised approach for gene expression data analysis," in *Proc. 2nd IEEE Int. Symp. Bioinform. Bioeng.*, 2001, pp. 41–48.
[12] S. Busygin, O. Prokopyev, and P. M. Pardalos, "Biclustering in data mining," *Comput. Oper. Res.*, vol. 35, no. 9, pp. 2964–2987, 2008.
[13] E. C. Chi, G. I. Allen, and R. G. Baraniuk, "Convex biclustering," 2014. [Online]. Available: http://arxiv.org/abs/1408.0856,arXiv:1408.0856 [stat.ME].
[14] R. R. Coifman and M. Gavish, "Harmonic analysis of digital data bases," in *Wavelets and Multiscale Analysis* (ser. Applied and Numerical Harmonic Analysis), J. Cohen and A. I. Zayed, Eds. Boston, MA, USA: Birkhäuser, 2011, pp. 161–197.
[15] M. Gavish and R. R. Coifman, "Sampling, denoising and compression of matrices by coherent matrix organization," *Appl. Comput. Harmon. Anal.*, vol. 33, no. 3, pp. 354–369, 2012.
[16] A. Singer and R. R. Coifman, "Non-linear independent component analysis with diffusion maps," *Appl. Comput. Harmon. Anal.*, vol. 25, no. 2, pp. 226–239, 2008.
[17] R. Talmon, I. Cohen, S. Gannot, and R. R. Coifman, "Supervised graph-based processing for sequential transient interference suppression," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 9, pp. 2528–2538, Nov. 2012.
[18] R. Talmon and R. R. Coifman, "Empirical intrinsic geometry for nonlinear modeling and time series filtering," *Proc. Nat. Acad. Sci. USA*, vol. 110, pp. 12535–12540, 2013.
[19] R. Talmon and R. R. Coifman, "Intrinsic modeling of stochastic dynamical systems using empirical geometry," *Appl. Comput. Harmon. Anal.*, vol. 39, pp. 138–160, 2014.
[20] A. Haddad, D. Kushnir, and R. R. Coifman, "Texture separation via a reference set," *Appl. Comput. Harmon. Anal.*, vol. 36, no. 2, pp. 335–347, Mar. 2014.
[21] G. Mishne, R. Talmon, and I. Cohen, "Graph-based supervised automatic target detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2738–2754, May 2015.
[22] W. E. Leeb, "Topics in metric approximation," Ph.D. dissertation, Dept. of Mathematics, Yale University, New Haven, CT, USA, 2015.
[23] K. Friston, R. Moran, and A. K. Seth, "Analysing connectivity with Granger causality and dynamic causal modelling," *Current Opinion Neurobiol.*, vol. 23, no. 2, pp. 172–178, 2013.
[24] O. Sporns, "Contributions and challenges for network models in cognitive neuroscience," *Nature Neurosci.*, vol. 17, no. 5, pp. 652–660, 2014.
[25] M. Ding, Y. Chen, and S. Bressler, "Granger causality: Basic theory and application to neuroscience," in *Handbook of Time Series Analysis: Recent Theoretical Developments and Applications*. New York, NY, USA: Wiley, 2006, pp. 437–460.
[26] Schreiber, "Measuring information transfer," *Phys. Rev. Lett.*, vol. 85, no. 2, pp. 461–464, 2000.

[27] W. Truccolo, U. Eden, M. Fellows, J. Donoghue, and E. Brown, "A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects," *J. Neurophysiol*, vol. 93, no. 2, pp. 1074–1089, 2005.
[28] K. V. Shenoy, M. Sahani, and M. M. Churchland, "Cortical control of arm movements: a dynamical systems perspective," *Annu. Rev. Neurosci.*, vol. 36, pp. 337–359, 2013.
[29] E. W. Archer, U. Koster, J. W. Pillow, and J. H. Macke, "Low-dimensional models of neural population activity in sensory cortical circuits," in *Proc. Adv. Neural Inform. Process. Syst.*, 2014, pp. 343–351.
[30] W. Wu, M. Black, D. Mumford, Y. Gao, E. Bienenstock, and J. Donoghue, "Modeling and decoding motor cortical activity using a switching Kalman filter," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 933–42, Jun. 2004.
[31] Y. Ahmadian, J. W. Pillow, and L. Paninski, "Efficient Markov chain Monte Carlo methods for decoding neural spike trains," *Neural Comput.*, vol. 23, no. 1, pp. 46–96, 2011.
[32] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, 2009.
[33] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, Sep. 1966.
[34] R. A. Harshman, "Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis," *UCLA Working Papers Phonetics*, vol. 16, pp. 1–84, 1970.
[35] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "MPCA: Multilinear principal component analysis of tensor objects," *IEEE Trans. Neural Netw.*, vol. 19, no. 1, pp. 18–39, Jan. 2008.
[36] R. Talmon, S. Mallat, H. Zaveri, and R. Coifman, "Manifold learning for latent variable inference in dynamical systems," *IEEE Trans. Signal Process.*, vol. 63, no. 15, pp. 3843–3856, Aug. 2015.
[37] M. Gavish, B. Nadler, and R. R. Coifman, "Multiscale wavelets on trees, graphs and high dimensional data: Theory and applications to semi supervised learning," in *Proc. Int. Conf. Mach. Learn.*, 2010, pp. 367–374.
[38] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[39] J. I. Ankenman, "Geometry and analysis of dual networks on questionnaires," Ph.D. dissertation, Dept. of Mathematics, Yale University, New Haven, CT, USA, 2014.
[40] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol. 466, no. 7307, pp. 761–764, 2010.
[41] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Comput. Soc. Conf. Comput Vis. Pattern Recognit.*, 2005, vol. 2, pp. 60–65.
[42] I. Ram, M. Elad, and I. Cohen, "Generalized tree-based wavelet transform," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4199–4209, Sep. 2011.
[43] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An optimal algorithm for approximate nearest neighbor searching fixed dimensions," *J. ACM*, vol. 45, no. 6, pp. 891–923, Nov. 1998.
[44] B.-K. Yi and C. Faloutsos, "Fast time sequence indexing for arbitrary l_p norms," in *Proc. Very Large Data Bases Conf.*, pp. 385–394, 2000.
[45] P. Burt and E. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Commun.*, vol. 31, no. 4, pp. 532–540, Apr. 1983.
[46] Y. Rubner, C. Tomasi, and L. J. Guibas, "A metric for distributions with applications to image databases," in *Proc. 6th Int. Conf. Comput. Vis.*, 1998, pp. 59–66.
[47] R. R. Coifman and W. E. Leeb, "Earth mover's distance and equivalent metrics for spaces with hierarchical partition trees," Yale University, New Haven, CT, USA, Tech. Rep. YALEU/DCS/TR1482, 2013.
[48] S. Shirdhonkar and D. Jacobs, "Approximate earth mover's distance in linear time," in *Proc. Conf. Comput. Vis. Pattern Recognit*, Jun. 2008, pp. 1–8.
[49] D. Pfau, E. A. Pnevmatikakis, and L. Paninski, "Robust learning of low-dimensional dynamics from large neural ensembles," in *Proc. Adv. Neural Inform. Process. Syst.*, 2013, pp. 2391–2399.
[50] 2016. [Online]. Available: http://github.com/gmishne/pyquest
[51] I. Q. Whishaw, S. M. Pellis, and B. P. Gorny, "Skilled reaching in rats and humans: evidence for parallel development or homology," *Behav. Brain Res.*, vol. 47, no. 1, pp. 59–70, 1992.
[52] T.-W. E. A. Chen, "Ultrasensitive fluorescent proteins for imaging neuronal activity," *Nature*, vol. 499, no. 7458, pp. 295–300, 2013.
[53] S. C. Rogan and B. L. Roth, "Remote control of neuronal signaling," *Pharmacol. Rev.*, vol. 63, no. 2, pp. 291–315, 2011.
[54] M. Kabra, A. A. Robie, M. Rivera-Alba, S. Branson, and K. Branson, "JAABA: Interactive machine learning for automatic annotation of animal behavior," *Nature Methods*, vol. 10, no. 1, pp. 64–67, 2013.

Gal Mishne received the B.Sc. degree (*summa cum laude*) in electrical engineering and in physics in 2009 from the Technion—Israel Institute of Technology, Haifa, Israel, where she is currently working toward the Ph.D. degree in electrical engineering. From 2008 to 2013, she was an Image Processing Engineer in the Israeli Defense Industry. Her research interests include signal processing, image processing, and geometric methods for data analysis. She received the Wolf Foundation Award for Ph.D. students, the Porat Award and the Jacobs-Qualcomm Fellowship in 2016, the Daniel Fellowship and the Freud Award in 2015, and the Ollendorff Fellowship in 2014.

Ronen Talmon received the B.A. degree (*cum laude*) in mathematics and computer science from the Open University, Ra'anana, Israel, in 2005, and the Ph.D. degree in electrical engineering from the Technion—Israel Institute of Technology, Haifa, Israel, in 2011.

From 2000 to 2005, he was a Software Developer and Researcher with the technological unit of the Israeli Defense Forces. From 2005 to 2011, he was a Teaching Assistant in the Department of Electrical Engineering, Technion. From 2011 to 2013, he was a Gibbs Assistant Professor in the Mathematics Department, Yale University, New Haven, CT, USA. In 2014, he joined the Department of Electrical Engineering, Technion. He is currently an Assistant Professor of electrical engineering with the Technion. His research interests include statistical signal processing, analysis and modeling of signals, speech enhancement, biomedical signal processing, applied harmonic analysis, and diffusion geometry.

Dr. Talmon received the Irwin and Joan Jacobs Fellowship, the Andrew and Erna Fince Viterbi Fellowship, and the Horev Fellowship.

Ron Meir received the B.Sc. degree in physics and mathematics from the Hebrew University, Jerusalem, Israel, in 1982, and the M.Sc. and Ph.D. degrees in theoretical physics from Weizmann Institute of Science, Rehovot, Israel, in 1984 and 1988, respectively. He was a Weizmann Research Fellow with California Institute of Technology during 1988–1990 and then he joined Bell Communications Research, Morristown, NJ, USA. Since 1994 he has been a Professor with the Faculty of the Electrical Engineering Department, Technion—Israel Institute of Technology, Haifa, Israel. His current research interests include information processing and control in neural systems, reinforcement learning in natural and artificial systems, the perception-action cycle, bottom-up and top-down inference and learning in deep neural network.

Jackie Schiller received the Ph.D. degree in physiology and biophysics from the Hebrew University, Jerusalem, Israel. From 1993 to 1995, she was a Postdoctoral Fellow in the Max-Planck-Institute for Medical with Prof. B. Sakmann. Later from 1995 to 1997, she joined Prof. D. E Clapham as a Postdoctoral Fellow in the Mayo Clinic, Rochester, MN, USA. In 2000, she established her lab at the Ruth and Bruce Rappaport Faculty of Medicine, Technion—Israel Institute of Technology, Haifa, Israel. She is a Professor in the Department of Neuroscience, Ruth and Bruce Rappaport Faculty of Medicine, Technion. Her work is recognized worldwide, and she is considered a Leader in the field of dendritic computation and cortical physiology. She is frequently invited to participate in important international conferences and forums.

Maria Lavzin received the B.Sc. degree (*cum laude*) in medical sciences in 2009 from the Technion—Israel Institute of Technology, Haifa, Israel, where she is a currently working toward the M.D./Ph.D. degree at the Ruth and Bruce Rappaport Faculty of Medicine. From 2007 to 2009, she was a Research Assistant and from 2010 to 2013 a Teaching Assistant with the Technion Research and Development Foundation, Faculty of Medicine, Department of Physiology. Her research interests include neuronal computation, cellular physiology, neuronal networks, sensory processing, motor learning and behavior. She received the Clore Israel Foundation Scholars fellowship and the Foulkes Foundation fellowship.

Uri Dubin received the B.Sc. and M.Sc. degrees in electrical engineering from the Technion—Israel Institute of Technology, Haifa, Israel, in 1995 and 2006, respectively. He is currently working toward the Ph.D. degree at the Faculty of Medicine, Technion. From 1998 to 2012, he was with Elbit Systems, working on multidisciplinary projects. From 2012 to 2016, he joined the medical startup company TytoCare. He research interest include in signal processing, image analysis, image understanding, and neural modeling.

Ronald R. Coifman received the Ph.D. degree from the University of Geneva, Geneva, Switzerland, in 1965. Prior to coming to Yale in 1980, he was a Professor with Washington University, St Louis, MO, USA. From 1986 to 1989, he was the Chairman of the Mathematics Department, Yale University, New Haven, CT, USA, where he is currently a Phillips Professor of mathematics. He is currently leading a research program to develop new mathematical tools for efficient transcription of data, with applications to feature extraction recognition, denoising, and information organization. His recent publications have been in the areas of nonlinear Fourier analysis, wavelet theory, numerical analysis, and scattering theory. He is a Member of the National Academy of Sciences, American Academy of Arts and Sciences, and the Connecticut Academy of Sciences and Engineering. He received the DARPA Sustained Excellence Award in 1996, the 1996 Connecticut Science Medal, the 1999 Pioneer award from the International Society for Industrial and applied Mathematics, the National Science Medal in 1999, and the Wavelet Pioneer Award in 2007.