

---

# TRACE: Contrastive learning for multi-trial time-series data in neuroscience

---

Lisa Schmors<sup>1</sup> Dominic Gonschorek<sup>2,3</sup> Jan Niklas Böhm<sup>1</sup> Yongrong Qiu<sup>7-9</sup>  
Na Zhou<sup>4,5</sup> Dmitry Kobak<sup>1</sup> Andreas Tolias<sup>7-10</sup> Fabian Sinz<sup>4-6</sup>

Jacob Reimer<sup>4,5</sup> Katrin Franke<sup>2,7-9</sup> Sebastian Damrich<sup>1</sup> Philipp Berens<sup>1</sup>

<sup>1</sup> Hertie Institute for AI in Brain Health, University of Tübingen, Tübingen, Germany

<sup>2</sup> Institute for Ophthalmic Research, University of Tübingen, Germany

<sup>3</sup> Centre for Integrative Neuroscience, University of Tübingen, Germany

<sup>4</sup> Department of Neuroscience, Baylor College of Medicine, Houston, USA

<sup>5</sup> Center for Neuroscience and Artificial Intelligence, Baylor College of Medicine, Houston, USA

<sup>6</sup> Institute for Computer Science and Campus Institute Data Science, University of Göttingen, Germany

<sup>7</sup> Department of Ophthalmology, Byers Eye Institute, Stanford University School of Medicine, USA

<sup>8</sup> Stanford Bio-X, Stanford University, USA

<sup>9</sup> Wu Tsai Neurosciences Institute, Stanford University, USA

<sup>10</sup> Department of Electrical Engineering, Stanford University, USA

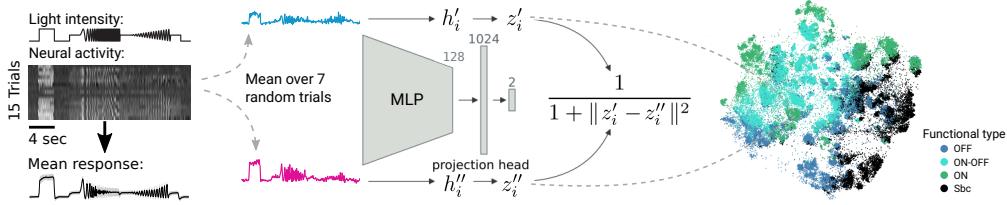
## Abstract

Modern neural recording techniques such as two-photon imaging or Neuropixel probes allow to acquire vast time-series datasets with responses of hundreds or thousands of neurons. Contrastive learning is a powerful self-supervised framework for learning representations of complex datasets. Existing applications for neural time series rely on generic data augmentations and do not exploit the multi-trial data structure inherent in many neural datasets. Here we present TRACE, a new contrastive learning framework that averages across different subsets of trials to generate positive pairs. TRACE allows to directly learn a two-dimensional embedding, combining ideas from contrastive learning and neighbor embeddings. We show that TRACE outperforms other methods, resolving fine response differences in simulated data. Further, using *in vivo* recordings, we show that the representations learned by TRACE capture both biologically relevant continuous variation, cell-type-related cluster structure, and can assist data quality control.

## 1 Introduction

With advances in recording techniques, datasets in neuroscience have grown in size and complexity [7, 38]. For example, two-photon imaging and Neuropixel probes have made it possible to record responses of tens of thousands of neurons from multiple cortical areas under comparable conditions [12, 36]. To summarize and visually explore such noisy, high-dimensional data, it is invaluable to represent it in two dimensions to identify functionally similar groups of neurons [47].

A prominent paradigm for learning informative representations of data is contrastive learning [17, 29]. Here, representations are created by contrasting similar samples (referred to as “positive pairs”) against dissimilar ones (“negative pairs”), ensuring that similar samples are grouped together, while dissimilar ones are separated. Contrastive learning has been popular for image data for some years [8], but has only recently seen first applications for neuroscience time-series data [35, 42]. For example, the contrastive method CEED [42] learns representations of extracellular action potential wave forms using generic data augmentations such as amplitude jitter. However, most contrastive learning methods, including CEED, do not directly embed into two dimensions necessary for visualization.



**Figure 1: TRACE embeds multi-trial time series using subset means for positive-pair generation.** *Left:* Typical experimental structure in neuroscience: multiple trials of neural activity in response to repeats of an identical stimulus (here, full-field light intensity modulations). For TRACE, positive pairs are generated using means of subsets of trials. *Middle:* Positive pairs are fed through an MLP and a fully-connected projection head to get representations  $z'_i$  and  $z''_i$ . The loss function pushes  $z'_i$  and  $z''_i$  together and maximizes their similarity. *Right:* Final embedding of large-scale neuroscience dataset recorded in superior colliculus. Color-coded according to their functional group.

Here, we exploit a common feature of neuroscience experiments – multiple recorded responses to identical stimuli – to develop a new contrastive learning framework tailored for multi-trial time series in neuroscience. This framework directly learns a two-dimensional (2D) representation of the data. Building on  $t$ -SimCNE [6], we present TRACE: **T**ime series **R**epresentation **A**nalysis through **C**ontrastive **E**mbeddings. Instead of using hand-tuned or generic data augmentations, TRACE creates positive pairs by averaging across different subsets of trials, better capturing the structure of time-series variation in neural responses (Fig. 1).

Using a synthetic dataset, we show that TRACE indeed automatically identifies informative parts of a time series, where other methods are “drowning in noise”. We then investigate the performance of TRACE on a real-world two-photon-imaging and a Neuropixels time-series dataset. TRACE produces visualizations that capture biological properties of the recorded cells better than other visualization techniques. Our code is available at <https://github.com/berenslab/TRACE/tree/neurips25> and [https://github.com/berenslab/TRACE\\_experiments](https://github.com/berenslab/TRACE_experiments).

In summary, our contributions are:

1. We introduce a novel contrastive self-supervised framework for visualizing multi-trial time-series data in neuroscience in 2D using the Cauchy similarity.
2. We develop a novel data augmentation technique for multi-trial time-series data based on averaging subsets of trials to create positive pairs.
3. We establish the superior performance of our visualization method in terms of quality metrics and alignment with biologically interesting cellular properties.
4. We show how the method can be used for identifying recording artifacts and investigating outliers.

## 2 Related work

Dimensionality reduction of neural data has been used in computational neuroscience for (1) assisting the discovery of cell types or continuous functional variation and (2) finding low-dimensional population dynamics. For (1), each point in the computed low-dimensional representation corresponds to a neuron, while for (2), each point corresponds to a time point. Our approach addresses task (1).

Towards this goal, two of the most prominent non-linear visualization methods are  $t$ -SNE [41] and UMAP [28]. While mostly used for single-cell transcriptomics data, they have also been employed for wave-shape classification for electrophysiological data [23].  $t$ -SNE and UMAP learn a non-parametric 2D embedding, guided by pairs of nearest neighbors in data space. However, nearest neighbors in the high-dimensional space can be a poor proxy for semantic similarity due to the curse of dimensionality [18]. Alternatively, self-supervised learning approaches train encoder networks to produce high-dimensional representations. For visual exploration, these need to be further reduced, e.g. using PCA [32]. Instead of pairs of nearest neighbors, these methods rely on data augmentations, better capturing the semantic similarities of samples. The type of augmentation depends on the data modality, e.g. randomly resized crops and flips for images [8] or sampling consecutive time steps in

speech recordings [29]. Notably, *t*-SimCNE [6] uses data augmentations to create positive pairs of images but embeds into 2D to visualize the data.

Contrastive learning has been applied to medical time-series data, in particular electrocorticogram (ECoG) data. In addition to general time-series augmentations like scaling, blurring [34], and jittering [15], frequency- [49], cutout- [9], and permutation-based augmentations [34] have been used, which are not applicable to multi-trial non-periodic, stimulus-response data. Some of these works [13, 21] also create positive pairs based on patient identity. Mix-up [48] creates augmentations by blending different samples [4], which was also applied to ECoG data [9, 44]. Patient-based positive pairs or mix-up are similar in spirit to our approach. An important difference is that they only mix or pair two single trial samples and not a larger subset of trials, as we do. Our approach balances positive pair variability and similarity to inference-time inputs.

In neuroscience, there has been comparatively little work on contrastive learning for time-series representations with the goal of identifying discrete cell types (i.e. clusters). Peterson et al. [33] use a self-supervised model to generate pseudo labels and enrich the collected data this way. Cho et al. [10] suggest a time-warping loss, which is not applicable to our experiments, which are aligned in time by construction. Related to our approach, the method CEED has been used for spike sorting and cell-type classification [42]. CEED generates representations using generic time-series as well as task-specific augmentations such as spike collision and channel subset selection. As is common for contrastive self-supervised learning, CEED does not embed into two dimensions directly, but relies on an additional dimensionality reduction step for a 2D visualization, unlike our method. Moreover, TRACE captures the local noise structure better than CEED’s generic augmentations (Fig. 2).

For learning representations of neural population dynamics, methods like LFADS [31], Swap-VAE [26], or the contrastive approaches MYOW [3], CEBRA [35], SinkDivLM [40], and Neuroformer [2] produce embeddings in which the temporal evolution of the population activity and behavioral modulation of neural activity can be explored. Here, each embedding point corresponds to a time point, not a single-neuron response like in TRACE. Thus, they tackle a fundamentally different problem and are not applicable in our setting (see point (2) in the beginning of this section).

Similarly, many general-purpose contrastive time-series models, like TNC [39], also produce time-point embeddings and are thus not applicable in our setting. Others, such as TS2Vec [46] contrast instances of time series, but learn high-dimensional embeddings, not geared towards visualization. We find that TRACE outperforms TS2Vec even when adapting the latter to a visualization setting.

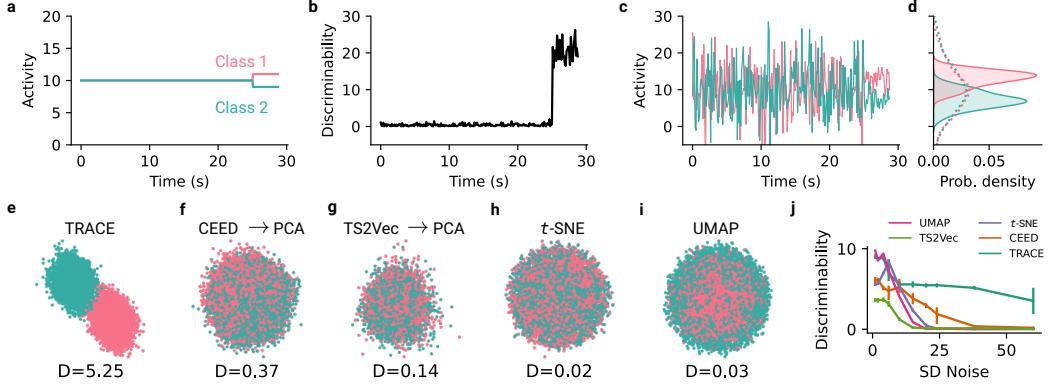
NEMO [45] and PhysMAP [24] contrast different electrophysiological modalities, while TRACE works by contrasting multiple recordings of the same modality.

### 3 TRACE: contrastive learning for multi-trial time-series data in neuroscience

Our goal is to create informative 2D embeddings of multi-trial time-series data from neuroscience experiments, supporting data exploration, cell-type discovery, and other clustering tasks. To do so, we developed TRACE, a method that makes use of the trial-based nature of many neuroscience experiments to create positive pairs by averaging over subsets of trials. It builds on the self-supervised method *t*-SimCNE [6], which is based on the SimCLR framework [8] and directly learns a 2D embedding. We briefly review contrastive learning and *t*-SimCNE before describing our approach. Finally, we will briefly describe the contrastive learning frameworks CEED and TS2Vec as our closest competitors using generic data augmentations for time series [42].

#### 3.1 Contrastive self-supervised visualizations

Given a dataset in a data space  $X$ , contrastive self-supervised learning trains the parameters  $\theta$  of a neural network  $f_\theta : X \rightarrow Z$  to obtain representations  $z = f_\theta(x)$  in embedding space  $Z$ . It learns salient features by making the embedding invariant to known data similarities encoded in positive pairs, which are typically obtained as two modifications  $x', x''$  of the same data sample  $x$ . Similarity in the embedding space is measured by a similarity function  $q : Z \times Z \rightarrow \mathbb{R}$ , Eqs. (2, 3). Positive pairs  $(x', x'')$  should have high similarity  $q(z', z'')$ , i.e.  $z'$  and  $z''$  should be close. This is typically



**Figure 2: Synthetic dataset shows ability of TRACE to identify cell types in noisy data.** (a) Simulation of two classes of neural responses with baseline (25 s) and type-specific response (5 s) periods. In the baseline period, both classes show 10 Hz activity, while in the type-specific response period they fire at 11 Hz and 9 Hz, respectively. Responses are corrupted by variable amounts of noise (Sec. 4.1). (b) Discriminability [14] of the two neuron classes per bin was measured using  $D = \frac{|\mu_1 - \mu_2|}{0.5(\sigma_1 + \sigma_2)}$  (mean across neurons shown). (c) Mean neural responses across 10 trials for two example neurons from the two types. (d) Distributions of baseline (dashed) and response periods (solid) for the two neuronal responses in (c). (e–i) Embeddings of 10k simulated neurons for baseline activity noise SD = 38 with 10 trials each. (j) Discriminability for simulated responses with increasing amounts of noise during the neural baseline activity. Error bars indicate 95% confidence intervals.

achieved with the InfoNCE loss, which for the  $i$ -th positive pair  $(x'_i, x''_i)$  of the training batch is

$$\mathcal{L}(x'_i, x''_i) = -\log \frac{q(z'_i, z''_i)}{q(z'_i, z''_i) + \sum_{\alpha \neq i} (q(z'_i, z'_\alpha) + q(z'_i, z''_\alpha))}. \quad (1)$$

Here,  $\alpha$  runs over all other positive pairs in the training batch. The pairs  $(z'_i, z'_\alpha)$  and  $(z'_i, z''_\alpha)$  are called negative pairs and their similarity is decreased, i.e.  $z'_i$  and  $z'_\alpha$  (or  $z''_\alpha$ ) pushed apart.

Usually, the embedding space is constrained to a high-dimensional hypersphere  $Z = \mathbb{S}^{d-1} \subset \mathbb{R}^d$  and the similarity function for  $z, \tilde{z} \in Z$  is based on the cosine similarity:

$$q(z, \tilde{z}) = \exp((z^\top \tilde{z}) / (\|z\| \|\tilde{z}\| \tau)), \quad (2)$$

where the temperature  $\tau$  is a hyperparameter. However, we need 2D representations for visualization. Setting  $d = 2$  would result in embedding everything into a circle, which is unsuitable for this purpose.

$t$ -SimCNE [6] learns a 2D visualization of image datasets with a parametric encoder and data augmentations as similarity source. The key ingredient is the  $t$ -SNE-inspired Cauchy kernel

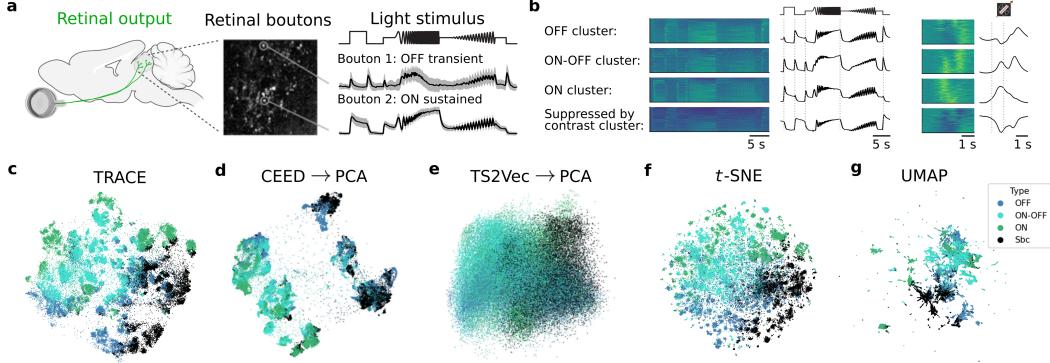
$$q(z, \tilde{z}) = (1 + \|z - \tilde{z}\|^2)^{-1} \quad (3)$$

in the 2D embedding space  $Z = \mathbb{R}^2$ . We adapt  $t$ -SimCNE to multi-trial time-series data from neuroscience by implementing positive pairs based on trials, replacing the original image-based transformations (see below). We adopt a simpler (one-stage) training procedure compared to  $t$ -SimCNE, directly learning the 2D embedding.

### 3.2 TRACE uses subset means as positive pairs

In many neuroscience experiments, the activity of a set of neurons is recorded repeatedly under identical stimulation conditions, e.g. presenting a visual stimulus multiple times (Fig. 1, left). We denote the time series of the  $l$ -th trial for neuron  $i$  as  $x_i^l[t]$ , with  $t \in \{1, \dots, T\}$  indexing discrete time. For all data points we had the same number of time steps, so we omit the time index  $t$  for clarity.

We reasoned that the variation between responses in different trials provides an estimate for the naturally occurring variability at each time point, e.g. due to fluctuating brain state or inaccuracies in the measurement. This is precisely the type of noise to which the embedding should be invariant.



**Figure 3: Neural dataset from retinal ganglion cells measured in superior colliculus and its embedding across different methods.** (a) Schematic of recording neural activity. Each neuron projecting to superior colliculus (left) has many retinal boutons (middle) from which we record activity in response to a full-field light stimulus using two-photon calcium imaging (right). (b) We functionally clustered the light-evoked neural activity into four major groups (OFF, ON-OFF, ON, Suppressed-by-contrast, Sbc) showing their mean response to the “chirp” (left) and the “moving bar” stimulus (right). (c) TRACE embedding of the neuronal time series colored by their functional group, (d, e) Same as (c), but for CEED / TS2Vec visualized in 2D using PCA, (f) t-SNE, and (g) UMAP.

Therefore, we use averages of two random, non-overlapping subsets of the trials as positive pairs, making the representation invariant to the trial-to-trial fluctuations. Formally, let  $r$  be the total number of trials, which are randomly split into two equisized, non-overlapping subsets  $S_1$  and  $S_2$  with

$$S_1 \cap S_2 = \emptyset \quad \text{and} \quad |S_1| = |S_2| = k \leq r/2. \quad (4)$$

Then the positive pair of neuron  $i$  consists of the means of these non-overlapping trial subsets:

$$x'_i = \frac{1}{k} \sum_{l \in S_1} x_i^l \quad \text{and} \quad x''_i = \frac{1}{k} \sum_{l \in S_2} x_i^l. \quad (5)$$

The subsets  $S_1, S_2$  are dynamically resampled per neuron in each epoch. A training batch is made up of the positive pairs of that batch’s neurons. This approach is inspired by mix-up augmentations [48], but generalized to averages of more than two samples and restricted to mixing only trial responses of the same neuron. The number  $k$  of trials to average over is a hyperparameter, but in practice we use  $k = \lfloor r/2 \rfloor$  as it provides a good trade-off between variability of the positive pairs, and similarity to the full mean which we embed at inference time. This choice also leads to the largest number of distinct positive pairs,  $\binom{r}{\lfloor r/2 \rfloor}$ . The subset means from other neurons in the batch were used as negative pairs.

### 3.3 Alternative contrastive learning framework for embedding time series

We compare against two other contrastive time-series models, CEED [42] and TS2Vec [46]. CEED provides the closest alternative to our framework, as it uses contrastive learning with general purpose time-series data augmentations and task-specific data augmentations, applicable to extracellular action potential wave forms [42]. To compare with CEED in the context of general neuroscience time series, we re-implemented their general purpose time-series augmentations: (1) amplitude jittering, (2) temporal jittering, and (3) correlated background noise. For more details see Appendix A.7. CEED embeds time series into five-dimensional (5D) space, which is mapped to 2D using PCA for visualization (alternative UMAP results are provided in Fig. S6, Table S3).

TS2Vec employs two types of contrastive losses: In the first, it contrasts representations of different time series (like CEED and TRACE). In the second, it contrasts representations of different time points from the same time series. These losses are applied through a hierarchical process — starting with two time segments and progressively subdividing it into smaller units — enabling both inter-series and temporal intra-series contrasting at multiple scales. TS2Vec outputs in  $\mathbb{R}^{320}$  and uses the cosine similarity in its loss function, like CEED. When applying CEED and TS2Vec to our neural data, we

use the mean neural response across trials. Both methods apply identical transformations for each recorded neuron  $i$ . In contrast, our approach of using the means across a subset of trials allows to automatically identify the local noise structure for each neuron (Sec. 3.2).

We compared our 2D TRACE embeddings with the default versions of CEED and TS2Vec using PCA (as suggested by the CEED paper) to obtain a 2D visualization of their 5D and 320D outputs, respectively. We call these CEED  $\rightarrow$  PCA and TS2Vec  $\rightarrow$  PCA (Fig. 3d,e). We also ablated TRACE’s two main contributions: generating positive pairs by averaging across a subset of trials and Cauchy similarity on 2D outputs. By modifying CEED and TS2Vec to output 2D embeddings with Cauchy similarity, we created hybrid models, TRACE + CEED and TRACE + TS2Vec (Fig. S3, App. A.6).

## 4 Experimental setup

### 4.1 Datasets

**Synthetic dataset** We simulated artificial neuronal responses of two cell types with distinct temporal structure: Each response consisted of 25 seconds of baseline activity followed by 5 seconds of class-specific signal, with either positive or negative amplitude defining the two cell types (Fig. 2a–d). To mimic realistic recording conditions, we added Gaussian noise with higher variance during the baseline activity (amplitude 10, standard deviation (SD) from 1 to 60 for single trial) and lower variance during the class-specific response period (amplitude 9 or 11, SD 8 for single trial), resembling known effects of stimulus onset in visual neurons [11]. We generated a typical number of 10 trials per neuron. While idealized, two neuron types may indeed differ in their response to only part of the stimulus, making long response periods uninformative or even detrimental for separating them.

**Large-scale neural calcium imaging dataset** We used a large-scale two-photon-imaging dataset from *in vivo* mouse retinal ganglion cell axon endings, measured in superior colliculus of awake, head-fixed mice (Fig. 3a). These neurons expressed the genetic calcium indicator GCaMP8m under the hSyn promoter. We presented two visual stimuli to the animal during recordings: (1) a full-field “chirp” consisting of a bright step and two sinusoidal intensity modulations (Fig. 1), and (2) local bright moving bars on a dark background in eight directions (Fig. 4f, right). In total, the dataset consisted of recordings of 71,021 individual retinal ganglion cell boutons measured at a sampling frequency of 8 Hz, leading to time series with 260 time bins. For the two stimuli, 15 and 10 repeated trials were recorded, respectively. We manually split responses into four groups using the responses to the moving bar stimulus (see App. A.1, Fig. 3b): ON, OFF, ON-OFF, and Suppressed-by-contrast (Sbc) and to evaluate the cluster structure of the embeddings we clustered the data using a Gaussian mixture model with 50 components. All procedures were approved by the the Baylor College of Medicine, Houston, USA, animal protocol number: AN-8132.

**Allen Institute Neuropixels spiking dataset** We used the Allen Institute Neuropixels “visual coding” dataset [36], which is part of the Allen Brain Observatory to test performance on a Neuropixels spiking dataset with action potential resolution. The dataset recorded activity of single neurons across visual cortical and thalamic structures in awake, head-fixed mice viewing diverse visual stimuli using Neuropixels silicon probes. After quality filtering and excluding non-visual neurons, we analyzed spiking responses of 10,322 neurons to light flashes and drifting gratings across visual brain areas (for details see App. A.5). To compute the *ARI score* and *kNN accuracy*, we used the labels of brain area. Biologically meaningful metrics provided by the Allen Institute were: orientation selective index (OSI), preferred orientation (PO), grating modulation ratio (F1/F0), natural image selectivity (NIS; based on responses to natural images not used to create the embeddings), behavioral modulation (correlation of firing rate with running speed of the mouse; not used to create the embeddings). Alignment of the embedding with biological metrics was measured with *kNN regression* ( $R^2$ ). For PO and behavior, we computed the radial correlation, as it better captured their global structure in the embedding.

### 4.2 Quantitative measures of embedding quality

We evaluated the embedding quality using different metrics:

1. The *ARI score* [20] between response groups identified in the 2D embedding using Mixture of Gaussian clustering and either the group labels of the calcium imaging dataset or brain region of

Table 1: **Quantitative model performance the large-scale superior colliculus dataset.** The columns are: model type (see text), the ARI score,  $k$ NN accuracy, Spearman correlation ( $r_S$ ), the maximum correlation for ON-OFF-index ( $r_{OOi}$ ), response transience index ( $r_{RTi}$ ), the recording depth  $r_{Depth}$ , and the average rank  $\mu_{Rank}$  of each method. For a definition of the measures, see Sec. 4.2. All metrics other than the rank are better when higher. Uncertainties for the ARI score were insignificant and we omitted them. The best values for each metric are **bold**.

Model	ARI	$k$ NN acc.	$r_S$	$r_{OOi}$	$r_{RTi}$	$r_{Depth}$	$\mu_{Rank}$
TRACE	<b>0.28</b>	$69.7 \pm 0.3\%$	$0.45 \pm 0.01$	$0.67 \pm 0.02$	<b><math>0.26 \pm 0.00</math></b>	$0.54 \pm 0.02$	<b>2.17</b>
+ CEED	0.25	$64.0 \pm 0.4\%$	$0.51 \pm 0.02$	$0.64 \pm 0.01$	<b><math>0.26 \pm 0.00</math></b>	$0.48 \pm 0.04$	2.67
+ TS2Vec	0.10	$23.8 \pm 1.1\%$	<b><math>0.60 \pm 0.01</math></b>	$0.28 \pm 0.04$	$0.23 \pm 0.06$	$0.51 \pm 0.00$	4.33
CEED → PCA	0.20	$60.7 \pm 0.6\%$	$0.45 \pm 0.05$	$0.54 \pm 0.10$	$0.17 \pm 0.00$	<b><math>0.55 \pm 0.07</math></b>	4.00
TS2Vec → PCA	0.04	$11.6 \pm 0.9\%$	$0.30 \pm 0.01$	$0.12 \pm 0.01$	$0.09 \pm 0.01$	$0.24 \pm 0.01$	6.83
$t$ -SNE	0.24	<b><math>70.7 \pm 0.2\%</math></b>	$0.50 \pm 0.01$	<b><math>0.69 \pm 0.01</math></b>	$0.17 \pm 0.02$	$0.42 \pm 0.03$	3.00
UMAP	0.23	$69.9 \pm 0.2\%$	$0.27 \pm 0.02$	$0.38 \pm 0.05$	<b><math>0.26 \pm 0.02</math></b>	$0.42 \pm 0.00$	4.00

the Neuropixels dataset, respectively. This quantifies how well the 2D embedding captured major response groups.

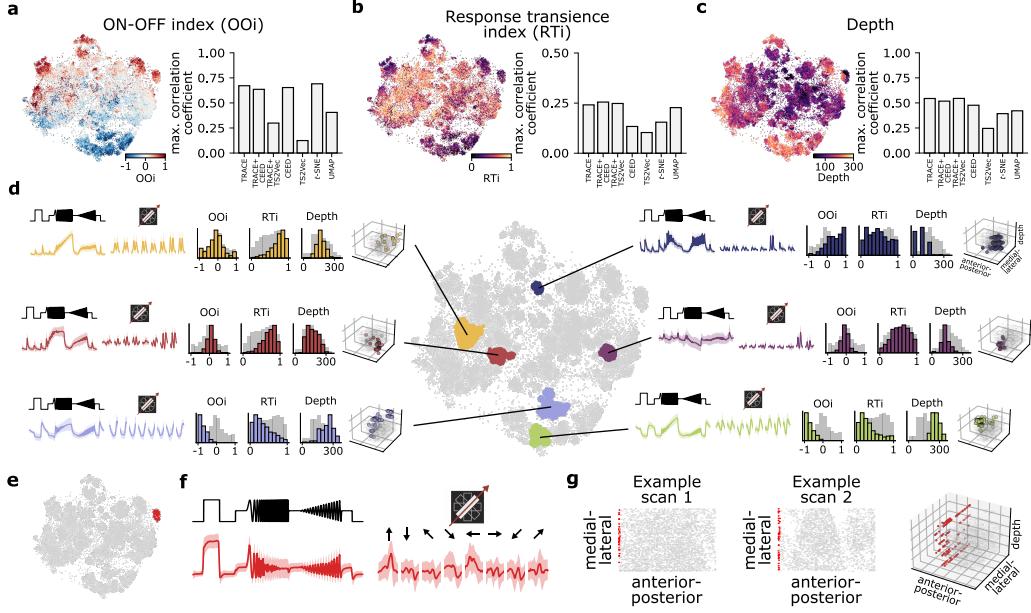
2. The  $k$ NN accuracy as a standard metric that quantifies embedding quality by predicting the response-type labels through a majority vote of each point’s 15 nearest neighbors in the 2D embedding space. This metric indicates how well points from the same response group cluster together and is commonly used in contrastive learning [30].
3. Spearman’s rank correlation  $r_S$  [37] between pairwise distances in the original time-series space and the low-dimensional embedding,  $r_S = \text{corr}(\|x_i - x_j\|, \|z_i - z_j\|)$  to quantify how well the embedding preserved the relative distances between data points. This is a common metric for visualization methods [22], but rests on the assumption that distances between the original data are meaningful, which may not be the case due to the curse of dimensionality ([1]).
4. The linear or radial correlation  $r$  between the 2D embedding and biological relevant variables such as the ON-OFF index ( $OOi \in [-1, 1]$ ), the response-transience index ( $RTi \in [0, 1]$ ) (see App. A.2, A.3), and the recording depth for the calcium imaging dataset (Fig. S4) and preferred orientation (PO) and behavioral modulation for the Neuropixels dataset. For linear gradients, we first fit a linear regression between the embedding coordinates ( $x, y$ ) and each variable to determine the rotation angle  $\theta$  that maximized correlation along one axis. The radial distance of each point was determined based on its Euclidean distance from the arithmetic mean of the embedding. Absolute Pearson correlations were computed between biological variables and both the  $\theta$ -direction (linear) and radial distances and the maximum reported in Table 1.
5. The  $k$ NN regression to measure how well the respective biological variable of a neuron is predicted by averaging the values of its  $k = 15$  nearest neighbors in the embedding (used for the Neuropixels dataset as it sometimes better captured the global structure in the embedding).
6. The average rank of the above metrics to provide an aggregated score.

## 5 Results

The goal of TRACE is to support exploratory data analysis of multi-trial time-series data in neuroscience to identify functional cell types or groups and to study continuous variation of neuronal function between cells. Here, we show that TRACE performs better than its competitors both on a synthetic dataset (Fig. 2) and on two large-scale *in vivo* neuroscience time-series datasets (Figs. 3,S8).

### 5.1 TRACE identifies informative regions in a synthetic dataset

In the synthetic dataset, responses of two artificial neuron types were constructed such that the first 25 seconds corresponded to baseline activity with high noise, while the final 5 seconds distinguished two neuron types with high signal-to-noise-ratio (Fig. 2a–d, Sec. 4.1). We compared TRACE against CEED, TS2Vec,  $t$ -SNE, and UMAP and found that TRACE successfully separated the two neuronal types, revealing differences in responses that the other methods failed to distinguish (Fig. 2e–i).



**Figure 4: The TRACE embedding reflects variables of biological interest and aids in identifying experimental artifacts.** (a) *Left*: TRACE embedding color-coded by the ON-OFF index (OOi). *Right*: Maximum correlation with OOI across all methods. (b) Same as (a), but for the response transience index (RTi), and (c) the recording depth in the superior colliculus. (d) TRACE embedding with example islands extracted using HDBSCAN clustering ( $\epsilon = 0$ , minimum numbers of samples per cluster 50, minimum cluster size 500). Each example shows (from *left* to *right*) the mean response to the chirp and the moving bar stimulus, the OOI, the RTi, the recorded depth and the anatomical location for the respective cluster. (e) TRACE embedding with artifact island (red). (f) Responses of the artifact island to the chirp (*left*) and the moving bars (*right*). (g) Two example scan fields with neuronal responses identified as artifacts in red and others in gray, as well as anatomical location across the entire recording volume.

TRACE even performed well when using a smaller number of trials  $k$  used for the non-overlapping subsets (Fig. S2a). In addition, we varied the noise levels by increasing the baseline standard deviation. TRACE outperformed all other methods and was able to successfully separate the two classes even at high noise levels (Fig. 2j, S1, Table S1). The other methods performed well at low noise but their performance rapidly deteriorated as noise increased.

To successfully separate the two classes, the methods needed to learn to ignore the first period of spontaneous activity. *t*-SNE and UMAP struggled because they compute distances between responses and these are dominated by uninformative distances induced by the spontaneous activity (Fig. 2c). In CEED, only the correlated background noise can have varying effects across different time bins. However, as the variance of the noise is estimated on the whole dataset without knowledge of the true label, the variance is inflated in the signal part of the time series because of inter-class variability. As a result, CEED’s covariance augmentation might transform a sample with positive response amplitude into one with negative response amplitude, destroying the class structure. In contrast, the sample specific subset mean augmentation used by TRACE does not have this problem: The subset means for a sample with positive response amplitude will retain this positive signal response, and only differ strongly on the low signal-to-noise part of the recording. Thus, TRACE specifically learned to ignore this part of the time series and only focuses on the part that encodes the different classes, explaining why it can produce a visualization with clearly separated classes (Fig. 2e, S1a).

## 5.2 TRACE represents biological structure better in superior colliculus two-photon dataset

Next, we applied TRACE to two-photon calcium recordings of retinal ganglion cell boutons measured in the mouse superior colliculus responding to light stimuli (Fig. 3a, Sec. 4.1 for details). Based on the recorded time series, we identified four major neural response groups based on the responses to the

moving bar stimulus: ON, OFF, ON-OFF, and Suppressed-by-contrast (Sbc) neurons (Fig. 3b). We compared TRACE against CEED, TS2Vec, *t*-SNE, and UMAP (Fig. 3c–g, Table 1) and additionally extended the TRACE setup to use either CEED-like augmentations or the TS2Vec approach (see Sec. 3.3, Fig. S3). We found that the embedding learned by TRACE and TRACE-variants visually exhibited the best balance between resolving cluster structure and retaining large-scale structure (Fig. S3). This was confirmed quantitatively: For example, TRACE most accurately reflected the manually identified response groups (ON, OFF, ON-OFF, and Suppressed-by-contrast), as indicated by the best ARI score (Table 1). In addition, TRACE showed comparable *k*NN accuracy to *t*-SNE, in contrast to CEED or TS2Vec, indicating that nearest neighbour in the embedding typically came from the same neuronal response group. Finally, TRACE and TRACE-variants showed the best correlation between time-series distances and embedding distances, showing that the embedding overall respected the structure of the high-dimensional space well. Computationally, TRACE was significantly faster than TS2Vec (Table S5) and more efficient than CEED. While CEED needs expensive data augmentations for each observation in the batch we pre-computed 10k noise samples to improve efficiency (not counted in the reported training time). This is not necessary for TRACE as the mean of a subset of trials is used for the positive pairs. Testing how reducing the number of trials ( $k$ ) in non-overlapping subsets affects the results we found that TRACE still achieved good performance even with  $k = 1$  (Table S2, Fig. S2b).

We next studied to what extent the different embeddings captured other neuronal properties such as the tendency to respond to light increments or decrements (measured by the ON-OFF index (OOi), Fig. 4a) or the kinetics of the response to a light step (measured by the response transience index (RTi), Fig. 4b) [5]. We found that TRACE and TRACE-variants captured these properties well or better than competing methods, yielding the highest correlation with RTi and comparable correlation values for OOi and depth (Table 1, Fig. 4a–c, Fig. S4). Notably, the depth is a completely independent measure never used during learning (whereas OOi and RTi were derived from the shape of the time-series activity used for training). When ranking methods by their mean rank ( $\mu_{\text{Rank}}$ ) across all evaluation metrics, TRACE came first among the evaluated models showing that it consistently performs well across all metrics, in line with the visual impression (Table 1, Fig. 3, Fig. S3). TRACE also performed competitively in higher embedding dimension (Table S4).

Next, we clustered the 2D TRACE representation using HDBSCAN [27] to explore the structure of the visualization in more detail (Fig. 4d). We found that distinct clusters in the embedding showed unique neural response characteristics, representing types within the manually defined response groups of ON, OFF, ON-OFF, and Sbc (examples shown in Fig. 4d). Interestingly, many of these identified subgroups clustered in specific, spatially distinct regions within the superior colliculus, suggesting that the TRACE embedding learned the known relationship between neural responses and anatomical location (without having access to the location during training) [25].

### 5.3 TRACE finds recording artifacts and outliers

In the TRACE embedding, we found an isolated island on the far right, which showed peculiar light responses (Fig. 4e), closely following the light intensity of the stimulus (Fig. 4f). We found that these responses were exclusively recorded on the far left of a scan field (Fig. 4g), suggesting that they corresponded to light artifacts. In the representations for TRACE + CEED and CEED this island was also visible but less clearly separated. Interestingly, the only other embedding clearly showing these artifacts as outliers was that of UMAP, while they did not stand out at all for TS2Vec (Fig. S5).

Finally, there were a few outlier points located in the white space between clusters, which did not seem to belong to any of the clusters (Fig. S7). To detect outliers we identified outliers as points that are both locally isolated (few neighbors) and globally sparse (low density) (App. A.4). We investigated these neural responses and found some obvious examples of outlier responses that show noisy responses for either of the two stimuli, suggesting TRACE can be used for data cleaning.

### 5.4 TRACE shows superior performance on Neuropixels spiking dataset

Next, we tested the generalizability of our approach across domains and applied it to the Allen Institute Neuropixels dataset [12, 36] that included diverse responses of visual neurons from different brain areas to visual stimuli such as dark and bright flashes and drifting gratings (Fig. S8a).

We evaluated embeddings using distance correlation  $r_S$  and clustering metrics (ARI score,  $k$ NN accuracy) with brain region as ground truth labels. While clear regional separation is not necessarily expected because hierarchical visual processing naturally creates overlapping response patterns across areas, the brain region provided the only available categorical ground truth for this dataset (Fig. S8b,c). We therefore additionally evaluated embeddings using biologically meaningful continuous variables (OSI, PO, F1/F0; see methods and Table S6). To test generalization beyond the stimulus features used to create embeddings, we also used the natural image selectivity (NIS) and behavioral modulation (correlation with running speed) to evaluate the embedding structure, neither of which were derived from the flash and grating responses used for creating the embeddings (Fig. S9).

While standard methods (*t*-SNE, UMAP) found some structure especially in terms of distance correlation and for OSI, TRACE outperformed them in most metrics and outperformed CEED in all metrics (Table S6; Fig. S8d–h). The performance of TS2Vec variants was mixed, with good preservation of some structure (OSI, NIS) but very poor global distance correlation (Fig. S9). Overall, TRACE achieved the best rank by far, indicating best overall performance on this dataset (Table S6). Notably, while TRACE was the only methods capturing both image selectivity and modulation by behavior well (despite these features not being used for training), only TRACE was able to capture this and produce clearly structured, interpretable embeddings (Fig. S9).

## 6 Conclusion, limitations, and future work

We presented TRACE, a new framework that combines contrastive learning with neighbor embeddings to directly generate interpretable 2D visualizations of large-scale neural time-series data. Using the inherent structure of multi-trial recordings common in neuroscience experiments, TRACE is able to separate subtle differences of simulated neural response types that competing methods fail to distinguish. When applied to a diverse neural dataset of two-photon recordings, TRACE captured both continuous variations in neural properties and discrete cell-type structures, and identified clusters with fine differences in functional responses and highlighted recording artifacts in the learned 2D representation (Sec. 5.3). TRACE proved especially valuable for Neuropixels spike train data, because its approach of trial averaging preserves the underlying Poisson statistics better than standard augmentations used by other contrastive methods (e.g. temporal jitter or additive Gaussian noise).

Conceptually, TRACE has two advantages over existing methods. First, methods like *t*-SNE and UMAP based on neighborhood relationships in the high-dimensional space are sensitive to uninformative, noisy baseline activity and may not be able to detect short class-informative response periods (Sec. 5.1). Contrastive frameworks such as CEED rely on general-purpose data transformations and may mask subtle response differences between cell types. Second, TRACE directly produces a 2D visualization while other contrastive methods require a separate reduction step. As a result, the embeddings produced by TRACE were the most biologically interpretable as revealed by their ability to reflect key response characteristics (Sec. 5.2).

A limitation of our work is the need to train an embedding model, increasing computational time compared to *t*-SNE or UMAP. However, this comes with the benefit of improved embedding quality and the ability to map new data points. Another limitation of our work is the need for repeated trials. However, this data collection approach data is standard for many neuronal time-series datasets. In some cases, between-trial variation may be of interest, e.g., when investigating adaptation over multiple trials. TRACE operates under the assumption that differences between trials are undesirable noise thus making it not the right tool for this type of question.

In future work, it will be interesting to apply TRACE to a much wider range of time-series data because many non-neuroscience datasets also have inherent multi-trial structure or can be reshaped into this format. For example, in sport analytics inertial measurements are often taken during repeated exercises, such as basketball free-throw drills [16, 19]. The the Google Speech Command Dataset [43] contains 5 audio recordings per speech commands and speaker. For medical data such as electrocardiography one could use different daily cycles or stress tests as trials. For financial market data, trading periods can be used as trials to detect unusual patterns or market anomalies. Another application domain could be climate data where years, seasons, or tidal cycles could be used as trials. These avenues highlight the broad applicability of leveraging multi-trial structure with TRACE.

## Acknowledgments

The authors thank the Hertie Foundation, the German Research Foundation (CRC 1233 “Robust Vision” and DFG/ANR EU 42/12-1, Project number 505379160) and the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for financial support. FS acknowledges the support of the Lower Saxony Ministry of Science and Culture (MWK) with funds from the Volkswagen Foundation’s zukunft.niedersachsen program (project name: CAIMed-Lower Saxony Center for Artificial Intelligence and Causal Methods in Medicine; grant number: ZN4257. KF was supported by the European Research Council (Starting Grant “Eye To Action” 101117156). PB and DK are members of the EXC 2064 “Machine Learning — New Perspectives for Science“.

## References

- [1] Aggarwal, C. C., Hinneburg, A., and Keim, D. A. On the surprising behavior of distance metrics in high dimensional space. In *Database theory—ICDT 2001: 8th international conference London, UK, January 4–6, 2001 proceedings* 8, pp. 420–434. Springer, 2001.
- [2] Antoniades, A., Yu, Y., Canzano, J. S., Wang, W. Y., and Smith, S. Neuroformer: Multimodal and multitask generative pretraining for brain data. In *The Twelfth International Conference on Learning Representations*, 2024.
- [3] Azabou, M., Azar, M. G., Liu, R., Lin, C.-H., Johnson, E. C., Bhaskaran-Nair, K., Dabagia, M., Avila-Pires, B., Kitchell, L., Hengen, K. B., et al. Mine your own view: Self-supervised learning through across-sample prediction. *arXiv preprint arXiv:2102.10106*, 2021.
- [4] Bachman, P., Hjelm, R. D., and Buchwalter, W. Learning representations by maximizing mutual information across views. *Advances in Neural Information Processing Systems*, 32, 2019.
- [5] Baden, T., Berens, P., Franke, K., Román Rosón, M., Bethge, M., and Euler, T. The functional diversity of retinal ganglion cells in the mouse. *Nature*, 529(7586):345–350, 2016.
- [6] Böhm, N., Berens, P., and Kobak, D. Unsupervised visualization of image datasets using contrastive learning. In *The Eleventh International Conference on Learning Representations*, 2023.
- [7] Chen, R., Canales, A., and Anikeeva, P. Neural recording and modulation technologies. *Nature Reviews Materials*, 2(2):1–16, 2017.
- [8] Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pp. 1597–1607. PMLR, 2020.
- [9] Cheng, J. Y., Goh, H., Dogrusoz, K., Tuzel, O., and Azemi, E. Subject-aware contrastive learning for biosignals. *arXiv preprint arXiv:2007.04871*, 2020.
- [10] Cho, C. J., Chang, E., and Anumanchipalli, G. Neural latent aligner: Cross-trial alignment for learning representations of complex, naturalistic neural data. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 5661–5676. PMLR, 23–29 Jul 2023.
- [11] Churchland, M. M., Yu, B. M., Cunningham, J. P., Sugrue, L. P., Cohen, M. R., Corrado, G. S., Newsome, W. T., Clark, A. M., Hosseini, P., Scott, B. B., et al. Stimulus onset quenches neural variability: a widespread cortical phenomenon. *Nature Neuroscience*, 13(3):369–378, 2010.
- [12] de Vries, S. E., Lecoq, J. A., Buice, M. A., Groblewski, P. A., Ocker, G. K., Oliver, M., Feng, D., Cain, N., Ledochowitsch, P., Millman, D., et al. A large-scale standardized physiological survey reveals functional organization of the mouse visual cortex. *Nature Neuroscience*, 23(1):138–151, 2020.
- [13] Diamant, N., Reinertsen, E., Song, S., Aguirre, A. D., Stultz, C. M., and Batra, P. Patient contrastive learning: A performant, expressive, and practical approach to electrocardiogram modeling. *PLoS Computational Biology*, 18(2):e1009862, 2022.

[14] Egan, J. P. *Psychophysics and signal detection*. Indiana University, Hearing and Communication Laboratory, 1962.

[15] Eldele, E., Ragab, M., Chen, Z., Wu, M., Kwoh, C. K., Li, X., and Guan, C. Time-series representation learning via temporal and contextual contrasting. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, 2021.

[16] García-de Villa, S., Jiménez-Martín, A., and García-Domínguez, J. J. A database of physical therapy exercises with variability of execution collected by wearable sensors. *Scientific Data*, 9(1):266, 2022.

[17] Gutmann, M. U. and Hyvärinen, A. Noise-Contrastive Estimation of Unnormalized Statistical Models, with Applications to Natural Image Statistics. *Journal of Machine Learning Research*, 13(2), 2012.

[18] Hinneburg, A., Aggarwal, C. C., and Keim, D. A. What is the nearest neighbor in high dimensional spaces? In *International Conference on Very Large Data Bases*, 2000.

[19] Hoelzemann, A., Romero, J. L., Bock, M., Laerhoven, K. V., and Lv, Q. Hang-time har: A benchmark dataset for basketball activity recognition using wrist-worn inertial sensors. *Sensors*, 23(13):5879, 2023.

[20] Hubert, L. and Arabie, P. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

[21] Kiyasseh, D., Zhu, T., and Clifton, D. A. Clocs: Contrastive learning of cardiac signals across space, time, and patients. In *International Conference on Machine Learning*, pp. 5606–5615. PMLR, 2021.

[22] Kobak, D. and Berens, P. The art of using t-sne for single-cell transcriptomics. *Nature Communications*, 10(1):5416, 2019.

[23] Lee, E. K., Balasubramanian, H., Tsolias, A., Anakwe, S. U., Medalla, M., Shenoy, K. V., and Chandrasekaran, C. Non-linear dimensionality reduction on extracellular waveforms reveals cell type diversity in premotor cortex. *eLife*, 10:e67490, 2021.

[24] Lee, E. K., Gül, A. E., Heller, G., Lakunina, A., Jaramillo, S., Przytycki, P. F., and Chandrasekaran, C. Physmap-interpretable in vivo neuronal cell type identification using multi-modal analysis of electrophysiological data. *bioRxiv preprint bioRxiv:2024.02.28.582461*, 2024.

[25] Li, Y.-t. and Meister, M. Functional cell types in the mouse superior colliculus. *eLife*, 12:e82367, 2023.

[26] Liu, R., Azabou, M., Dabagia, M., Lin, C.-H., Gheshlaghi Azar, M., Hengen, K., Valko, M., and Dyer, E. Drop, swap, and generate: A self-supervised approach for generating neural activity. *Advances in Neural Information Processing Systems*, 34:10587–10599, 2021.

[27] McInnes, L., Healy, J., Astels, S., et al. HDBSCAN: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205, 2017.

[28] McInnes, L., Healy, J., and Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[29] Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[30] Oquab, M., Darct, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[31] Pandarinath, C., O’Shea, D. J., Collins, J., Jozefowicz, R., Stavisky, S. D., Kao, J. C., Trautmann, E. M., Kaufman, M. T., Ryu, S. I., Hochberg, L. R., et al. Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature Methods*, 15(10):805–815, 2018.

[32] Pearson, K. Liii. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

- [33] Peterson, S. M., Rao, R. P. N., and Brunton, B. W. Learning neural decoders without labels using multiple data streams. *Journal of Neural Engineering*, 19(4), aug 2022. doi: 10.1088/1741-2552/ac857c.
- [34] Sarkar, P. and Etemad, A. Self-supervised ecg representation learning for emotion recognition. *IEEE Transactions on Affective Computing*, 13(3):1541–1554, 2020.
- [35] Schneider, S., Lee, J. H., and Mathis, M. W. Learnable latent embeddings for joint behavioural and neural analysis. *Nature*, 617(7960):360–368, 2023.
- [36] Siegle, J. H., Jia, X., Durand, S., Gale, S., Bennett, C., Graddis, N., Heller, G., Ramirez, T. K., Choi, H., Luviano, J. A., et al. Survey of spiking in the mouse visual system reveals functional hierarchy. *Nature*, 592(7852):86–92, 2021.
- [37] Spearman, C. The proof and measurement of association between two things. *The American Journal of Psychology*, 1904.
- [38] Stringer, C. and Pachitariu, M. Analysis methods for large-scale neuronal recordings. *Science*, 386(6722):eadp7429, 2024.
- [39] Tonekaboni, S., Eytan, D., and Goldenberg, A. Unsupervised representation learning for time series with temporal neighborhood coding. In *International Conference on Learning Representations*, 2021.
- [40] Urzay, C., Ahad, N., Azabou, M., Schneider, A., Atamkuri, G., Hengen, K., and Dyer, E. Detecting change points in neural population activity with contrastive metric learning. In *International IEEE/EMBS Conference on Neural Engineering*, 2023.
- [41] van der Maaten, L. and Hinton, G. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11), 2008.
- [42] Vishnubhotla, A., Loh, C., Srivastava, A., Paninski, L., and Hurwitz, C. Towards robust and generalizable representations of extracellular data using contrastive learning. *Advances in Neural Information Processing Systems*, 37, 2023.
- [43] Warden, P. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209*, 2018.
- [44] Wickstrøm, K., Kampffmeyer, M., Mikalsen, K. Ø., and Jenssen, R. Mixing up contrastive learning: Self-supervised representation learning for time series. *Pattern Recognition Letters*, 155:54–61, 2022.
- [45] Yu, H., Lyu, H., Xu, Y., Windolf, C., Lee, E. K., Yang, F., Shelton, A. M., Winter, O., Dyer, E. L., Chandrasekaran, C., et al. In vivo cell-type and brain region classification via multimodal contrastive learning. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [46] Yue, Z., Wang, Y., Duan, J., Yang, T., Huang, C., Tong, Y., and Xu, B. TS2Vec: Towards universal representation of time series. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 8980–8987, 2022.
- [47] Zeng, H. and Sanes, J. R. Neuronal cell-type classification: challenges, opportunities and the path forward. *Nature Reviews Neuroscience*, 18(9):530–546, 2017.
- [48] Zhang, H., Cisse, M., Dauphin, Y., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, pp. 1–13, 2018.
- [49] Zhang, X., Zhao, Z., Tsiligkaridis, T., and Zitnik, M. Self-supervised contrastive pre-training for time series via time-frequency consistency. *Advances in Neural Information Processing Systems*, 35:3988–4003, 2022.

## A Experimental details

### A.1 Identifying neural response groups for the calcium imaging dataset

A typical coarse classification of retinal ganglion cells is by their response to On- and Off-sets in the light stimulus and can typically be obtained with high confidence from simple statistics of the mean trial response of the moving bar stimulus [5]. In particular, suppressed-by-contrast responses were identified as the ones with negative area under the curve, calculated from their mean response across all directions of the moving bar. ON-OFF responses were identified using the characteristic double-peak (2nd component in PCA captured this double-peak). Finally, ON and OFF responses were classified based on their response direction (captured in 1st component of PCA).

### A.2 ON-OFF index

The ON-OFF index ( $OO_i \in [-1, 1]$ ) was calculated to quantify the response polarity of the recorded visual cells to the “chirp” light stimulus. Increased activity during light increments would indicate a response characteristic as ‘ON’ polarity ( $OO_i \sim 1$ ), increased activity during light decrements as ‘OFF’ polarity ( $OO_i \sim -1$ ), and a response to increments and decrements as ‘ON-OFF’ polarity ( $OO_i \sim 0$ ).

For a mean response  $x$ , the ON-OFF index ( $OO_i$ ) was computed as

$$OO_i = \frac{r_{ON} - r_{OFF}}{r_{ON} + r_{OFF}}, \quad (6)$$

where  $r_{OFF}$  and  $r_{ON}$  are defined as the integrated response over the short time interval ( $\Delta = 1$  s) after the start ( $t_{ON} = 2.5$  s) and the end ( $t_{OFF} = 5.5$  s) of the initial light stimulus during the “chirp”:

$$r_{ON} = \sum_{t=t_{ON}}^{t_{ON}+\Delta} x[t] \quad \text{and} \quad r_{OFF} = \sum_{t=t_{OFF}}^{t_{OFF}+\Delta} x[t] \quad (7)$$

Then, metric was clipped within the range  $[-1, 1]$  to obtain a bounded metric.

### A.3 Response transience index

The response transience index ( $RT_i \in [0, 1]$ ) quantifies the response kinetic of a visual cell during the first step response of the “chirp” stimulus. Cells with a sustained response characteristics have a  $RT_i = 0$ , whereas transient cells with a response decay back to baseline have  $RT_i = 1$ . The  $RT_i$  was computed as

$$RT_i = 1 - \frac{x[\text{peak} + a]}{x[\text{peak}]}, \quad (8)$$

where *peak* defined the time point of peak response and  $a = 400$  ms the response following the peak.

For the  $RT_i$ , we tested both direct and inverse  $(1 + RT_i)^{-1}$  relationships and report the maximum of both.

### A.4 Outlier detection

First, the number of neighbors for each point  $\mathbf{z}_i$  was calculated within a fixed radius  $d = 4$ . Neighbors were defined as points  $\mathbf{z}_j$  satisfying  $\|\mathbf{z}_i - \mathbf{z}_j\| \leq d$ , excluding  $\mathbf{z}_i$  itself. Points with more than one neighbor ( $N_i > 1$ ) were excluded, as they were considered part of dense regions. Next, KDE was used to estimate the density of the remaining points. Using a Gaussian kernel with bandwidth  $h = 0.5$ , the density at each point  $\mathbf{z}_i$  was calculated as

$$f(\mathbf{z}_i) = \frac{1}{nh^2} \sum_{j=1}^n \exp \left( -\frac{\|\mathbf{z}_i - \mathbf{z}_j\|^2}{2h^2} \right), \quad (9)$$

where  $n$  is the number of filtered points. Points with log-density values  $\log f(\mathbf{z}_i) < -6.0$  were classified as outliers.

## A.5 Allen Institute Neuropixels spiking dataset post-processing

Out of the stimulus battery available in the Neuropixels dataset, we used responses to two stimuli: light flashes (250 ms duration,  $n_{\text{trials}} = 75$ ) and drifting gratings (200 ms duration,  $n_{\text{trials}} = 14$ , 8 orientations, 5 frequencies), binning the spike trains at a temporal resolution of 25 ms. We grouped neurons by brain area using anatomical annotations from the Allen Institute. Brain areas were the following: posterior accessory optic nucleus (APN), dorsal lateral geniculate nucleus (LGD), ventral lateral geniculate nucleus (LGv), lateral posterior nucleus (LP), unspecified visual cortex (VIS), anterolateral visual cortex (VISal), anteromedial visual cortex (VISam), lateral visual cortex (VISl), primary visual cortex (VISp), posteromedial visual cortex (VISpm), rostralateral visual cortex (VISrl) and excluded non-visual areas.

## A.6 Implementation details

TRACE uses a lightweight multi-layer perceptron (MLP) that can run efficiently on a GPU. To ensure a fair comparison, we matched CEED’s architecture [42] and used an MLP consisting only of four layers with sizes [768, 512, 256, 128] and ReLU activations between them. We added a 1024-dimensional projection head. Due to TS2Vec’s specialized augmentations and hierarchical contrastive scheme, harmonizing its architecture was more difficult and we stuck to its original, dilated CNN architecture.

For the synthetic datasets, we ran all methods with batch size 512 for 100 epochs unless for TS2Vec→PCA, where we kept the default batch size and of number of epochs. We trained TRACE and CEED→PCA with learning rate 0.3 and TS2Vec with its default learning rate.

For the calcium imaging dataset, we used optimal hyperparameters found with a grid search with batch sizes ranging from 1024 to 3200 and learning rates from 0.1 to 0.2 for both TRACE and CEED. The best hyperparameter setting was chosen based on the final loss. For TRACE the best batch size was 1280, while for CEED it was 1024. A learning rate of 0.1 was optimal for both. For TRACE + CEED we used hyperparameters as in the unmodified TRACE version.

For the Neuropixels dataset we ran a separate grid search and identified optimal hyperparameters for for TRACE and CEED (0.03 learning rate, batch size 512) and for TRACE + CEED (0.08 learning rate, batch size 1024).

Due to the much longer run time of TS2Vec such a grid search was infeasible. For a fair comparison, we massively increased the number of epochs to 1000 for TRACE+TS2Vec and increased the batch size to 768 (larger sizes exceeded memory constraints). To stay close to TS2Vec’s original implementation for TS2Vec→PCA, we kept the short default runtime (< 1 epoch) and batch size 16. The learning rate for TS2Vec remained at its default value of 0.001. We used the same setting for both neural datasets.

Unless otherwise specified, we trained all embeddings for 1000 epochs for the calcium imaging dataset.

Computations were performed on an NVIDIA A40 GPU 48 GB.

## A.7 CEED augmentations

Here we give details on the general-purpose time-series augmentations that we used when running CEED and TRACE+CEED. We apply these augmentation to the mean of all trials,

$$x_i = \frac{1}{r} \sum_{l=1}^r x_i^l.$$

Amplitude jittering randomly scales the signal between 0.7 and 1.3 using a uniform distribution,

$$x'_i = r' \cdot x_i \text{ and } x''_i = r'' \cdot x_i \quad \text{with } r', r'' \sim \mathcal{U}(0.7, 1.3), \quad (10)$$

while temporal jittering shifts the signal by up to 3 time bins (up to 370 ms for 8 Hz sampling frequency)

$$x'_i[t] = x_i[t - s'] \text{ and } x''_i[t] = x_i[t - s''] \quad \text{for } s', s'' \sim \mathcal{U}(\{\pm 1, \pm 2, \pm 3\}). \quad (11)$$

To efficiently handle the computationally expensive correlated noise generation, we pre-computed 10,000 noise samples from the covariance matrix of the data using a multivariate normal distribution and added them to the original time series, mimicking the augmentation

$$x'_i = x_i + \varepsilon' \text{ and } x''_i = x_i + \varepsilon'' \quad \text{for } \varepsilon', \varepsilon'' \sim \mathcal{N}(\mathbf{0}, \text{Cov}(\{x_1, \dots, x_n\})). \quad (12)$$

We applied each transformation independently, with probabilities of 0.7 for amplitude jittering, 0.6 for temporal jittering, and 0.5 for correlated noise injection. Hyperparameters of the transformations were adapted to our datasets.

## A.8 Supplementary tables and figures

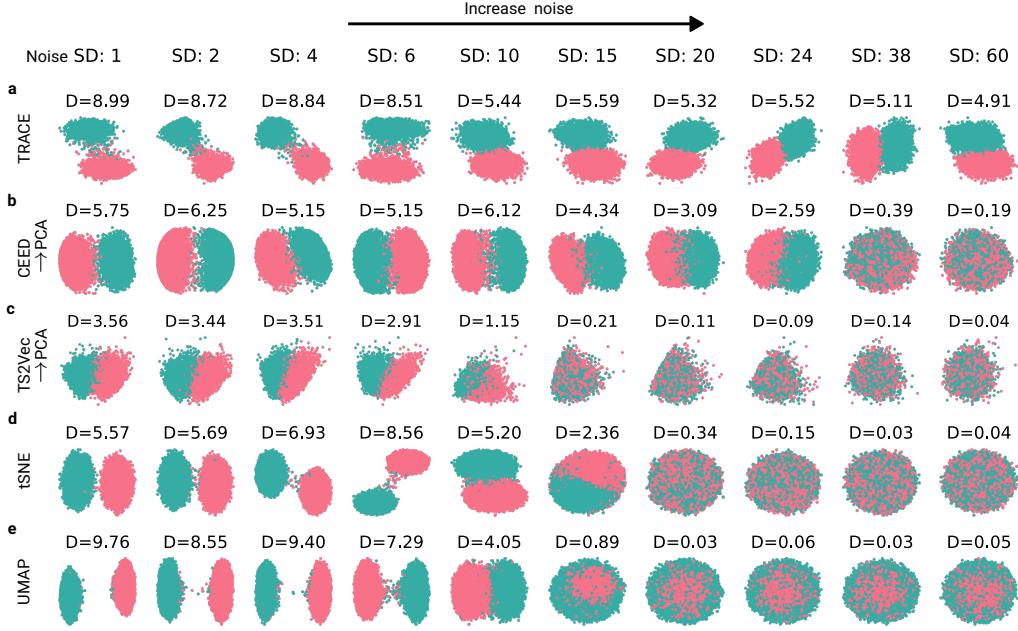


Figure S1: **TRACE is able to separate classes in the synthetic dataset even for high noise levels and outperforms other methods.** (a) TRACE embedding for increasing amounts of noise during the neural baseline activity. Discriminability (D) values indicated at the *top* of each embedding. Standard deviation (SD) of noise is indicated at the *top* of the figure. (b) Same as (a), but for CEED → PCA, (c) TS2Vec → PCA, (d) *t*-SNE, and (e) UMAP.

Table S1: Mean discriminability for simulated responses across three seeds for different noise levels (standard deviation of baseline response). Best performing method in **bold**.

Noise SD	1	2	4	6	10	15	20	24	38	60
TRACE	9.01	<b>8.75</b>	8.87	6.93	<b>5.57</b>	<b>5.57</b>	<b>5.45</b>	<b>5.47</b>	<b>5.14</b>	<b>3.52</b>
CEED → PCA	6.01	6.13	5.13	4.76	5.26	3.84	2.98	1.90	0.37	0.18
TS2Vec → PCA	3.65	3.61	3.68	3.16	1.24	0.22	0.11	0.10	0.13	0.05
<i>t</i> -SNE	5.56	5.65	6.94	<b>8.40</b>	5.24	2.35	0.45	0.09	0.03	0.03
UMAP	<b>9.76</b>	8.55	<b>9.40</b>	7.29	4.05	0.89	0.03	0.06	0.03	0.05

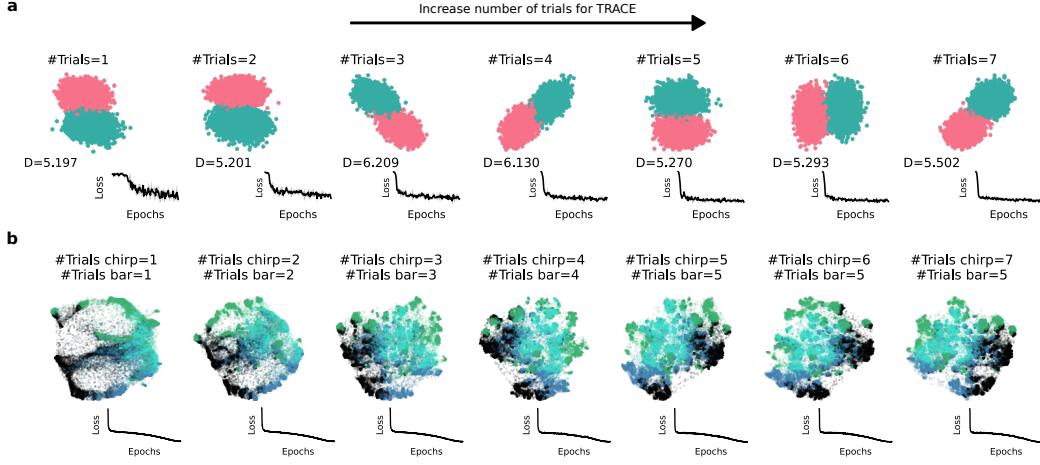


Figure S2: **TRACE embeddings with varying numbers trials  $k$  for averaging.** (a) *From left to right:* Embeddings for the synthetic dataset with increasing numbers of trials  $k$  from 1 to 7 used for the non-overlapping subsets. Discriminability ( $D$ ) between class 1 and class 2 neurons was computed as in Fig. 2. *Bottom:* Loss over epochs. (b) *From left to right:* Embeddings for the calcium imaging dataset with increasing numbers of trials per subset mean  $k$  from 1 to 7 for the "chirp" stimulus and from 1 to 5 for the moving bars stimulus. Neurons are colored by the four broad groups OFF (blue), ON-OFF (turquoise), ON (green), Suppressed-by-contrast (black). *Bottom:* Loss over epochs.

Table S2: Reducing number of trials ( $k$ ) used for the non-overlapping subsets using the calcium imaging dataset. The maximum number of trials was 7 for the "chirp" stimulus ( $k_{\text{chirp}}$ ) and 5 for the moving bar stimulus ( $k_{\text{moving bar}}$ ). Best performance in **bold**.

$k_{\text{chirp}}$	1	2	3	4	5	6	7
$k_{\text{moving bar}}$	1	2	3	4	5	5	5
ARI	0.24	0.27	0.27	0.28	<b>0.29</b>	0.28	<b>0.29</b>
$k\text{NN}$ accuracy	51.9	62.4	68.1	68.9	68.6	69.3	<b>69.5</b>
$r_S$	<b>0.48</b>	0.43	0.45	0.45	0.42	0.47	0.47
$r_{\text{OOi}}$	<b>0.71</b>	0.69	0.67	0.55	0.64	0.70	0.67
$r_{\text{RTi}}$	<b>0.30</b>	0.24	0.28	0.27	0.29	<b>0.30</b>	0.24
$r_{\text{Depth}}$	0.58	<b>0.62</b>	0.58	0.48	0.55	0.57	0.54

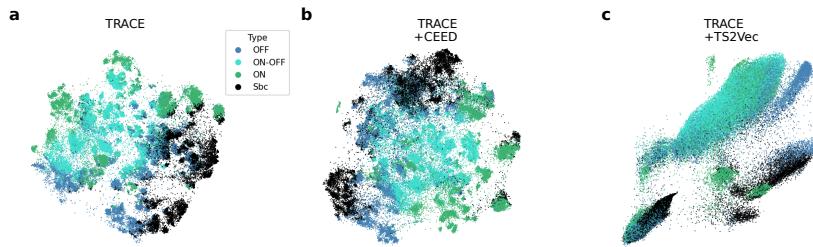
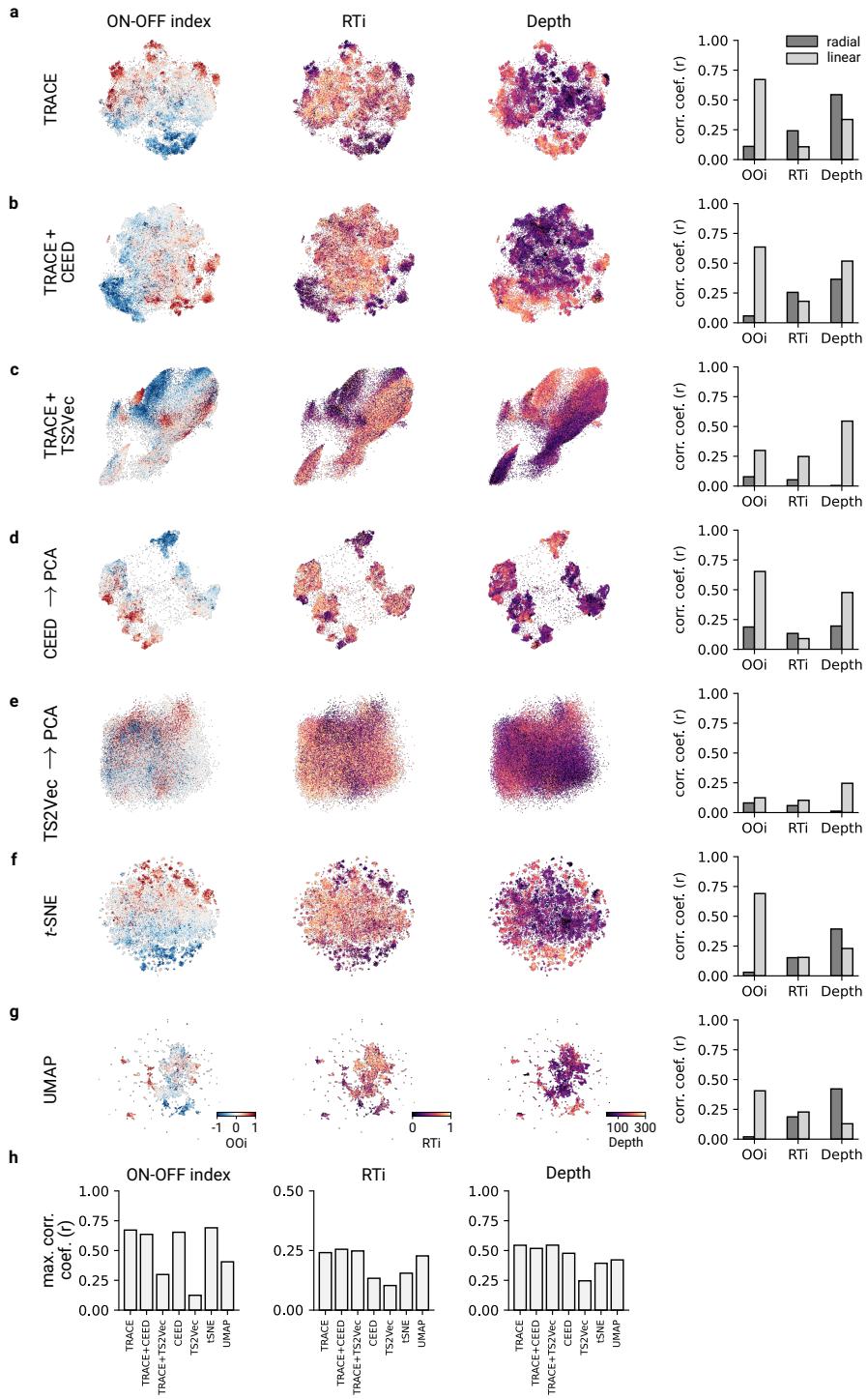
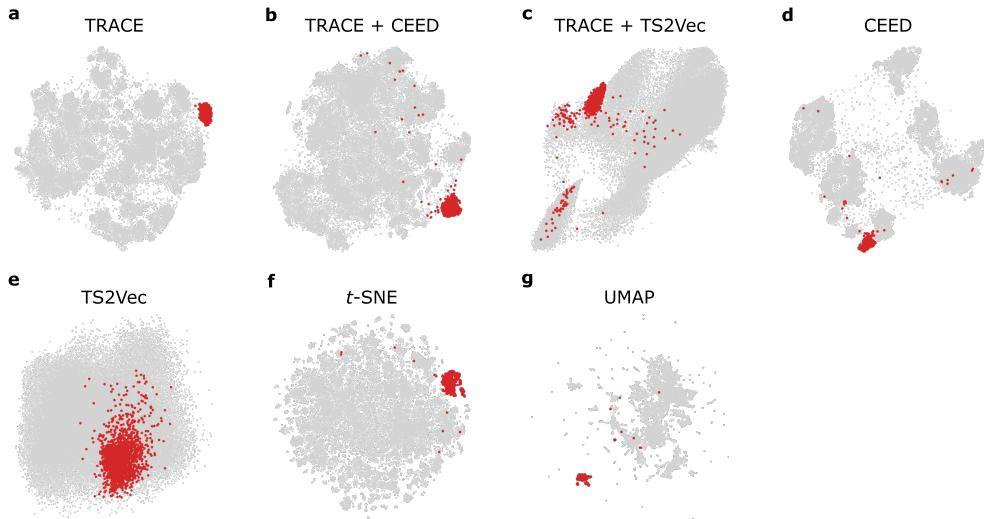


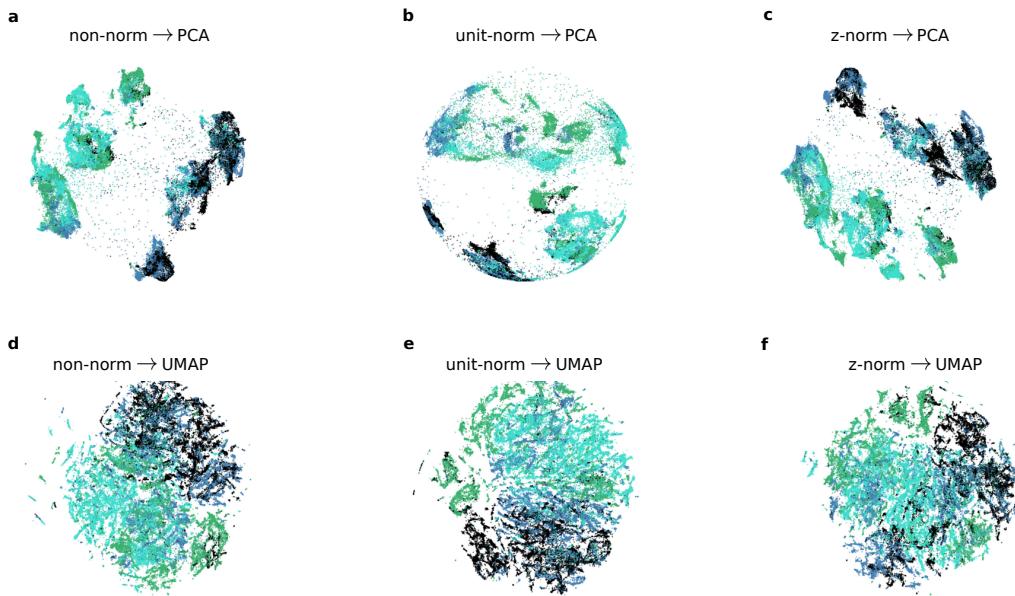
Figure S3: **TRACE-variants visually exhibit a good balance between resolving finer cluster structure and retaining large-scale structure of the calcium imaging dataset.** (a–c) Two-dimensional embeddings of (a) TRACE, (b) TRACE+CEED, (c) TRACE+TS2Vec. Color-coded according to their functional group.



**Figure S4: Evaluating low dimensional embeddings of the calcium imaging dataset by their correlation with biological variables.** (a) TRACE embedding colored by the ON-OFF-index (OOi, *left*), response transience index (RTi, *mid-left*), and recording depth in the superior colliculus (*mid-right*). *Right*: Absolute Pearson correlation coefficients between TRACE embedding and the three biological variables using radial and linear transformations to correlate. (b) Same as (a), but using the TRACE + CEED embedding, (c) TRACE + TS2Vec, (d) CEED → PCA, (e) TS2Vec → PCA, (f) t-SNE, and (g) UMAP. (h) Maximum correlation coefficients for each embedding for OOI (*left*), RTi (*middle*), and depth (*right*).



**Figure S5: Identified artifacts in the calcium imaging dataset across all methods.** (a) TRACE embedding with identified artifact island in red. (b) Same as in (a) but for TRACE + CEED, (c) TRACE + TS2Vec, (d) CEED → PCA, (e) TS2Vec → PCA, (f) t-SNE, and (g) UMAP.



**Figure S6: Different dimensionality reduction options for visualization of CEED's 5D output in 2D space.** (a–c) 2D PCA visualizations of the original CEED outputs in  $\mathbb{R}^5$  (a), CEED outputs normalized to the hypersphere  $S^4 \subset \mathbb{R}^d$  (b), the 5D CEED output with each dimension scaled to unit variance (c). (d–f) 2D UMAP visualizations of the same 5D data as in (a–c).

Table S3: Evaluating dimensionality reduction options for visualizing CEED’s 5-dimensional embeddings in 2D space. For the different normalizations, see the caption of Fig. S6. Best performance in **bold**. All metrics other than the aggregated rank are better when higher.

Model	$r_{OOi}$	$r_{RTi}$	$r_{Depth}$	$\mu_{Rank}$
CEED (non-norm PCA)	0.58	0.14	0.54	<b>2.33</b>
CEED (non-norm UMAP)	0.36	<b>0.19</b>	0.54	3
CEED (unit-norm PCA)	0.55	0.11	<b>0.55</b>	3
CEED (unit-norm UMAP)	0.38	0.16	0.31	4.33
CEED (z-norm PCA)	<b>0.59</b>	0.12	0.53	3
CEED (z-norm UMAP)	0.42	0.09	0.39	5

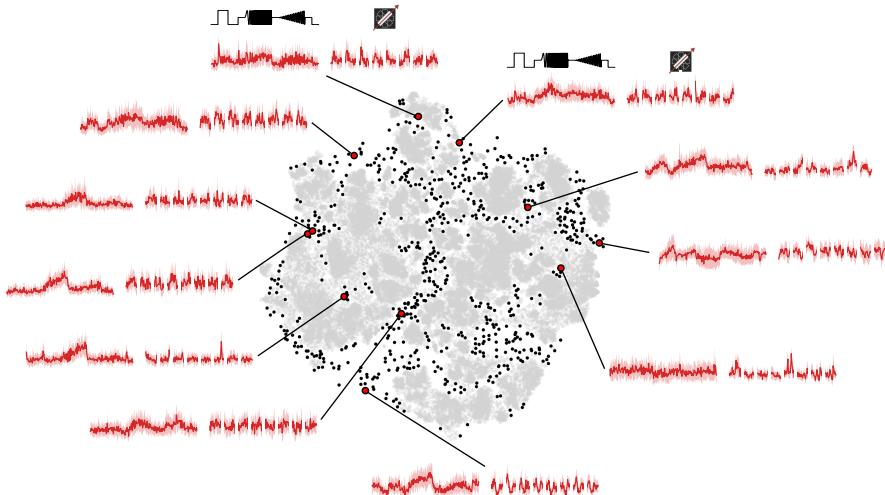


Figure S7: **Outliers in the TRACE embedding of retinal output neurons are noisy.** Isolating and detecting outliers (black) in the TRACE embedding space (gray). Example outliers are highlighted in red showing their highly noisy “chirp” and moving bar responses.

Table S4: Results for different methods on the calcium image data for varying embedding dimension. TRACE performed best in terms of  $k$ NN accuracy in all dimensions and maintained its performance in terms of Spearman correlation behind CEED, while TS2Vec degraded significantly. TS2Vec used the cosine similarity for  $d > 2$ .

Model	kNN accuracy				Spearman correlation $r_S$			
	$d = 2$	$d = 16$	$d = 128$	$\mu_{Rank}$	$d = 2$	$d = 16$	$d = 128$	$\mu_{Rank}$
TRACE	<b>69.7</b>	<b>75.3</b>	<b>75.5</b>	<b>1</b>	0.45	0.4	0.41	2.33
CEED	64.0	72.3	72.2	2	0.51	<b>0.58</b>	<b>0.51</b>	<b>1.33</b>
TS2Vec	23.8	54.9	66.4	3	<b>0.60</b>	0.13	0.18	2.33

Table S5: Training time for all methods on the calcium imaging dataset. Data augmentations for CEED and TRACE + CEED were pre-computed and sampled from during training to improve training time (see Sec. A.7). Please note that no data augmentations are necessary for TRACE. Each experiment was run on a NVIDIA A40 GPU 48 GB. For implementation details, see Sec. A.6.

Model	Epochs	Time (mm:ss)
TRACE	1000	23:17 $\pm$ 11.8s
TRACE + CEED	1000	25:40 $\pm$ 5.1s
TRACE + TS2Vec	1000	962:43 $\pm$ 2668.8s
CEED $\rightarrow$ PCA	1000	25:17 $\pm$ 19.7s
TS2Vec $\rightarrow$ PCA	<1	1:23 $\pm$ 2.0s
<i>t</i> -SNE	1000	6:59 $\pm$ 9.4s
UMAP	1000	4:56 $\pm$ 8.3s

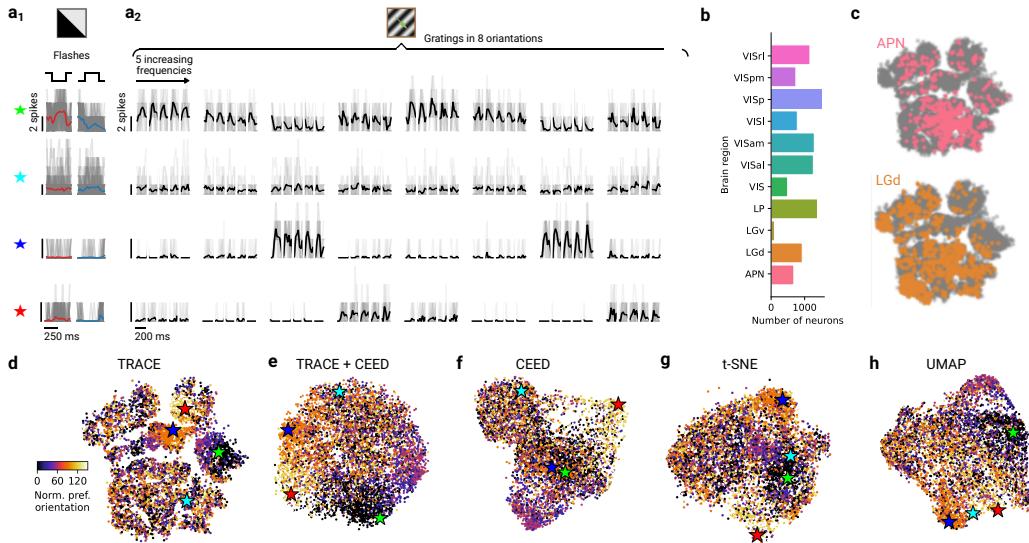
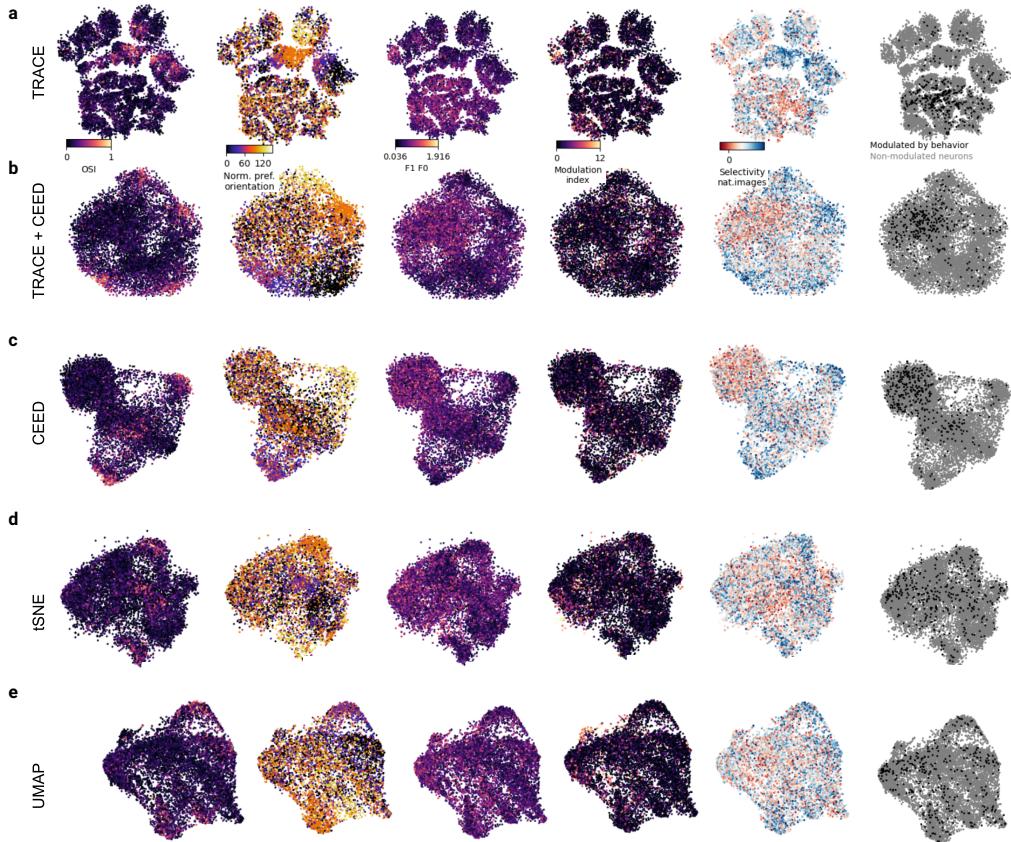


Figure S8: **Neuropixels dataset.** (a) Mean responses to dark flashes (red) and bright flashes (blue) ( $a_1$ ), as well as drifting gratings ( $a_2$ , black) for four example neurons. Single trial responses in gray. (b) Number of neurons per brain area. (c) Coverage of two example brain areas (APN, LGd) in the TRACE embedding. (d) TRACE embedding of the neuronal time series color-coded according to their preferred orientation. Example neurons from (a) are marked with stars. (e) Same as (d), but for TRACE+CEED, (f) CEED visualized in 2D using PCA, (g) *t*-SNE, and (h) UMAP.

Table S6: **Quantitative model performance on the Neuropixels dataset.** The columns are: model type (see text), the ARI score,  $k$ NN accuracy, Spearman correlation ( $r_S$ ), correlations with biologically meaningful variables (orientation selective index (OSI), grating modulation ratio (F1/F0), natural image selectivity (NIS),  $k$ NN regression for preferred orientation (PO), and mouse behavior, and the average rank  $\mu_{\text{Rank}}$  of each method. For a definition of the measures, see Sec. 4.2. All metrics other than the rank are better when higher. Uncertainties were insignificant and we omitted them. The best values for each metric are **bold**.

Model	ARI	$k$ NN acc	$r_S$	OSI	PO	F1/F0	NIS	behavior	$\mu_{\text{Rank}}$
TRACE	<b>0.04</b>	<b>23.7%</b>	0.40	0.38	<b>0.10</b>	0.17	0.12	<b>0.11</b>	<b>1.8</b>
+ CEED	0.03	20.7%	0.21	0.35	0.02	0.09	0.08	0.03	3.5
+ TS2Vec	0.01	16.9%	0.03	0.36	0.01	<b>0.29</b>	0.25	0.01	4.5
CEED $\rightarrow$ PCA	0.03	18.9%	0.20	0.27	0.01	0.09	0.05	0.07	4.3
TS2Vec $\rightarrow$ PCA	0.03	18.5%	0.04	<b>0.41</b>	0.01	0.14	<b>0.29</b>	0.02	4.0
<i>t</i> -SNE	0.01	21.8%	0.50	0.23	0.09	0.06	0.03	0.01	4.5
UMAP	0.01	18.6%	<b>0.52</b>	0.13	0.06	0.03	-0.02	0.01	5.5



**Figure S9: TRACE finds biologically relevant structure in Neuropixels data set. (a)** TRACE embedding colored by the OSI, the normalized preferred direction, the F1/F0 ratio, the modulation index, the selectivity index for natural images, and the neurons that are modulated by behavior vs. non-modulated by behavior. **(b)** Same as in (a) but for TRACE + CEED, **(c)** CEED → PCA, **(d)** t-SNE, and **(e)** UMAP.

## NeurIPS Paper Checklist

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: The main claims are our proposed new method TRACE, which is clearly mentioned in both the abstract and introduction. We mention the methods application domain of neuroscientific time-series data and outline why this contribution is both new and significant.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

### 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: We highlight the limitations of our method clearly and detail avenues for future work in the conclusion. To that end, we mention that the recorded data needs to exhibit multi-trial structure, as we leverage this pattern for formulating our contrastive loss function.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

### 3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: Our paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

#### 4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide the experimental setup and describe how our method works. In addition, we ran the methods multiple times and state the variability of the results. The code to reproduce our experiments is publicly available and we will de-anonymize it upon acceptance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

## 5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The code is made publicly available (see previous answer to “Experimental result reproducibility”). The neuronal dataset will be made publicly available in the future when the accompanying journal paper describing the experimental methods and biological findings is published. This approach to data release timing is necessary as the dataset represents new, unpublished research recording from axonal endings of retinal ganglion cells in superior colliculus via a novel chronic cranial window technique. All implementation details required for reproducibility are provided in the supplementary with specific implementation details for our method and all baselines.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We describe our experimental setup, as well as the implementation details. Furthermore, we make the code available, to make it possible to understand how we ran the experiments in detail.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[Yes\]](#)

Justification: We provide the mean and standard deviation across multiple runs for all experimental results (see e.g. Table 1). We verified that the presented embeddings are representative of all of the runs. For the synthetic dataset comparisons across different noise levels, we additionally computed the 95% confidence intervals.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [\[Yes\]](#)

Justification: We provide the details about our computing infrastructure, on what hardware we ran the experiments, and exemplary run times.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

## 9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [\[Yes\]](#)

Justification: We have reviewed the NeurIPS Code of Ethics and our paper and the research conducted confirms to this code.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

## 10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: This paper presents work whose goal is to advance foundational research on neuro-scientific representation-learning algorithms. There are no direct positive or negative societal impacts specific to our method.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our research contribution does not carry the potential for misuse, as we do not use data that could be implicated in any privacy or copyright violations. We do not work with language models, image generators or scraped datasets.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [\[Yes\]](#)

Justification: We properly cite all owners of software and code. We are the owners of the neural data shown.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

### 13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: We provide the code and document it in this submission in order to make it usable to the broader research community. We include the details about training and limitations (see answer to earlier questions) as well as use a license that allows usage of our code.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

### 14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

**15. Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

**16. Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorosity, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.