Efficient Neural Representations in Spatial Navigation and Pharmacokinetic Imaging: Information-Theoretic and Machine Learning Approaches

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Jared Deighton

May 2025

© by Jared Deighton, 2025
All Rights Reserved.

"I will not say: do not weep; for not all tears are an evil." - Gandalf, The Return of the King by J.R.R Tolkien

Acknowledgments

I want to thank my parents, Jeremy and Heather, for everything, including their endless reminders to "stay safe, get smart, and love others." Thank you, Alison, my sister, for believing in me, showing me how to succeed, and always seeing the bright side.

I also want to thank my advisor, Dr. Vasileios Maroulas, for guiding my studies and growth as a researcher, as well as Dr. Piotr Franaszczuk, Dr. Catherine Schuman, and Dr. Ioannis Sgouralis for serving on my dissertation committee. Furthermore, I would like to thank my elementary school teachers, high school teachers, professors, and anyone else who invested their time and energy in my learning.

To my friends, both new and old, whose fellowship kept me going through twenty years of learning, I thank you. This includes Camren Turner, Isaiah Keeler, Tucker Besch, Kyle Harkema, Matthew Blanker, Chris Ralph, Zach Reitsma, Kevin Sisco, Brett O'Brien, Peter Simala, Jonathan Kalman, Jacob Nguyen, Chiara Mattamira, Patrick Gillespie, Brittany Story, Wyatt Mackey, and Hayden Everett.

Finally, I want to thank my fiancée, Samantha, for her unwavering love and support, without which I would falter.

This work has been partially supported by the U.S. Army Research Laboratory Cooperative Agreement No. W911NF-21-2-0186.

Abstract

Mammalian spatial navigation relies on specialized neurons, such as place and grid cells, which encode position based on self-motion and environmental cues. While extensive research has explored the computational role of grid cells, the principles underlying efficient place cell coding remain less understood. Existing spatial information measures primarily assess single-neuron encoding, limiting insights into population-level representations. To address this gap, we introduce novel information-theoretic measures that quantify the encoding efficiency of multiple neurons, including the joint stimulus information rate for neuron pairs and the spectral-stimulus information for arbitrary populations. The spectral-stimulus information, defined as the leading eigenvalue of the stimulus information matrix, is maximized when neurons exhibit localized, non-overlapping firing fields—mirroring place cell and head direction cell activity in biological systems. We demonstrate that these measures can be used to train recurrent neural networks (RNNs) via self-supervised learning, leading to the emergence of place cells and head direction cells. Our findings highlight how neural populations collectively encode stimuli, offering a more comprehensive framework for understanding place cell formation and optimizing artificial navigation systems in novel environments. Additionally, functional ultrasound imaging (fUSI) provides high spatiotemporal resolution for monitoring cerebral blood volume (CBV) by detecting backscattered echoes from red blood cells. While fUSI has been widely used in preclinical neuroscience, many studies focus on predetermined regions of interest (ROIs), potentially overlooking relevant brain activity. To address this, we combined fUSI with three machine learning models—convolutional neural networks (CNNs), Vision Transformers (ViTs), and Support Vector Machines (SVMs)—to analyze the pharmacokinetics of Dizocilpine (MK-801), an NMDA receptor antagonist. CNNs demonstrated superior performance in detecting

and localizing drug-induced changes in brain hemodynamics. Using class activation mapping (CAM), CNNs identified anatomically specific patterns in cortical and hippocampal regions that align with known NMDA receptor distributions. Quantitative analysis confirmed significant drug-induced CBV reductions in CNN-identified regions. While all models distinguished between drug and control conditions, CNNs uniquely maintained anatomical precision while tracking drug effects over time. By integrating advanced machine learning with fUSI and spectral-spatial measures, our work provides novel insights into both neural representation and pharmacological effects on brain function. These approaches enhance our ability to model spatial coding in artificial systems and improve the detection of drug-induced neural changes, advancing both computational neuroscience and biomedical imaging.

Table of Contents

1 Introduction **1**

2 Self-Supervised Stimulus Encoding Via Stimulus Information Rates **5**

2.1 Preliminaries . . . 6

2.1.1 Entropy and Mutual Information . . . 6

2.2 Stimulus Information Rates . . . 7

2.2.1 Spectral Analysis . . . 13

2.3 Self-Supervised Stimulus Encoding . . . 15

2.3.1 Recurrent Neural Networks for Stimulus Encoding . . . 15

2.3.2 Place cell score . . . 17

2.3.3 Neural Recordings . . . 18

2.4 Results . . . 19

2.4.1 Place cells arise via self-supervised learning . . . 19

2.4.2 Neural decoding . . . 21

2.4.3 Path invariance and uniformity . . . 23

2.4.4 Self-Supervised Head Direction Cell Learning . . . 25

2.4.5 Eigenvector Relative Importance . . . 25

3 Dynamic Drug Decoding with Functional Ultrasound Imaging **30**

3.1 Materials and methods . . . 30

3.1.1 Animals . . . 30

3.1.2 Surgical Procedures . . . 30

3.1.3 Data Acquisition . . . 31

3.1.4 Data Pre-processing and Registration . . . 31
3.1.5 Models and Feature selection . . . 33
3.1.6 Training and Testing Data Sets . . . 37
3.1.7 Model performance . . . 38
3.1.8 Statistical Analysis . . . 39
3.2 Results . . . 39
3.2.1 Comparative model performance in detecting MK-801 effects . . . 39
3.2.2 CNN captures anatomically relevant MK-801 induced brain hemodynamic responses . . . 41
3.2.3 SVM and ViT show limited spatial specificity . . . 44

4 Conclusions **47**

Bibliography **52**

Appendices **65**

A Spectral Spatial Information . . . 66
B Additional Proofs . . . 73
B.1 Proof of Theorem 2.2.1 . . . 73
B.2 Proof of Theorem 2.2.3 . . . 73
B.3 Proof of Theorem 2.2.5 . . . 74
B.4 Proof of Theorem 2.2.6 . . . 75
B.5 Proof of Theorem 2.2.8 . . . 76
B.6 Proof of Corollary 2.2.9 . . . 77
C Experimental details . . . 78
D Additional place cell maps . . . 82
E Joint Spatial Information Rate on real place cell recordings . . . 83
F Sensitivity . . . 84
G Dynamic Drug Decoding Data and Code Availability . . . 86
H Dynamic Drug Decoding Supplemental Material . . . 86

Vita **87**

List of Tables

3.1 Hyperparameters used in the CNN parameter sweep. 35
3.2 Hyperparameters used in the Vision Transformer parameter sweep. 37

1 Mathematical Notation 66
2 Hyperparameters used for training the networks. 79

List of Figures

2.1 (a) Place cells recorded in hippocampal subarea CA3. Firing locations are shown as red dots on the path of a rat (black). (b) Activity maps of the cells. Red is high firing rate, and blue is no firing. (c) Diagram of the process of path integration. The agent integrates four velocities to determine its current location via the integrated path (red). (a-b) adapted from [97]. 6

2.2 Example of the spatial information rate (bits/spike) of place cells recorded in [46] and directional information rates (bits/spike) of head direction cells recorded in [79]. Both scenarios use Equation 2.4, albeit with different stimuli S. 9

2.3 (a) The Skaggs' stimulus information rate for a neuron firing at 1spike/s on half and a quarter of the spatial domain. While the stimulus information rate in bits/s is the same in each scenario, the stimulus information rate in bits/s is larger when A fires in a quarter of the domain. (b) The sum of the Skaggs' stimulus information rates of two neurons A and B in two scenarios. 9

2.4 The joint stimulus information rate (Equation 2.2) evaluated in four scenarios. As opposed to the sum of Skaggs' stimulus rates, the joint spatial information rate prefers scenario (b) over scenario (a). Furthermore, the joint stimulus information rate is maximized in scenario (d), when A and B fire in small, non-overlapping fields. 11

2.5 Leading eigenvalue of stimulus information matrix generated from four populations of 40 simulated Gaussian head direction cell firing patterns. The final scenario is evenly distributed firing fields. A larger leading eigenvalue indicates superior direction encoding. 14

2.6 (a) Diagram of RNN architecture used to perform self-supervised learning. Shown here over n_x steps. (b) As training progresses over n epochs, the neuron's firing fields become more place cell-like. Here the firing fields of four neurons are shown. 16

2.7 (a) Activation maps of top 64 learned place cells from a model trained with spectral-spatial information, normalized to be between 0 and 1, and sorted by place cell score (Equation (2.13)). (b) Original and trained place cell score box plots and histograms. Place cell scores are recorded across 50 total models for each loss function with varying place cell sizes. (c) Mean ($\pm$ standard deviation) sparsity, smoothness, binarity ($\epsilon = 0.1$), and spatial information rate (Equations (2.14- 2.17)) of trained place cells as the number of place cells increases. 20

2.8 Comparison of decoding capabilities of binarized neurons before and after training. (a) Simulated and predicted trajectories leave-one-out classification with 128 neurons. (b) Skaggs vs spectral leave-one-out decoding mean square error (MSE) as the number of trained place cells increases. The mean ($\pm$ standard deviation) of MSE across experiments is shown. (c) Correctly (orange) and incorrectly (grey) classified activations from a support vector machine (SVM) for quadrant classification using 16 neurons. (d) Skaggs vs spectral quadrant classification accuracy. The mean ($\pm$ standard deviation) of accuracy across experiments is shown. 22

2.9 (a) Two trajectories formed by shuffling the order of velocities. Since both trajectories end at the same location, neural representations at the final position should be identical despite different paths to get there. (b) Histogram of path invariance discrepancy across 1000 trajectories from all untrained models, models trained with Skaggs' spatial information, and models trained with spectral-spatial information. 24

2.10 (a) Comparison of two place cell firing fields in which the initial cells fire in the same region. Standard deviation in active place cells decreases as firing fields separate. (b) Mean (± standard deviation) of the standard deviation in the number of place cells with high activity (defined as exceeding 20% of the peak firing rate of the rate map) as the number of trained place cells varies. . 25

2.11 (a) Top 25 (sorted by directional information rate) Skaggs and Spectral-directional information rate trained neurons from models with 256 output neurons. (b). Mean (± standard deviation) sparsity, smoothness, binarity ($\epsilon = 0.1$), and directional information rate (Equations 2.14- 2.17) of trained head direction cells as the number of head direction cells increases. 26

2.12 Directional (left) and spatial (right) information versus eigenvector entry magnitude from head direction cell recordings done in [79] and place cell recordings done in [46]. 28

2.13 Mean (± standard deviation) test median absolute error(Eqaution (2.19)) versus number of neurons used for head direction decoding. Neurons are either removed randomly or via the magnitude of the stimulus information matrix's leading eigenvector entry. Removing neurons via the latter technique yields greater predictability in decoding performance and superior performance when removing less than $\approx 65\%$ of the neurons. 29

3.1 Experimental setup and fUSI acquisition. (A) fUSI motorized system setup for signal acquisition. (B) Power Doppler-based 2D vascular map image of a mouse brain in a sagittal plane. (C) Diagram of the experimental protocol for the 60 min of continuous fUSI acquisition. After 5 min, saline or 1.5 mg/kg MK-801 was intraperitoneally injected to the animals. 32

3.2 Convolutional Neural Network (CNN) architecture and class activation mapping (CAM), shown here for MK-801/Saline classification. 36

3.3 Flowchart of analysis pipeline. (A) fUSI mice brain signal acquisition. (B) Mice 2D power Doppler (pD) image time series alignment to a common reference image. (C) Data partitioning for CNN. (D) CNN and feature selection. (E) Class activation mapping. 40
3.4 Model classification accuracy for distinguishing MK-801 from saline groups using ViT, SVM, and CNN. Shaded areas represent standard error derived from averaging across animals. The vertical dotted line at 5 min marks the injection time, and the gray shaded area (minutes 3-5) indicates the baseline period. 42
3.5 (A) Mean CAM overlay on the reference image, highlighting regions of importance in CNN-based classification of MK-801 effects (threshold $\theta = 0.65$). Brighter regions indicate higher relevance. (B-C) Time series of $\%pD$ (i.e., CBV) changes in CAM-selected regions (red curves) compared to non-selected control regions (blue curves) for (B) MK-801 and (C) Saline groups. The shaded areas represent standard error derived from averaging across animals. The red dashed line at 5 min marks the injection time. A gray patch (3-5 min) indicates the pre-injection baseline window. 43
3.6 (A) Mean feature weights overlay on the reference image, highlighting regions of importance in SVM-based classification (threshold $\theta = 0.65$). Negative weights were extracted and inverted to specifically identify regions driving MK-801 (versus saline) classification, then normalized using percentile-based thresholding. Brighter regions indicate higher relevance for MK-801 detection. (B-C) Time series of $\%pD$ (i.e., CBV) changes in SVM-selected regions (red curves) compared to non-selected control regions (blue curves) for (B) MK-801 and (C) Saline groups. The shaded areas represent standard error derived from averaging across animals. The red dashed line at 5 min marks the injection time. A gray patch (3-5 min) indicates the pre-injection baseline window. . . 45

3.7 (A) Mean attention map overlay on the reference image, highlighting regions of importance in ViT-based classification (threshold $\theta = 0.025$). Brighter regions indicate higher relevance. (B-C) Time series of $\%pD$ (i.e., CBV) changes in ViT-selected regions (green curves) compared to non-selected control regions (blue curves) for (B) MK-801 and (C) Saline groups. The shaded areas represent standard error derived from averaging across animals. The red dashed line at 5 min marks the injection time. A gray patch (3-5 min) indicates the pre-injection baseline window. 46

1 (a) Place cell box plots and histograms across 20 models (10 trained with 32 place cells, 10 with 64). Comparison in place cell score (Equation (2.13)) is shown between models trained via maximizing $S(J)$, the summation of the entries of J versus maximizing the leading eigenvalue of J. (b) Comparison between the leading eigenvalue of the spatial information matrix J, λ_1, and $S(J)$ across four scenarios. Consider four neurons, A, B, C, and D in a simple four-quadrant square arena. If the letter of a neuron appears in a quadrant, then we set the firing rate of the neuron in that quadrant to 1 spike/sec. The corresponding leading eigenpair (λ_1, v_1) as well as $S(J)$ are shown for each scenario. 71

2 Mean ($\pm$ standard deviation) train and test loss from 10 experiments for each N_p (number of place cells) and loss function. 80

3 Pre-trained and trained (with spectral spatial information) place cell activations for 128 place cells. Each place cell's activations are normalized between 0 and 1. The top 64 are displayed in Fig. 2.7. 82

4 (a) Skaggs' spatial information rate for six place cells recording from [46] located in the right hippocampal CA1 subfield of a male Japanese monkey performing a shuttling task on a rectangular (4×0.9 m) track. (b) Joint spatial information, $I(A, B)$, compared to the sum of Skaggs' spatial information, $I(A) + I(B)$, for all possible pairs of neurons. The leading eigenvalue of the corresponding spatial information matrix is $\lambda_1 = 24.21$. . . 83

5 Additional performance metrics for distinguishing MK-801 from saline groups using ViT, SVM, and CNN. (A) Precision, (B) Recall, (C) F1 Score, and (D) ROC AUC. For all panels, shaded areas represent standard error derived from averaging across animals. The vertical dotted line at 5 min marks the injection time, and the gray shaded area (minutes 3-5) indicates the baseline period. 86

Chapter 1

Introduction

In recent years, there has been significant interest in building computational models that mimic mammalian spatial representation, as a proxy for understanding the utility of underlying neural dynamics [8, 13, 20, 22, 37, 43, 60, 63, 64, 65, 66, 70, 88, 92, 93]. Many implementations train recurrent neural networks (RNNs) for path integration via supervised learning. These models are trained to predict position or to match their output to that of predetermined place cells [22, 92, 93]. By training RNNs to perform path integration based on velocity inputs, grid cell-like representations may form in the hidden layer [8, 22, 88, 92]. This suggests that grid cells, border cells, and other cells observed in the medial entorhinal cortex (MEC), are a natural solution for representing space efficiently given the predominant recurrent connections in the neural circuits [22]. However, it has been recently suggested that the emergence of grid cells in RNNs with velocity inputs may be strongly driven by particular, non-fundamental, and post-hoc implementation choices as opposed to fundamental truths about neural circuits or the loss function(s) they optimize [87]. That is to say, grid cells are not necessarily a natural solution for efficient space representation. Recent normative modeling approaches suggest grid cells construct a basis for predicting future outcomes [96, 107], and encode not only variables of interest but also how the variable transforms [28].

Self-supervised learning, which does not rely on external targets, has just recently been applied to the brain's spatial representations and is not well explored in either the machine learning or neuroscience literature [88]. Previous studies (both supervised and self-supervised) focus on the emergence of grid-cell-like representations.

In contrast to grid cells, place cells have received little attention from the computational neuroscience and machine learning communities. Theoretical and computational studies on optimal place field location and the emergence of place cells have been conducted but are limited to a *single* place cell [90, 104]. Meanwhile, studies in mice and rats have shown that place cells use several factors when determining where to fire in space, including self-motion and the presence of boundaries [17, 55, 71]. In 2017, Keinath et al. [51] showed that the hippocampal map realigns after disorientation, preserving place cell anti-correlated firing, even after realignment within the environment. This suggests that when seeking to understand how place cells encode space, analyzing a pair or group of place cells rather than each one individually is more useful, as they collectively act to form a cognitive map [73, 81]. Information theory [89] has emerged as a technique to determine how much information neurons carry about stimuli [23, 81, 90]. In particular, Quian Quiroga and Panzeri [81] showed that the capacity of a set of neurons to decode position relates to the amount of spatial information conveyed by the neurons. Paramount in the relationship between information theory and neuroscience is the formulation of the spatial information rate by Skaggs et al. [90], which has seen tremendous use in characterizing how well single neurons encode space in both experimental studies [5, 7, 31, 32, 34, 35, 39, 45, 67, 74] and in theoretical studies [94, 104].

In Chapter 2, we discuss novel information-theoretic measures to measure neuron population encoding efficiency and train RNNs to encode stimuli through self-supervised learning [24]. The novel measures include the joint stimulus information rate for two neurons and the spectral-stimulus information for an arbitrary number of neurons. The spectral-stimulus information is the leading eigenvalue of the stimulus information matrix. We prove that this is maximized when neurons fire in small, non-overlapping regions, similar to place cell firings throughout space, and head direction cell firings across directionality.

We show that our metrics can be used to train RNNs to encode stimuli through self-supervised learning. In particular, our models are capable of learning many place cells and head direction cells. We also demonstrate that learned place cells exhibit strong decoding capabilities, outperforming neurons trained with traditional Skaggs' spatial information rate.

Next, in Chapter 3 we turn our attention to dynamic drug decoding via Functional ultrasound imaging (fUSI) [25]. fUSI is an emerging hemodynamic-based neuroimaging technology that measures cerebral blood volume (CBV) changes by detecting backscattered echoes from red blood cells moving within its field of view (FOV) [58, 59]. It provides a unique combination of large spatial coverage, high spatiotemporal resolution ($\sim$ 100 µm and $\sim$ 100 ms), and sufficient sensitivity to detect slow blood flow changes (less than 1 mm/s). The relative simplicity and portability of ultrasound scanners have allowed fUSI to be performed in a wide range of preclinical and clinical studies, providing minimally invasive neural imaging in species ranging from mice to humans [75, 76, 48, 27, 68, 42]. Among its various applications, fUSI has been employed in preclinical drug development to elucidate the mechanisms of action of drugs targeting the central nervous system. In particular, the mechanisms of various drugs, including anesthetics, cholinesterase agonists and antagonists, N-methyl-D-aspartate (NMDA) receptor antagonists, and selective norepinephrine reuptake inhibitors (SNRIs), have been investigated with the goal of better understanding their effects on brain function and vascular dynamics and accelerating the development of new drug therapies [82, 100, 101, 21].

Despite the important contribution of these neuropharmacological studies, they predominantly focus on how drugs influence the hemodynamic signal in specific regions of interest (ROIs) and/or the functional connectivity between these regions. This approach inherently carries the risk of bias, as potentially relevant neural activity outside these predetermined areas may be ignored. Therefore, there is a clear and distinct need for developing advanced analytical methods with the capability to assess the dynamic effects of drugs on the brain without relying on a priori region specification.

To address this, we develop and compare multiple machine-learning approaches combined with fUSI to elucidate the pharmacokinetics of Dizocilpine (MK-801) in the brain. MK-801 is a potent and selective NMDA receptor antagonist that was initially employed as a pharmacological model of psychosis in rodents [6] and is still extensively utilized to model schizophrenia [108]. Our comparative analysis demonstrates that CNNs provide superior performance in both detecting and localizing drug-induced changes in brain hemodynamics, offering anatomically precise insights into the temporal evolution of drug effects. The

CNN approach successfully identified key regions showing differential responses to MK-801, particularly in cortical and hippocampal areas, which align with known distributions of NMDA receptors and previous findings on MK-801's effects.

While all three machine learning approaches could distinguish between drug and control conditions, CNNs proved particularly effective due to their ability to capture hierarchical spatial features while maintaining anatomical specificity. This specificity is enhanced through Class Activation Mapping (CAM) [109], which provides clear visualization of the regions most relevant for drug effect classification. By combining CNN with Axiom-Based Grad-Class Activation Mapping (XGRAD-CAM) [36], which integrates principles of sensitivity and conservation, we validate the biological relevance of CNN-identified regions through quantitative analysis of CBV changes. This integrated approach offers a powerful new methodology for investigating drug actions in the brain, enabling unbiased detection and localization of drug effects while maintaining anatomical precision and biological interpretability.

Chapter 2

Self-Supervised Stimulus Encoding Via Stimulus Information Rates

The brain possesses several specialized neurons that fire in response to given stimuli, such as location, head direction, speed, noise, and distance from walls [53, 54, 72, 98, 103]. O'Keefe and Dostrovsky [72] (1971) discovered spatially selective neurons in the hippocampus, which they termed place cells. Place cells form a sparse representation of space by firing at one or several seemingly random locations in small and large environments [84] (Figure 2.1 (a)). In 2005, Hafting et al. [44] discovered another set of spatially selective neurons in the MEC, termed grid cells. In contrast to place cells, grid cells exhibit regular hexagonal spatial firing fields. Together, the MEC and hippocampus can encode relative spatial location, without reference to external cues, by integrating linear and angular self-motion [64]. This process is described as path integration or dead reckoning (Figure 2.1 (b)).

The process of firing in response to stimuli is known as neural encoding. To measure the efficiency of encoding, neuroscientists often turn to information theory [89], the mathematical study of the transmission and processing of information. These techniques have been used to determine how much information neurons carry about stimuli [23, 81, 90]. In particular, Quian Quiroga and Panzeri [81] showed that the capacity of a set of neurons to decode position relates to the amount of spatial information conveyed by the neurons. Paramount in the relationship between information theory and neuroscience is the formulation of the spatial information rate by Skaggs et al. [90], which has seen tremendous use in characterizing

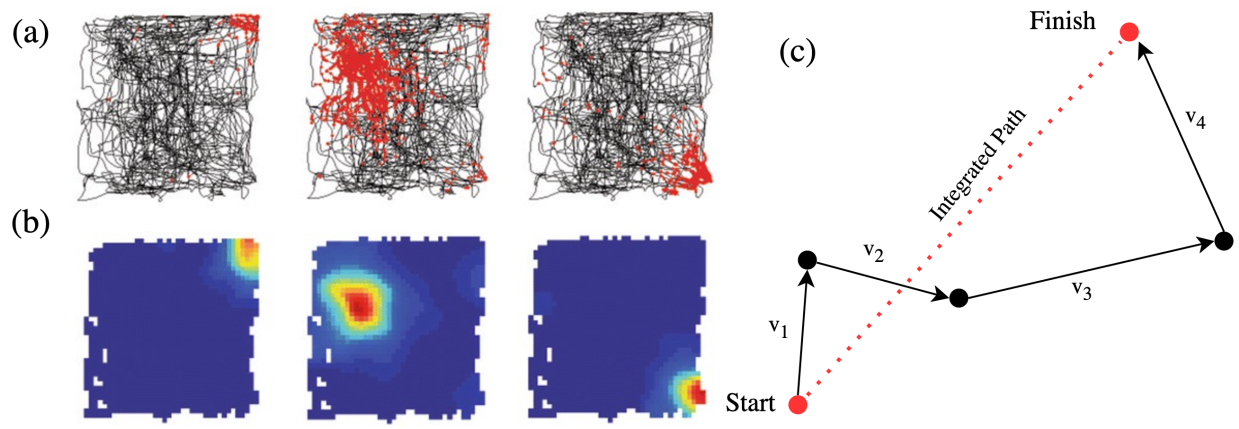

Figure 2.1: (a) Place cells recorded in hippocampal subarea CA3. Firing locations are shown as red dots on the path of a rat (black). (b) Activity maps of the cells. Red is high firing rate, and blue is no firing. (c) Diagram of the process of path integration. The agent integrates four velocities to determine its current location via the integrated path (red). (a-b) adapted from [97].

how well single neurons encode space in both experimental studies [5, 7, 31, 32, 34, 35, 39, 45, 67, 74] and in theoretical studies [94, 104]. Further works introduced information-theoretic measures for general stimuli, such as the stimulus-specific information [15, 16], which has been applied to gauge auditory encoding [50].

2.1 Preliminaries

The brain is an information-processing unit. Oftentimes, neuroscientists incorporate information theory to describe neural coding—characterizing the relationship between stimulus and neural responses. Here, we introduce the joint stimulus information rate and spectral-stimulus information. Before introducing the stimulus information rates, we review some common information measures and discuss their use in building the stimulus and joint stimulus information rates.

2.1.1 Entropy and Mutual Information

The most fundamental measurement in information theory is **entropy** [89], or the average level of uncertainty. For a discrete random variable X with probability mass function $p(x)$

for $x \in \mathcal{X}$ the entropy, $H(X)$ is given by

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log_2(p(x)), \tag{2.1}$$

where $\mathcal{X}$ is the set of possible outcomes of X, and $p(x)$ is the probability of outcome x. The units of $H(X)$ are *bits* if the logarithm is base 2, or *nats* if using the natural logarithm. High entropy indicates uncertainty in the distribution. Low entropy indicates predictability or certainty in the distribution. In other words, when entropy is low, the random variable's outcomes are more concentrated. A uniform distribution maximizes entropy.

To conceptualize how information about a random variable changes uncertainty in another random variable, we recall the concept of **mutual information** [89]. For two discrete random variables, X and Y, the mutual information is calculated via a double sum:

$$I(X : Y) = \sum_{x \in \mathcal{X},\ y \in \mathcal{Y}} p(x, y) \log_2 \left(\frac{p(x, y)}{p(x)p(y)} \right). \tag{2.2}$$

The mutual information of X and Y is the information X provides about Y, or vice versa. In this sense, it is a measure of independence. In fact, $I(X : Y)$ is equivalent to the Kullback Leiber divergence between the joint distribution $p(x, y)$ and the joint distribution under the assumption the data are independent $p(x)p(y)$.

In this work, we will be interested in information regarding some stimulus space S representing the set of possible values a stimulus can take. For example, S may be locations, directions, or speeds. Focusing on discrete stimulus spaces, the *occupancy distribution* is the probability mass function such that $p(s)$ for $s \in S$ is the probability that stimulus s occurs.

2.2 Stimulus Information Rates

To understand the effectiveness of neural encoding, we proceed with stimulus information rates [12]. These measures depend only on the occupancy distribution and the neuron firing rates across the stimulus space. To obtain the stimulus information rate of a neuron A for stimulus space S, we model the probability of neuron A firing during a short time interval

Δt as a Bernoulli distribution with parameter $\lambda_A(s)\Delta t$ where $\lambda_A(s)$ is the mean firing rate of neuron A given stimulus $s \in S$.

Definition 2.1. *The **stimulus information rate** of a single neuron A, across stimulus space S, is*

$$I_{sec}(A : S) = \sum_{s \in S} \lambda_A(s) \log_2 \left(\frac{\lambda_A(s)}{\bar{\lambda}_A} \right) p(s), \tag{2.3}$$

where s is the stimulus vector, $p(s)$ is the probability of the agent receiving stimulus s, $\lambda_A(s)$ is the mean firing rate of the neuron A with stimulus s, and $\bar{\lambda}_A = \sum_{s \in S} \lambda_A(s)p(s)$ is the mean firing rate of neuron A over the entire stimulus space [12].

At this moment, the stimulus information rate has units of bits/s, bits per second. Measuring the information quantity in bits/spike is often more useful for distinguishing information content. This is denoted as I_{spike}, and defined by

$$I_{spike}(A : S) = \frac{1}{\bar{\lambda}_A} I_{sec}(A : S) = \sum_{s \in S} \frac{\lambda_A(s)}{\bar{\lambda}_A} \log_2 \left(\frac{\lambda_A(s)}{\bar{\lambda}_A} \right) p(s), \tag{2.4}$$

I_{spike} measures the specificity of a neuron with respect to stimulus S [90].

As demonstrated in Figure 2.2, if S represents a set of spatial locations, the stimulus information rate is referred to as the spatial information rate. Similarly, if S represents a set of orientations, it is referred to as the directional information rate. This nomenclature applies to all future information measures in this work, where stimulus may be replaced with the specific variable whose encoding by a neuron we wish to evaluate. We present an example of the stimulus information rate, and its shortcomings, in Figure 2.3.

Theorem 2.2.1. *For a fixed occupancy distribution such that $\min_i p(i) = \epsilon$, the maximum of the stimulus information rate is $\log_2(1/\epsilon)$.*

Proof. See Appendix B.1. □

Corollary 2.2.2. *For a uniform occupancy distribution, the maximum of the stimulus information rate is $\log_2(n_s)$.*

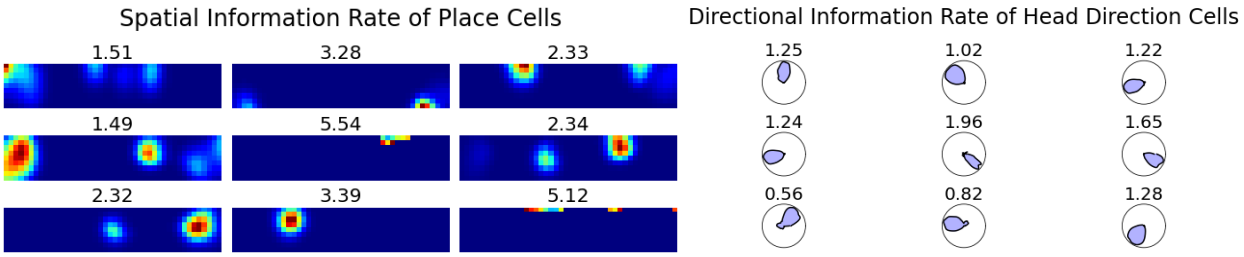

Figure 2.2: Example of the spatial information rate (bits/spike) of place cells recorded in [46] and directional information rates (bits/spike) of head direction cells recorded in [79]. Both scenarios use Equation 2.4, albeit with different stimuli S.

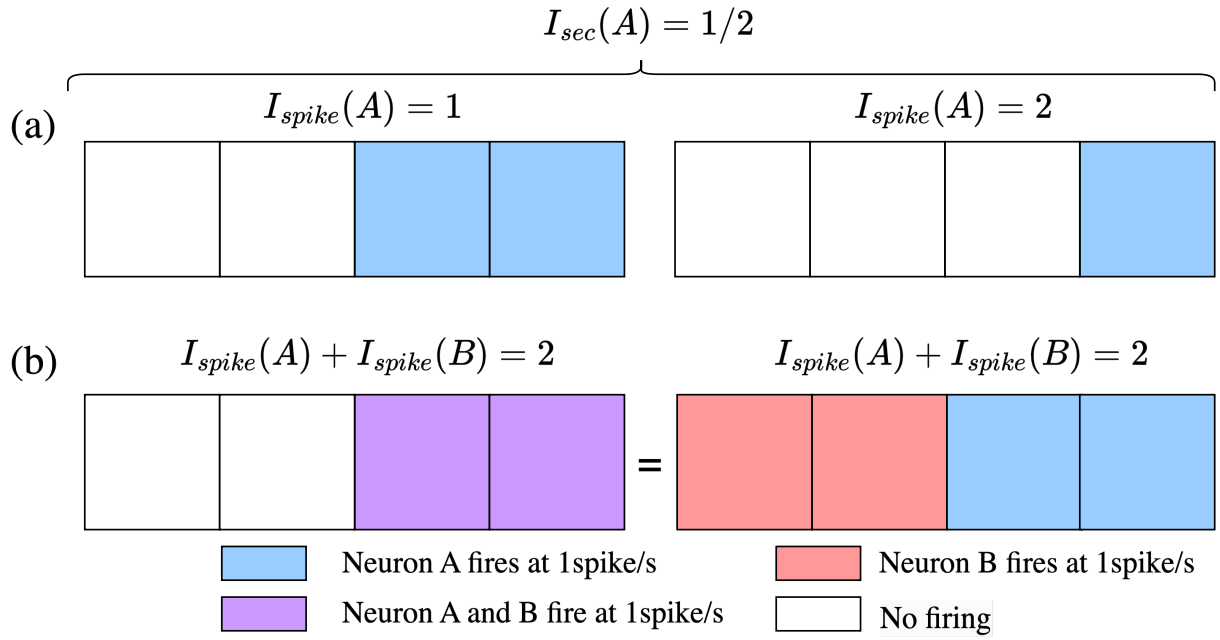

Figure 2.3: (a) The Skaggs' stimulus information rate for a neuron firing at 1spike/s on half and a quarter of the spatial domain. While the stimulus information rate in bits/s is the same in each scenario, the stimulus information rate in bits/s is larger when A fires in a quarter of the domain. (b) The sum of the Skaggs' stimulus information rates of two neurons A and B in two scenarios.

Proof. Use Theorem 2.2.1 with $\epsilon = \frac{1}{n_s}$ □

Theorem 2.2.3. *For a fixed discrete occupancy distribution p such that $\min_i p(i) = \epsilon$ is reasonably large, the stimulus information rate is L-Lipschitz. The stimulus information rate is not L-Lipschitz for all p.*

Proof. See Appendix B.2. □

Although the stimulus information rate effectively gauges a single neuron's encoding efficiency, it is poorly suited for analyzing pairs or populations of neurons. As shown in Figure 2.3, the sum of the stimulus information rates is the same regardless of whether A and B have identical or opposite firing patterns. To accommodate this, we derive the joint stimulus information rate of two neurons in Construction A.1.

Definition 2.2. *Consider two neurons A and B, and let $r \in [-1, 1]$ be the Pearson's correlation coefficient between neurons over some stimulus domain S. Without assuming firing independence, we can explicitly write the joint distribution for neurons A and B firing (or not firing) over a short time frame Δt. Then, if p is a probability mass function over S, then the* ***joint stimulus information rate*** *(in bits/s) of neurons A and B is*

$$\begin{aligned} I_{sec}(A, B : S) = \sum_{s \in S} \Bigg[& p(s) r \sqrt{\lambda_{A,B}(s)} \log_2 \left(\frac{\sqrt{\lambda_{A,B}(s)}}{\tilde{\lambda}_{A,B}} \right) \\ & + p(s) \left(\lambda_A(s) - r\sqrt{\lambda_{A,B}(s)} \right) \log_2 \left(\frac{\lambda_A(s) - r\sqrt{\lambda_{A,B}(s)}}{\bar{\lambda}_A - r\tilde{\lambda}_{A,B}} \right) \\ & + p(s) \left(\lambda_B(s) - r\sqrt{\lambda_{A,B}(s)} \right) \log_2 \left(\frac{\lambda_B(s) - r\sqrt{\lambda_{A,B}(s)}}{\bar{\lambda}_B - r\tilde{\lambda}_{A,B}} \right) \Bigg], \end{aligned} \tag{2.5}$$

where $\lambda_A(s)\lambda_B(s) = \lambda_{A,B}(s)$ and $\sum_{s \in S} p(s)\sqrt{\lambda_{A,B}(s)} = \tilde{\lambda}_{A,B}$.

We can likewise define $I_{spike}(A, B : S)$ by choosing an appropriate normalization term:

$$I_{spike}(A, B : S) = \frac{1}{\Lambda_{A,B}} I_{sec}(A, B), \tag{2.6}$$

where

$$\Lambda_{A,B} = \sum_{s \in S} p(s) \frac{\lambda_A(s) + \lambda_B(s)}{2} = \frac{\bar{\lambda}_A + \bar{\lambda}_B}{2}.$$

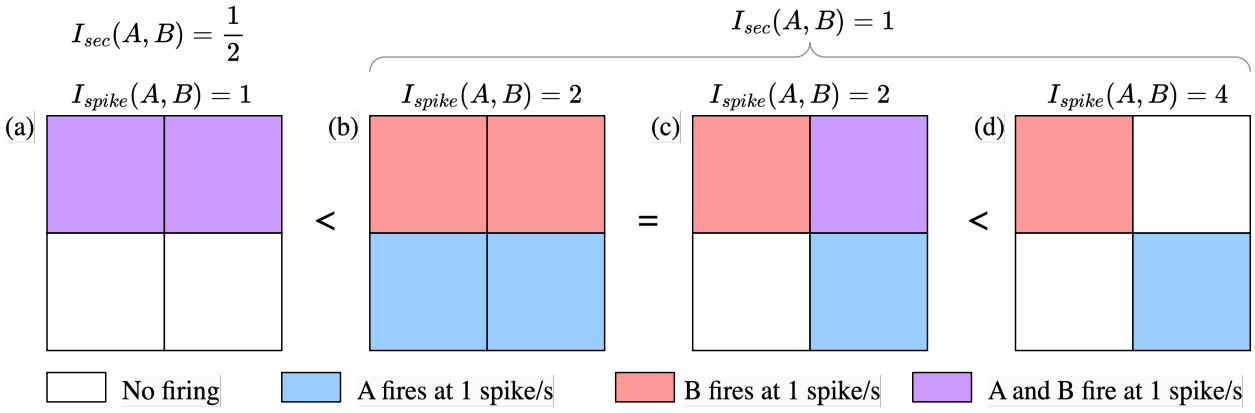

Figure 2.4: The joint stimulus information rate (Equation 2.2) evaluated in four scenarios. As opposed to the sum of Skaggs' stimulus rates, the joint spatial information rate prefers scenario (b) over scenario (a). Furthermore, the joint stimulus information rate is maximized in scenario (d), when A and B fire in small, non-overlapping fields.

As desired, $I_{spike}(A, B : S)$ *is in units of bits/spike. Furthermore, as shown in Theorem A.4,* $I_{spike}(A, B : S) = I_{spike}(B, A : S)$*, and* $I_{spike}(A, A : S) = I_{spike}(A : S)$*.*

Our definition of $I_{spike}(A, B : S)$ has two main limitations. First, it is not well-defined for all possible collections of firing rates $\{\lambda_A(x_i)\}_{i=1}^{n_x}$, $\{\lambda_B(x_i)\}_{i=1}^{n_x}$. We circumvent this by assigning the value 0 to terms not well-defined, as is commonly done for $0 \log(0)$ in information-theoretic quantities. This choice is well-motivated, as $\lim_{x \to 0} x \log(x) = 0$. Furthermore, we have yet to construct a proof that $I_{spike}(A, B : S)$ is non-negative when only considering well-defined terms. However, our experiments have not encountered a negative joint spatial information rate. Thus, to proceed with analytical results, we assume the following conjecture.

Conjecture 2.2.4. *The joint stimulus information rate,* $I_{spike}(A, B : S)$ *is nonnegative.*

When two neurons do not fire in the same part of the stimulus domain, the joint stimulus information rate is proportional to the sum of the individual stimulus information rates of the neurons.

Theorem 2.2.5. *If* $\{s \in S : \lambda_A(s) > 0\} \cap \{s \in S : \lambda_B(s) > 0\} = \emptyset$, *then*

$$I_{sec}(A, B : S) = I_{sec}(A : S) + I_{sec}(B : S)$$
$$I_{spike}(A, B : S) = \frac{1}{\Lambda_{A,B}}(I_{sec}(A : S) + I_{sec}(B : S))$$

Proof. See Appendix B.3. □

Importantly, the joint stimulus information rate can quantify the efficiency of a pair of neurons encoding a stimulus (Figure 2.4). That is, it prefers when neurons fire in *different* regions of the stimulus space. Furthermore, it retains the desirable properties of the stimulus information rate— it is maximized when neurons fire in small regions of the stimulus space.

Theorem 2.2.6. *For a fixed occupancy distribution with smallest entries* $p(s_i) = \epsilon_1 > 0$ *and* $p(s_j) = \epsilon_2 > 0$, $\epsilon_1 \geq \epsilon_2$, *the maximum of the joint stimulus information rate is*

$$\frac{2\left(\epsilon_1 \lambda_A(s_1) \log_2(1/\epsilon_1) + \epsilon_2 \lambda_B(s_2) \log_2(1/\epsilon_2)\right)}{(\bar{\lambda}_A + \bar{\lambda}_B)}$$

Furthermore, this occurs precisely when neurons A *and* B *fire in distinct stimulus bins. That is, the maximum is achieved when, without loss of generality,* $\{s \in S : \lambda_A(s) > 0\} = s_i$ *and* $\{s \in S : \lambda_B(s) > 0\} = s_j$ *for some* $s_i, s_j \in S$ *where* $i \neq j$.

Proof. See Appendix B.4 □

Corollary 2.2.7. *For a uniform occupancy distribution, the maximum of the joint stimulus information rate is* $2\log_2(n_s)$.

Proof. Use Theorem 2.2.6 with $\epsilon_1 = \epsilon_2 = \frac{1}{n_s}$. □

For $n_p \geq 3$ neurons, attempts to derive an information metric with techniques identical to those used for the joint stimulus information rate result in an underdetermined system. Therefore, we turn toward spectral analysis to measure neuron encoding efficiency at the population level.

2.2.1 Spectral Analysis

Definition 2.3. *The **stimulus information matrix** $J \in \mathbb{R}^{n_p \times n_p}$ for n_p neurons is the matrix whose entries are given by*

$$J_{i,j} = I_{spike}(A_i, A_j : S). \tag{2.7}$$

J is symmetric, and its diagonal consists of the stimulus information rates of the individual neurons, $I_{spike}(A_i : S)$ for $i = 1, \ldots, n_p$. If $\lambda_1, \ldots \lambda_{n_p}$ are the eigenvalues of J such that $|\lambda_1| \geq |\lambda_2| \geq, \ldots, \geq |\lambda_{n_p}|$, we have that $\lambda_i \in \mathbb{R}$ for all $i = 1, \ldots, n_p$. Furthermore, if $J_{i,j} = I_{spike}(A_i, A_j : S) \geq 0$ for all i, j, then $\lambda_1 \geq 0$ by the Perron-Frobenius Theorem [38]. We call this eigenvalue the **spectral-stimulus information**. The spectral-stimulus information increases as neuron firing fields disperse over the stimulus domain (Figure 2.5). As before, we may replace stimulus with the appropriate stimulus, such as spectral-spatial, or spectral-directional information, to reference the variable whose encoding by neurons we wish to evaluate.

When no two neurons fire in the same portion of the stimulus domain, we may write the stimulus information matrix explicitly as well as the spectral-stimulus information.

Theorem 2.2.8. *Suppose we have a stimulus space with n_s stimulus bins and a uniform occupancy distribution. Given $n_p = n_s$ neurons such that $\{s \in S : \lambda_i(s) > 0\} \neq \emptyset$ for all $i = 1, \ldots, n_p$ and $\{s \in S : \lambda_i(s) > 0\} \cap \{s \in S : \lambda_j(s) > 0\} = \emptyset$ for all $i, j = 1, \ldots, n_p$, $i \neq j$, we have*

$$J_{i,j} = \log_2(n_s)(2 - \delta_{i,j})$$

where n_s is the number of stimulus bins and $\delta_{i,j}$ is the Kronecker delta, equal to 1 when $i = j$ and zero otherwise.

Proof. See Appendix B.5. □

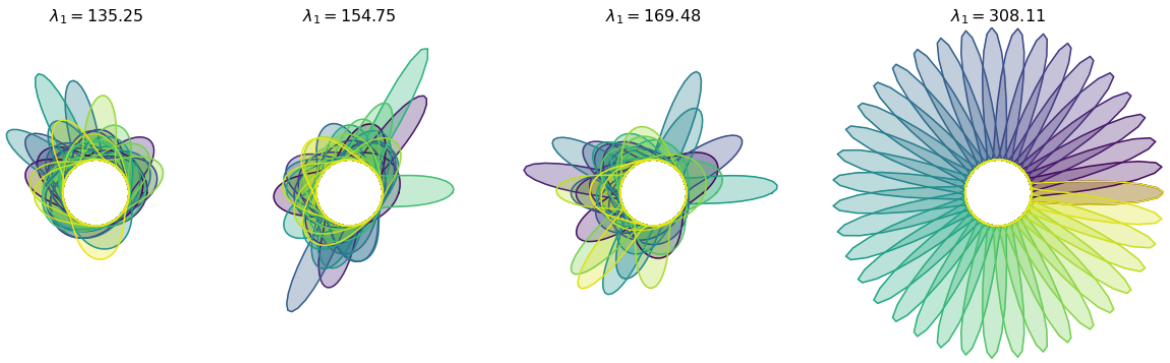

Figure 2.5: Leading eigenvalue of stimulus information matrix generated from four populations of 40 simulated Gaussian head direction cell firing patterns. The final scenario is evenly distributed firing fields. A larger leading eigenvalue indicates superior direction encoding.

Corollary 2.2.9. *Under the same assumptions, the leading eigenvalue and eigenvector pair* (λ_1, v_1) *of* J *is*

$$(\lambda_1, v_1) = (\log_2(n_s)(2n_p - 1), \ \mathbf{1}) \tag{2.8}$$

where $\mathbf{1} \in \mathbb{R}^{n_p}$ *is the vector of all ones.*

Proof. See Appendix B.6. □

Theorem 2.2.10. *Given a uniform occupancy distribution, this is the largest eigenvalue possible.*

Proof. Let $\hat{J}$ be the matrix described in Theorem 2.2.8. For a general stimulus information matrix J, by Theorem 2.2.6, $J_{i,j} \leq \hat{J}_{i,j}$ for all i, j. Now, since J is symmetric with nonnegative entries, the leading eigenvector of J, v, has nonnegative components and maximizes the Rayleigh Quotient of J. Thus,

$$\max_{x, \ ||x||=1} x^T J x = \lambda_1 = v^T J v \leq v^T \hat{J} v \leq \max_{x, \ ||x||=1} x^T \hat{J} x = \log_2(n_s)(2n_p - 1)$$

So, the leading eigenvalue J is less or equal to that of $\hat{J}$. □

Theorem 2.2.10 states that spectral-stimulus information is optimized when all neurons fire in non-overlapping regions of the stimulus domain. That is, given a stimulus domain

with n_s stimulus bins and $n_p = n_s$ neurons, it is best for each neuron to fire in its unique stimulus bin. This result motivates the use of spectral-stimulus information for self-supervised encoding in machine learning.

2.3 Self-Supervised Stimulus Encoding

Self-supervised stimulus encoding refers to training neurons to represent a particular feature of the sensory environment without external targets. Here, we encode a stimulus into the output neurons of a recurrent neural network (RNN) via information maximization.

2.3.1 Recurrent Neural Networks for Stimulus Encoding

Recurrent networks are a class of artificial neural networks that process data sequentially. These networks maintain an internal representation, known as the hidden state, which is recursively updated as the network receives inputs. For an Elman RNN [30] with a sequence of input data $\{v_1, \ldots, v_{n_x}\}$, the internal state is updated via

$$G_t = \sigma(W_{ih} v_t + W_{hh} G_{t-1}) \tag{2.9}$$

where G_t is the hidden state vector, W_{ih} and W_{hh} are the input and hidden layer weight matrices, respectively, and σ is the activation function, such as $ReLU(\cdot)$ (Figure 2.6). The output of the network at time t is given by

$$P_t = \sigma(W G_t) \tag{2.10}$$

For self-supervised stimulus encoding, our data comes in the form of alterations to the stimulus space. That is, for location in two dimensions, the input data will be velocities in two dimensions. For head direction, input data takes the form of one-dimensional angular velocities. The initial activations $P_0 \in \mathbb{R}^{n_p}$ are chosen by calculating distances from uniformly randomly sampled locations in the arena and applying a difference of softmaxes (Appendix C) and encoded by E.

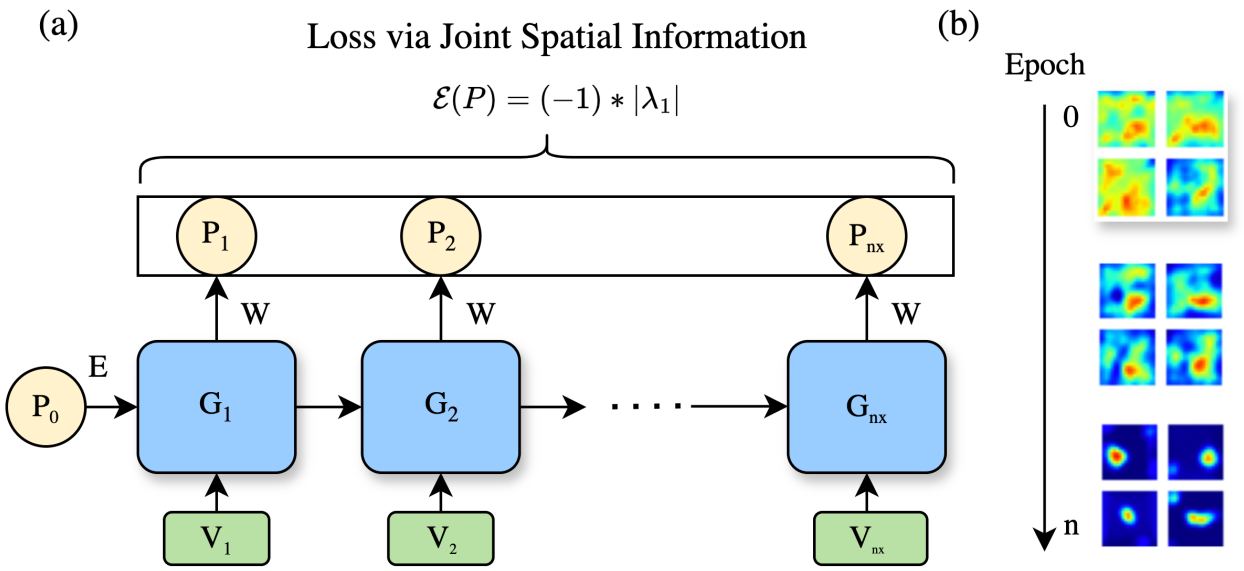

Figure 2.6: (a) Diagram of RNN architecture used to perform self-supervised learning. Shown here over n_x steps. (b) As training progresses over n epochs, the neuron's firing fields become more place cell-like. Here the firing fields of four neurons are shown.

We use the stimulus information matrix J (Definition 2.3), associated with neuron output activations $P = [P_1, \cdots, P_{n_x}]^T$ to analyze global information and calculate epoch loss (Figure 2.6 (a)). In detail, each column of P serves as a distribution of firing rates $\{\lambda(s)\}_{s \in S}$ across the trajectory, from which spectral-stimulus information can be calculated. To learn place or head direction cells via J, one could choose to optimize the entries of J or spectral-stimulus information, λ_1. These values are highly correlated (Remark A.3), and optimizing along either choice leads to similar results (Figure 1). For our experiments, we choose to focus on the optimization of the spectral-stimulus information. That is, to increase information efficiency, we minimize

$$\mathcal{E}_{spectral}(P) = (-1) * |\lambda_1|, \tag{2.11}$$

where λ_1 is the leading eigenvalue of J. The joint stimulus information rate, I_{spike}, depends on the firing rates of the neurons, P, and the occupancy distribution, which is the probability mass function across n_x steps along the trajectory. For simplicity, we assume this to

be a discrete uniform distribution for all trajectories in the stimulus domain. Additional experimental details, including hyperparameters, are listed in Appendix C.

As a baseline, we compare the results of maximizing spectral information (Equation (2.11)) with the results of maximizing single-order information according to Skaggs' stimulus information rate [90]. This latter loss function is given by,

$$\mathcal{E}_{skaggs}(P) = (-1) * \sum_{i=1}^{n_p} I_{spike}(P[:, i]), \tag{2.12}$$

where $P[:, i] \in \mathbb{R}^{n_x}$ is the ith column of P.

2.3.2 Place cell score

To compare place cells produced by optimizing along our spectral-spatial information measure with control experiments, we construct a **place cell score**. Our place cell score is an approximately equal combination of three characterizing features of place cells: (i) smoothness of the firing rate; (ii) binary firing rates across space; and (iii) sparse activations. Each of these factors is a key feature of *in vitro* place cells [14, 62]. The place cell score PC of a place cell A is then given by

$$\text{PC}_{\text{score}}(A, \epsilon) = 10 * \text{Binary}(A, \epsilon) - 100 * \text{Smoothness}(A) - 10 * \text{Sparsity}(A), \tag{2.13}$$

where ϵ is a small number and a hyperparameter of the system. To compute the smoothness, binary activity, and sparsity scores of a place cell, we measure the firing rates of the place cell P across the entire arena and normalize values between 0 and 1. Then, if $\Pi \in \mathbb{R}^{h \times h}$ is the binned matrix of normalized firing rates across a square arena from place cell P at a

spatial resolution h, we use the following metrics:

$$\text{Smoothness}(A) = \frac{1}{h^2}\sum_{i=1}^{h}\sum_{j=1}^{h}|\Pi_{i,j} - \tilde{\Pi}_{i,j}|, \tag{2.14}$$

$$\text{Binary}(A, \epsilon) = P(\Pi < \epsilon) + P(\Pi > 1 - \epsilon), \tag{2.15}$$

$$\text{Sparsity}(A) = \frac{(\sum_{i=1}^{h}\sum_{j=1}^{h}\Pi_{i,j})^2}{\sum_{i=1}^{h}\sum_{j=1}^{h}\Pi_{i,j}^2}, \tag{2.16}$$

where $\tilde{\Pi} \in \mathbb{R}^{h\times h}$ is the smoothed matrix of normalized firing rates from place cell A. For our purposes, $\tilde{\Pi}$ is acquired via Gaussian Blur with kernel size $(5, 5)$ and kernel standard deviations $\sigma_x = \sigma_y = 1$. We may also calculate the spatial information rate of the learned place cells across the entire arena via

$$I_{spike}(A) = \sum_{i=1}^{h}\sum_{j=1}^{h}\frac{\Pi_{i,j}}{\bar{\Pi}}\log_2\left(\frac{\Pi_{i,j}}{\bar{\Pi}}\right)p(\text{bin}_{i,j}), \tag{2.17}$$

where $\bar{\Pi} = \sum_{i=1}^{h}\sum_{j=1}^{h}p(\text{bin}_{i,j})\Pi_{i,j}$. For simplicity, we assume the agent is equally likely to be in any spatial bin, i.e. $p(\text{bin}_{i,j}) = 1/h^2$ for all i, j.

2.3.3 Neural Recordings

Head direction cells

Head direction cells encode the direction the head is facing at any moment in time, independent of body orientation. These cells are not one-to-one, meaning that a cell can fire in response to multiple head directions; thus, it takes multiple head direction cells firing synchronously to decode head direction from cell activity. We look at a subset of head direction cell spiking data collected by Peyrache et al. in [79]. Specifically, we analyze 20 minutes of spiking data collected from 'Mouse28-140313' during the foraging portion of the experiment. During this time, the mouse was allowed to roam around it's rectangular environment. Researchers recorded the mouse's ground truth head direction and spiking data from 22 head direction cells within the anterodorsal thalamic nucleus (ADn) binned into 100ms-wide bins.

Place cells

Place cells are neurons in the hippocampus that fire when an animal visits specific regions of its environment [73, 99]. We apply our information-theoretic measures to place cells recording collected by Hazama et al. in [46]. Spike-timestamps were recorded of 18 place cells located in the right hippocampal CA1 subfield of a male Japanese monkey performing a shuttling task on a rectangular (4×0.9 m) track. The monkey was required to shuttle back and forth on the track multiple times to get rewards.

2.4 Results

RNNs receiving velocity inputs can produce place cell-like responses via self-supervised learning with information theory. These responses can be achieved by training the model to maximize either the eigenvalue of the stimulus information matrix (Equation (2.11)), which we refer to as *spectral-spatial information*, or the sum of the Skaggs' spatial information rates (Equation (2.12)). Here, we compare the techniques by analyzing the learned models and their corresponding encoding of space.

2.4.1 Place cells arise via self-supervised learning

Training with spectral-spatial information significantly improves spatial encoding by enhancing the quantity and quality of place cells. Figure 2.7 (a) shows 64 learned place activations from a model of 128 neurons. The neurons learn to fire in small regions, which generally are non-overlapping. Figure 3 compares this model's 128 place cell activation maps to the activation maps of the model identical random initialization but trained with Skaggs' spatial information. In Figure 2.7 (b-c), we compare place cell scores of place cells learned via spectral-spatial information to those learned via Skaggs' spatial information.

Overall, neurons trained with spectral-spatial information yielded a greater median place cell score (0.971) than neurons trained with Skaggs (-0.656), while both improve upon untrained neurons (-13.28) (Figure 2.7 (b)). In a sensitivity analysis (Appendix F), we show that spectral-spatial information consistently outperforms Skaggs spatial information

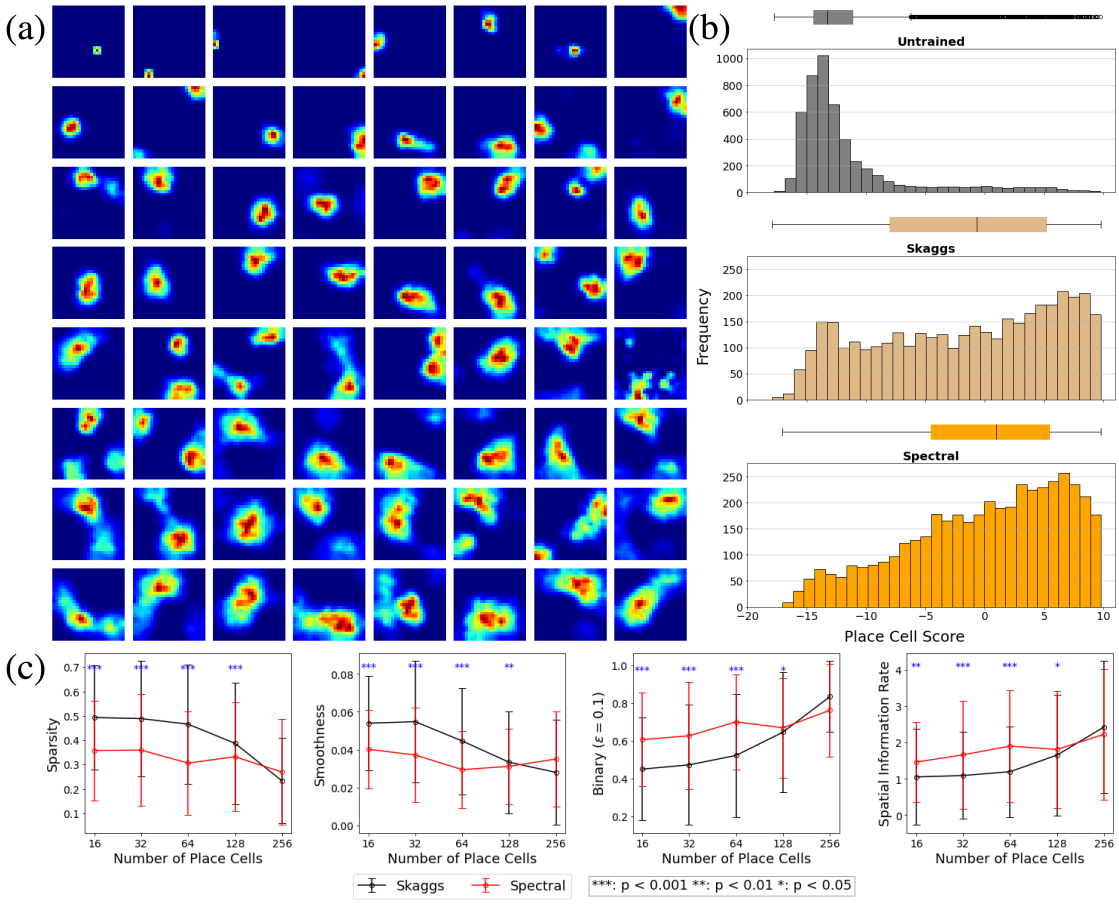

Figure 2.7: (a) Activation maps of top 64 learned place cells from a model trained with spectral-spatial information, normalized to be between 0 and 1, and sorted by place cell score (Equation (2.13)). (b) Original and trained place cell score box plots and histograms. Place cell scores are recorded across 50 total models for each loss function with varying place cell sizes. (c) Mean ($\pm$ standard deviation) sparsity, smoothness, binarity ($\epsilon = 0.1$), and spatial information rate (Equations (2.14- 2.17)) of trained place cells as the number of place cells increases.

in learning place-cell-like representations regardless of changes in training parameters (batch size, number of hidden units, etc.). These improvements underscore the effectiveness of including spectral-spatial information when optimizing neural representations of space. Place cell activation maps from all models are available in the supplementary materials.

It is worth noting, however, that while spectral-spatial information yields, on average, better spatial information, sparsity, binarity, and smoothness for models with 16, 32, 64, and 128 neurons, the opposite is true for models trained with 256 neurons (Figure 2.7 (c)). We hypothesize that as the number of neurons increases, spectral-spatial information is dominated by the Skaggs' spatial information of the neurons, thus yielding similar results.

2.4.2 Neural decoding

We consider two experiments to ensure the decoding abilities of the learned place cells. In each experiment, we attempt to decode the agent's location from binarized place cell activations, one if the neuron firing rate is nonzero, and zero otherwise. By turning the activations into binary values, spatial decoding must be done by relating a combination of active neurons to an area in the arena. Consequently, each decoding algorithm operates with the sole knowledge of which neurons are in an "active" state. The experiment is run for each of the 50 models, i.e. 10 times for a given number of place cells, to obtain an average score. We display the results in Figure 2.8.

The first decoding task is to decode the position in the arena using leave-one-out classification. In leave-one-out classification, one data point is reserved for testing and the remaining data points are used to train a Naive Bayes' Classifier (see [94] for details). This procedure is repeated multiple times so that each firing rate value is used once for testing. In Figure 2.8, we apply this using data from 10 trajectories (1000 positions) for each model. Neurons trained with spectral-spatial information can predict the position with high precision, mirroring the simulated trajectory in all areas of the arena (Figure 2.8 (a)). Furthermore, these neurons consistently outperform neurons trained with Skaggs' spatial information rate (Figure 2.8 (b)).

To generate Figure 2.8 (c - d), we train a support vector machine (SVM) with a linear kernel to predict which quadrant of the region the agent is in from the binarized neural

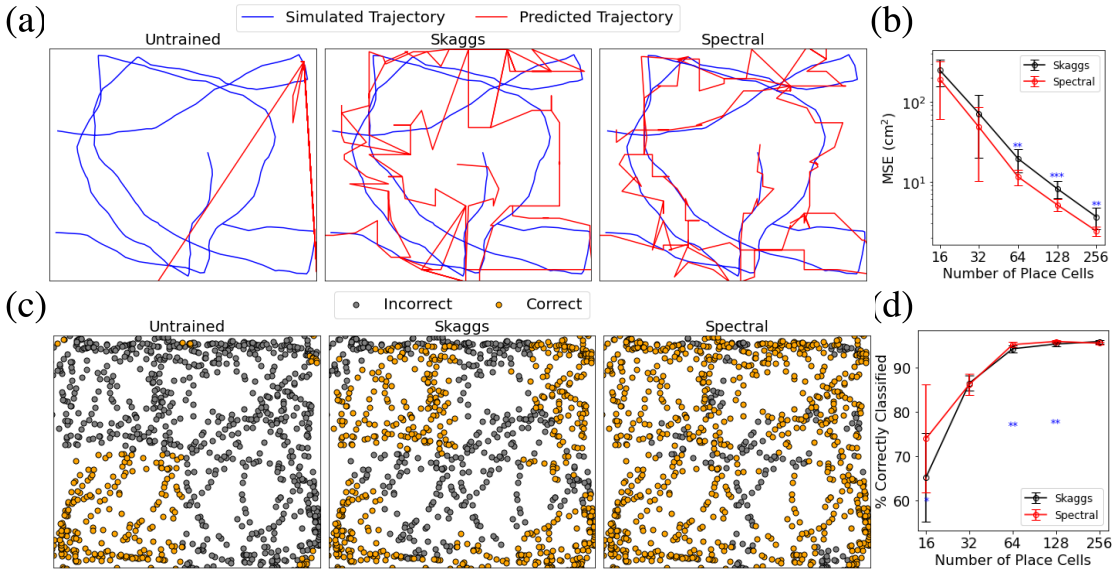

Figure 2.8: Comparison of decoding capabilities of binarized neurons before and after training. (a) Simulated and predicted trajectories leave-one-out classification with 128 neurons. (b) Skaggs vs spectral leave-one-out decoding mean square error (MSE) as the number of trained place cells increases. The mean (± standard deviation) of MSE across experiments is shown. (c) Correctly (orange) and incorrectly (grey) classified activations from a support vector machine (SVM) for quadrant classification using 16 neurons. (d) Skaggs vs spectral quadrant classification accuracy. The mean (± standard deviation) of accuracy across experiments is shown.

input. Spectral-spatial information outperforms Skaggs spatial information by nearly 10% when training just 16 neurons, and performs comparably for greater than 16 neurons.

These experiments demonstrate that the place cells learned via self-supervised learning with spectral-spatial information encode space better than neurons trained with Skaggs' spatial information. Combined with the results from Figure 2.7, we conclude that self-supervised learning with spectral-spatial information allows for precise spatial decoding with more place cell-like neural representations than previously attainable via Skaggs' spatial information.

2.4.3 Path invariance and uniformity

A key feature of any collection of neurons encoding spatial representation is its path invariance [88]. That is, two trajectories that begin and end at the same position should have identical final neural representations (Figure 2.9 (a)). Furthermore, the firing fields of place cells are known to create a sparse representation of space, with consistent activity responses in the absence of additional stimuli [86], a property we refer to as uniformity.

To test for path invariance of our place cell representations, we form two trajectories by randomly shuffling a sequence of velocity vectors. We then compare the place cell activity by measuring the Euclidean distance of the discrepancy (Equation (2.18)) between the two trajectories: if $P_{n_x}, P_{\text{shuffled}\,n_x} \in \mathbb{R}^{Np}$ are re-scaled activations at the final position, then the path invariance discrepancy D is

$$D = ||P_{n_x} - P_{\text{shuffled}\,n_x}||_2^2. \tag{2.18}$$

We measure D for 1000 pairs of trajectories for each of the 50 models, and display results in Figure 2.9 (b).

Our models trained with spatial information demonstrate exceptional performance (Figure 2.9 (b)). In detail, the models achieved an average ($\pm$ standard deviation) path invariance discrepancy of 0.004 ± 0.024 for Skaggs' optimization and 0.022 ± 0.066 for spectral optimization, while untrained models had a path invariance discrepancy of 1.024 ± 0.859. We view path invariance as a binary property that is present in both models trained

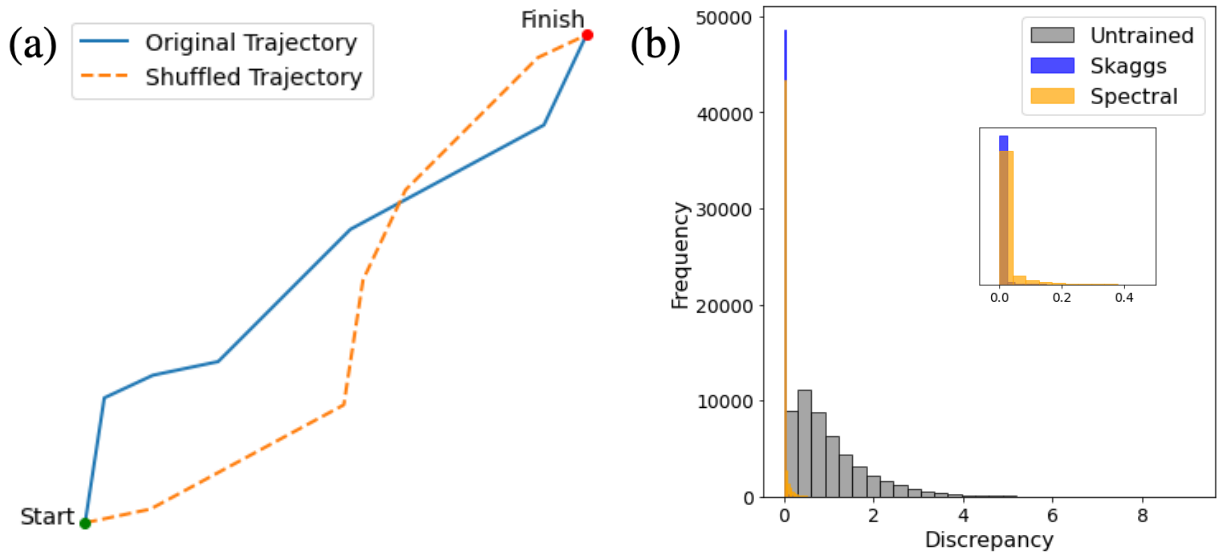

Figure 2.9: (a) Two trajectories formed by shuffling the order of velocities. Since both trajectories end at the same location, neural representations at the final position should be identical despite different paths to get there. (b) Histogram of path invariance discrepancy across 1000 trajectories from all untrained models, models trained with Skaggs' spatial information, and models trained with spectral-spatial information.

with Skaggs' and spectral optimization. Thus, the difference in average path invariance discrepancy between Skaggs' and spectral-optimized models is negligible. Altogether, path invariance further indicates that self-supervised learning combined with spatial information optimization results in models that successfully encode space.

To measure the uniformity of spatial representations, we calculate the standard deviation in the number of highly active place cells at each discretized spatial region. A lower standard deviation suggests that the place cells have become more uniform as their firing fields disperse evenly across the domain. Place cells learn anti-correlated firing only by training with spectral-spatial information (Equation (2.11)), as shown in Figure 2.10 (a). Training with the Skaggs loss function optimizes each cell's spatial information rate (Equation (2.12)) and is agnostic toward the firing of the other cells. In Figure 2.10 (b), we show that place cells trained with spectral-spatial information result in greater uniformity than those trained with Skaggs' spatial information.

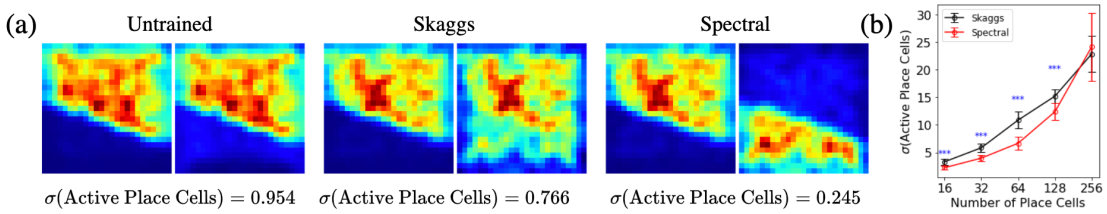

Figure 2.10: (a) Comparison of two place cell firing fields in which the initial cells fire in the same region. Standard deviation in active place cells decreases as firing fields separate. (b) Mean ($\pm$ standard deviation) of the standard deviation in the number of place cells with high activity (defined as exceeding 20% of the peak firing rate of the rate map) as the number of trained place cells varies.

2.4.4 Self-Supervised Head Direction Cell Learning

Using techniques described in subsection 2.3, we may also train RNNs to encode direction without external targets. Here, the input to the RNN is given by a sequence of turns $\{v_1, \dots v_{n_x}\}$ where $v_i \in \mathbb{R}^1$ and an initial head direction. Now, similar to what was done for position encoding, we create the *directional* information matrix via the RNN output and calculate epoch loss. Again, we are optimizing the spectral-directional information λ_1 or the Skaggs' directional information. While neurons before training fire sporadically, without preferred directions, after training, neurons resemble head-direction cells by firing only when the agent is facing a particular direction (Figure 2.11 (a)). Furthermore, the neurons often fire in non-overlapping directions, as predicted by Theorem 2.2.10.

Contrary to place cell learning, spectral-directional information shows no statistically significant improvement over Skaggs' directional information for head direction cell learning (Figure 2.11 (b)). Nonetheless, we demonstrate that information theory serves as a valuable tool for encoding stimuli through self-supervised learning.

2.4.5 Eigenvector Relative Importance

Corollary 2.2.9 and Theorem 2.2.10 suggest that information is ideally encoded equally among all neurons. Here, all entries of the leading eigenvector of the stimulus information matrix are identical. This perfect balance is rarely the case when analyzing recordings of biological neurons such as head direction or place cells. Instead, the leading eigenvector

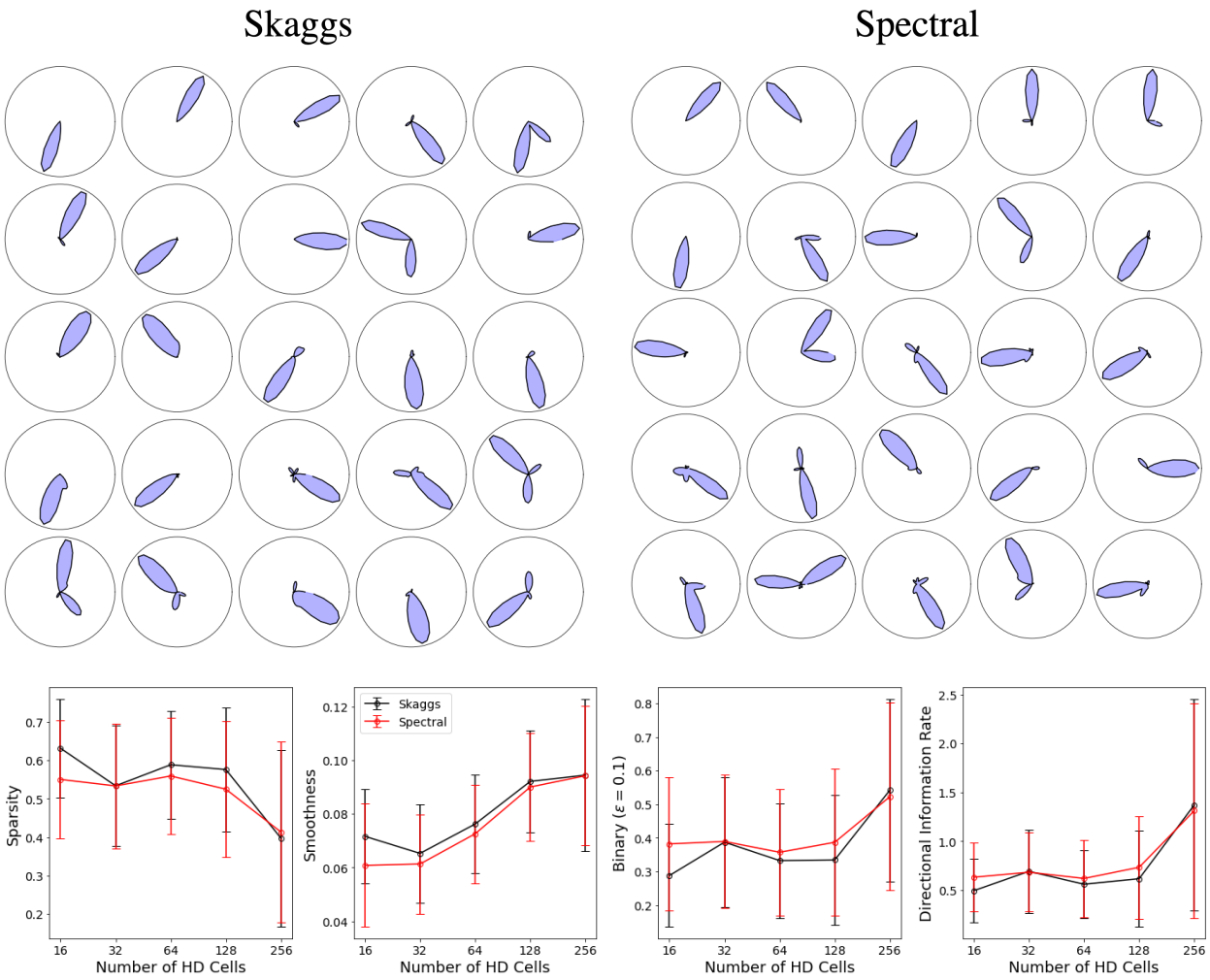

Figure 2.11: (a) Top 25 (sorted by directional information rate) Skaggs and Spectral-directional information rate trained neurons from models with 256 output neurons. (b). Mean ($\pm$ standard deviation) sparsity, smoothness, binarity ($\epsilon = 0.1$), and directional information rate (Equations 2.14- 2.17) of trained head direction cells as the number of head direction cells increases.

entries will vary in magnitude. This serves as a feature, by telling us something about the underlying system. In particular, neurons corresponding to larger entries of the leading eigenvector play a larger role in encoding. As shown in Figure 2.12, the stimulus information rate of a neuron correlates with its corresponding entry of the leading eigenvector of the stimulus information matrix. This phenomenon is also demonstrated in Figure 1 (b).

Given this geometric interpretation, we introduce the idea of *eigen-pruning*—or using this eigenvector to decide which neurons are relatively important stimuli for decoding. Should eigen-pruning be successful, we could remove neurons with the smallest eigenvector entries and retain some level of stimulus decoding capabilities. We tested this using a simple three-layer feed-forward neural network (FFNN), which is trained to predict head direction from head direction cell spike data from [79]. The average median absolute error (MAE) is recorded across 5 experiments for each number of input neurons. The MAE is given by

$$MAE = \underset{n=1,2,\ldots,N}{median} |rescale(\theta_{dec}(n) - \theta_{true}(n))| \tag{2.19}$$

where N is the number of predictions, and $\theta_{dec}, \theta_{true}$ are the decoded and true head directions, and the mapping rescale accounts for the ring structure of HD [65].

Eigen-pruning results in better decoding performance over randomly removing neurons when removing less than $\approx 65\%$ of the neurons (Figure 2.13). As shown in Figure 2.13, the FFNN is able to roughly maintain decoding accuracy (12-16 degrees) until it is given inputs from just five of the original twenty-two head direction cells. These five head direction cells are vital for effective decoding, and removing them yields a drop in decoding accuracy. This demonstrates that eigen-pruning gives greater predictability of decoding performance as opposed to randomly pruning the input.

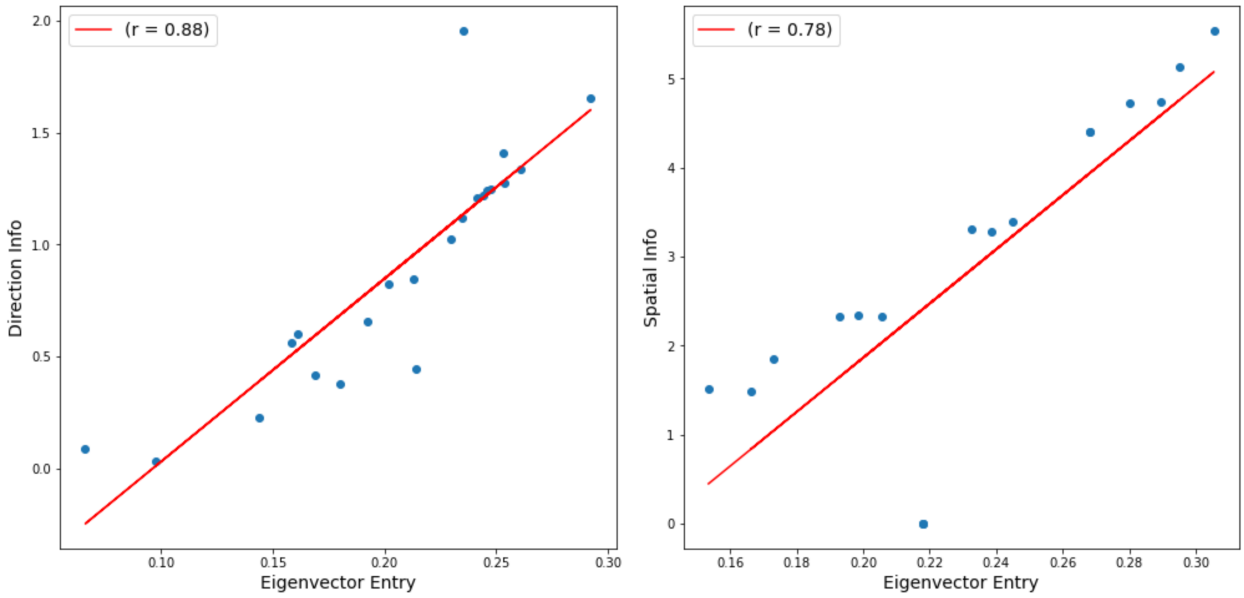

Figure 2.12: Directional (left) and spatial (right) information versus eigenvector entry magnitude from head direction cell recordings done in [79] and place cell recordings done in [46].

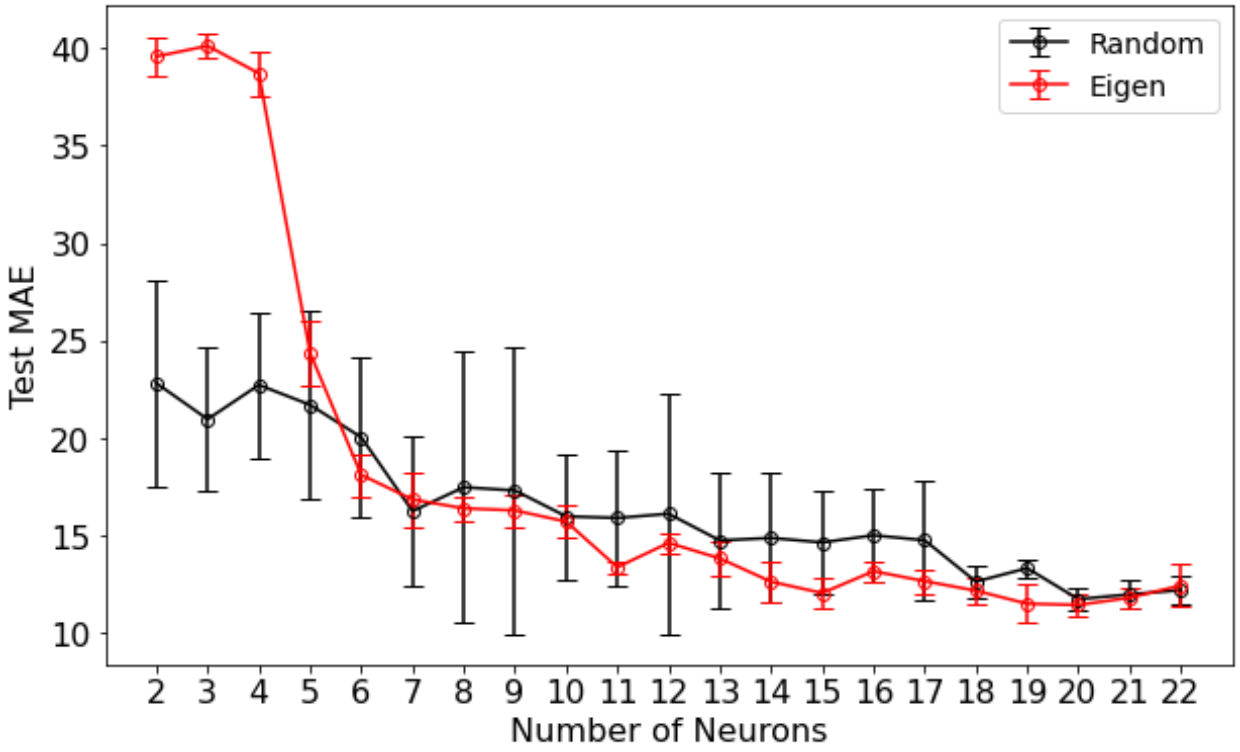

Figure 2.13: Mean ($\pm$ standard deviation) test median absolute error(Eqaution (2.19)) versus number of neurons used for head direction decoding. Neurons are either removed randomly or via the magnitude of the stimulus information matrix's leading eigenvector entry. Removing neurons via the latter technique yields greater predictability in decoding performance and superior performance when removing less than $\approx 65\%$ of the neurons.

Chapter 3

Dynamic Drug Decoding with Functional Ultrasound Imaging

3.1 Materials and methods

3.1.1 Animals

Twenty-three (23) 8-12 weeks-old male mice were utilized in this study (C57BL/6, Charles River Laboratories, Hollister, CA). All mice were group-housed, fed *ad libitum*, and maintained at a regular light-dark cycle of 12 hours. The animals were divided into two groups: MK-801 drug group (n=10) and saline vehicle control group (n=13).

3.1.2 Surgical Procedures

The mice were anesthetized using 5% isoflurane solution delivered in a mixture of oxygen and nitrous oxide (1:2 ratio) and then maintained at a constant isoflurane concentration of 1.5 - 2 % throughout the experiment. Body temperature was kept stable using an electric warming pad. The animals were head-fixed in a stereotaxic frame with ear bars to minimize head movement and reduce motion artifacts. A commercially available depilatory cream (Nair, Pharmapacks) was utilized to remove hair from the scalp followed by application of an echographic ultrasonic gel on the intact scalp-skin to enhance acoustic coupling during fUSI signal acquisition. All the experimental and surgical protocols were approved by the

Institutional Animal Care and Use Committee of University of Southern California (IACUC #21006).

3.1.3 Data Acquisition

Transcranial power Doppler (pD) images were obtained using the Iconeus One scanner (Iconeus, Paris, France). A 128-element linear array transducer probe with a 15.6 MHz center frequency and 0.1 mm pitch was placed on intact mice skull. The probe was fixed in a motorized system during the course of the experiment (Figure 3.1A). Prior to recording, the target sagittal plane for imaging was determined by performing a 3-D whole-brain fUSI fast scan for each animal. The plane was then aligned with the standard Allen Mouse Common Coordinate Framework brain atlas using the specialized software provided with the Iconeus One system[102]. The details of fUSI parameters are described in [21]. Briefly, each image was constructed from 200 compounded frames, acquired at 500 Hz using 11 tilted plane waves ($-10°$ to $10°$, $2°$ increments). The pulse repetition frequency was 5.5 kHz, with continuous acquisition of 400 ms blocks of compounded images, separated by 600 ms intervals. This approach enables pD image acquisition with in-plane spatial resolution of 100 µm × 100 µm, slice thickness of 400 µm, FOV of 12.8 mm in width and 9.1 mm in depth, and an overall image-frame production rate of 1Hz (Figure 3.1B). The fUSI data acquisition protocol consisted of 5 minutes of baseline recording, followed by intraperitoneal (i.p.) injection of either 0.2 ml of saline or MK-801 (1.5 mg/kg) at the 5 minutes mark, followed by an additional 55 minutes of recording, post-saline/MK-801 injection (Figure 3.1C).

3.1.4 Data Pre-processing and Registration

We implemented the NoRMCorre motion correction method [80], combined with our in-house algorithms for filtering breathing and high-frequency signal to reduce noise. Specifically, we utilized a lowpass filter featuring a normalized passband frequency of 0.02 Hz and a stopband attenuation of 60 dB, which also offsets any delay caused by the filtering process, thereby eliminating high-frequency fluctuations.

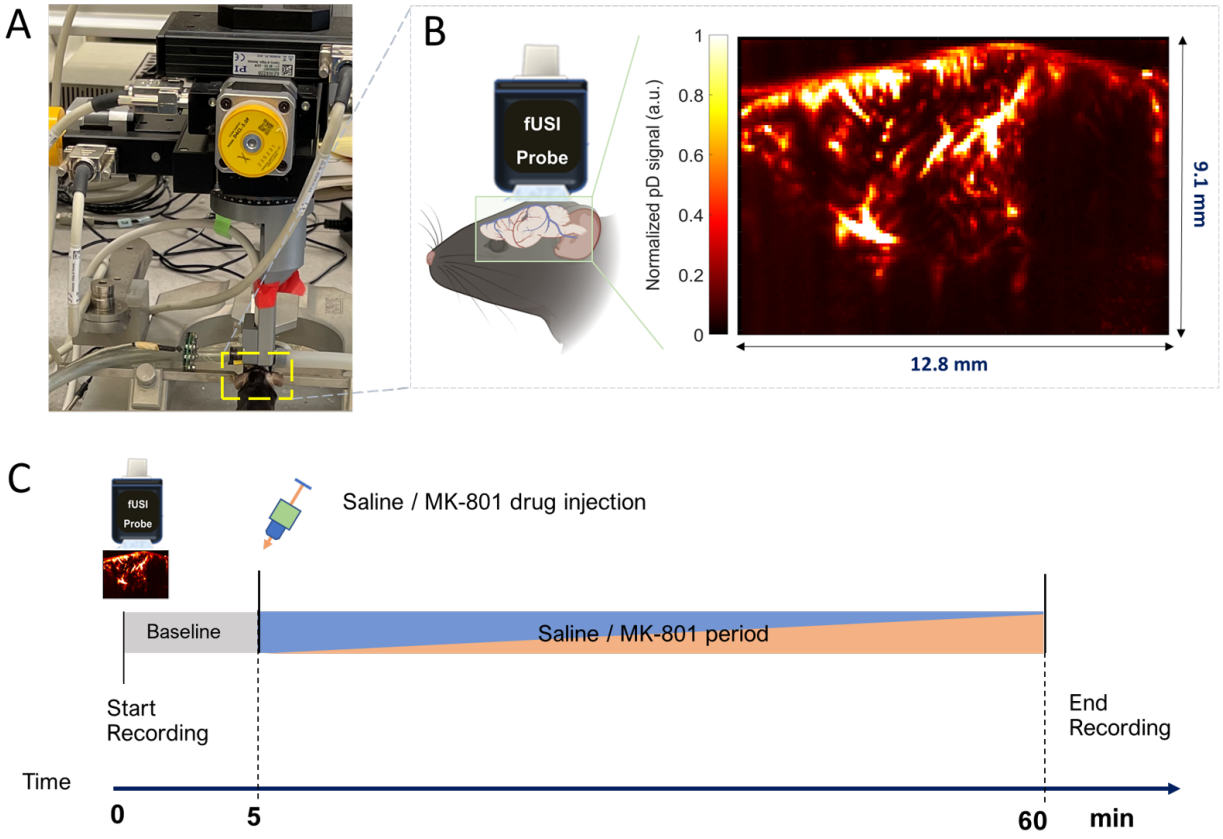

Figure 3.1: Experimental setup and fUSI acquisition. (A) fUSI motorized system setup for signal acquisition. (B) Power Doppler-based 2D vascular map image of a mouse brain in a sagittal plane. (C) Diagram of the experimental protocol for the 60 min of continuous fUSI acquisition. After 5 min, saline or 1.5 mg/kg MK-801 was intraperitoneally injected to the animals.

Furthermore, we registered all the images to a common reference image-frame to ensure that the CNN does not rely on the varying brain structures of different mice for image classification We first identified the reference animal based on the imaging quality-i.e., we selected an animal with clear sub-cortical structures as the reference one. Then we computed the mean image of the first ten images beginning from the third minute of the baseline period to create a stable fixed image (reference image), in order to mitigate any potential variability that might arise from using a single image as a reference. We then employed the *Imregdeform* algorithm, which is part of the Matlab Image Processing Toolbox (since 2022b). This algorithm utilizes a parametric approach for non-rigid image registration with total variation regularization. All images were aligned to the reference frame using the *Imregdeform* algorithm, which aids in simplifying subsequent data analysis.The default parameters were used for *Imregdeform*.

3.1.5 Models and Feature selection

Implementation and Parameter Selection

The SVMs were implemented using scikit-learn [78], while CNNs and Vision Transformers were implemented in PyTorch [77]. The majority of the code was run using a 2.8 GHz Quad-Core Intel Core i7 processor with 16 GB of memory. Parameter sweeps were run using a Tesla K80 GPU with 64GB of memory. Parameter sweep was performed using Weights and Biases [10] grid sweep. After the grid sweep, we subjected the 15 best hyperparameter combinations to 10-fold cross-validation, from which the best model (one that has the highest peak accuracy) was chosen.

Convolutional Neural Network

The CNN progressively reduces the spatial dimensions of the image input while increasing the depth of feature maps through multiple convolutional layers (Figure 3.2).To ensure compatibility with class activation mapping, the architecture includes a global average pooling (GAP) layer followed by a single linear layer, which outputs two features corresponding to the binary classification task. We conducted extensive parameter sweeps to determine the

optimal architecture and hyperparameters (Table 3.1, best-performing values in bold). The final architecture consists of four convolutional layers with 32 filters each, employing the $ELU(\cdot)$ activation function.The $ELU(\cdot)$ activation function, which was selected over ReLU and LeakyReLU alternatives, is defined by:

$$ELU(x) = \begin{cases} x & \text{if } x > 0 \\ \exp(x) - 1 & \text{if } x \leq 0 \end{cases} \tag{3.1}$$

This activation function is known to speed up learning and lead to higher classification accuracies by pushing the network's mean activation toward zero [19]. The model is trained using the Adam optimizer with a learning rate of 1e-3 and batch size of 32.

Class Activation Mapping

Class activation mapping was performed using XGRAD-CAM [57] with a sliding time window of one minute. The localization map for a given input is computed as follows

$$L^{(c)}(x, y) = ReLU\left(\sum_k w_k^{(c)} A_k(x, y)\right) \tag{3.2}$$

where $ReLU(x) = \max(0, x)$ and the coefficient $w_k^{(c)}$ being defined as

$$w_k^{(c)} = \sum_{i=1}^{H}\sum_{j=1}^{W}\left(\frac{\partial Y^{(c)}}{\partial A_k(i,j)} \cdot \frac{A_k(i,j)}{\sum_{m=1}^{H}\sum_{n=1}^{W} A_k(m,n)}\right) \tag{3.3}$$

where $A_k(x, y)$ is the activation of node k in the target layer of the model at position (x, y) and $Y^{(c)}$ is the model output score for class c before activation. Here, H and W are the height and width of the feature maps, respectively.

The target layer for this analysis was chosen to be the final convolutional layer, which contains higher-level feature representations and is commonly chosen for CAM [110].

The weights $w_k^{(c)}$ found in Equation (3.3) were computed using the product of the gradient of the model's raw score for the class (c) with respect to each activation and a normalization term. This encodes the importance of each activation in the final convolutional layer

Table 3.1: Hyperparameters used in the CNN parameter sweep.

Hyperparameter	**Value**
Batch size	**32**, 64
Learning rate	1e-4, **1e-3**, 1e-2
CNN nonlinearity	ReLU$(\cdot)$, **ELU**$(\cdot)$, LeakyReLU$(\cdot)$
Number of Filters	**32**, 64
Number of Convolutional Layers	2, 3, **4**
Optimizer	Adam [52]
Visualization Technique	XGRAD-CAM [57]
Loss Function	Cross-entropy

regarding the final classification decision for class (c). In our case, the final convolutional layer produces 128 feature maps with height, $H = 16$, and width $W = 12$.

As described in Equation (3.2), after calculating the weights for the given class and node, the weighted average was computed and the $ReLU(\cdot)$ activation function was applied to obtain the activation at each region. The result is a 12×16 activation map, corresponding to the dimensions of the feature maps from the final convolutional layer. To obtain a better picture for comparison with the initial image, the 12×16 image is up-sampled to the original image size of 91×128 via bilinear interpolation.

We generated class activation maps to identify brain regions that were important for detecting MK-801 effects. The up-sampled activation maps were averaged across all MK-801 samples to obtain a summary map. After normalizing the map to have values between 0 and 1, we applied an experimentally selected threshold, θ, to identify important regions for classification. The normalized class activation map CAM at each point (x, y) was computed as

$$CAM(x,y) = \begin{cases} L^{MK}(x,y) & \text{if } L^{MK}(x,y) \geq \theta \\ 0 & \text{if } L^{MK}(x,y) < \theta \end{cases} \tag{3.4}$$

where $L^{MK}(x, y)$ represents the activation map for the MK-801 condition. These maps highlight pixels that are most influential in identifying drug-induced changes in brain activity.

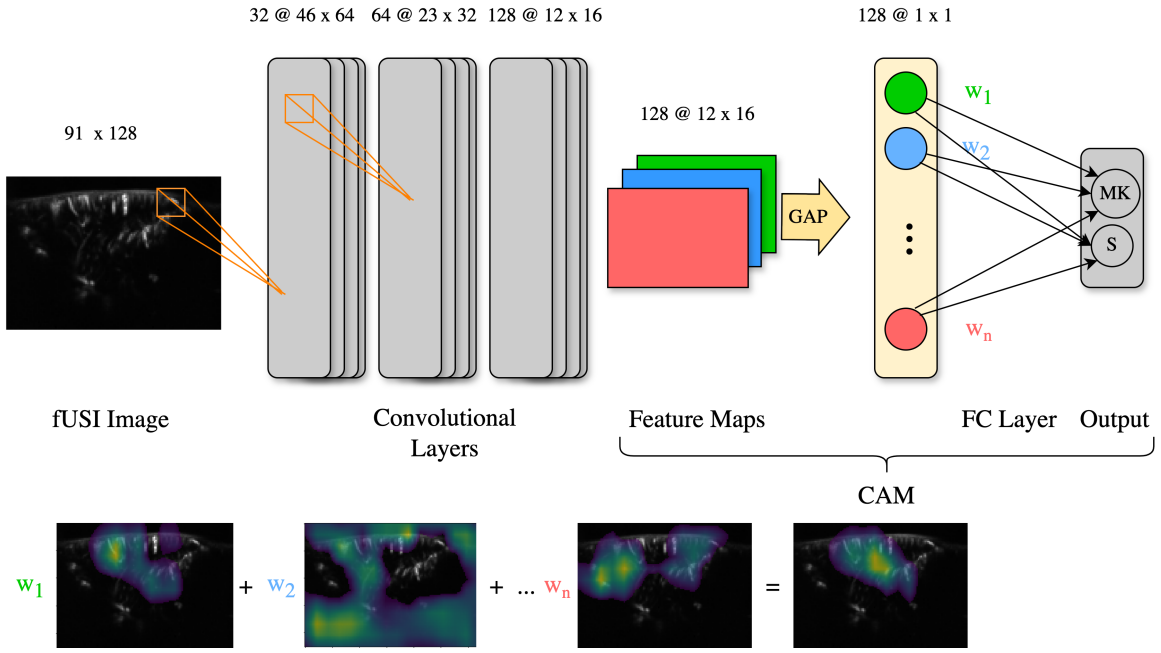

Figure 3.2: Convolutional Neural Network (CNN) architecture and class activation mapping (CAM), shown here for MK-801/Saline classification.

Vision Transformer

Vision Transformers (ViT) are deep learning models designed for computer vision tasks [4]. Unlike CNNs, ViTs process images by first splitting them into fixed-size patches and then applying a transformer architecture to the sequence of these patches. The architecture details of our ViT implementation are determined through parameter sweeps (Table 3.2, best-performing values in bold). The final model uses 16 transformer layers with 4 attention heads and an embedding dimension of 256, trained with a batch size of 16 and learning rate of 1e-4.

To visualize the regions important for classification, we employed Attention Rollout [1] with max fusion and a discard ratio of 0.9. For an in-depth description of attention rollout, see [1]. Briefly, attention rollout recursively computes the token attentions in each layer of a trained transformer model, resulting in an *attention map*, analogous to class activation maps in CNNs.

Table 3.2: Hyperparameters used in the Vision Transformer parameter sweep.

Hyperparameter	Value
Batch size	**16**, 32
Learning rate	1e-5, **1e-4**, 1e-3
Depth	4, 8, **16**
Number of Heads	2, **4**, 8
Embedding Dimension	128, **256**, 512
Patch size	(7, 8)
Kernel size	(7, 8)
Optimizer	Adam [52]
Visualization Technique	Attention-Rollout [1]
Loss Function	Cross-entropy

Support Vector Machine

Support Vector Machines (SVMs) are supervised learning models that perform classification by finding an optimal hyperplane to separate data points into different classes. To provide a baseline comparison with deep learning approaches, we implemented a linear SVM classifier that directly processes the registered fUSI data without any predefined regions of interest. SVMs assign a weight to each feature used to fit the model. In our implementation, each pixel in the fUSI images serves as an individual feature for the SVM, allowing for unbiased feature selection across the entire imaging field.

3.1.6 Training and Testing Data Sets

All MK-801 vs. saline classification task models were subjected to 10-fold cross-validation for robust performance assessment. In each cross-validation fold, data from 5 out of 10 MK-801 injected animals and 7 out of 13 saline-injected animals was randomly selected for training, while the remaining data was held out for testing.

This near 50-50 training/testing was chosen to ensure that a substantial number of animals were allocated to the test set for each cross-validation fold, which is particularly important for generating reliable class activation maps, feature weights visualizations, and attention maps. For each cross-validation fold, these visualizations are generated via averaging single-animal maps from the given testing set. While increasing the training set size

could improve classification accuracy, it would also reduce the number of animals available for generating these visualizations, potentially biasing the results. Specifically, with fewer test subjects, the class activation maps/attention maps would be averaged over a smaller sample, making them less representative and more susceptible to individual variability. For smaller datasets, a more significant portion of data may need to be allocated to ensure the models learn effectively.

CNN and ViT models were trained on fUSI data from the final five minutes of post-injection recordings since our previous study shows that the effects of MK-801 on brain activity become more potent over time [21]. Testing was conducted on the entire post-injection period using a one-minute sliding window.

For SVM, we trained and tested a separate model on each successive one-minute interval to capture temporal variations in feature importance. This is done to compare the resuliting *feature weight visualization* with the activation maps and attention maps from the CNN and ViT, which are dynamic despite static training data. For a given SVM, the weights for each feature depend only on data used to *train* the model. Therefore, to measure temporal variations in feature performance we must train a novel model for each one-minute interval. Each model generates a feature weight map corresponding to the SVM trained on the given one-minute interval. The feature maps are concatenated to form the feature weight visualization. The same train-test split as done for CNN and ViT is used to fit each SVM, with the only difference an SVM is trained and tested for each minute, as opposed to the CNN or ViT approach, in which a single model is trained on the final five minutes.

3.1.7 Model performance

Model performance was assessed using 10-fold cross-validation. For each iteration, the model was trained on the designated training set and tested on the remaining test set, and a sliding window of one minute was utilized to analyze the model's performance over time. The adequacy of the model was evaluated using the following metrics:

- Accuracy: (TP + TN)/ (P + N).
- Precision: TP / (TP + FP).

- Recall: TPR = TP/P.
- Area Under Curve (AUC): Area under the receiver operating characteristic (ROC), which is the plot of the true positive rate (TPR) against the false positive rate (FPR). FPR is defined as FN/P.
- F1-Score: Harmonic mean of precision and recall: 2TP/(2TP + FP + FN).

Where P and N represent the number of positive and negative instances, respectively, and TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives. To reduce potential biases from individual training/testing set selections, we averaged evaluation metrics in each temporal window across all cross-validation iterations.

3.1.8 Statistical Analysis

Repeated measures ANOVA was performed to assess the significance of CBV changes between selected and non-selected regions for each model (CNN, SVM, and ViT), with Region (selected vs. non-selected) and Time as factors. For all analyses, statistical significance was set at $p < 0.05$. The analyses examined both main effects and region $\times$ time interactions to evaluate both overall differences between regions and how these differences evolved over time. Statistical analyses were conducted using Matlab R2020b.

3.2 Results

3.2.1 Comparative model performance in detecting MK-801 effects

To investigate the spatiotemporal effects of MK-801 on brain hemodynamics, we developed an analysis pipeline combining functional ultrasound imaging with machine learning approaches (Figure 3.3. Power Doppler (pD) images were recorded from anesthetized mice receiving either MK-801 or saline injections. Our pipeline consisted of three main stages: data acquisition and preprocessing, model development and training, and performance

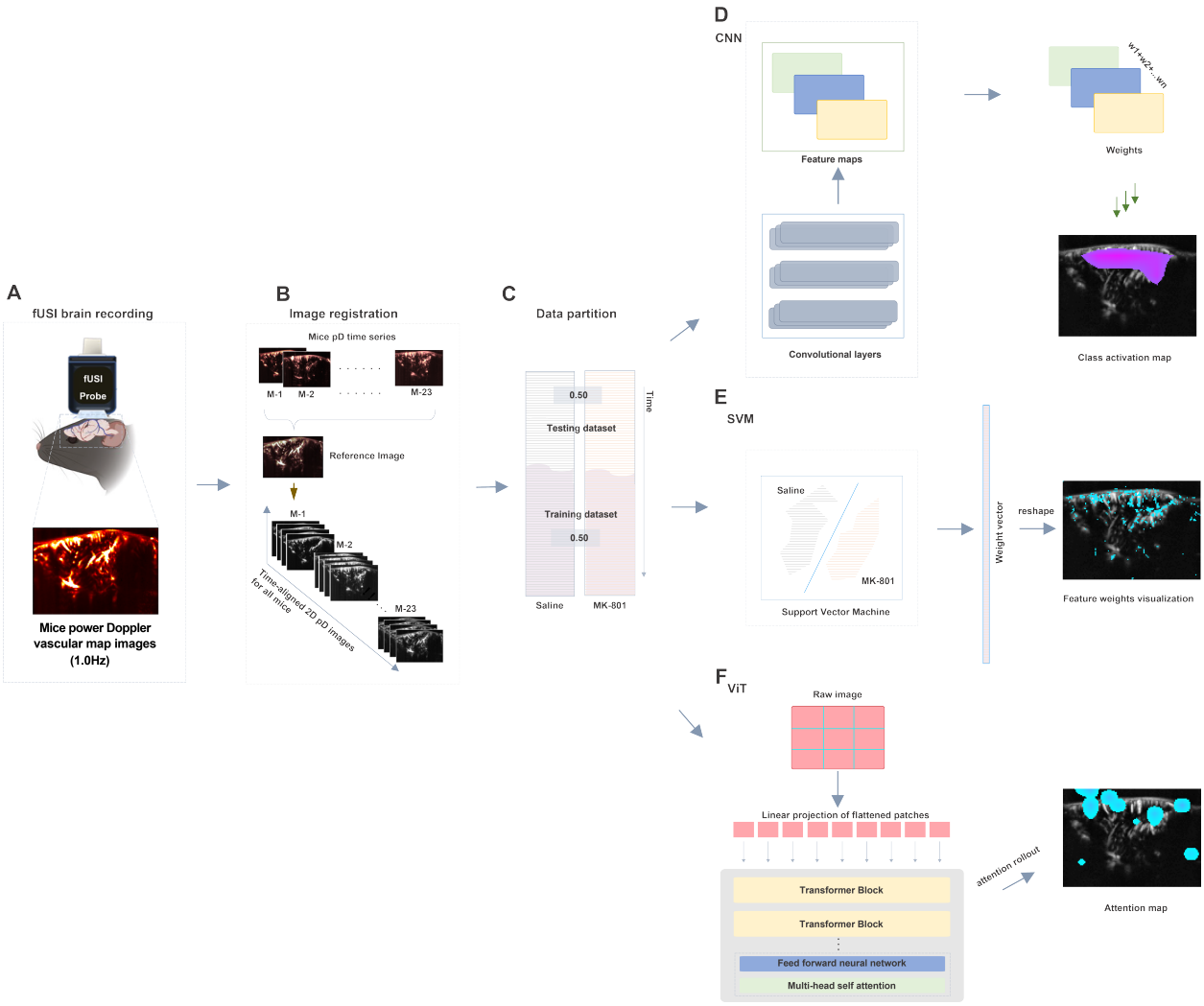

Figure 3.3: Flowchart of analysis pipeline. (A) fUSI mice brain signal acquisition. (B) Mice 2D power Doppler (pD) image time series alignment to a common reference image. (C) Data partitioning for CNN. (D) CNN and feature selection. (E) Class activation mapping.

evaluation (Figure 3.3A-C). In the preprocessing stage, we implemented an advanced non-rigid image registration method *Imregdeform* with the default parameters. To establish a common reference frame, we first selected a mouse with high quality vascular maps to serve as reference mouse. We then created a stable reference image by averaging ten consecutive frames from the second minute of baseline recording from this animal. All images were subsequently aligned to this reference frame, ensuring the models did not rely on varying brain anatomical structures across animals for classification.

The CNN and ViT architectures underwent extensive parameter sweep through grid search followed by cross-validation of the top performing configurations (Tables 3.1, 3.2, while a linear SVM provided a baseline comparison. We trained each model using pD images from the final five minutes of recording (55-60 minutes post-injection) when the effects of MK-801 should be most pronounced, then evaluated their performance across successive one-minute intervals spanning the entire 60-minute experiment. We assessed each model's ability to distinguish the effect of MK-801 over time (Figure 3.4. All three models showed dynamic classification performance that evolved with the progression of MK-801 induced effects. Consistent with our hypothesis that classification accuracy would improve as drug effects developed, at the time of injection (minute 5), all models performed near chance level (CNN: 55.9 ± 1.4%, ViT: 55.7 ± 2.4%, SVM: 52.8 ± 2.2 %, mean ± SD). As MK-801 gradually took effect, both the CNN and ViT demonstrated robust performance improvements, with CNN reaching a peak accuracy of 79.2 ± 6.2% at 32 minutes post-injection, and ViT achieving a comparable peak of 79.5 ± 5.8 % at 37 minutes post-injection. The SVM showed more modest performance, reaching a maximum accuracy of 74.2 ± 12.8 % at minute 35 post-injection. Additional model performance metrics including precision, recall, F1-score, and ROC-AUC are presented in Figure s1.

3.2.2 CNN captures anatomically relevant MK-801 induced brain hemodynamic responses

The class activation maps for the CNN (generated using XGRAD-CAM) revealed consistent activation patterns centered on cortical regions and extending into specific subcortical structures, notably the hippocampus and midbrain(Figure 3.5A). These activation patterns provides and unbiased view of the brain regions most important for distinguishing drug effects from control (saline injection) conditions.

To validate the biological relevance of these CNN-identified regions, we quantified their hemodynamic responses using cerebral blood volume (CBV) measurements. CBV changes were computed as the percent change in power Doppler signal ($\%pD$) relative to baseline, which was defined as the mean pD signal intensity during minutes 3-5 of the pre-injection

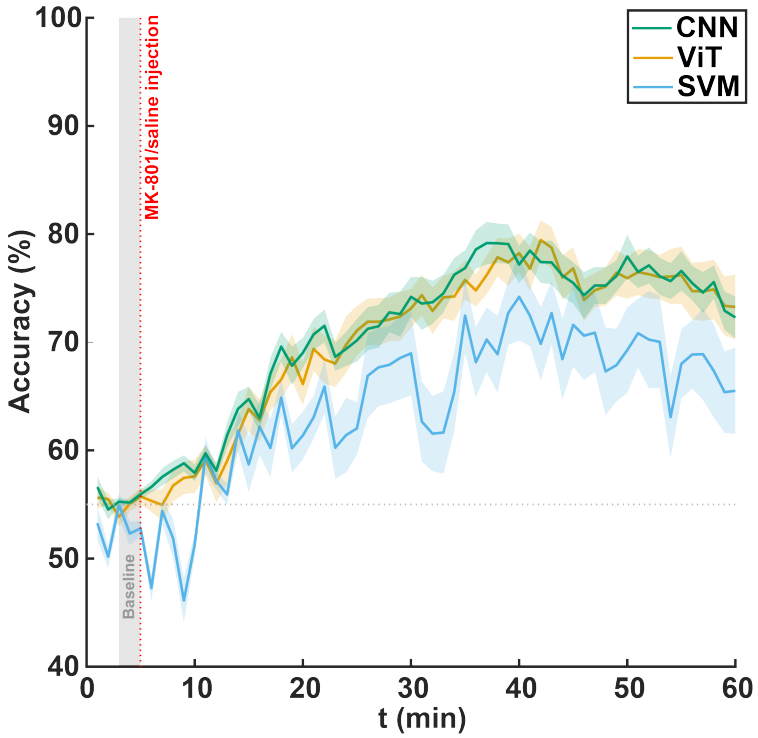

Figure 3.4: Model classification accuracy for distinguishing MK-801 from saline groups using ViT, SVM, and CNN. Shaded areas represent standard error derived from averaging across animals. The vertical dotted line at 5 min marks the injection time, and the gray shaded area (minutes 3-5) indicates the baseline period.

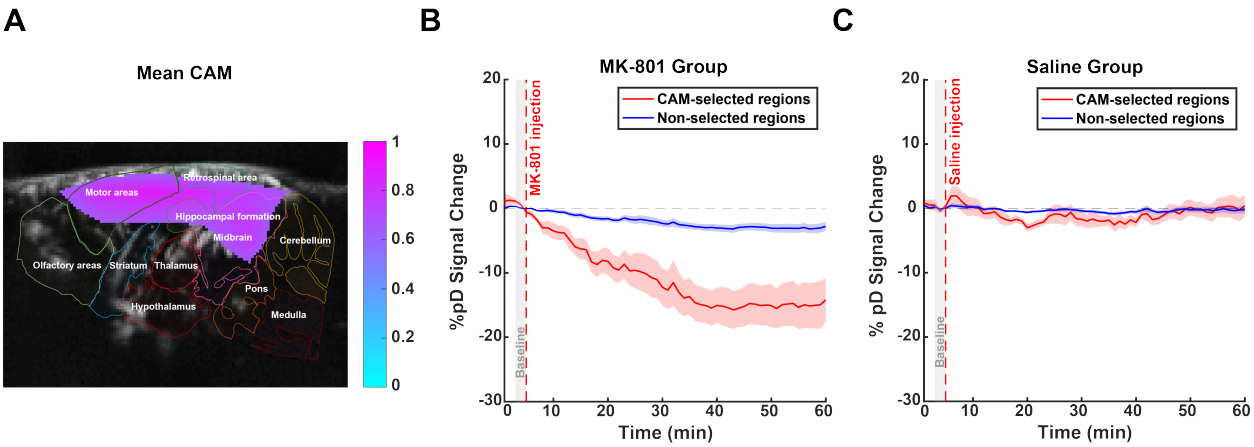

Figure 3.5: (A) Mean CAM overlay on the reference image, highlighting regions of importance in CNN-based classification of MK-801 effects (threshold $\theta = 0.65$). Brighter regions indicate higher relevance. (B-C) Time series of $\%pD$ (i.e., CBV) changes in CAM-selected regions (red curves) compared to non-selected control regions (blue curves) for (B) MK-801 and (C) Saline groups. The shaded areas represent standard error derived from averaging across animals. The red dashed line at 5 min marks the injection time. A gray patch (3-5 min) indicates the pre-injection baseline window.

period. This specific baseline window was chosen to ensure signal stability prior to drug injection. Notably, in MK-801 injected mice, CNN-selected brain regions exhibited a progressive decrease in CBV, reaching -14.21 $\pm$ 10.75% (mean $\pm$ SD) from baseline at 55 minutes post-injection, compared to -2.79 $\pm$ 2.18%. A repeated measures ANOVA confirmed a significant main effect of Region ($F_{(1,9)} = 20.62$, $p < 0.01$) and a Region $\times$ Time interaction ($F_{(1,9)} = 20.44$, $p < 0.01$), indicating that the CBV reduction was more pronounced in CNN-selected regions and evolved over time (Figure 3.5B). On the other hand, analysis of the same regions in saline-injected animals showed minimal changes at 60 minutes post-injection (0.41 $\pm$ 6.33% in selected regions vs -0.21 $\pm$ 2.08% in non-selected regions) with no significant CBV differences between CNN-selected and non-selected regions ($F_{(1,12)} = 0.94$, $p = 0.35$), nor a significant Region $\times$ Time interaction ($F_{(1,12)} = 0.70$, $p = 0.42$) (Figure3.5C). This spatial distribution of drug-sensitive regions aligns with previous findings demonstrating MK-801-induced CBV reduction in the hippocampus and medial prefrontal cortex (mPFC) [21], areas known for their high density of NMDA receptors [105].

3.2.3 SVM and ViT show limited spatial specificity

In contrast to the CNN's anatomically specific activation patterns, the SVM's feature weights revealed diffuse activation across the entire imaging field (Figure 3.6A). Quantifying CBV changes as percent change in pD signal relative to minutes 3-5 of the pre-injection baseline period, in MK-801 injected mice, the CBV in the SVM-selected regions showed a significant decrease in CBV compared to non-selected regions. A repeated measures ANOVA revealed a significant main effect of Region ($F_{(1,9)} = 84.05$, $p < 0.001$), as well as a significant Region $\times$ Time interaction ($F_{(1,9)} = 82.35$, $p < 0.001$), indicating that the CBV responses in SVM-selected regions differed significantly over time compared to non-selected regions (Figure 3.6B). For saline-injected animals, the CBV change in SVM-selected regions showed a trend toward being different from non-selected regions, with a significant main effect of Region ($F_{(1,12)} = 12.10$, $p < 0.01$) and a significant Region $\times$ Time interaction ($F_{(1,12)} = 14.60$, $p < 0.01$), indicating some temporal divergence between SVM-selected and non-selected regions (Figure 3.6C). These findings suggest that despite the diffuse pattern of SVM feature weights, the selected regions do exhibit distinct hemodynamic responses that evolve differently over time compared to non-selected regions. While capturing meaningful temporal dynamics in hemodynamic responses following MK-801 administration, the SVM relies on a broader spatial distribution of features compared to the CNN.

The ViT also detected drug-induced changes in brain hemodynamics, though with distinct spatial characteristics (Figure 3.7A). Measuring CBV changes as percent change in power Doppler signal relative to minutes 3-5 of the pre-injection baseline period, we found that in MK-801 injected mice, ViT-selected regions exhibited a substantial decrease in CBV, reaching -9.23 $\pm$ 6.14% (mean $\pm$ std) at 60 minutes post-injection, compared to -3.60 $\pm$ 2.84% in non-selected regions. A repeated measures ANOVA revealed a significant main effect of Region ($F_{(1,9)} = 31.90$, $p < 0.001$) and a significant Region $\times$ Time interaction ($F_{(1,9)} = 34.40$, $p < 0.001$), indicating that the CBV reduction was more pronounced in ViT-selected regions and evolved distinctly over time (Figure 3.7B). In contrast, analysis of saline-injected animals showed no significant differences between ViT-selected and non-selected regions (-0.18 $\pm$ 4.62% vs -0.13 $\pm$ 2.28%, respectively; Region effect: $F_{(1,12)} = 0.99$,

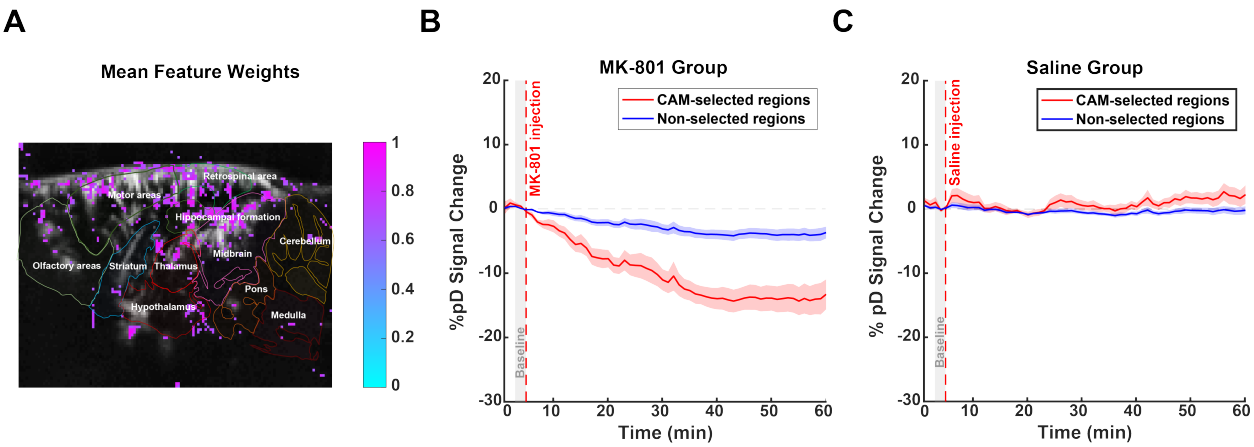

Figure 3.6: (A) Mean feature weights overlay on the reference image, highlighting regions of importance in SVM-based classification (threshold $\theta = 0.65$). Negative weights were extracted and inverted to specifically identify regions driving MK-801 (versus saline) classification, then normalized using percentile-based thresholding. Brighter regions indicate higher relevance for MK-801 detection. (B-C) Time series of $\%pD$ (i.e., CBV) changes in SVM-selected regions (red curves) compared to non-selected control regions (blue curves) for (B) MK-801 and (C) Saline groups. The shaded areas represent standard error derived from averaging across animals. The red dashed line at 5 min marks the injection time. A gray patch (3-5 min) indicates the pre-injection baseline window.

$p = 0.34$; Region $\times$ Time interaction: $F_{(1,12)} = 0.83$, $p = 0.38$) (Figure 3.7C). While the ViT successfully identified regions showing differential drug responses, its attention maps included areas outside the brain structure (Figure 3.7A), potentially limiting the biological interpretation of these findings. This "checkerboard" phenomenon has been highlighted as a drawback of attention rollout [2]. Together, these results suggest that while SVM and ViT could effectively distinguish between MK-801 and saline groups, they lacked anatomical precision in their feature selection compared to the CNN's more targeted identification of the corresponding brain regions.

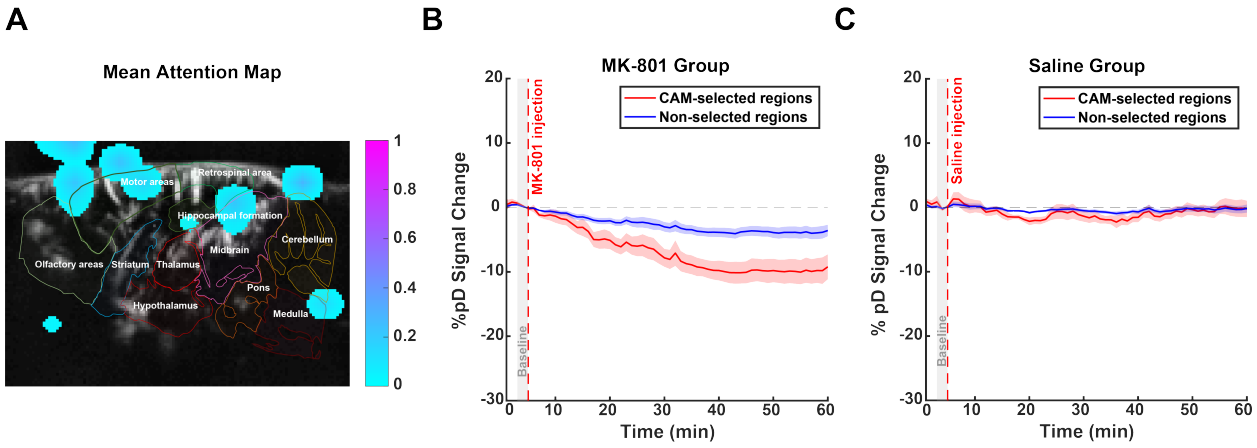

Figure 3.7: (A) Mean attention map overlay on the reference image, highlighting regions of importance in ViT-based classification (threshold $\theta = 0.025$). Brighter regions indicate higher relevance. (B-C) Time series of $\%pD$ (i.e., CBV) changes in ViT-selected regions (green curves) compared to non-selected control regions (blue curves) for (B) MK-801 and (C) Saline groups. The shaded areas represent standard error derived from averaging across animals. The red dashed line at 5 min marks the injection time. A gray patch (3-5 min) indicates the pre-injection baseline window.

Chapter 4

Conclusions

To create an internal map of the world, the brain uses a multitude of specialized neurons to encode spatial information. Here, we formulated the first rigorous definition of the joint stimulus information rate (Definition 2.2) and spectral-spatial information (Definition 2.3). This crucial step allows us to effectively quantify the efficiency of a population of neurons in conveying information about a stimulus. We demonstrated that this novel criterion can be used to train RNNs to produce place cell-like and head direction cell-like representations capable of decoding position and head direction.

Our results show that spectral-spatial information optimization generally produces a superior place cell-like encoding than the traditional Skaggs' stimulus information rate, as exhibited by larger place cell scores, better decoding performance, and more uniform firing across the domain. However, it is worth noting that for models trained with 256 neurons spectral-spatial information optimization is, at best, as good as Skaggs' stimulus information optimization. This suggests that spectral-spatial information may not be superior for training much larger numbers of neurons. Furthermore, despite serving as an excellent metric for place cell population encoding efficiency in both artificial and real neural systems, there is no evidence suggesting that spectral-spatial information is biologically plausible.

Contrary to previous works, the units in the hidden layer of our models bore little resemblance to that of grid cells. This phenomenon has been observed in supervised models with realistic place cell-like heterogeneous responses [87]. It has been suggested that for place cell-based recurrent architectures to learn grid cell-like hidden layer representations,

the readout code must be translation invariant [8, 92]. This property is not likely present in *in vivo* place cells, which over-represent certain locations, and is not present in those learned by our models [29, 40]. In total, a model that simultaneously produces grid and place cells in an unsupervised fashion is still at large.

A natural extension of our work is to construct stimulus information measures for $n \geq 3$ place cells *without* the use of eigenvalue analysis. However, such measures may be less computationally efficient than our proposed method for large n, as they would likely scale exponentially with the number of place cells (as opposed to our method, which scales quadratically). This will require further mathematical analysis, possibly diverging from the techniques used thus far.

Our results lend themselves nicely to the investigation of place cell remapping. Studies have shown that place cells form statistically independent representations of pairs of environments [5]. Furthermore, evidence suggests that different behavioral strategies can affect place cell firing [56, 62]. Our model can achieve different representations in geometrically identical environments by altering the agent's occupancy distribution. Here, we assumed the occupancy distribution to be a discrete uniform distribution for all trajectories. Future work on the impact of non-uniform occupancy distributions on place cell firing may prove beneficial. Another avenue for future research is merging information-theoretic methods with models of episodic memory in the hippocampus [9, 95]. Future computational directions could include implementing RNNs capable of encoding multiple stimuli at once, a property present in the mammalian brain [85]. This could incorporate an attention mechanism that allows the model to dynamically decide how much to focus on various stimuli. Recent interest has also been in the relationship between synergy, redundancy [106], and the phenomenon where neural networks during training suddenly generalize, known as grokking [18]. Furthermore, combining information-theoretic approaches with topological methods [41, 61, 69] is an additional area of interest.

Overall, our work displays the success of unifying advanced information theory with neuroscience and its applications to machine learning. The result is a widely applicable formula for place cell efficiency, which lays the foundation for more questions regarding place cell formation, function, and role in the mammalian brain.

Functional ultrasound imaging (fUSI) enables high-resolution measurement of cerebral blood volume changes [58, 59]. While its applications in neuropharmacology have grown [82, 100, 101, 21], most analyses rely on predefined regions of interest, which may miss important drug effects in other brain areas and introduce investigator bias in result interpretation. To address these limitations, we developed and evaluated multiple machine learning approaches for automated, whole-brain analysis of drug-induced hemodynamic changes. Our systematic comparison of CNNs, ViTs, and SVMs revealed distinct capabilities in analyzing fUSI data. Convolutional networks achieved both superior classification performance and biological specificity, demonstrating their unique ability to capture meaningful spatial patterns in brain hemodynamics. While SVMs generated diffuse activation patterns and ViTs included regions outside brain boundaries, CNNs consistently identified concentrated activity in cortical and hippocampal regions - areas with high NMDA receptor expression [105]. The biological relevance of CNN-identified regions was validated through quantitative analysis, which revealed significant CBV reductions specifically within these areas following MK-801 administration, while changes in other brain regions remained minimal. This precise spatial localization, combined with robust classification performance, suggests that CNNs can effectively capture both the location and magnitude of drug-induced changes in brain activity.

Our approach offers notable advantages over traditional ROI-dependent methods through its data-driven framework that processes the entire fUSI field of view. Conventional analyses typically require manual or atlas-based ROI selection before signal extraction, which inherently constrains the analysis to predetermined regions. In contrast, our CNN approach evaluates spatial patterns across the complete imaging field, identifying relevant hemodynamic changes based on functional responses rather than anatomical assumptions. This methodology helps reduce potential selection bias in neuropharmacological imaging analysis. The class activation mapping technique provides visualization of regions contributing to classification decisions, effectively generating data-driven ROIs based solely on functional responses to drug administration. This approach allows for the detection of drug effects that might occur outside canonical regions associated with a drug's known mechanism of action.

While our study demonstrates the potential of combining fUSI with CNN-based techniques for investigating drug-induced changes in brain activity, our approach presents important considerations and opportunities for refinement. The current study focused on a single drug (MK-801) in a specific animal model, which limits the generalizability of our findings. The use of isoflurane anesthesia presents another significant consideration, as anesthetics can modulate neurovascular coupling and potentially interact with the same neural systems targeted by experimental drugs [33]. Future studies should extend this methodology to awake, freely moving animals to eliminate these confounding factors, though this would require substantial adaptations to both imaging protocols and motion correction algorithms.

Regarding the machine learning models, several limitations warrant consideration. Despite reducing reliance on predefined ROIs, our CNN may exhibit bias toward regions with higher vascular density or signal-to-noise ratios. The overlap between CNN-identified regions and known areas of high NMDA receptor density suggests that for well-characterized drugs like MK-801, hypothesis-driven approaches might yield comparable results. Our image registration pipeline also presents opportunities for improvement. While we used default parameters for the *Imregdeform* algorithm which provided adequate alignment for our analysis, optimized registration parameters could potentially enhance the detection of subtle drug-induced changes. Future work could explore systematic tuning of registration parameters or alternative registration approaches to further improve cross-subject alignment. Additionally, the interpretability of deep learning models remains challenging despite our use of class activation mapping. While our implementation offers valuable insights into spatial aspects of drug effects, future work could explore methods for improving model interpretability and better capturing the complex temporal evolution of drug responses in the brain.

Beyond methodological refinements, the translation of this approach to clinical applications presents promising opportunities. Functional ultrasound imaging has already been successfully applied to human brain imaging in various contexts [49, 26, 91, 3, 83].Our machine learning framework could be adapted to these clinical scenarios, providing novel insights into physiological processes, pathological conditions, and therapeutic interventions

in the human central nervous system. While challenges exist—including the need for larger training datasets and potentially more sophisticated registration techniques—these represent technical instead of fundamental limitations. This minimally invasive nature, high spatiotemporal resolution and cost-effectiveness of ultrasound make this technique favorable for clinical applications.

Overall, our study establishes a novel data analysis framework integrating deep learning with fUSI for neuropharmacological research. The proposed methodology offers an unbiased approach for detecting and localizing drug-induced changes without relying on predetermined ROIs. Our comparison of machine learning approaches demonstrates that CNNs provide superior biological specificity while maintaining classification performance, establishing a foundation for future fUSI data analysis. This methodological innovation has the potential to accelerate drug development by revealing previously overlooked drug effects and enabling more comprehensive characterization of therapeutic compounds across both preclinical and clinical settings.

Bibliography

[1] Abnar, S. and Zuidema, W. (2022). Quantifying attention flow in transformers. arxiv 2020. *arXiv preprint arXiv:2005.00928*. 36, 37

[2] Achtibat, R., Hatefi, S. M. V., Dreyer, M., Jain, A., Wiegand, T., Lapuschkin, S., and Samek, W. (2024). Attnlrp: attention-aware layer-wise relevance propagation for transformers. *arXiv preprint arXiv:2402.05602*. 45

[3] Agyeman, K. A., Lee, D. J., Russin, J., Kreydin, E. I., Choi, W., Abedi, A., Lo, Y. T., Cavaleri, J., Wu, K., Edgerton, V. R., et al. (2024). Functional ultrasound imaging of the human spinal cord. *Neuron*, 112(10):1710–1722. 50

[4] Alexey, D. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv: 2010.11929*. 36

[5] Alme, C. B., Miao, C., Jezek, K., Treves, A., Moser, E. I., and Moser, M.-B. (2014). Place cells in the hippocampus: eleven maps for eleven rooms. *Proceedings of the National Academy of Sciences*, 111(52):18428–18435. 2, 6, 48

[6] Andiné, P., Widermark, N., Axelsson, R., Nyberg, G., Olofsson, U amd Mårtensson, E., and M, S. (1999). Characterization of mk-801-induced behavior as a putative rat model of psychosis. *J Pharmacol Exp Ther*, 290:1393–408. 3

[7] Aronov, D., Nevers, R., and Tank, D. W. (2017). Mapping of a non-spatial dimension by the hippocampal–entorhinal circuit. *Nature*, 543(7647):719–722. 2, 6

[8] Banino, A., Barry, C., Uria, B., Blundell, C., Lillicrap, T., Mirowski, P., Pritzel, A., Chadwick, M. J., Degris, T., Modayil, J., et al. (2018). Vector-based navigation using grid-like representations in artificial agents. *Nature*, 557(7705):429–433. 1, 48

[9] Benna, M. K. and Fusi, S. (2021). Place cells may simply be memory cells: Memory compression leads to spatial tuning and history dependence. *Proceedings of the National Academy of Sciences*, 118(51):e2018422118. 48

[10] Biewald, L. (2020). Experiment tracking with weights and biases. Software available from wandb.com. 33

[11] Bonnici, V. (2020). Kullback-leibler divergence between quantum distributions, and its upper-bound. *arXiv preprint arXiv:2008.05932*. 73

[12] Brenner, N., Strong, S. P., Koberle, R., Bialek, W., and Steveninck, R. R. d. R. v. (2000). Synergy in a neural code. *Neural computation*, 12(7):1531–1552. 7, 8

[13] Burak, Y. and Fiete, I. R. (2009). Accurate path integration in continuous attractor network models of grid cells. *PLoS computational biology*, 5(2):e1000291. 1

[14] Bures, J., Fenton, A. A., Kaminsky, Y., and Zinyuk, L. (1997). Place cells and place navigation. *Proceedings of the National Academy of Sciences*, 94(1):343–350. 17

[15] Butts, D. A. (2003). How much information is associated with a particular stimulus? *Network: Computation in Neural Systems*, 14(2):177. 6

[16] Butts, D. A. and Goldman, M. S. (2006). Tuning curves, neuronal variability, and sensory coding. *PLoS biology*, 4(4):e92. 6

[17] Chen, G., King, J. A., Burgess, N., and O'Keefe, J. (2013). How vision and movement combine in the hippocampal place code. *Proceedings of the National Academy of Sciences*, 110(1):378–383. 2

[18] Clauw, K., Stramaglia, S., and Marinazzo, D. (2024). Information-theoretic progress measures reveal grokking is an emergent phase transition. *arXiv preprint arXiv:2408.08944*. 48

[19] Clevert, D.-A. (2015). Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*. 34

[20] Couey, J. J., Witoelar, A., Zhang, S.-J., Zheng, K., Ye, J., Dunn, B., Czajkowski, R., Moser, M.-B., Moser, E. I., Roudi, Y., et al. (2013). Recurrent inhibitory circuitry as a mechanism for grid formation. *Nature neuroscience*, 16(3):318–324. 1

[21] Crown, L. M., Agyeman, K. A., Choi, W., Zepeda, N., Iseri, E., Pahlavan, P., Siegel, S. J., Liu, C., Christopoulos, V., and Lee, D. J. (2024). Theta-frequency medial septal

nucleus deep brain stimulation increases neurovascular activity in mk-801-treated mice. *Frontiers in Neuroscience*, 18:1372315. 3, 31, 38, 43, 49

[22] Cueva, C. J. and Wei, X.-X. (2018). Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. *arXiv preprint arXiv:1803.07770*. 1, 78

[23] Dayan, P. and Abbott, L. F. (2005). *Theoretical neuroscience: computational and mathematical modeling of neural systems*. MIT press. 2, 5

[24] Deighton, J., Mackey, W., Schizas, I., Boothe Jr, D. L., and Maroulas, V. (2024a). Higher-order spatial information for self-supervised place cell learning. *arXiv preprint arXiv:2407.06195*. 2

[25] Deighton, J., Zhong, S., Agyeman, K., Choi, W., Liu, C., Lee, D., Maroulas, V., and Christopoulos, V. (2024b). Monitoring drug-induced brain activity changes with functional ultrasound imaging and convolutional neural networks. *arXiv preprint arXiv:2410.09523*. 3

[26] Demene, C., Baranger, J., Bernal, M., Delanoe, C., Auvin, S., Biran, V., Alison, M., Mairesse, J., Harribaud, E., Pernot, M., et al. (2017a). Functional ultrasound imaging of brain activity in human newborns. *Science translational medicine*, 9(411):eaah6756. 50

[27] Demene, C., Baranger, J., Bernal, M., Delanoe, C., Auvin, S., Biran, V., Alison, M., Mairesse, J., Harribaud, E., Pernot, M., Tanter, M., and O, B. (2017b). Functional ultrasound imaging of brain activity in human newborns. *Sci Transl Med.*, 9. 3

[28] Dorrell, W., Latham, P. E., Behrens, T. E., and Whittington, J. C. (2022). Actionable neural representations: Grid cells from minimal constraints. *arXiv preprint arXiv:2209.15563*. 1

[29] Dupret, D., O'neill, J., Pleydell-Bouverie, B., and Csicsvari, J. (2010). The reorganization and reactivation of hippocampal maps predict spatial memory performance. *Nature neuroscience*, 13(8):995–1002. 48

[30] Elman, J. L. (1991). Distributed representations, simple recurrent networks, and grammatical structure. *Machine learning*, 7:195–225. 15

[31] Fenton, A. A., Kao, H.-Y., Neymotin, S. A., Olypher, A., Vayntrub, Y., Lytton, W. W., and Ludvig, N. (2008). Unmasking the ca1 ensemble place code by exposures to small and large environments: more place cells and multiple, irregularly arranged, and expanded place fields in the larger space. *Journal of Neuroscience*, 28(44):11250–11262. 2, 6

[32] Finkelstein, A., Derdikman, D., Rubin, A., Foerster, J. N., Las, L., and Ulanovsky, N. (2015). Three-dimensional head-direction coding in the bat brain. *Nature*, 517(7533):159–164. 2, 6

[33] Franceschini, M. A., Radhakrishnan, H., Thakur, K., Wu, W., Ruvinskaya, S., Carp, S., and Boas, D. A. (2010). The effect of different anesthetics on neurovascular coupling. *Neuroimage*, 51(4):1367–1377. 50

[34] Frank, L. M., Brown, E. N., and Wilson, M. (2000). Trajectory encoding in the hippocampus and entorhinal cortex. *Neuron*, 27(1):169–178. 2, 6

[35] Fu, H., Rodriguez, G. A., Herman, M., Emrani, S., Nahmani, E., Barrett, G., Figueroa, H. Y., Goldberg, E., Hussaini, S. A., and Duff, K. E. (2017). Tau pathology induces excitatory neuron loss, grid cell dysfunction, and spatial memory deficits reminiscent of early alzheimer's disease. *Neuron*, 93(3):533–541. 2, 6

[36] Fu, R., Hu, Q., Dong, X., Guo, Y., Gao, Y., and Li, B. (2020). Axiom-based grad-cam: Towards accurate visualization and explanation of cnns. *arXiv preprint arXiv:2008.02312*. 4

[37] Fuhs, M. C. and Touretzky, D. S. (2006). A spin glass model of path integration in rat medial entorhinal cortex. *Journal of Neuroscience*, 26(16):4266–4276. 1

[38] Gantmakher, F. R. (2000). *The theory of matrices*, volume 131. American Mathematical Soc. 13

[39] Gardner, R. J., Hermansen, E., Pachitariu, M., Burak, Y., Baas, N. A., Dunn, B. A., Moser, M.-B., and Moser, E. I. (2022). Toroidal topology of population activity in grid cells. *Nature*, 602(7895):123–128. 2, 6

[40] Gauthier, J. L. and Tank, D. W. (2018). A dedicated population for reward coding in the hippocampus. *Neuron*, 99(1):179–193. 48

[41] Gillespie, P., Maroulas, V., and Schizas, I. (2024). Bayesian sheaf neural networks. *arXiv preprint arXiv:2410.09590*. 48

[42] Griggs, W. S., Norman, S. L., Deffieux, T., Segura, F., Osmanski, B.-F., Chau, G., Christopoulos, V., Liu, C., Tanter, M., Shapiro, M. G., et al. (2023). Decoding motor plans using a closed-loop ultrasonic brain–machine interface. *Nature Neuroscience*, pages 1–12. 3

[43] Guanella, A., Kiper, D., and Verschure, P. (2007). A model of grid cells based on a twisted torus topology. *International journal of neural systems*, 17(04):231–240. 1

[44] Hafting, T., Fyhn, M., Molden, S., Moser, M.-B., and Moser, E. I. (2005). Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052):801–806. 5

[45] Hayman, R., Verriotis, M. A., Jovalekic, A., Fenton, A. A., and Jeffery, K. J. (2011). Anisotropic encoding of three-dimensional space by place cells and grid cells. *Nature neuroscience*, 14(9):1182–1188. 2, 6

[46] Hazama, Y. and Tamura, R. (2019). Data on the activity of place cells in the hippocampal ca1 subfield of a monkey performing a shuttling task. *Data in brief*, 26:104467. x, xii, xiv, 9, 19, 28, 83

[47] Horn, R. and Johnson, C. (1990). *Matrix Analysis*. Cambridge University Press. 70

[48] Imbault, M., Chauvet, D., Gennisson, J., Capelle, L., and M, T. (2017a). Intraoperative functional ultrasound imaging of human brain activity. *Sci Rep.*, 7. 3

[49] Imbault, M., Chauvet, D., Gennisson, J.-L., Capelle, L., and Tanter, M. (2017b). Intraoperative functional ultrasound imaging of human brain activity. *Scientific reports*, 7(1):7304. 50

[50] Kayser, C., Logothetis, N. K., and Panzeri, S. (2010). Millisecond encoding precision of auditory cortex neurons. *Proceedings of the National Academy of Sciences*, 107(39):16976–16981. 6

[51] Keinath, A. T., Julian, J. B., Epstein, R. A., and Muzzio, I. A. (2017). Environmental geometry aligns the hippocampal map during spatial reorientation. *Current Biology*, 27(3):309–317. 2

[52] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. 35, 37, 79

[53] Kropff, E., Carmichael, J. E., Moser, M.-B., and Moser, E. I. (2015). Speed cells in the medial entorhinal cortex. *Nature*, 523(7561):419–424. 5

[54] Lever, C., Burton, S., Jeewajee, A., O'Keefe, J., and Burgess, N. (2009). Boundary vector cells in the subiculum of the hippocampal formation. *Journal of Neuroscience*, 29(31):9771–9777. 5

[55] Lever, C., Wills, T., Cacucci, F., Burgess, N., and O'Keefe, J. (2002). Long-term plasticity in hippocampal place-cell representation of environmental geometry. *Nature*, 416(6876):90–94. 2

[56] Levy, S. J. and Hasselmo, M. E. (2023). Hippocampal remapping induced by new behavior is mediated by spatial context. *bioRxiv*, pages 2023–02. 48

[57] Lu, F., Wu, F., Hu, P., Peng, Z., and Kong, D. (2017). Automatic 3d liver location and segmentation via convolutional neural network and graph cut. *International journal of computer assisted radiology and surgery*, 12:171–182. 34, 35

[58] Macé, E., Montaldo, G., Cohen, I., Baulac, M., Fink, M., and Tanter, M. (2011). Functional ultrasound imaging of the brain. *Nature methods*, 8(8):662–664. 3, 49

[59] Mace, E., Montaldo, G., Osmanski, B.-F., Cohen, I., Fink, M., and Tanter, M. (2013). Functional ultrasound imaging of the brain: theory and basic principles. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 60(3):492–506. 3, 49

[60] Mackey, W., Schizas, I., Deighton, J., Boothe Jr, D. L., and Maroulas, V. (2024). Geometric sparsification in recurrent neural networks. *arXiv preprint arXiv:2406.06290*. 1

[61] Marchese, A., Maroulas, V., and Mike, J. (2017). K- means clustering on the space of persistence diagrams. In *Wavelets and Sparsity XVII*, volume 10394, pages 218–227. SPIE. 48

[62] Markus, E. J., Barnes, C. A., McNaughton, B. L., Gladden, V. L., and Skaggs, W. E. (1994). Spatial information content and reliability of hippocampal ca1 neurons: effects of visual input. *Hippocampus*, 4(4):410–421. 17, 48

[63] Maroulas, V., Micucci, C. P., and Nasrin, F. (2020). Bayesian topological learning for classifying the structure of biological networks. *arXiv preprint arXiv:2009.11974*. 1

[64] McNaughton, B. L., Battaglia, F. P., Jensen, O., Moser, E. I., and Moser, M.-B. (2006). Path integration and the neural basis of the'cognitive map'. *Nature Reviews Neuroscience*, 7(8):663–678. 1, 5

[65] Mitchell, E. C., Story, B., Boothe, D., Franaszczuk, P. J., and Maroulas, V. (2024). A topological deep learning framework for neural spike decoding. *Biophysical Journal*. 1, 27

[66] Nasrin, F., Oballe, C., Boothe, D., and Maroulas, V. (2019). Bayesian topological learning for brain state classification. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 1247–1252. IEEE. 1

[67] Nieh, E. H., Schottdorf, M., Freeman, N. W., Low, R. J., Lewallen, S., Koay, S. A., Pinto, L., Gauthier, J. L., Brody, C. D., and Tank, D. W. (2021). Geometry of abstract learned knowledge in the hippocampus. *Nature*, 595(7865):80–84. 2, 6

[68] Norman, S. L., Maresca, D., Christopoulos, V. N., Griggs, W. S., Demene, C., Tanter, M., Shapiro, M. G., and Andersen, R. A. (2021). Single-trial decoding of movement intentions using functional ultrasound neuroimaging. *Neuron*, 109(9):1554–1566. 3

[69] Oballe, C., Boothe, D., Franaszczuk, P. J., and Maroulas, V. (2021a). Tofu: Topology functional units for deep learning. *Foundations of Data Science*. 48

[70] Oballe, C., Cherne, A., Boothe, D., Kerick, S., Franaszczuk, P. J., and Maroulas, V. (2021b). Bayesian topological signal processing. *Discrete & Continuous Dynamical Systems-S*. 1

[71] O'Keefe, J. (1976). Place units in the hippocampus of the freely moving rat. *Experimental neurology*, 51(1):78–109. 2

[72] O'Keefe, J. and Dostrovsky, J. (1971). The hippocampus as a spatial map: preliminary evidence from unit activity in the freely-moving rat. *Brain research*. 5

[73] O'keefe, J. and Nadel, L. (1978). *The hippocampus as a cognitive map*. Oxford university press. 2, 19

[74] Ormond, J. and O'Keefe, J. (2022). Hippocampal place cells have goal-oriented vector fields during navigation. *Nature*, 607(7920):741–746. 2, 6

[75] Osmanski, B., Martin, C., Montaldo, G., Laniéce, P., Pain, F., Tanter, M., and H, G. (2014a). Functional ultrasound imaging reveals different odor-evoked patterns of vascular activity in the main olfactory bulb and the anterior piriform cortex. *NeuroImage*, 95:176–84. 3

[76] Osmanski, B.-F., Pezet, S., Ricobaraza, A., Lenkei, Z., and Tanter, M. (2014b). Functional ultrasound imaging of intrinsic connectivity in the living rat brain with high spatiotemporal resolution. *Nature Communications*, 5(1):5023. 3

[77] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32. 33, 78

[78] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830. 33, 81

[79] Peyrache, A., Lacroix, M. M., Petersen, P. C., and Buzsáki, G. (2015). Internally organized mechanisms of the head direction sense. *Nature Neuroscience*, 18(4):569–575. x, xii, 9, 18, 27, 28

[80] Pnevmatikakis, E. and Giovannucci, A. (2017). Normcorre: An online algorithm for piecewise rigid motion correction of calcium imaging data. *Journal of Neuroscience Methods*, 291:83–94. 31

[81] Quian Quiroga, R. and Panzeri, S. (2009). Extracting information from neuronal populations: information theory and decoding approaches. *Nature Reviews Neuroscience*, 10(3):173–185. 2, 5

[82] Rabut, C., Ferrier, J., Bertolo, A., Osmanski, B., Mousset, X., Pezet, S., Deffieux, T., Lenkei, Z., and Tanter, M. (2020). Pharmaco-fus: Quantification of pharmacologically-induced dynamic changes in brain perfusion and connectivity by functional ultrasound imaging in awake mice. *Neuroimage*, 222:117231. 3, 49

[83] Rabut, C., Norman, S. L., Griggs, W. S., Russin, J. J., Jann, K., Christopoulos, V., Liu, C., Andersen, R. A., and Shapiro, M. G. (2024). Functional ultrasound imaging of human brain activity through an acoustically transparent cranial window. *Science translational medicine*, 16(749):eadj3143. 50

[84] Rich, P. D., Liaw, H.-P., and Lee, A. K. (2014). Large environments reveal the statistical structure governing hippocampal representations. *Science*, 345(6198):814–817. 5

[85] Sanders, H., Ji, D., Sasaki, T., Leutgeb, J. K., Wilson, M. A., and Lisman, J. E. (2019). Temporal coding and rate remapping: Representation of nonspatial information in the hippocampus. *Hippocampus*, 29(2):111–127. 48

[86] Save, E., Cressant, A., Thinus-Blanc, C., and Poucet, B. (1998). Spatial firing of hippocampal place cells in blind rats. *Journal of Neuroscience*, 18(5):1818–1826. 23

[87] Schaeffer, R., Khona, M., and Fiete, I. (2022). No free lunch from deep learning in neuroscience: A case study through models of the entorhinal-hippocampal circuit. *Advances in Neural Information Processing Systems*, 35:16052–16067. 1, 47

[88] Schaeffer, R., Khona, M., Ma, T., Eyzaguirre, C., Koyejo, S., and Fiete, I. (2024). Self-supervised learning of representations for space generates multi-modular grid cells. *Advances in Neural Information Processing Systems*, 36. 1, 23

[89] Shannon, C. E. (1948). A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423. 2, 5, 6, 7

[90] Skaggs, W., Mcnaughton, B., and Gothard, K. (1992). An information-theoretic approach to deciphering the hippocampal code. *Advances in neural information processing systems*, 5. 2, 5, 8, 17

[91] Soloukey, S., Vincent, A. J., Satoer, D. D., Mastik, F., Smits, M., Dirven, C. M., Strydis, C., Bosch, J. G., van der Steen, A. F., De Zeeuw, C. I., et al. (2020). Functional ultrasound (fus) during awake brain surgery: the clinical potential of intra-operative functional and vascular brain mapping. *Frontiers in neuroscience*, 13:1384. 50

[92] Sorscher, B., Mel, G., Ganguli, S., and Ocko, S. (2019). A unified theory for the origin of grid cells through the lens of pattern formation. *Advances in neural information processing systems*, 32. 1, 48, 78, 81

[93] Sorscher, B., Mel, G. C., Ocko, S. A., Giocomo, L. M., and Ganguli, S. (2023). A unified theory for the computational and mechanistic origins of grid cells. *Neuron*, 111(1):121–137. 1, 78, 81

[94] Souza, B. C., Pavão, R., Belchior, H., and Tort, A. B. (2018). On information metrics for spatial coding. *Neuroscience*, 375:62–73. 2, 6, 21

[95] Squire, L. R., Stark, C. E., and Clark, R. E. (2004). The medial temporal lobe. *Annu. Rev. Neurosci.*, 27:279–306. 48

[96] Stachenfeld, K. L., Botvinick, M. M., and Gershman, S. J. (2017). The hippocampus as a predictive map. *Nature neuroscience*, 20(11):1643–1653. 1

[97] Stensola, T. and Moser, E. I. (2016). Grid cells and spatial maps in entorhinal cortex and hippocampus. *Micro-, meso-and macro-dynamics of the brain*, pages 59–80. x, 6

[98] Taube, J. S., Muller, R. U., and Ranck, J. B. (1990). Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis. *Journal of Neuroscience*, 10(2):420–435. 5

[99] Tolman, E. C. (1948). Cognitive maps in rats and men. *Psychological review*, 55(4):189. 19

[100] Vidal, B., Droguerre, M., Valdebenito, M., Zimmer, L., Hamon, M., Mouthon, F., and Charvériat, M. (2020a). Pharmaco-fus for characterizing drugs for alzheimer's disease - the case of thn201, a drug combination of donepezil plus mefloquine. *Front Neurosci.*, 14:835. 3, 49

[101] Vidal, B., Droguerre, M., Venet, L., Zimmer, M., Valdebenito, M., Mouthon, F., and Charvériat, M. (2020b). Functional ultrasound imaging to study brain dynamics: Application of pharmaco-fus to atomoxetine. *Neuropharmacology*, 179. 3, 49

[102] Wang, Q., Ding, S.-L., Li, Y., Royall, J., Feng, D., Lesnar, P., Graddis, N., Naeemi, M., Facer, B., Ho, A., et al. (2020). The allen mouse brain common coordinate framework: a 3d reference atlas. *Cell*, 181(4):936–953. 31

[103] Wang, X., Lu, T., Snider, R. K., and Liang, L. (2005). Sustained firing in auditory cortex evoked by preferred stimuli. *Nature*, 435(7040):341–346. 5

[104] Wang, Y., Xu, X., and Wang, R. (2019). The place cell activity is information-efficient constrained by energy. *Neural Networks*, 116:110–118. 2, 6

[105] Watanabe, M., Mishina, M., and Inoue, Y. (1994). Distinct distributions of five nmda receptor channel subunit mrnas in the brainstem. *Journal of Comparative Neurology*, 343(4):520–531. 43, 49

[106] Williams, P. L. and Beer, R. D. (2010). Nonnegative decomposition of multivariate information. *arXiv preprint arXiv:1004.2515*. 48

[107] Yu, C., Behrens, T. E., and Burgess, N. (2020). Prediction and generalisation over directed actions by grid cells. *arXiv preprint arXiv:2006.03355*. 1

[108] Zepeda, N., Crown, L., Medvidovic, S., Choi, W., Sheth, M., Bergosh, M., Gifford, R., Folz, C., Lam, P., Lu, G., Featherstone, R., Liu, C., Siegel, S., and DJ, L. (2022). Frequency-specific medial septal nucleus deep brain stimulation improves spatial memory in mk-801-treated male rats. *Neurobiol Dis.*, 170. 3

[109] Zhou, B., Khosla, A., A., L., Oliva, A., and Torralba, A. (2016a). Learning Deep Features for Discriminative Localization. *CVPR*. 4

[110] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016b). Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929. 34

Appendices

A Spectral Spatial Information

Table 1: Mathematical Notation

Notation	Value
J	Spatial information matrix
P_t	Output of RNN at step t
G_t	Hidden state of RNN at step t
W	RNN Hidden state update matrix
$\lambda_A(x)$	Firing rate of place cell A at spatial location x
λ_1	Leading eigenvalue of spatial information matrix
r	Pearson's correlation coefficient

Construction A.1. *Here, we construct the joint spatial information rate of two place cells A and B. For some discrete space X, let $\lambda_a(x)$ and $\lambda_b(x)$ be the firing rate of place cells A and B at location $x \in X$, respectively. Let A_x and B_x denote the events where A and B spike at position x during Δt and A_x^c and B_x^c denote the complements.*

Now, we estimate the correlation between two place cells r as the Pearson's correlation over the entire spatial domain and create the joint distribution of the cells firing during a small time Δt by treating each distribution of a place cell firing during Δt as a Bernoulli distribution. Then,

$$R = COV(A, B) = r\sqrt{\lambda_a(x)\Delta t(1-\lambda_a(x)\Delta t)\lambda_b(x)\Delta t(1-\lambda_b(x)\Delta t)}$$

where r is the correlation coefficient between place cells A and B, and we have

$$\begin{aligned}
P(A_x \cap B_x) &= R + \lambda_a(x)\lambda_b(x)(\Delta t)^2 \\
P(A_x \cap B_x^c) &= \lambda_a(x)\Delta t(1-\lambda_b(x)\Delta t) - R \\
P(A_x^c \cap B_x) &= (1-\lambda_a(x)\Delta t)\lambda_b(x)\Delta t - R \\
P(A_x^c \cap B_x^c) &= (1-\lambda_a(x)\Delta t)(1-\lambda_b(x)\Delta t) + R
\end{aligned}$$

Neglecting terms higher than Δt, we have $R \approx r\sqrt{\lambda_a(x)\lambda_b(x)}\Delta t$ and

$$
\begin{aligned}
P(A_x \cap B_x) &= r\sqrt{\lambda_a(x)\lambda_b(x)}\Delta t \\
P(A_x \cap B_x^c) &= \lambda_a(x)\Delta t - r\sqrt{\lambda_a(x)\lambda_b(x)}\Delta t \\
P(A_x^c \cap B_x) &= \lambda_b(x)\Delta t - r\sqrt{\lambda_a(x)\lambda_b(x)}\Delta t \\
P(A_x^c \cap B_x^c) &= 1 - \lambda_a(x)\Delta t - \lambda_b(x)\Delta t + r\sqrt{\lambda_a(x)\lambda_b(x)}\Delta t
\end{aligned}
$$

We also have, regardless of location $E[(P(A_x \cap B_x))] = \sum_x p(x) r\sqrt{\lambda_a(x)\lambda_b(x)}\Delta t$ and similarly for the other scenarios.

Then, the mutual information between the joint distribution and space is

$$
I(A,B) = \sum_x \sum_{\omega_x \in \Omega_x} p(x)P(\omega_x)\log_2\left(\frac{P(\omega_x)}{E[P(\omega_x)]}\right) \tag{1}
$$

where $\Omega_x = \{A_x \cap B_x, \ldots, A_x^c \cap B_x^c\}$ is the set of possible outcomes at location x. In Lemma A.2, we show that the final term ($\omega_x = A_x^c \cap B_x^c$) goes to zero when dropping terms higher than Δt. Thus, expanding Equation (1) and a short simplification yields,

$$
\begin{aligned}
I(A,B) = \Delta t \sum_x p(x) r\sqrt{\lambda_a(x)\lambda_b(x)}\log_2\left(\frac{\sqrt{\lambda_a(x)\lambda_b(x)}}{\sum_x p(x)\sqrt{\lambda_a(x)\lambda_b(x)}}\right) \\
+ p(x)[\lambda_a(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}]\log_2\left(\frac{\lambda_a(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}}{\sum_x p(x)[\lambda_a(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}]}\right) \\
+ p(x)[\lambda_b(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}]\log_2\left(\frac{\lambda_b(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}}{\sum_x p(x)[\lambda_b(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}]}\right)
\end{aligned}
$$

Then, using the series expansion of $I(A,B)$ around t, we have that the first time derivative is approximated by

$$\begin{aligned} I(A,B)' \approx & \sum_x p(x) r \sqrt{\lambda_a(x)\lambda_b(x)} \log_2 \left(\frac{\sqrt{\lambda_a(x)\lambda_b(x)}}{\sum_x p(x)\sqrt{\lambda_a(x)\lambda_b(x)}} \right) \\ & + p(x)[\lambda_a(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}] \log_2 \left(\frac{\lambda_a(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}}{\sum_x p(x)[\lambda_a(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}]} \right) \\ & + p(x)[\lambda_b(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}] \log_2 \left(\frac{\lambda_b(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}}{\sum_x p(x)[\lambda_b(x) - r\sqrt{\lambda_a(x)\lambda_b(x)}]} \right) \end{aligned}$$

This is the definition of $I_{sec}(A,B)$ the joint spatial information rate. We may write more succinctly,

$$\begin{aligned} I_{sec}(A,B) = & \sum_x p(x) r \sqrt{\lambda_{a,b}(x)} \log_2 \left(\frac{\sqrt{\lambda_{a,b}(x)}}{\tilde{\lambda}_{a,b}} \right) \\ & + p(x) \left(\lambda_a(x) - r\sqrt{\lambda_{a,b}(x)} \right) \log_2 \left(\frac{\lambda_a(x) - r\sqrt{\lambda_{a,b}(x)}}{\bar{\lambda}_a - r\tilde{\lambda}_{a,b}} \right) \\ & + p(x) \left(\lambda_b(x) - r\sqrt{\lambda_{a,b}(x)} \right) \log_2 \left(\frac{\lambda_b(x) - r\sqrt{\lambda_{a,b}(x)}}{\bar{\lambda}_b - r\tilde{\lambda}_{a,b}} \right) \end{aligned}$$

where $\sum_x p(x)\lambda_a(x) = \bar{\lambda}_a$, $\sum_x p(x)\lambda_b(x) = \bar{\lambda}_b$, $\sum_x p(x)\sqrt{\lambda_{a,b}}(x) = \tilde{\lambda}_{a,b}$, and $\lambda_a(x)\lambda_b(x) = \lambda_{a,b}(x)$. Notice, if A, B are independent ($r = 0$), then,

$$\begin{aligned} I_{sec}(A,B) &= \sum_x p(x)0 + p(x)\left(\lambda_a(x) - 0\right) \log_2 \left(\frac{\lambda_a(x) - 0}{\bar{\lambda}_a - 0} \right) + p(x)\left(\lambda_b(x) - 0\right) \log_2 \left(\frac{\lambda_b(x) - 0}{\bar{\lambda}_b - 0} \right) \\ &= \sum_x p(x)\lambda_a(x) \log_2 \left(\frac{\lambda_a(x)}{\bar{\lambda}_a} \right) + \sum_x p(x)\lambda_b(x) \log_2 \left(\frac{\lambda_b(x)}{\bar{\lambda}_b} \right) \\ &= I_{sec}(A) + I_{sec}(B) \end{aligned}$$

Finally as is done for I_{sec}, we may write the joint information of two PCs in units of bits/spike by defining,

$$I_{spike}(A,B) = \frac{1}{\Lambda_{A,B}} I_{sec}(A,B)$$

where $\Lambda_{A,B} = \sum_x p(x)\frac{\lambda_a(x)+\lambda_b(x)}{2} = \frac{\bar{\lambda}_a+\bar{\lambda}_b}{2}$. *This is chosen so that* $I_{spike}(A,A) = I_{spike}(A)$.

Lemma A.2 (No-firing term provides no information)**.** *Here we show that the no-firing term from Equation (1) goes to zero.*

Proof. First, notice

$$\begin{aligned}
&\log_2\left(\frac{1-\Delta t(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)})}{\sum_x p(x)[1-\Delta t(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)})}\right)\\
&=\log_2\left(1-\Delta t(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)}\right)\\
&-\log_2\left(\sum_x p(x)-\Delta t\sum_x p(x)[\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)}]\right)\\
&=\log_2\left(1-\Delta t(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)})\right)\\
&-\log_2\left(1-\Delta t\sum_x p(x)[\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)}]\right)
\end{aligned}$$

Using the Taylor series $\log_2(1-x\Delta t) \approx -\frac{x\Delta t}{\ln(2)}$ we can approximate these terms by

$$\begin{aligned}
&\log_2\left(\frac{1-\Delta t(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)})}{\sum_x p(x)[1-\Delta t(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)})}\right)\\
&\approx -\Delta t\frac{\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)}}{\ln(2)}+\Delta t\frac{\sum_x p(x)[\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)}]}{\ln(2)}
\end{aligned}$$

Thus, focusing on the last term of $I(A,B)$ in Equation (1),

$$\begin{aligned}
&\sum_x p(x)[1-\Delta t(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)})]\\
&\cdot\log_2\left(\frac{1-\Delta t(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)})}{\sum_x p(x)[1-\Delta t(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)})]}\right)\\
&=\sum_x p(x)\frac{\Delta t}{\ln(2)}\left(\sum_x p(x)[\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)}]-(\lambda_a(x)+\lambda_b(x)-r\sqrt{\lambda_a(x)\lambda_b(x)})\right)\\
&=0
\end{aligned}$$

□

Remark A.3 (Eigenvalues of Spatial Information Matrix)**.** *Suppose we have a data matrix* $P \in \mathbb{R}^{n_x \times n_p}$ *representing* n_p *place cell firing rates across a trajectory of length* n_x*. That is,* $P_{i,j} = \lambda_j(x_i)$*, place cell* j*'s mean firing rate at spatial location* x_i*. From* P*, we construct* $J \in \mathbb{R}^{n_p \times n_p}$ *with* $J_{i,j} = I_{spike}(P_i, P_j)$*, the joint spatial information rate between place cells* i, j*, and the trajectory. Let,* $R(J, v)$ *be the the Rayleigh quotient [47] of* J *with nonzero vector* v *such that* $||v||_2 = 1$*, and* $S(J)$ *be the sum of the entries of* J*. That is,*

$$R(J, v) = v^T J v = \sum_{i=1}^{n_p} \sum_{i=1}^{n_p} v_i J_{ij} v_j$$
$$S(J) = \sum_{i=1}^{n_p} \sum_{i=1}^{n_p} J_{ij}$$

It follows that $\max_{v,\ ||v||_2=1} R(J, v) = \lambda_1$ *where* λ_1 *is the leading eigenvalue of* J*. When* $v_i > 0$ *for all* $i = 1, \ldots, n_p$*, we have that* λ_1 *is positively correlated with* $S(J)$ *(Fig. 1).*

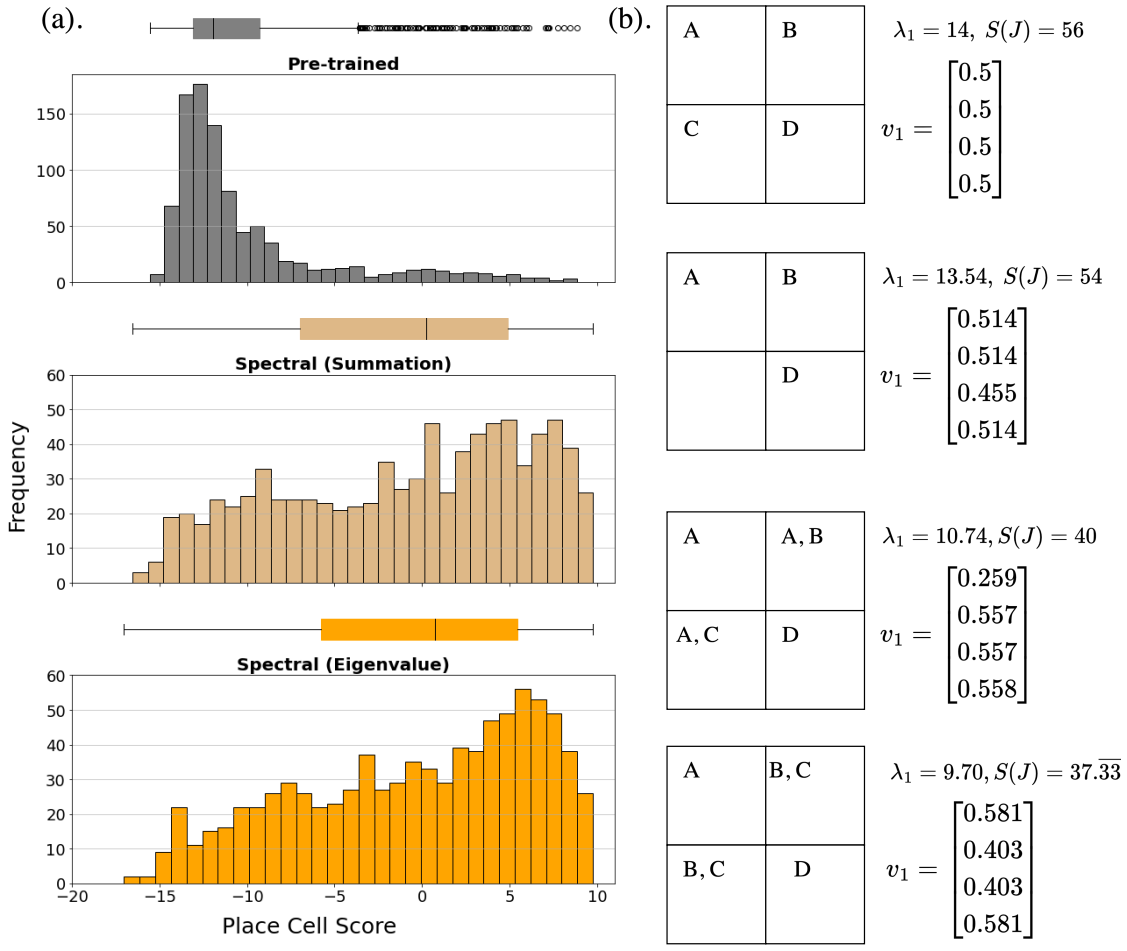

Figure 1: (a) Place cell box plots and histograms across 20 models (10 trained with 32 place cells, 10 with 64). Comparison in place cell score (Equation (2.13)) is shown between models trained via maximizing $S(J)$, the summation of the entries of J versus maximizing the leading eigenvalue of J. (b) Comparison between the leading eigenvalue of the spatial information matrix J, λ_1, and $S(J)$ across four scenarios. Consider four neurons, A, B, C, and D in a simple four-quadrant square arena. If the letter of a neuron appears in a quadrant, then we set the firing rate of the neuron in that quadrant to 1 spike/sec. The corresponding leading eigenpair (λ_1, v_1) as well as $S(J)$ are shown for each scenario.

Theorem A.4 (Stimulus Information Properties)**.** *We show that the described properties hold. First,* $I_{spike}(A,B) = I_{spike}(B,A)$ *holds trivially. Next, we show that* $I_{spike}(A,A) = I_{spike}(A)$

Proof. Notice that if $A = B$, then we have the following:

$$\begin{aligned}
&\bar{\lambda}_B = \bar{\lambda}_A \\
&\sqrt{\lambda_{A,B}(x)} = \sqrt{\lambda_a(x)\lambda_b(x)} = \sqrt{\lambda_a^2(x)} = \lambda_A(x) \\
&\tilde{\lambda}_{A,B} = \sum_x p(x)\sqrt{\lambda_{A,B}(x)} = \sum_x p(x)\lambda_{a(x)} = \bar{\lambda}_A \\
&\lambda_{A,B} = \frac{\bar{\lambda}_A + \bar{\lambda_B}}{2} = \frac{\bar{\lambda}_A + \bar{\lambda_A}}{2} = \bar{\lambda}_A \\
&r = 1
\end{aligned}$$

Thus,

$$\begin{aligned}
I_{spike}(A,A) = \frac{1}{\bar{\lambda}_A} I_{sec}(A,A) = \frac{1}{\bar{\lambda}_A} \sum_{s\in S} \Bigg[& p(s)\lambda_a(s)\log_2\left(\frac{\lambda_a(s)}{\bar{\lambda}_A}\right) \\
& + p(s)\left(\lambda_A(s) - \lambda_A(s)\right)\log_2\left(\frac{\lambda_A(s) - \lambda_A(s)}{\bar{\lambda}_A - \bar{\lambda}_A}\right) \\
& + p(s)\left(\lambda_A(s) - \lambda_A(s)\right)\log_2\left(\frac{\lambda_A(s) - \lambda_A(s)}{\bar{\lambda}_A - \bar{\lambda}_A}\right) \Bigg] \\
= \sum_{s\in S} \Bigg[& p(s)\frac{\lambda_a}{\bar{\lambda}_A}(s)\log_2\left(\frac{\lambda_a(s)}{\bar{\lambda}_A}\right) + p(s)0\log(0/0) + p(s)0\log(0/0) \Bigg]
\end{aligned}$$

Lastly, setting terms which are not well-defined to zero,

$$I_{spike}(A,A) = \sum_{s\in S} p(s)\frac{\lambda_a}{\bar{\lambda}_A}(s)\log_2\left(\frac{\lambda_a(s)}{\bar{\lambda}_A}\right) = I_{spike}(A)$$

□

B Additional Proofs

B.1 Proof of Theorem 2.2.1

Proof of Theorem 2.2.1.

Proof. Let $\alpha(s) = \frac{\lambda_A(s)}{\bar{\lambda}_A}$. Then, $\alpha(s) \geq 0$ and $\sum_{s\in S} \alpha(s)p(s) = 1$. Then, equation 2.4 becomes

$$I_{spike}(A:S) = \sum_{s\in S} \alpha(s)p(s)\log_2(\alpha(s)) = D_{KL}(q(s) \mid\mid p(s))$$

where $q(s) = \alpha(s)p(s)$ and D_{KL} is the Kullback-Leiber divergence. This grows when $q(s)$ becomes more concentrated compared to $p(s)$ and is maximized when $q(s)$ is completely concentrated to a single state s^* [11]. That is

$$q(s) = \begin{cases} 1 & s = s^* \\ 0 & s \neq s^* \end{cases}$$

In this scenario, we have $\lambda_A(s^*)p(s^*) = \bar{\lambda}_A$, thus,

$$\begin{aligned} I_{spike}(A:S) &= \sum_{s\in S} \frac{\lambda_A(s)}{\bar{\lambda}_A}\log_2\left(\frac{\lambda_A(s)}{\bar{\lambda}_A}\right)p(s) = \frac{\lambda_A(s^*)}{\lambda_A(s^*)p(s^*)}\log\left(\frac{\lambda_A(s^*)}{\lambda_A(s^*)p(s^*)}\right)p(s^*) \\ &= \log_2\left(\frac{1}{p(s^*)}\right) \end{aligned}$$

This is clearly largest when the chosen state $s^* = \arg\min_{s\in S} p(s)$ and $p(s^*) = \epsilon$.

□

B.2 Proof of Theorem 2.2.3

Proof of Theorem 2.2.3.

Proof. Let $X = \{x_i\}$ and $Y = \{y_i\}$ be vectors in $\mathbb{R}^n$. We hope to show

$$|I_{spike}(X) - I_{spike}(Y)| \leq L||X - Y||_p,$$

for some p-norm $||\cdot||_p$. Now,

$$|I_{spike}(X) - I_{spike}(Y)| = \sum_i \left[\frac{x_i}{\bar{x}} \log_2\left(\frac{x_i}{\bar{x}}\right) - \frac{y_i}{\bar{y}} \log_2\left(\frac{y_i}{\bar{y}}\right)\right] p(i)$$

where $\bar{x} = \sum_i p(i)x_i$ and $\bar{y} = \sum_i p(i)y_i$.

Let $f(t) = t\log_2(t)$ for $t > 0$. This is Lipschitz as there exists C such that $|f'(t)| \leq C$ on any closed and bounded domain $[a, b]$. Thus,

$$|I_{spike}(X) - I_{spike}(Y)| \leq L_1 \sum_i \left|\frac{x_i}{\bar{x}} - \frac{y_i}{\bar{y}}\right| \leq L_1 L_2 \sum_i |x_i - y_i| = L||X - Y||_1.$$

This holds when $\frac{x_i}{\bar{x}}$ is bounded above for all i, which is true for a fixed occupancy distribution. For a general occupancy distribution in which $p(i)$ can be made arbitrarily small, this does not hold.

Thus, for a fixed occupancy distribution, the stimulus information rate is L-Lipschitz.

□

B.3 Proof of Theorem 2.2.5

Proof of Theorem 2.2.5.

Proof. If $\{s \in S : \lambda_A(s) > 0\} \cap \{s \in S : \lambda_B(s) > 0\} = \emptyset$, then $\lambda_{A,B}(s) = \lambda_A(s)\lambda_B(s) = 0$ for all $s \in S$. Therefore, $\tilde{\lambda}_{A,B} = \sum_{s\in S} p(s)\sqrt{\lambda_{A,B}(s)} = 0$ and $\bar{\lambda}_{A,B} = \sum_{s\in S} p(s)\lambda_{A,B}(s) = 0$. Thus,

$$\begin{aligned} I_{sec}(A, B : S) &= \sum_{s\in S} \left[p(s)0 + p(s)\left(\lambda_A(s) - 0\right)\log_2\left(\frac{\lambda_A(s) - 0}{\bar{\lambda}_A - 0}\right) + p(s)\left(\lambda_B(s) - 0\right)\log_2\left(\frac{\lambda_B(s) - 0}{\bar{\lambda}_B - 0}\right)\right] \\ &= \sum_{s\in S} p(s)\left(\lambda_A(s)\right)\log_2\left(\frac{\lambda_A(s)}{\bar{\lambda}_A}\right) + \sum_{s\in S} p(s)\left(\lambda_B(s)\right)\log_2\left(\frac{\lambda_B(s)}{\bar{\lambda}_B}\right) \\ &= I_{sec}(A : S) + I_{sec}(B : S) \end{aligned}$$

And

$$\begin{aligned} I_{spike}(A, B : S) &= \frac{1}{\Lambda_{A,B}} I_{sec}(A, B : S) \\ &= \frac{1}{\Lambda_{A,B}} (I_{sec}(A : S) + I_{sec}(B : S)) \end{aligned}$$

□

B.4 Proof of Theorem 2.2.6

Proof of Theorem 2.2.6.

Proof. Since $I_{spike}(A, B)$ does not change when either neuron's firing rate is multiplied by a constant, we may assume without loss of generality, that $\bar{\lambda}_A = \bar{\lambda}_B = 1$. In this case $I_{spike}(A, B : S) = I_{sec}(A, B : S)$ and $I_{spike}(A : S) = I_{sec}(A : S)$. So we may find the maximum of $I_{sec}(A, B : S)$ subject to $\bar{\lambda}_A = \bar{\lambda}_B = 1$.

Now, recall

$$\begin{aligned} I_{sec}(A, B) = &\sum_s p(s) r \sqrt{\lambda_{A,B}(s)} \log_2 \left(\frac{\sqrt{\lambda_{A,B}(s)}}{\tilde{\lambda}_{A,B}} \right) \\ &+ p(s) \left(\lambda_A(s) - r\sqrt{\lambda_{A,B}(s)} \right) \log_2 \left(\frac{\lambda_A(s) - r\sqrt{\lambda_{A,B}(s)}}{\bar{\lambda}_A - r\tilde{\lambda}_{A,B}} \right) \\ &+ p(s) \left(\lambda_B(s) - r\sqrt{\lambda_{A,B}(s)} \right) \log_2 \left(\frac{\lambda_B(s) - r\sqrt{\lambda_{A,B}(s)}}{\bar{\lambda}_B - r\tilde{\lambda}_{A,B}} \right) \end{aligned}$$

where $\sum_s p(s)\sqrt{\lambda_{A,B}}(s) = \tilde{\lambda}_{A,B}$, and $\lambda_A(s)\lambda_B(s) = \lambda_{A,B}(s)$. The second and third terms dominate this sum, and are maximized when

$$\begin{aligned} \lambda_A(s_1) - r\sqrt{\lambda_{A,B}(s_1)} \neq 0 \\ \lambda_B(s_2) - r\sqrt{\lambda_{A,B}(s_2)} \neq 0 \end{aligned}$$

for particular $s_1, s_2 \in S$. Then, for all remaining $s \in S$,

$$\lambda_A(s) = \lambda_B(s) = r\sqrt{\lambda_{A,B}(s)}$$
$$\lambda_A^2(s) = \lambda_B^2(s) = r^2\lambda_A(s)\lambda_B(s)$$

This can be achieved when $r = 1$ and $\lambda_A(s) = \lambda_B(s)$ for all $s \in S$ or when $r = -1$ and $\lambda_{A,B}(s) = 0$ for all $s \in S$.

In the first case $r = 1$, we have $I_{spike}(A, B) = I_{spike}(A) = \log(1/\epsilon_1)$ where $p(s_1) = \epsilon_1$.

In the second case,

$$\begin{aligned} I_{sec}(A, B : S) &= \sum_s p(s)\lambda_A(s)\log_2\left(\frac{\lambda_A(s)}{\bar{\lambda}_A}\right) + p(s)\lambda_B(s)\log_2\left(\frac{\lambda_B(s)}{\bar{\lambda}_B}\right) \\ &= I_{sec}(A : S) + I_{sec}(B : S) \end{aligned}$$

which, by Theorem 2.2.1, is maximized when $s_1 \neq s_2$. The resulting maximum is

$$\epsilon_1\lambda_A(s_1)\log(1/\epsilon_1) + \epsilon_2\lambda_B(s_2)\log_2(1/\epsilon_2)$$

Now, in this scenario, $\epsilon_1\lambda_A(s_1) = \bar{\lambda}_A = 1$ and $\epsilon_2\lambda_B(s_2) = \bar{\lambda}_B = 1$. For $\epsilon_2 \in (0, 1)$,

$$\log(1/\epsilon_1) + \log_2(1/\epsilon_2) > \log(1/\epsilon_1)$$

Thus, the maximum is achieved when neurons fire in single and separate stimulus bins. It follows that, for a general $\bar{\lambda}_A(s)$ and $\bar{\lambda}_B(s)$, the maximum of $I_{spike}(A, B : S)$ is

$$\frac{2\left(\epsilon_1\lambda_A(s_1)\log_2(1/\epsilon_1) + \epsilon_2\lambda_B(s_2)\log_2(1/\epsilon_2)\right)}{(\bar{\lambda}_A + \bar{\lambda}_B)}$$

□

B.5 Proof of Theorem 2.2.8

Proof of Theorem 2.2.8.

Proof. There are two cases.

First when $i = j$, by we have $I_{spike}(P_i, P_j : S) = I_{spike}(P_i, P_i : S) = I_{spike}(P_i : S)$. Since each neuron only fires in one stimulus bin, by Corollary 2.2.2, $J_{i,i} = I_{spike}(P_i : S) = \log_2(n_s)$ for all $i = 1, \ldots, n_p$.

Secondly, when $i \neq j$, we may invoke Theorem 2.2.5 together. That is,

$$\begin{aligned} J_{i,j} &= \frac{2}{\bar{\lambda}_A + \bar{\lambda}_B} \left(\sum_{s \in S} \lambda_A(s) \log_2 \left(\frac{\lambda_A(s)}{\bar{\lambda}_A} \right) \frac{1}{n_s} + \lambda_B(s) \log_2 \left(\frac{\lambda_B(s)}{\bar{\lambda}_B} \right) \frac{1}{n_s} \right) \\ &= \frac{2}{\bar{\lambda}_A + \bar{\lambda}_B} \left(\bar{\lambda}_A \log_2(n_s) + \bar{\lambda}_B \log_2(n_s) \right) = 2 \log_2(n_s) \end{aligned}$$

Together, we may write

$$J_{i,j} = \log_2(n_s)(2 - \delta_{i,j})$$

where $\delta_{i,j}$ is the Kronecker delta, equal to 1 when $i = j$ and zero otherwise.

□

B.6 Proof of Corollary 2.2.9

Proof of Corollary 2.2.9.

Proof. We can write

$$J = \log_2(n_s) \begin{bmatrix} 1 & 2 & 2 & \ldots & 2 \\ 2 & 1 & 2 & \ldots & 2 \\ 2 & 2 & 1 & \ldots & 2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 2 & 2 & 2 & \ldots & 1 \end{bmatrix} = \log_2(n_s)(2\mathbf{1}\mathbf{1}^T - I)$$

Now, since I and $\mathbf{1}\mathbf{1}^T$ are diagonalizable and commute, they share the same eigenvectors. The only eigenvalue of I is 1. The eigenvalues of $\mathbf{1}\mathbf{1}^T$ are n_p with multiplicity 1 and 0 with multiplicity $n_p - 1$. Thus, the leading eigenvalue of J is $\log_2(n_s)(2n_p - 1)$.

The leading eigenvector corresponds to the eigenvector with eigenvalue n_p of the unit matrix $\mathbf{1}\mathbf{1}^T$, which is the vector $\mathbf{1}$. □

C Experimental details

Our code was implemented in PyTorch [77] and is available at https://github.com/JaredDeightonUTK/Information-PC-Paper. Our methodology closely aligns with the frameworks established in the publicly available codes used in [92, 93] and in [22]. Hyperparameters for our experiments are listed in Table 2. The majority of the code was run using a 2.8 GHz Quad-Core Intel Core i7 processor with 16 GB of memory. Here, training a single spectral spatial information model with 16 place cells and 256 hidden units for 100 epochs takes roughly 16 seconds. Training 64 place cells and 1028 hidden units for 30 epochs takes roughly 2 minutes. Code to train models with 256 place cells and to perform the sensitivity analysis was run using a Tesla K80 GPU with 64GB of memory.

Table 2: Hyperparameters used for training the networks.

Hyperparameter	**Value**
Batch size	40
Sequence length	100
Learning rate	1e-4
Number of place cells	16, 32, 64, 128, 256
Number of hidden units	256, 1028, 1028, 1028, 2048
Periodic boundary conditions	False
Arena Size (m)	0.5
RNN nonlinearity	$ReLU(\cdot)$
Optimizer	Adam [52]
Weight decay	None

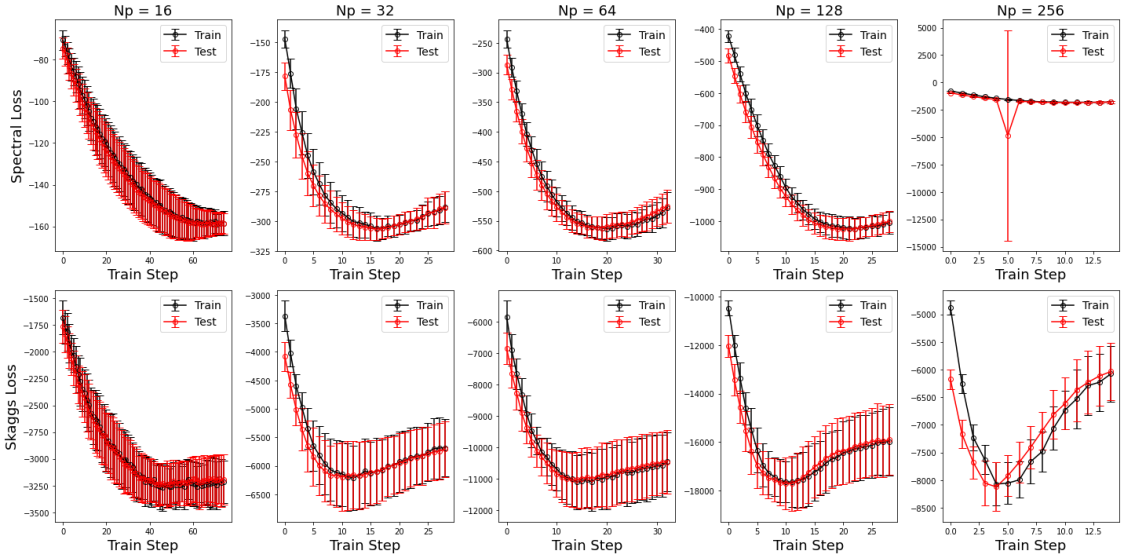

Figure 2: Mean (± standard deviation) train and test loss from 10 experiments for each N_p (number of place cells) and loss function.

Trajectory generation: We generate trajectories in a two-dimensional square arena using previously published codes found in [92, 93]. In short, the agent's initial position is randomly drawn from a uniform distribution across the set of possible positions. Then, a sequence of random speeds and turns is constructed to create the trajectory. Trajectories can be freely generated. We reserve 40 unique trajectories for testing and train the model on 40 unique trajectories for each epoch. Training and testing summaries are in Fig. 2.

Initial Activation The intial place cell activatons P_0 are generated using a difference of softmaxes. First, we randomly choose n_p two-dimensional points in the arena, denoted as $y \in \mathbb{R}^{n_p \times 2}$. Then, for a position $x \in \mathbb{R}^2$, we have

$$(P_0)_i = \text{softmax}\left(\frac{-||x-y_i||^2}{2\sigma_E^2}\right) - \text{softmax}\left(\frac{-||x-y_i||^2}{2\sigma_I^2}\right)$$

for $i = 1, 2, \ldots, n_p$. P_0 is then shifted and scaled such that outputs so that they lie in [0,1].

Leave-One-out classification. Leave-one-out classification for spatial decoding was performed using Scikit-learn's [78] Naive Bayes classifier and KBinsDiscretizer. For each trained model, we pass 10 trajectories (1000 positions) through the network to obtain place cell activations. The 1000 positions are binned using the KBinsDiscretizer with 100 bins in each dimension. Then, all but one of the firing rate-location pairs are used to estimate posterior probabilities (i.e. to fit the classifier). Then, the Naive Bayes classifier predicts the position of the left-out firing rate. This process is repeated so that every firing rate value is used for testing. The predicted positions (given as a predicted bin) are then transformed back to a predicted position in $\mathbb{R}^2$ using KBinsDiscretizer's inverse transform, from which the mean-squared error with the true positions is calculated.

Support Vector Machine. Quadrant classification for spatial decoding was performed using Scikit-learn's [78] support vector machine (SVM) and KBinsDiscretizer. First, we generate many trajectories (1000) to obtain place cell activations at various positions for each model. Now, to create the quadrant classes the positions are binned using the KBinsDiscretizer with 2 bins in each dimension. We use 10000 firing rate-quadrant pairs to fit an SVM with a linear kernel. Then, the SVM is tested on 1000 separate samples to obtain the quadrant classification accuracy.

D Additional place cell maps

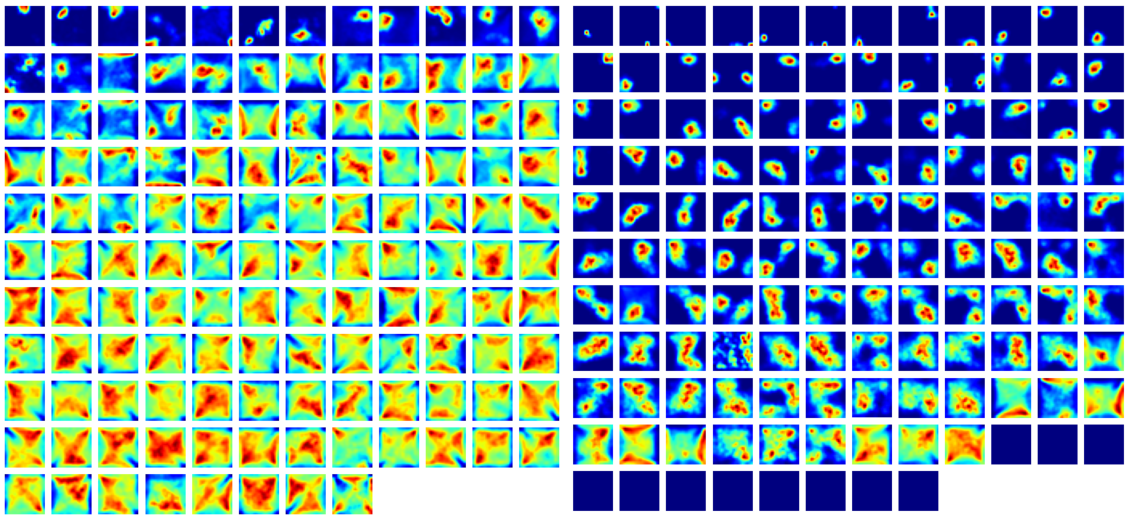

Figure 3: Pre-trained and trained (with spectral spatial information) place cell activations for 128 place cells. Each place cell's activations are normalized between 0 and 1. The top 64 are displayed in Fig. 2.7.

E Joint Spatial Information Rate on real place cell recordings

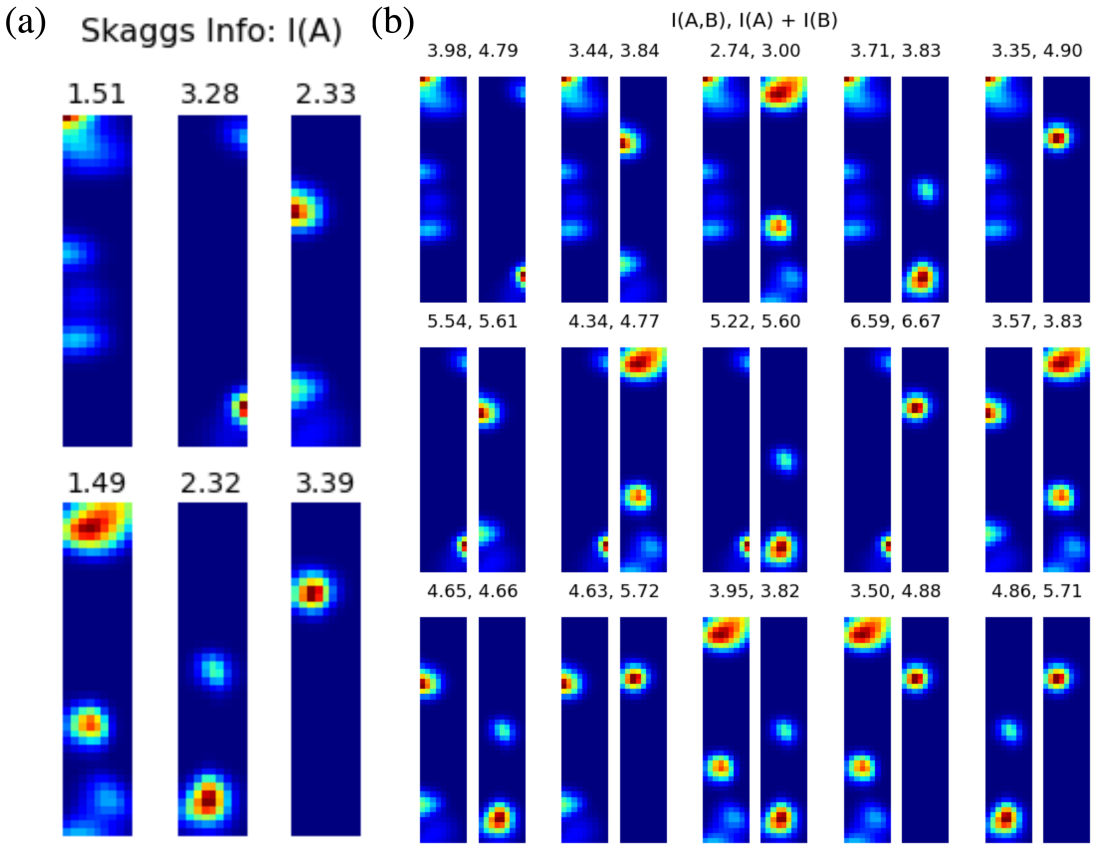

Figure 4: (a) Skaggs' spatial information rate for six place cells recording from [46] located in the right hippocampal CA1 subfield of a male Japanese monkey performing a shuttling task on a rectangular (4×0.9 m) track. (b) Joint spatial information, $I(A, B)$, compared to the sum of Skaggs' spatial information, $I(A) + I(B)$, for all possible pairs of neurons. The leading eigenvalue of the corresponding spatial information matrix is $\lambda_1 = 24.21$.

F Sensitivity

To test the effect of training with spectral spatial information as compared to Skaggs spatial information rate across training parameters, we train three distinct models (each with 32 place cells) for all possible combinations of the following parameters:

- Arena Size (in meters): 0.25, 0.75, 1, 1.5
- Number of Hidden Units (Ng): 256, 512, 1028
- Learning Rate: 0.001, 0.0001, 0.00001
- Batch Size: 10, 40, 80
- Sequence Length: 40, 80, 120

We implement a simple stopping criterion during training, in which training halts if the current epoch loss is greater than the mean of the previous three epoch losses (provided the model has trained for at least five epochs). The mean place cell score (± standard deviation) is presented below across pairs of parameters by averaging over remaining parameters.

Skaggs PC Score ($N_p = 32$)

Arena Size (m) / Ng	0.25	0.75	1	1.5
256	-2.81 ± 6.01	-4.22 ± 6.04	-3.24 ± 5.98	-1.52 ± 4.61
512	-1.93 ± 5.45	-3.65 ± 5.36	-2.6 ± 5.52	-0.48 ± 3.99
1028	-1.67 ± 5.22	-2.98 ± 5.15	-2.11 ± 5.5	0.42 ± 3.64

Learning Rate / Ng	1e-03	1e-04	1e-05
256	**-6.62 ± 5.49**	-0.16 ± 5.45	-2.07 ± 6.03
512	**-4.48 ± 5.26**	-1.43 ± 4.92	-0.58 ± 5.06
1028	**-1.93 ± 4.96**	-1.65 ± 4.93	-1.17 ± 4.75

Batch Size / Ng	10	40	80
256	-6.3 ± 6.66	-4.54 ± 6.01	1.99 ± 4.31
512	-6.09 ± 6.32	-3.75 ± 5.59	3.34 ± 3.34
1028	-5.68 ± 6.38	-3.16 ± 5.47	4.08 ± 2.79

Spectral PC Score ($N_p = 32$)

Arena Size (m) / Ng	0.25	0.75	1	1.5
256	**-1.6 ± 5.54**	**-2.9 ± 6.25**	**-2.21 ± 5.93**	**-0.61 ± 4.47**
512	**-0.46 ± 5.48**	**-2.27 ± 5.96**	**-1.22 ± 5.91**	**0.2 ± 4.1**
1028	**-0.33 ± 5.52**	**-2.35 ± 5.3**	**-1.28 ± 5.76**	**0.68 ± 3.96**

Learning Rate / Ng	1e-03	1e-04	1e-05
256	-6.79 ± 5.3	**2.16 ± 5.75**	**-0.86 ± 5.59**
512	-4.78 ± 5.17	**1.03 ± 5.96**	**0.93 ± 4.96**
1028	-2.25 ± 4.76	**-1.28 ± 5.46**	**1.09 ± 5.18**

Batch Size / Ng	10	40	80
256	**-4.76 ± 6.37**	**-3.16 ± 6.08**	**2.43 ± 4.19**
512	**-4.1 ± 6.67**	**-2.42 ± 5.95**	**3.71 ± 3.47**
1028	**-4.48 ± 6.55**	**-2.16 ± 5.8**	**4.2 ± 3.05**

Sequence Length Ng	40	80	120
256	-1.87 ± 5.28	-2.76 ± 5.68	-4.22 ± 6.02
512	-1.25 ± 4.87	-2.1 ± 4.95	-3.15 ± 5.42
1028	-1.03 ± 4.81	-1.52 ± 4.84	-2.21 ± 4.99

Sequence Length Ng	40	80	120
256	**-0.82 ± 5.26**	**-1.94 ± 5.65**	**-2.73 ± 5.73**
512	**-0.22 ± 5.13**	**-0.88 ± 5.31**	**-1.7 ± 5.65**
1028	**-0.43 ± 5.0**	**-0.82 ± 5.14**	**-1.2 ± 5.27**

Learning Rate Arena Size (m)	1e-03	1e-04	1e-05
0.25	**-3.57 ± 4.67**	1.24 ± 3.66	0.75 ± 3.92
0.75	**-4.03 ± 5.39**	-0.94 ± 5.77	-1.44 ± 5.52
1	**-4.36 ± 5.41**	-1.78 ± 5.76	-1.81 ± 5.83
1.5	**-5.42 ± 5.48**	-2.83 ± 5.22	-2.6 ± 5.85

Learning Rate Arena Size (m)	1e-03	1e-04	1e-05
0.25	-3.68 ± 4.57	**2.12 ± 4.16**	**1.83 ± 3.8**
0.75	-4.3 ± 5.18	**1.42 ± 6.1**	**0.49 ± 5.26**
1	-4.73 ± 5.29	**0.13 ± 6.45**	**-0.1 ± 5.87**
1.5	-5.72 ± 5.27	**-1.13 ± 6.18**	**-0.66 ± 6.06**

Batch Size Arena Size (m)	10	40	80
0.25	-3.54 ± 4.73	-2.0 ± 4.58	3.96 ± 2.94
0.75	-5.73 ± 6.82	-3.32 ± 6.09	2.62 ± 3.77
1	-6.32 ± 7.09	-4.32 ± 6.16	**2.7 ± 3.76**
1.5	-8.5 ± 7.16	-5.63 ± 5.93	3.28 ± 3.46

Batch Size Arena Size (m)	10	40	80
0.25	**-2.66 ± 4.79**	**-1.49 ± 4.81**	**4.43 ± 2.93**
0.75	**-3.7 ± 6.65**	**-1.67 ± 6.12**	**2.97 ± 3.77**
1	**-4.69 ± 7.29**	**-2.68 ± 6.33**	2.67 ± 3.99
1.5	**-6.74 ± 7.4**	**-4.49 ± 6.52**	**3.72 ± 3.58**

Sequence Length Arena Size (m)	40	80	120
0.25	-0.3 ± 4.05	-0.35 ± 3.89	-0.93 ± 4.31
0.75	-1.21 ± 5.34	-2.04 ± 5.44	-3.16 ± 5.9
1	-1.41 ± 5.19	-2.9 ± 5.7	-3.63 ± 6.11
1.5	-2.61 ± 5.36	-3.21 ± 5.61	-5.03 ± 5.58

Sequence Length Arena Size (m)	40	80	120
0.25	**0.1 ± 4.24**	**0.36 ± 4.03**	**-0.18 ± 4.26**
0.75	**-0.09 ± 5.14**	**-0.99 ± 5.65**	**-1.32 ± 5.75**
1	**-0.34 ± 5.5**	**-1.85 ± 5.95**	**-2.51 ± 6.16**
1.5	**-1.63 ± 5.65**	**-2.38 ± 5.82**	**-3.5 ± 6.04**

Batch Size Learning Rate	10	40	80
1e-03	-5.2 ± 6.95	-2.81 ± 5.8	4.77 ± 2.55
1e-04	-5.07 ± 6.71	-2.56 ± 5.66	3.81 ± 3.47
1e-05	**-7.79 ± 5.69**	**-6.08 ± 5.61**	**0.84 ± 4.42**

Batch Size Learning Rate	10	40	80
1e-03	**-2.43 ± 7.63**	**-0.68 ± 6.56**	**5.02 ± 2.97**
1e-04	**-2.73 ± 6.64**	**-0.76 ± 5.8**	**4.65 ± 3.29**
1e-05	-8.18 ± 5.33	-6.31 ± 5.46	0.67 ± 4.44

Sequence Length Learning Rate	40	80	120
1e-03	-0.76 ± 5.25	-0.82 ± 4.9	-1.66 ± 5.15
1e-04	-0.77 ± 4.98	-1.18 ± 5.29	-1.88 ± 5.57
1e-05	**-2.62 ± 4.72**	**-4.38 ± 5.29**	**-6.03 ± 5.7**

Sequence Length Learning Rate	40	80	120
1e-03	**0.86 ± 5.75**	**0.74 ± 5.54**	**0.3 ± 5.88**
1e-04	**0.58 ± 5.14**	**0.31 ± 5.38**	**0.28 ± 5.21**
1e-05	-2.92 ± 4.5	-4.69 ± 5.17	-6.21 ± 5.56

Sequence Length Batch Size	40	80	120
10	4.31 ± 2.83	3.45 ± 3.43	1.65 ± 4.18
40	-3.09 ± 5.55	-3.9 ± 5.78	-4.46 ± 5.74
80	-5.38 ± 6.58	-5.92 ± 6.27	-6.77 ± 6.51

Sequence Length Batch Size	40	80	120
10	**4.47 ± 2.9**	**3.56 ± 3.6**	**2.31 ± 4.21**
40	**-1.97 ± 5.85**	**-2.86 ± 5.98**	**-2.92 ± 6.0**
80	**-3.98 ± 6.65**	**-4.35 ± 6.51**	**-5.02 ± 6.44**

G Dynamic Drug Decoding Data and Code Availability

The datasets generated and analyzed during the current study are available from the corresponding authors on reasonable request. The datasets cannot be used for commercial purposes, and we are unable to fulfill requests that would violate patient confidentiality, ethical guidelines, or patient consent agreements. The code will be made available on GitHub upon publication.

H Dynamic Drug Decoding Supplemental Material

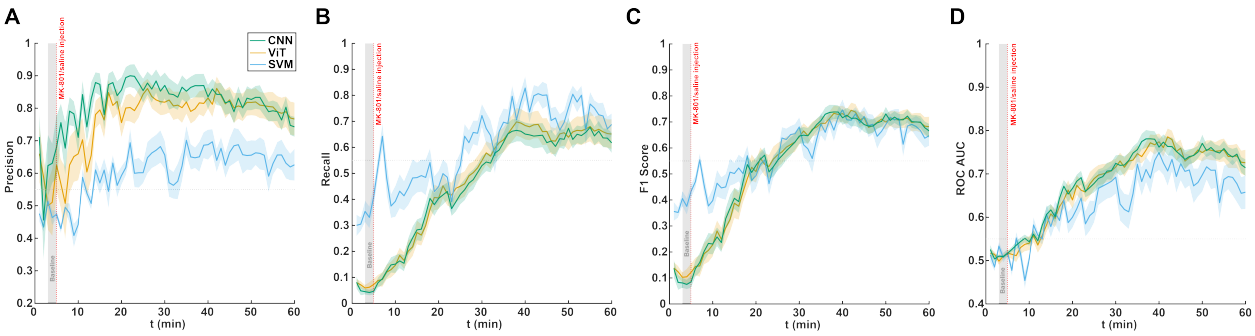

Figure 5: Additional performance metrics for distinguishing MK-801 from saline groups using ViT, SVM, and CNN. (A) Precision, (B) Recall, (C) F1 Score, and (D) ROC AUC. For all panels, shaded areas represent standard error derived from averaging across animals. The vertical dotted line at 5 min marks the injection time, and the gray shaded area (minutes 3-5) indicates the baseline period.

Vita

Jared Deighton was born in Marion, Michigan. He first became interested in mathematics in high school when, during an engineering co-op, he realized he was more interested in mathematics than engineering. He attended Calvin University in Grand Rapids, MI, where he became interested in mathematical modeling. In 2020 he graduated with a Bachelor of Science in Mathematics and proceeded to the University of Tennessee Knoxville for his graduate studies.

In January 2023, he joined the Maroulas Research Group to begin collaboration with the U.S. Army Research Lab.

Jared has accepted a position as an Assistant Professor at Simmons University in the Computer, Data & Mathematical Sciences Department.