Distinguishing examples while building concepts in hippocampal and artificial networks

Louis Kang*[1] and Taro Toyoizumi[2]

[1]Neural Circuits and Computations Unit, RIKEN Center for Brain Science
[2]Laboratory for Neural Computation and Adaptation, RIKEN Center for Brain Science

19 January 2023

Abstract

The hippocampal subfield CA3 is thought to function as an autoassociative network that stores sensory information as memories. This information arrives via the entorhinal cortex (EC), which projects to CA3 directly as well as indirectly through the dentate gyrus (DG). DG sparsifies and decorrelates the information before also projecting to CA3. The computational purpose for receiving two encodings of the same sensory information has not been firmly established. We model CA3 as a Hopfield-like network that stores both correlated and decorrelated encodings and retrieves them at low and high inhibitory tone, respectively. As more memories are stored, the dense, correlated encodings merge along shared features while the sparse, decorrelated encodings remain distinct. In this way, the model learns to transition between concept and example representations by controlling inhibitory tone. To experimentally test for the presence of these complementary encodings, we analyze the theta-modulated tuning of phase-precessing place cells in rat CA3. In accordance with our model's prediction, these neurons exhibit more precise spatial tuning and encode more detailed task features during theta phases with sparser activity. Finally, we generalize the model beyond hippocampal architecture and find that feedforward neural networks trained in multitask learning benefit from a novel loss term that promotes hybrid encoding using correlated and decorrelated representations. Thus, the complementary encodings that we have found in CA3 can provide broad computational advantages for solving complex tasks.

Introduction

The hippocampus is believed to underlie our ability to form episodic memories, through which we can recount personally experienced events from our daily lives (Scoville and Milner, 1957). In particular, the subfield CA3 is thought to provide this capability as an autoassociative network (McNaughton and Morris, 1987; O'Reilly and Rudy, 2001; Rolls and Kesner, 2006). Its pyramidal cells contain abundant recurrent connections exhibiting spike-timing-dependent plasticity (Bi and Poo, 1998; Mishra et al., 2016). These features allow networks to perform pattern completion and recover stored patterns of neural activity from noisy cues. Sensory information to be stored as memories arrives to CA3 via the entorhinal cortex (EC), which serves as the major gateway between hippocampus and neocortex (Fig. 1A). Neurons from layer II of EC project to CA3 via two different pathways (Amaral and Pierre, 2006). First, they synapse directly onto the distal dendrites of CA3 pyramidal cells through the perforant path (PP). Second, before reaching CA3, perforant path axons branch towards the dentate gyrus (DG) and synapse onto granule cells. Granule cell axons form the mossy fibers (MF) that also synapse onto CA3 pyramidal cells, though at more proximal dendrites.

*louis.kang@riken.jp

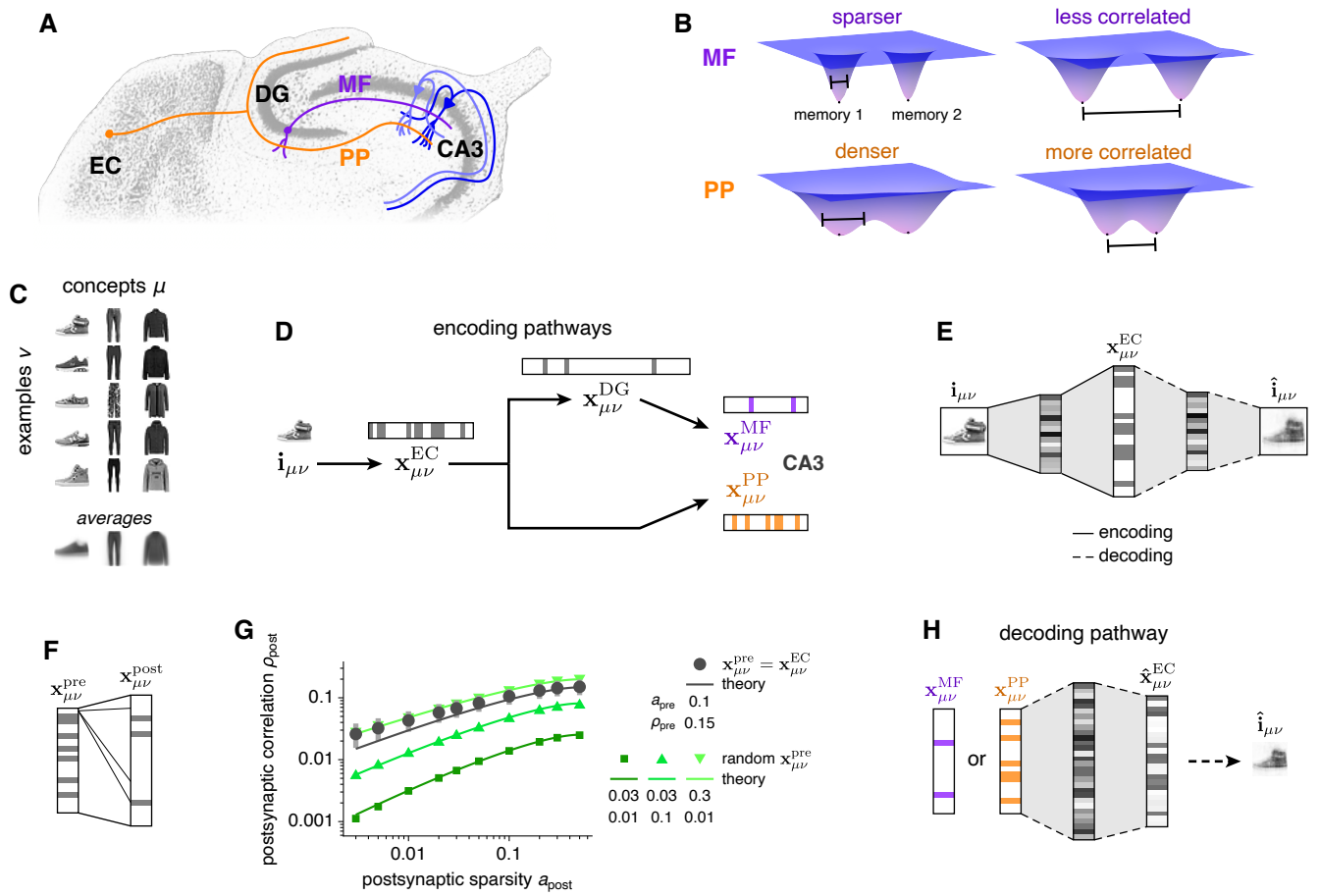

Figure 1: Transformations of memory representations along hippocampal pathways to CA3. (**A**, **B**) Biological observations. (**A**) Entorhinal cortex (EC) projects to CA3 directly via the perforant path (PP, orange) as well as indirectly through the dentate gyrus (DG) via mossy fibers (MF, purple). Adapted from The Mouse Brain Library (Rosen et al., 2000, free use license). (**B**) MF memory encodings are believed to be sparser and less correlated compared to PP encodings. In an autoassociative network, attractor basins of the former tend to remain separate and those of the latter tend to merge. (**C**–**G**) Our model. (**C**) Memories are FashionMNIST images, each of which is an example of a concept. (**D**) Overview of encoding pathways corresponding to **A**. (**E**) We use an autoencoder with a binary middle layer to transform each memory $\mathbf{i}_{\mu\nu}$ into an EC pattern $\mathbf{x}^{\mathrm{EC}}_{\mu\nu}$. (**F**) From EC to CA3, we use random binary connectivity matrices to transform each presynaptic pattern $\mathbf{x}^{\mathrm{pre}}_{\mu\nu}$ to a postsynaptic pattern $\mathbf{x}^{\mathrm{post}}_{\mu\nu}$. (**G**) Enforcing sparser postsynaptic patterns in **F** promotes decorrelation. Dark gray indicates use of $\mathbf{x}^{\mathrm{EC}}_{\mu\nu}$ as presynaptic patterns. Points indicate means and bars indicate standard deviations over 8 random connectivity matrices. Green indicates randomly generated presynaptic patterns at various sparsities a_{pre} and correlations ρ_{pre}. Theoretical curves depict Eq. 1. (**H**) To visualize CA3 encodings, we pass them through a feedforward network trained to produce the corresponding $\mathbf{x}^{\mathrm{EC}}_{\mu\nu}$ for each $\mathbf{x}^{\mathrm{MF}}_{\mu\nu}$ and $\mathbf{x}^{\mathrm{PP}}_{\mu\nu}$. Images are then decoded using the autoencoder in **E**.

Along these pathways, information is transformed by each projection in addition to being simply relayed. DG sparsifies encodings from EC by maintaining high inhibitory tone across its numerous neurons (Engin et al., 2015). Sparsification in feedforward networks generally decorrelates activity patterns as well (Marr, 1971; O'Reilly and McClelland, 1994; Vinje and Gallant, 2000; Pitkow and Meister, 2012; Cayco-Gajic et al., 2017). The sparse, decorrelated nature of DG encodings is preserved by the MF pathway because its connectivity is also sparse; each CA3 pyramidal cell receives input from only ≈50 granule cells (Amaral et al., 1990). In contrast, PP connectivity is dense with each CA3 pyramidal cell receiving input from ≈4000 EC neurons (Amaral et al., 1990), so natural correlations between similar sensory stimuli should be preserved. Thus, CA3 appears to receive two encodings of the same sensory information with different

properties: one sparse and decorrelated through MF and the other dense and correlated through PP. What is the computational purpose of this dual-input architecture? Previous theories have proposed that the MF pathway is crucial for pattern separation during memory storage, but retrieval is predominantly mediated by the PP pathway and can even be hindered by MF inputs (Treves and Rolls, 1992; McClelland and Goddard, 1996; Kaifosh and Losonczy, 2016). In these models, MF and PP encodings merge during storage and one hybrid pattern per memory is recovered during retrieval.

Instead, we consider the possibility that CA3 can store both MF and PP encodings for each memory and retrieve either of them. Inhibitory tone selects between the two; with a higher activity threshold, sparser MF patterns are more likely to be recovered, and the opposite holds for denser PP patterns. By encoding the same memory in two different ways, each can be leveraged for a different computational purpose. Conceptually, in terms of energy landscapes, sparser patterns have narrower attractor basins than denser patterns because fewer neurons actively participate (Fig. 1B). Moreover, less correlated patterns are located farther apart compared to more correlated patterns. Thus, MF energy basins tend to remain separate with barriers between them, a property called pattern separation that maintains distinctions between similar memories and is known to exist in DG (Leutgeb et al., 2007; Aimone et al., 2011). In contrast, PP energy basins tend to merge, which enables the clustering of similar memories into concepts. This proposed ability for CA3 to recall both individual experiences and generalizations across would explain observed features of hippocampal function. For instance, remembering the details of an individual experience with your grandmother is a classic example of hippocampus-dependent episodic memory. Meanwhile, recent research has found that the hippocampus also participates in semantic memory, as evidenced by *grandmother cells* that generalize over your visits and respond to many different representations of your grandmother (Quiroga et al., 2005, 2009).

To instantiate these ideas, we will present a model for EC, DG, and CA3 in which CA3 stores both MF and PP encodings of each memory. We will see that MF encodings remain distinct, whereas PP encodings perform concept learning by merging similar memories. Our model predicts relationships between coding properties and network sparsity across phases of the theta oscillation, which modulates inhibitory tone in the hippocampal region. We will test these predictions across two publicly available datasets (Mizuseki et al., 2013; Karlsson et al., 2015), and each analysis will reveal that tuning of CA3 neurons is sharper during sparse theta phases and broader during dense phases. This supports our model and enriches our understanding of phase coding in hippocampus. While our model does not include CA1, we present comparative experimental analyses for this subfield in various Supplementary Figures. Finally, we will apply inspiration from CA3 toward machine learning and introduce a novel plug-and-play loss function that endows artificial neural networks with both correlated and decorrelated representations. These networks can perform better in multitask learning compared to networks with single representation types, which suggests a promising strategy for helping neural networks to solve complex tasks.

Results

MF encodings remain distinct while PP encodings build concepts in our model for CA3

We model how representations of memories are transformed along the two pathways from EC to CA3 and then how the resultant encodings are stored and retrieved in CA3. First, we focus on the transformations

between memories and their CA3 encodings. The sensory inputs that constitute memories in our model are FashionMNIST images (Xiao et al., 2017), each of which is an example belonging to one of three concepts: sneakers, trousers, and coats (Fig. 1C). They are converted to neural activity patterns along each projection from EC to CA3 (Fig. 1D). Our neurons are binary with activity values of 0 or 1. Each image $\mathbf{i}_{\mu\nu}$ representing example ν in concept μ is first encoded by EC using a binary autoencoder (Fig. 1E). Its middle hidden layer activations represent the patterns $\mathbf{x}^{\mathrm{EC}}_{\mu\nu}$. From EC, we produce DG, MF, and PP encodings with random, binary, and sparse connectivity matrices between presynaptic and postsynaptic regions (Fig. 1F). Each matrix transforms presynaptic patterns into postsynaptic inputs, which are converted into postsynaptic patterns at a desired sparsity using a winners-take-all approach. That is, the postsynaptic neurons receiving the largest inputs are set to 1 and the others are set to 0. Enforcing a desired postsynaptic sparsity is equivalent to adjusting an activity threshold. At CA3, two encodings for each image converge: $\mathbf{x}^{\mathrm{MF}}_{\mu\nu}$ with sparsity 0.02 and $\mathbf{x}^{\mathrm{PP}}_{\mu\nu}$ with sparsity 0.2. Sparsity is the fraction of active neurons, so lower values correspond to sparser patterns. The network parameters in our model are fully described in the Methods section; they are chosen to follow biologically observed trends.

Not only are MF patterns sparser, they are less correlated with average correlation 0.01, compared to a corresponding value of 0.09 for PP patterns. Such an association between sparsification and decorrelation has been widely reported across many theoretical models and brain regions (O'Reilly and McClelland, 1994; Vinje and Gallant, 2000; Pitkow and Meister, 2012; Cayco-Gajic et al., 2017). It is also captured by our simulations in which we either take $\mathbf{x}^{\mathrm{EC}}_{\mu\nu}$ to serve as $\mathbf{x}^{\mathrm{pre}}_{\mu\nu}$ or randomly generate $\mathbf{x}^{\mathrm{pre}}_{\mu\nu}$'s at various sparsities and correlations (Fig. 1G). Decreasing postsynaptic sparsity (sparsification) correspondingly decreases the postsynaptic correlation (decorrelation) for any presynaptic statistics. We contribute further insight by deriving an explicit mathematical formula that connects sparsities and correlations of patterns in presynaptic and postsynaptic networks:

$$\rho_{\mathrm{post}} = \frac{\Gamma\left[\sqrt{2}\,\mathrm{erfc}^{-1}(2a_{\mathrm{post}}),\ a_{\mathrm{pre}} + \rho_{\mathrm{pre}} - a_{\mathrm{pre}}\rho_{\mathrm{pre}}\right] - a_{\mathrm{post}}^2}{a_{\mathrm{post}}(1 - a_{\mathrm{post}})},$$
$$\text{where} \quad \Gamma[\phi, \sigma] = \frac{1}{2\pi}\int_{\arccos\sigma}^{\pi} \mathrm{d}\psi\, \exp\left[-\frac{\phi^2}{1+\cos\psi}\right] \tag{1}$$

and erfc^{-1} is the inverse complementary error function. In other words, given the sparsity a_{pre} and correlation ρ_{pre} of the presynaptic patterns and the desired sparsity a_{post} of the postsynaptic patterns, the postsynaptic correlation ρ_{post} is determined. Equation 1 is remarkable in that only these four quantities are involved, revealing that at least in some classes of feedforward networks, other parameters such as network sizes, synaptic density, and absolute threshold values do not contribute to decorrelation. It is derived in Supplementary Methods, and its behavior is further depicted in Fig. S1A, B.

Ultimately, the encoding pathways in Fig. 1D–G provide CA3 with a sparse, decorrelated $\mathbf{x}^{\mathrm{MF}}_{\mu\nu}$ and a dense, correlated $\mathbf{x}^{\mathrm{PP}}_{\mu\nu}$ for each memory, in accordance with our biological understanding (Fig. 1A, B). Next, we will store these patterns in an autoassociative model of CA3. To intuitively evaluate memory retrieval, it will be useful to decode CA3 representations back into images. To do so, we train a continuous-valued feedforward network to associate each MF and PP pattern with its corresponding EC pattern (Fig. 1H). From there, the reconstructed EC pattern can be fed into the decoding half of the autoencoder in Fig. 1E to recover the image encoded by CA3. This decoding pathway is for visualization and is not designed to mimic biology, although there may be parallels with the output pathway from CA3 to deep layers of EC through CA1 (Amaral and Pierre, 2006).

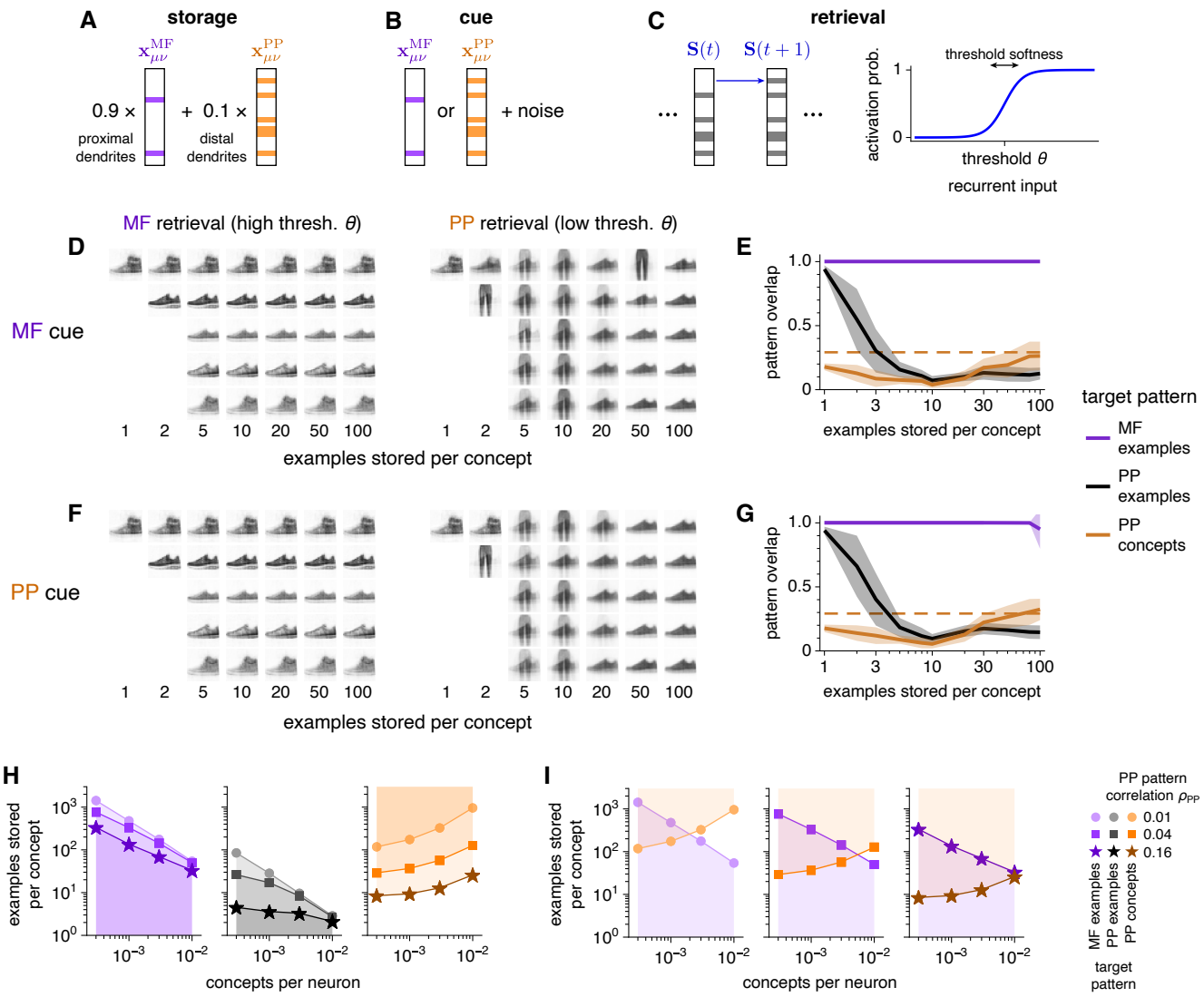

Figure 2: In a model for CA3 that stores both MF and PP encodings of the same memories, MF examples remain distinct while PP examples build concept representations. (**A**–**C**) Overview of the Hopfield-like model for CA3. (**A**) We store linear combinations of MF and PP encodings, with greater weight on the former because MF inputs are stronger. (**B**) Retrieval begins by initializing the network to a stored pattern corrupted by flipping the activity of randomly chosen neurons. (**C**) During retrieval, the network is asynchronously updated with a threshold θ that controls the desired sparsity of the recalled pattern. (**D**, **E**) Retrieval behavior using MF cues. Examples from the three concepts depicted in Fig. 1C are stored. (**D**) Visualizations of retrieved patterns. MF encodings, retrieved at high θ, maintain distinct representations of examples. PP encodings, retrieved at low θ, merge into concept representations as more examples are stored (compare with average image in Fig. 1C). (**E**) Overlap of retrieved patterns with target patterns: MF examples, PP examples, or PP concepts defined by averaging over PP examples and binarizing (Methods). Solid lines indicate means, shaded regions indicate standard deviations, and the dashed orange line indicates the theoretically estimated maximum value for concept retrieval (Methods). In all networks, up to 30 cues are tested. (**F**, **G**) Similar to **D**, **E**, but using PP cues. (**H**) Network capacities computed using random MF and PP patterns instead of FashionMNIST encodings. Shaded regions indicate regimes of high overlap between retrieved patterns and target patterns (Supplementary Methods). MF patterns have sparsity 0.01 and correlation 0. PP patterns have sparsity 0.5. (**I**) Similar to **H**, but overlaying capacities for MF examples and PP concepts to highlight the existence of regimes in which both can be recovered.

Now, we model memory storage in the CA3 autoassociative network. For each example ν in concept μ, its MF encoding $\mathbf{x}^{\mathrm{MF}}_{\mu\nu}$ arrives at the proximal dendrites and its PP encoding $\mathbf{x}^{\mathrm{PP}}_{\mu\nu}$ arrives at the distal dentrites of CA3 pyramidal cells. For simplicity, we assume linear integration of inputs from the two dendritic

compartments (Fig. 2A). The relative strength of PP inputs is weaker because since PP synapses are located more distally and are observed to be much weaker than MF synapses, which are even called detonator synapses (Amaral and Pierre, 2006; Henze et al., 2002; Vyleta et al., 2016). The superimposed activities are stored in a Hopfield-like network (Hopfield, 1982), with connectivity

$$W_{ij} \sim \sum_{\mu\nu} \left(0.9\, x^{\mathrm{MF}}_{\mu\nu i} + 0.1\, x^{\mathrm{PP}}_{\mu\nu i}\right)\left(0.9\, x^{\mathrm{MF}}_{\mu\nu j} + 0.1\, x^{\mathrm{PP}}_{\mu\nu j}\right), \tag{2}$$

where i and j are respectively postsynaptic and presynaptic neurons. Equation 2 captures the most crucial terms in W_{ij}; see Methods for the full expression.

In previous models, CA3 would retrieve only MF encodings, only PP encodings, or only the activity common between MF–PP pairs (Treves and Rolls, 1992; McClelland and Goddard, 1996; Kaifosh and Losonczy, 2016). We assess the ability of the network to retrieve either $\mathbf{x}^{\mathrm{MF}}_{\mu\nu}$ or $\mathbf{x}^{\mathrm{PP}}_{\mu\nu}$ using either encoding as a cue (Fig. 2B). Each cue is corrupted by flipping randomly chosen neurons between active and inactive and is set as the initial network activity. During retrieval, the network is asynchronously updated via Glauber dynamics (Amit et al., 1985). That is, at each simulation timestep, one neuron is randomly selected to be updated (Fig. 2C). If its total input from other neurons exceeds a threshold θ, then it is more likely to become active. The width of the sigmoid function in Fig. 2C determines the softness of the threshold. A large width implies that activation and inactivation are almost equally likely for recurrent input near threshold. A small width implies that activation is almost guaranteed for recurrent input above threshold and almost impossible for input below threshold. See Methods for the full expression of this update rule.

The threshold θ represents the general inhibitory tone of CA3 and plays a key role in retrieval. At high θ, neural activity is disfavored, so we expect the network to retrieve the sparser, more strongly stored MF encoding of the cue. Upon lowering θ, more neurons are permitted to activate, so those participating in the denser, more weakly stored PP encoding should become active as well. This combined activity of both encodings almost the same as the PP encoding alone, which contains many more active neurons. Thus, we expect the network to approximately retrieve the PP encoding at low θ.

Figure 2D–G illustrates the central behavior of our CA3 model; see Fig. S2A, B for trouser and coat visualizations, which behave similarly to the sneaker visualizations shown here. First using MF encodings as cues, we seek to retrieve either MF or PP encodings by respectively setting a high or low threshold. As we load the network with increasingly more stored examples, distinct MF examples can consistently be retrieved with high threshold (Fig. 2D). Meanwhile, retrieval of PP examples with low threshold fails above 1–2 examples stored per concept. At large example loads, the network again retrieves a sneaker memory when cued with sneaker examples. However, this memory is the same for all sneaker cues and, in fact, appears similar to the average image over all sneaker examples (Fig. 1C). Thus, the network is retrieving a representation of the sneaker *concept*. Notably, concepts are never directly presented to the network; instead, the network builds them through the unsupervised accumulation of correlated examples. The retrieval properties visualized in Fig. 2D are quantified in Fig. 2E by computing the overlap between retrieved and target patterns. Across all example loads shown, retrieved MF patterns overlap with target examples. As example load increases, retrieved PP patterns transition from encoding examples to representing concepts. We define the target pattern for a PP concept by activating the most active neurons across PP examples within that concept until the PP sparsity is reached, and the dotted line in Fig. 2E estimates the largest overlap achievable (Methods).

The network capabilities observed for MF cues are preserved when we instead use PP cues (Fig. 2F, G)

or cues combining the neurons active in either encoding (Fig. S2C–E); again, these latter two are similar because MF encodings are sparse. Thus, retrieval behavior is driven largely by the level of inhibition rather than the encoding type of cues. This feature implies that our model is agnostic to whether memory retrieval in the hippocampus is mediated by the MF pathway, the PP pathway, or both. Computationally, it implies that our model can not only retrieve two encodings for each memory but also perform heteroassociation between them.

We investigate retrieval more comprehensively by randomly generating MF and PP patterns across a broader range of statistics instead of propagating images along the hippocampal pathways in Fig. 1D (Methods). For simplicity, we take MF examples to be uncorrelated. In Fig. 2H, I, we show regimes for successful retrieval of MF examples, PP examples, and PP concepts. For MF and PP examples, the network has a capacity for stored patterns above which they can no longer be retrieved (Fig. 2H). For PP concepts, the network requires storage of a minimum number of examples below which concepts cannot be built. As expected intuitively, fewer examples are needed if they are more correlated, since common features can be more easily deduced. Figure 2I overlays retrieval regimes for MF examples and PP concepts. When the number of concepts is low, there exists a regime at intermediate numbers of stored examples in which both examples and concepts can be retrieved. This multiscale retrieval regime corresponds to the network behavior observed in Fig. 2D–G, and it is larger for more correlated PP encodings. On the other hand, its size does not substantially change with the sparsity of MF patterns (Fig. S2F, G). Using techniques from statistical physics, we can calculate the capacity for each type of pattern, and these theoretical results agree with our simulation (Kang and Toyoizumi, 2023).

To further explore the heteroassociative capability of our network, we cue the network with an MF pattern and apply a time-varying threshold during retrieval. The network representation can then alternate between the PP concept of the original cue during oscillation phases with low threshold and various MF examples of that concept during phases with high threshold (Fig. 3A, B). Sharply and sinusoidally varying threshold values both produce this behavior. From one oscillation cycle to the next, the MF encoding can hop among different examples because concept information is preferentially preserved over example information during low-threshold phases. If we weakly apply the MF cue as additional neural input throughout the simulation (Methods), the network will only alternate between the target MF example and the target PP concept. This condition can represent memory retrieval with ongoing sensory input. We quantify the distribution of network behaviors during high- and low-threshold phases in Fig. 3C. The proportion of simulations in which single MF patterns are retrieved, the persistence of the target PP concept, and other retrieval properties vary with network parameters. In Fig. S3A, B, we present analogous results for randomly generated MF and PP patterns demonstrating that these retrieval properties also depend on MF pattern sparsity. All in all, while our network can represent either examples or concepts at each moment in time, an oscillating threshold provides access to a range of representations over every oscillation cycle.

Place cell data reveals predicted relationships between encoding properties and theta phase

The central feature of our CA3 model is that an activity threshold determines whether the network retrieves example or concept encodings. We claim that the theta oscillation in CA3 physiologically implements this threshold and drives changes in memory scale. To be specific, our model predicts that single neurons should convey more information per spike about example identity during epochs of sparser activity (Fig. 4A). This single-neuron prediction can be tested by analyzing publicly available datasets of CA3 place cells. Figure 4B

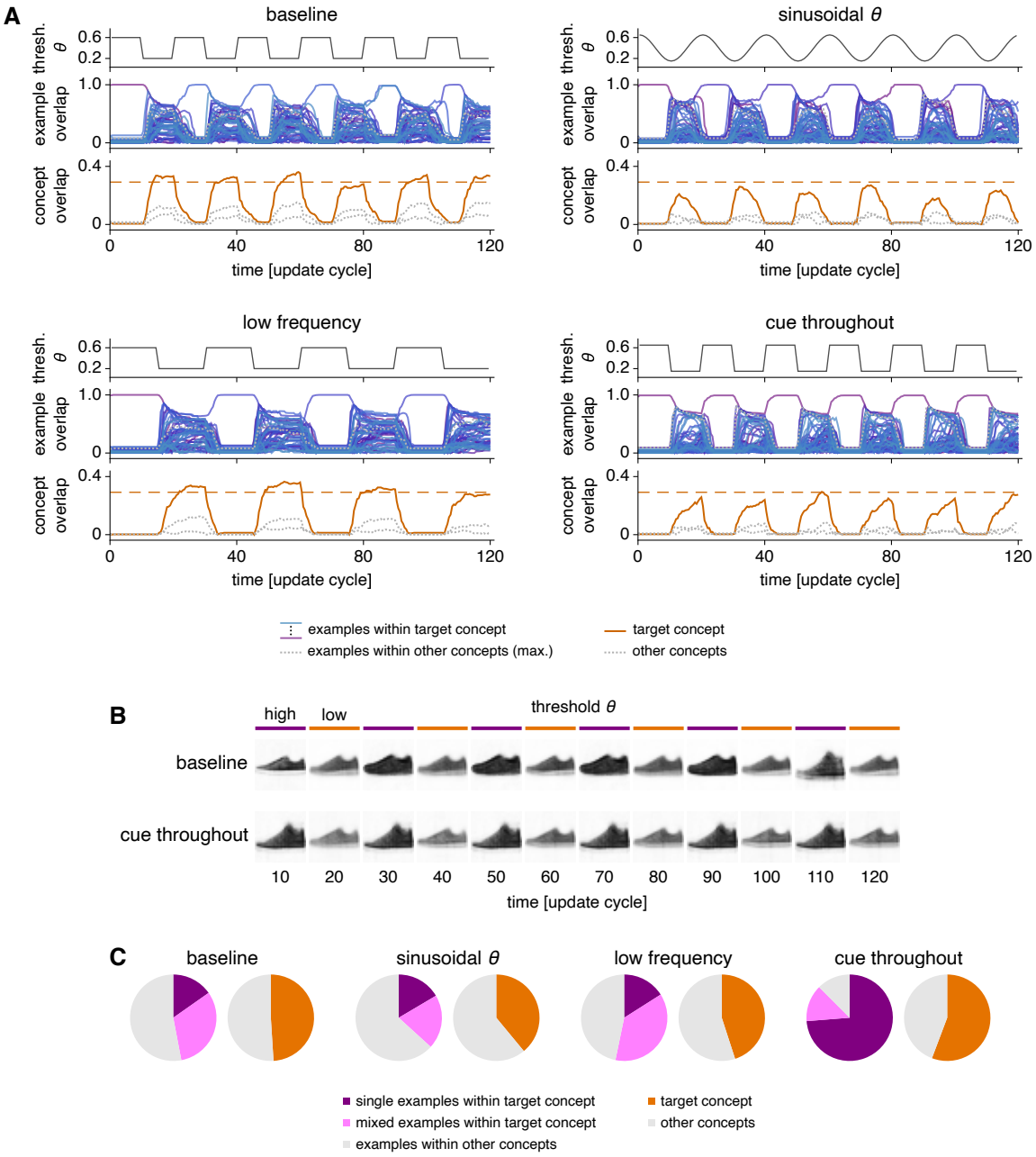

Figure 3: The CA3 model can alternate between MF example and PP concept representations under an oscillating threshold. Four scenarios are considered: a baseline condition with abrupt threshold changes, sinusoidal threshold changes, less frequent threshold changes, and the weak input of an MF cue throughout the simulation instead of only at the beginning. (**A**) Pattern overlap dynamics. Each panel shows, from top to bottom, the threshold, overlaps with MF examples, and overlaps with PP concepts. The dashed orange line indicates the theoretically estimated maximum value for concept retrieval (Methods). (**B**) Visualizations of retrieved patterns show alternation between examples and concepts. In the baseline case, various examples are explored; in the cue-throughout case, the same cued example persists. (**C**) Summary of retrieval behavior between update cycles 60 to 120. For each scenario, 20 cues are tested in each of 20 networks. Each panel depicts the fractions of simulations demonstrating various example (left) and concept (right) behaviors. In all networks, 50 randomly chosen examples from each of the 3 concepts depicted in Fig. 1C are stored. One update cycle corresponds to the updating of every neuron in the network (Methods).

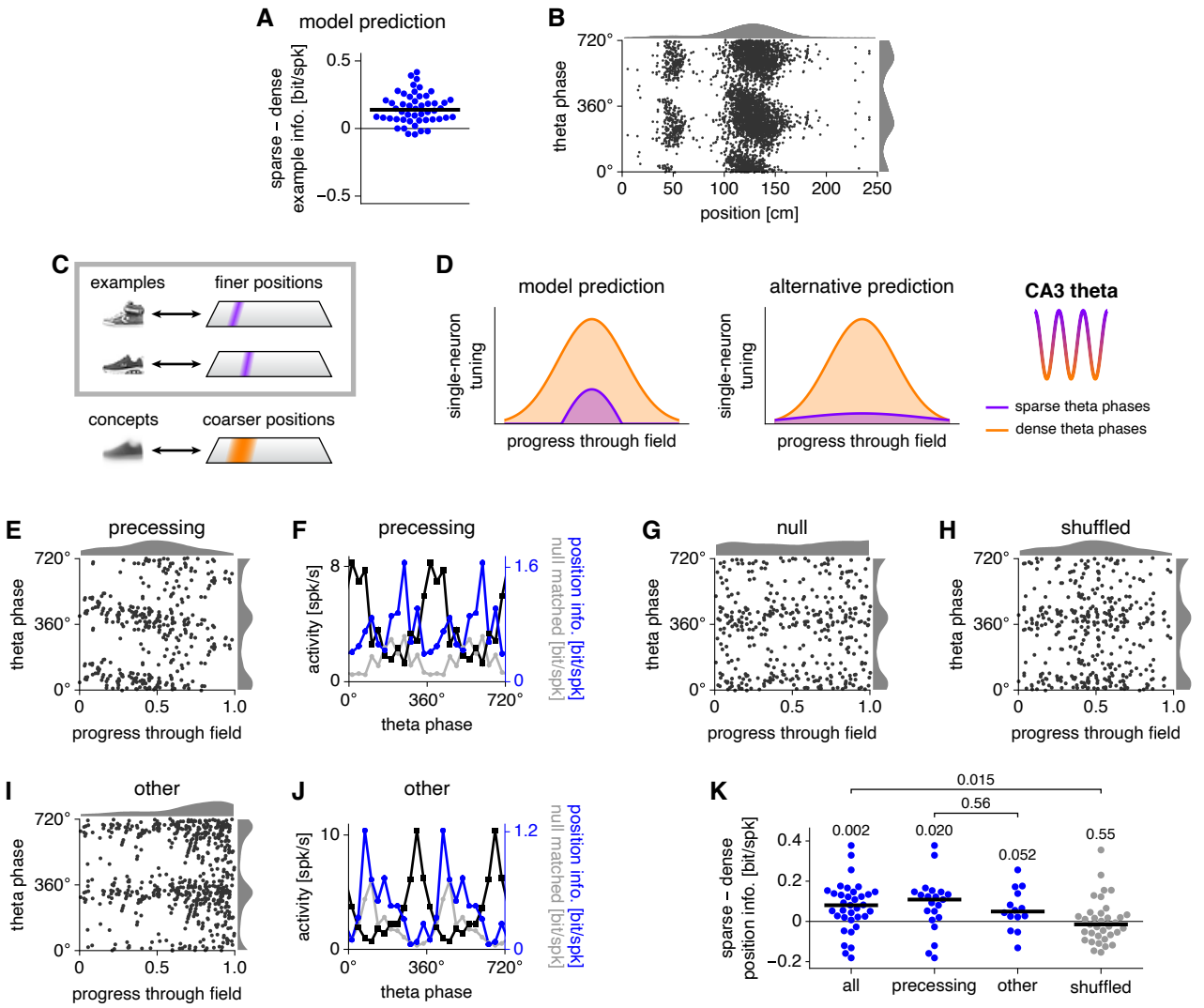

Figure 4: Place field data support the model prediction that sparser theta phases should preferentially encode finer, example-like positions. (**A**) Our CA3 model predicts that single neurons convey more information per spike about example identity during sparse regimes. Each point represents a neuron. (**B**) Example CA3 place cell activity along a linear track. Each spike is represented by two points at equivalent phases. Histogram over position (top) reveals place fields. Histogram over theta phase (right) reveals variation in sparsity. (**C**) To test our model, we construe CA3 place cells to store fine positions as examples, which can combine into coarser regions as concepts. Here, we focus on example encoding. (**D**) Our model predicts that CA3 place fields are more sharply tuned during sparse theta phases. An alternative hypothesis is sharper tuning during dense phases. (**E**) Example phase-precessing place field. (**F**) Activity (black), raw position information per spike (blue), and mean null-matched position information (gray) by theta phase for the field in **E**. Sparsity-corrected position information is the difference between the raw and mean null-matched values. (**G**) Null-matched place field obtained by replacing spike positions, but not phases, with uniformly distributed random values. (**H**) Shuffled place field obtained by permuting spike phases and positions. (**I**, **J**) Similar to **E**, **F**, but for a place field that is not precessing. (**K**) Average difference in position information between the sparsest and densest halves of theta phases. For all cell populations, sparse phases convey more position information per spike. Each point represents a field. Numbers indicate p-values calculated by Wilcoxon signed-rank tests except for the comparison between precessing and other, which is calculated by the Mann-Whitney U test. For all results, spikes during each traveling direction are separately analyzed. In **A** and **K**, information is sparsity-corrected with horizontal lines indicating medians.

shows one example place cell recorded while a rat traverses a linear track (Mizuseki et al., 2013, 2014). Note that its activity is strongly modulated by the theta oscillation; we use single-neuron activity as an indicator of network sparsity since a direct relationship between the two has been observed (Fig. 3 in Skaggs et al.,

1996). We assume an equivalence between the encoding of images by our CA3 model and the encoding of spatial positions by CA3 place cells (Fig. 4C). Examples are equivalent to fine positions along the linear track. Just as similar examples merge into concepts, nearby positions can aggregate into coarser regions of space. Through this equivalence, we can translate the prediction about example information per spike (Fig. 4A) into a prediction about spatial tuning (Fig. 4D). During denser theta phases, place fields should be broader, which corresponds to lower position information per spike. This prediction relies on our claim that the theta oscillation in CA3 acts as the inhibitory threshold of our model. A priori, the alternative prediction that place fields are sharper during dense theta phases is equally valid. Higher activity may result from strong drive by external stimuli that the neuron serves to encode, while lower activity may reflect noise unrelated to neural tuning. The sharpening of visual tuning curves by attention is an example of this alternative prediction (McAdams and Maunsell, 1999).

First, we investigate the encoding of fine, example-like positions by analyzing phase-dependent tuning within single place fields. We use the Collaborative Research in Computational Neuroscience (CRCNS) hc-3 dataset contributed by György Buzsáki and colleagues (Mizuseki et al., 2013, 2014). Figure 4E shows one extracted field that exhibits phase precession (for others, see Fig. S4A). At each phase, we compute the total activity as well as the information per spike conveyed about position within the field (Fig. 4F and Fig. S4B). It is well known that the estimation of information per spike is strongly biased by sparsity. Consider the null data in Fig. 4G that is matched in spike phases; spike positions, however, are randomly chosen from a uniform distribution. In the large spike count limit, uniformly distributed activity should not convey any information. Yet, the null data show more position information per spike during sparser phases (Fig. 4F). To correct for this bias, we follow previous protocols and subtract averages over many null-matched samples from position information (Dotson and Yartsev, 2021). In all of our comparisons of information between sparse and dense phases, including the model prediction in Fig. 4A, we report sparsity-corrected information. For further validation, we generate a shuffled dataset that disrupts any relationship between spike positions and phases found in the original data (Fig. 4H). Figure 4I, J illustrates a second place field whose tuning also depends on theta phase but does not exhibit precession. For each theta-modulated CA3 place field, we partition phases into sparse and dense halves based on activity, and we average the sparsity-corrected position information per spike across each partition. CA3 place fields convey significantly more information during sparse phases than dense phases (Fig. 4K). This relationship is present in both phase-precessing and other fields (although slightly non-significantly in the latter) and is absent in the shuffled data. Thus, experimental data support our model's prediction that CA3 encodes information in a finer, example-like manner during sparse theta phases. Notably, CA1 place fields do not convey more information per spike during sparse phases, which helps to show that our prediction is nontrivial and demonstrates that the phase behavior in CA3 is not just simply propagated forward to CA1 (Fig. S4C).

To characterize the relationship between information and theta phase more precisely, we aggregate spikes over phase-precessing fields in CA3 and in CA1 (Fig. S4D–G). This process implicitly assumes that each phase-precessing field is a sample of a general distribution characteristic to each region. These aggregate fields recapitulate the single-neuron results that CA3 spikes are uniquely more informative during sparse phases (Fig. S4H). They also reveal how position information varies with other field properties over theta phases (Fig. S4I, J). For example, information is inversely correlated with field width, confirming the interpretation that more informative phases have sharper tuning curves (Fig. 4D). In CA3, information is greatest during early progression through the field, which corresponds to future locations, with a smaller peak during late progression, which corresponds to past locations. In contrast, past locations are more sharply tuned in CA1.

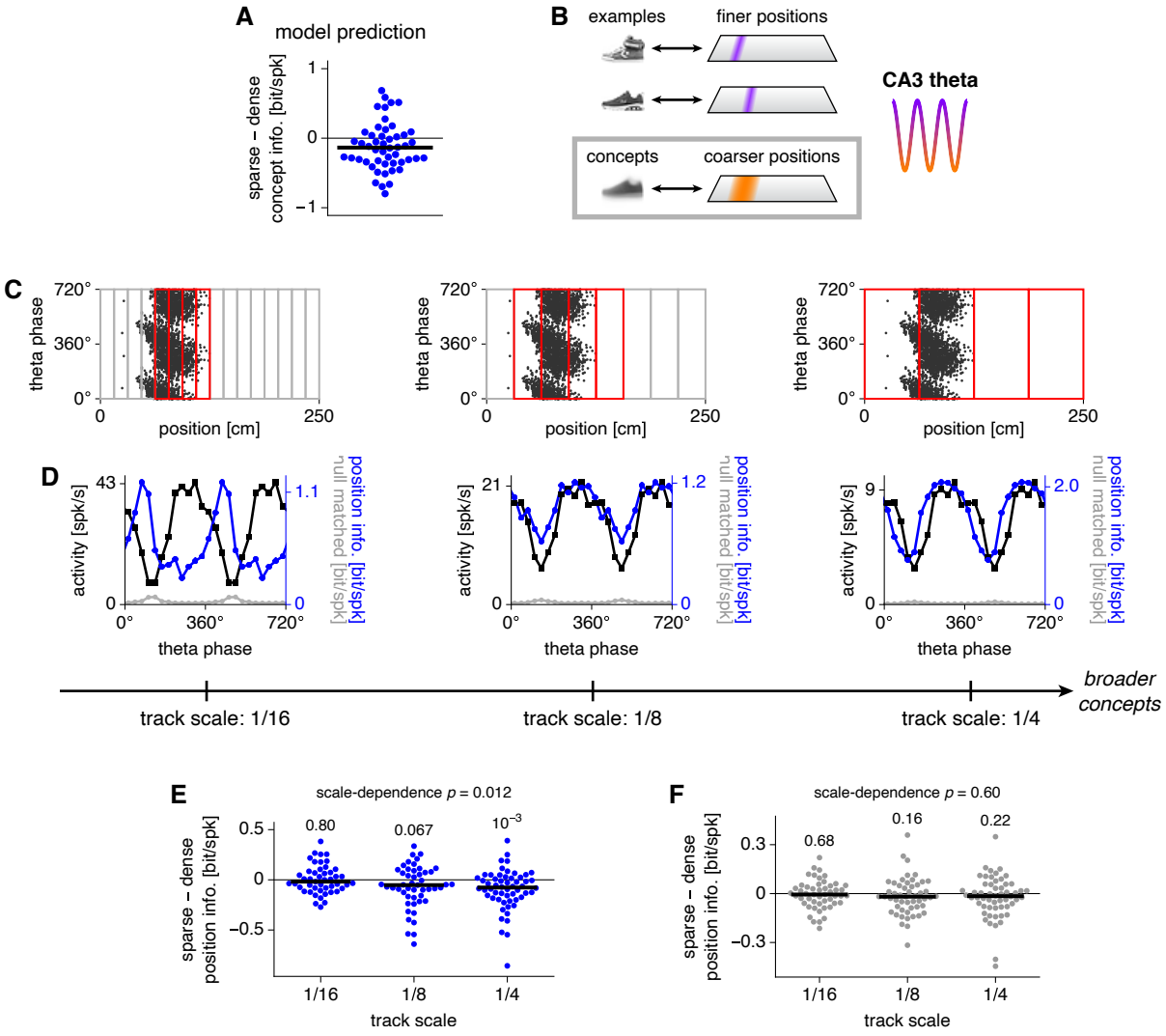

Figure 5: Place cell data support the model prediction that denser theta phases should preferentially encode coarser, concept-like positions. (**A**) Our CA3 model predicts that single neurons convey more information per spike about concept identity during dense regimes. Each point represents a neuron. (**B**) To test our model, we construe CA3 place cells to store fine positions as examples, which can combine into coarser regions as concepts. Here, we focus on concept encoding. (**C**) We calculate position information at various track scales over windows of 4 contiguous bins. (**D**) Activity (black), raw position information per spike (blue), and mean null-matched position information (gray) by theta phase for the red windows in **C**. Sparsity-corrected position information is the difference between the raw and mean null-matched values. (**E**) Average difference in position information between the sparsest and densest halves of theta phases. For coarser scales, dense phases convey more position information per spike. Each point represents values from a place cell averaged over all windows. Numbers indicate p-values calculated by Wilcoxon signed-rank tests for each scale and by Spearman's ρ for the trend across scales. (**F**) Similar to **E**, but for shuffled data whose spike phases and positions are permuted. For all results, spikes during each traveling direction are separately analyzed. In **A**, **E**, and **F**, information is sparsity-corrected with horizontal lines indicating medians.

Thus, different hippocampal subfields may differentially encode past and future positions across the theta cycle; we will return to this topic in the Discussion.

Next, we turn our attention to the representation of concepts instead of examples. Our model predicts that single neurons exhibit more concept information per spike during dense activity regimes (Fig. 5A). To test this prediction using the same CRCNS hc-3 dataset, we invoke the aforementioned equivalence between concepts in our model and coarser positions along a linear track (Fig. 5B). Thus, single CA3 neurons should

encode more information per spike about coarse positions during dense theta phases. Previously, to test for finer position encoding in Fig. 4, we divided single place fields into multiple position bins during the computation of information. Here, we analyze encoding of coarser positions by choosing large position bins across the whole track (Fig. 5C, D and Fig. S5A). We consider different bin sizes to characterize at which scale the merging of examples into concepts occurs. When we again compute the average difference in sparsity-corrected position information per spike between sparse and dense theta phases, we find that dense phases are the most preferentially informative at the coarsest scales (Fig. 5E). CA1 place cells also exhibit this property (Fig. S5B). Crucially, differences between sparse and dense phases are not seen in shuffled data, which supports the validity of our analysis methods (Fig. 5F). Our results are further bolstered by their preservation under a different binning procedure (Fig. S4C–E). Thus, coarse positions along a linear track can be best distinguished during dense theta phases, in agreement with our model. Note that we always consider 4 bins at a time even for track scales smaller than 1/4, because changing the number of bins across scales introduces a bias in the shuffled data (Fig. S5F–H). At the finest scales, this process sometimes fails to capture entire fields and artificially partitions them (Fig. 5C, left), which explains why sparse phases do not convey more information as they do in Fig. 4K.

In our model, concepts are formed by merging examples across all correlated features. While track position can be one such feature, we now assess whether our predictions also apply to another one. In the CRCNS hc-6 dataset contributed by Loren Frank and colleagues, CA3 place cells are recorded during a W-maze alternation task in which mice must alternately visit left and right arms between runs along the center arm (Karlsson et al., 2015). It is known that place cells along the center arm can encode the turn direction upon entering or leaving the center arm in addition to position (Frank et al., 2000; Wood et al., 2000). Again, our model predicts that sparse theta phases preferentially encode specific information (Fig. 6A), so they should be more tuned to a particular turn direction (Fig. 6B). During dense phases, they should generalize over turn directions and solely encode position.

Figure 6C shows spikes from one CA3 place cell accumulated over outward runs along the center arm followed by either left or right turns (Fig. S6A). For each theta phase, we compute the total activity, the turn information per spike (ignoring position), and the mean information of null-matched samples used for sparsity correction (Fig. 6D). Figure 6E, F show similar results for inward runs (for others, see Fig. S6B, C). For both outward and inward runs, sparsity-corrected turn information per spike is greater during sparse theta phases compared to dense phases (Fig. 6G). This finding is not observed in data in which theta phase and turn direction are shuffled (Fig. 6G, H). Not only do these results support our model, they also reveal that in addition to *splitter cells* that encode turn direction over all theta phases (Duvelle et al., 2023), CA3 contains many more place cells that encode it only at certain phases (Fig. S6D). The difference between sparse and dense phases is significantly greater in CA3 than it is in CA1 (Fig. S6E, F). Thus, our subfield-specific results for example encoding are consistent across position and turn direction. Aggregate neurons, formed by combining spikes from more active turn directions and those from less active turn directions, demonstrate similar tuning properties to individual neurons (Fig. S6G–I).

Beyond the single-neuron results presented above, we seek to test our predictions at the population level. To do so, we perform phase-dependent Bayesian population decoding of turn direction during runs along the center arm (Fig. 6I). This analysis requires multiple neurons with sufficiently sharp tuning to be simultaneously active across all theta phases; it can be used to decode left versus right turns, whereas an analogous decoding of track position, which spans a much broader range of values, is intractable with our datasets. We find that the CA3 population likelihood exhibits greater confidence during sparse phases

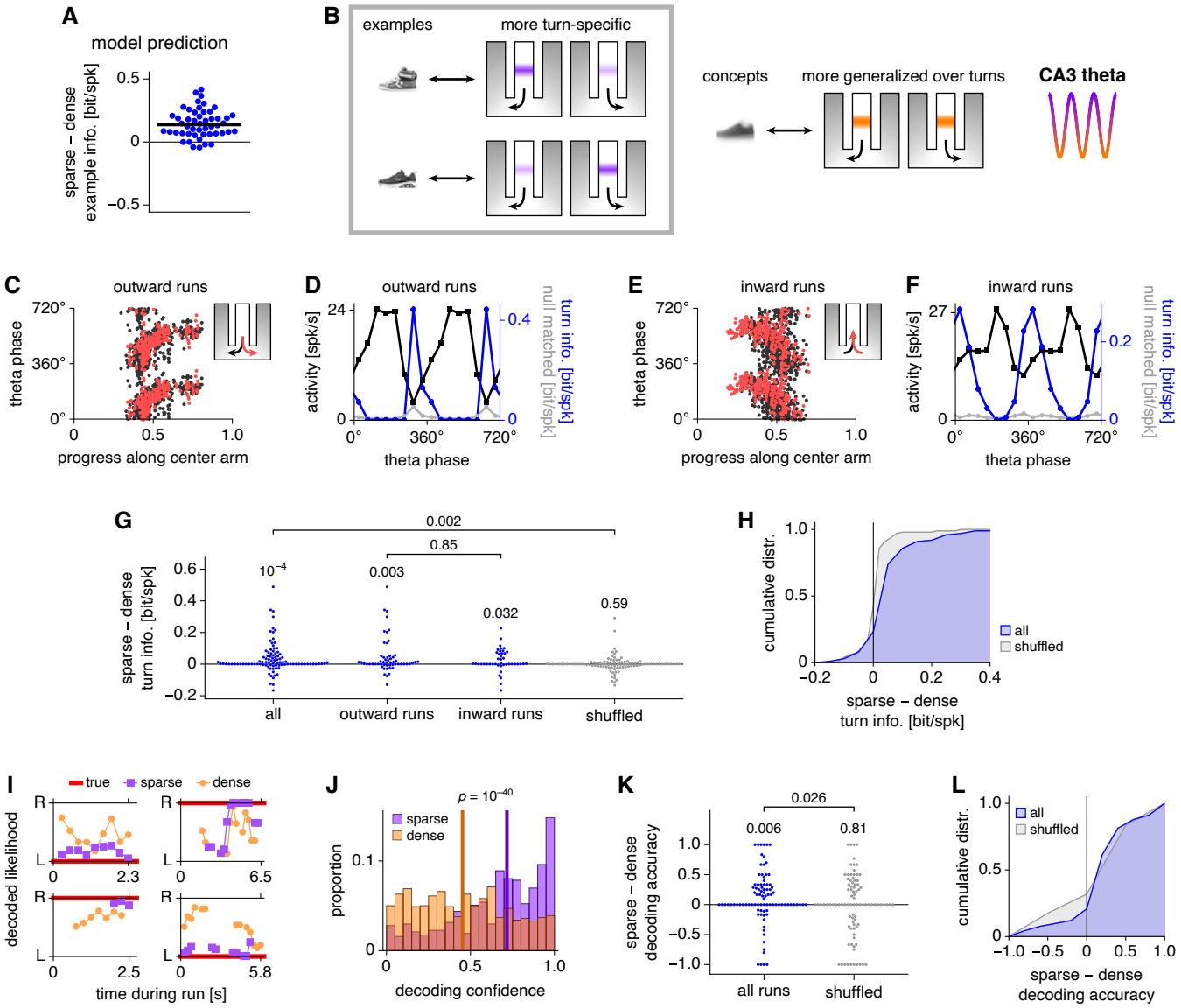

Figure 6: W-maze data support the model prediction that sparser theta phases should preferentially encode turn direction in addition to position. (**A**) Same as Fig. 4A. (**B**) To test our model, we construe CA3 place cells to store turn directions during the central arm of a W-maze alternation task as examples. By combining examples, concepts that generalize over turns to solely encode position can be formed. (**C**–**H**) Single-neuron information results. (**C**) Example place cell that is active during outward runs. Each spike is represented by two points at equivalent phases with different colors representing different future turn directions. (**D**) Activity (black), raw turn information (blue), and mean null-matched turn information (gray) by theta phase for the neuron in **C**. Sparsity-corrected turn information is the difference between the raw and mean null-matched values. (**E**, **F**) Similar to **C**, **D**, but for a place cell that is active during inward runs with colors representing past turn directions. (**G**) Average difference in turn information between the sparsest and densest halves of theta phases. For all cell populations, sparse phases convey more turn information per spike. Each point represents a place cell. Numbers indicate p-values calculated by Wilcoxon signed-rank tests except for the comparison between outward and inward runs, which is calculated by the Mann-Whitney U test. (**H**) Cumulative distribution functions for values in **G**. (**I**–**L**) Bayesian population decoding results. (**I**) Likelihood of left (L) or right (R) turns during four runs along the center arm using spikes from either the sparsest or densest halves of theta phases. (**J**) Sparse encodings exhibit greater confidence about turn direction. (**K**) Average difference in accuracy of maximum likelihood estimation between the sparsest and densest halves of theta phases. Sparse phases encode turn direction more accurately. Each point represents values for one run averaged over decoded timepoints. Numbers indicate p-values calculated by Wilcoxon signed-rank tests. (**L**) Cumulative distribution functions for values in **K**. For all results, spikes during each traveling direction are separately analyzed. In **A**, **G**, and **H**, information is sparsity-corrected with horizontal lines indicating medians.

(Fig. 6J). From a Bayesian perspective, the population expresses stronger beliefs about turn direction during sparse phases and is more agnostic during dense phases. If pressed to choose the direction with higher likelihood as its estimate, CA3 is also more accurate during sparse phases (Fig. 6K, L). These results match our predictions in Fig. 6A, B and bolster our single-neuron results. Moreover, they are specific to CA3, as similar conclusions cannot be made about the CA1 place cell population (Fig. S6J–L).

In summary, extensive data analysis reveals experimental support for our CA3 model over two datasets collected by different research groups, across two encoding modalities, for both example and concept representations, and at both the single-neuron and population level.

CA3-like complementary encodings improve neural network performance in multitask machine learning

We have observed how CA3 encodes behaviorally relevant information at different scales across theta phases. Can these different types of encodings be useful for solving different types of tasks? Can they even benefit neural networks designed for machine learning, abstracting away from the hippocampus? To address these questions, we turn to a classic paradigm in machine learning: a multilayer perceptron trained on MNIST digits (LeCun et al., 1998). First, we augment the MNIST dataset by randomly assigning an additional label to each image: a set number (Fig. 7A). We train the fully connected feedforward network to perform one of two tasks: classification of the written digit or identification of the assigned set (Fig. 7B). The former requires clustering of images based on common features, which resembles concept learning in our CA3 model, and the latter requires discerning differences between similar images, which resembles example learning in our CA3 model (Fig. 7C). We use a held-out test dataset to evaluate digit classification performance and corrupted images from the train dataset to evaluate set identification performance.

In our CA3 model, we found that examples were preferentially encoded by the decorrelated MF pathway and concepts by the correlated PP pathway (Fig. 2). In an analogous fashion, we seek to manipulate the correlation properties within the final hidden layer of our perception, whose activations $\mathbf{s}_\alpha$ serve as encodings of the input images $\mathbf{i}_\alpha$. In particular, we apply a novel DeCorr loss function, which penalizes correlations in $\mathbf{s}_\alpha$ between every pair of items α, β in a training batch (Fig. 7D):

$$\mathcal{L}_{\text{DeCorr}} \approx \frac{1}{2} \sum_{\substack{\alpha,\beta \in \\ \text{batch}}} \text{Pearson}(\mathbf{s}_\alpha, \mathbf{s}_\beta)^2. \tag{3}$$

DeCorr mimics the MF pathway; the equation is approximate due to a slight modification of the Pearson correlation formula to aid numerical convergence (Methods). Alternatively, we consider the baseline condition with no loss function on hidden layer activations, which preserves natural correlations between similar images and mimics the PP pathway. Indeed, we observe that different encoding properties are suited for different tasks. Baseline networks perform better in concept learning (Fig. 7E) while DeCorr networks perform better in example learning (Fig. 7F), and these effects vary consistently with the strength of the DeCorr loss function (Fig. S7A, B). Thus, DeCorr allows us to tune encoding correlations in neural networks to highlight input features at either broader or finer scales. Tasks can be solved more effectively by matching their computational requirements with the appropriate encoding scale. Note that DeCorr is different from the DeCov loss function previously developed to reduce overfitting (Cogswell et al., 2015). DeCorr decorrelates pairs of inputs across all neurons in the specified layer, whereas DeCov decorrelates pairs of neurons across all inputs. As a regularizer that promotes generalization, DeCov improves digit classification and does not

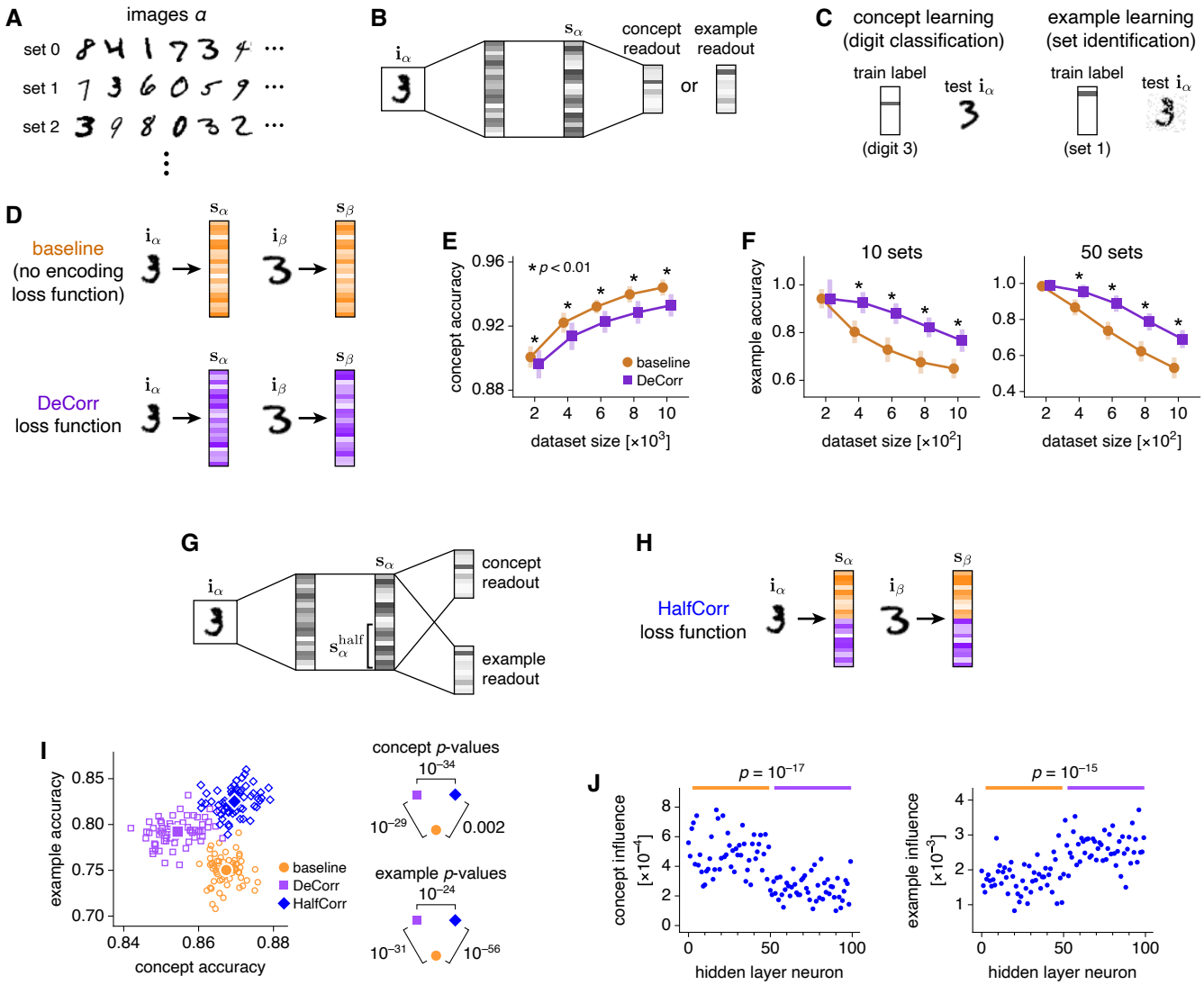

Figure 7: Complementary encodings inspired by CA3 can improve machine learning performance in a complex task. (**A**) We extend the MNIST dataset by randomly assigning an additional set label to each image. (**B**–**F**) We train a multilayer perceptron to either classify digits or identify sets. (**B**) Network architecture. Each hidden layer contains 50 neurons. (**C**) Task structures. Digit classification requires building concepts and is tested with held-out test images. Set identification requires distinguishing examples and is tested with noisy train images. (**D**) We apply the DeCorr loss function (Eq. 3) to decorrelate encodings in the final hidden layer, in analogy with MF patterns in CA3. Without an encoding loss function, image correlations are preserved, in analogy with PP patterns. (**E**, **F**) DeCorr decreases concept learning performance and increases example learning performance. Points indicate means and bars indicate standard deviations over 32 networks. (**G**–**J**) We train a multilayer perceptron to simultaneously classify digits and identify sets. (**G**) Network architecture. Each hidden layer contains 100 neurons. The train dataset contains 1000 images and 10 sets. (**H**) We apply the HalfCorr loss function (Eq. 4) to decorrelate encodings only among the second half of the final hidden layer. Correlated and decorrelated encodings are both present, in analogy with MF and PP patterns across the theta cycle in CA3. (**I**) DeCorr networks generally perform better at example learning but worse at concept learning compared to baseline. HalfCorr networks exhibit high performance in both tasks. Open symbols represent individual networks and filled symbols represent means over 64 networks. (**J**) Influence of each neuron in HalfCorr networks on concept and example learning, defined as the average decrease in accuracy upon clamping its activation to 0. Correlated neurons (orange bars) are more influential in concept learning, and decorrelated neurons (purple bars) are more influential in example learning. For all results, p-values are computed using unpaired t-tests.

substantially improve set identification, which contrasts with the effect of DeCorr (Fig. S7C, D).

Complex tasks, including those performed by biological systems, may require information to be processed at different scales of correlation. In CA3, a spectrum of encodings is available during each theta cycle. Can neural networks take advantage of multiple encodings? We tackle this question by asking a perceptron to simultaneously perform digit classification and set identification (Fig. 7G). In addition to the baseline and DeCorr networks, we define a HalfCorr loss function (Fig. 7H):

$$\mathcal{L}_{\text{HalfCorr}} \approx \frac{1}{2} \sum_{\substack{\alpha,\beta \in \\ \text{batch}}} \text{Pearson}(\mathbf{s}_\alpha^{\text{half}}, \mathbf{s}_\beta^{\text{half}})^2, \tag{4}$$

where $\mathbf{s}_\alpha^{\text{half}}$ represents the second half of neurons in the final hidden layer. After training with this loss function, the neural representation consists of both a correlated, PP-like component in the first half and a decorrelated, MF-like component in the second half. When we evaluate these networks on both digit classification and set identification, we see that baseline and DeCorr networks behave similarly to how they did on single tasks. Compared to baseline, DeCorr networks perform better in example learning at the cost of poorer concept learning (Fig. 7I). However, HalfCorr networks do not suffer from this tradeoff and perform well at both tasks. Their superior performance is maintained over a variety of network and dataset parameters (Fig. S7E). Moreover, HalfCorr networks learn to preferentially use each type of encoding for the task to which it is better suited. We use the decrease in task accuracy upon silencing a neuron as a metric for its influence on the task. Correlated neurons are more influential in concept learning and decorrelated neurons in example learning (Fig. 7J).

Note that we do not manipulate pattern sparsity in these artificial networks. Sparsification can be useful in the hippocampus because it provides a biologically tractable means of achieving decorrelation. It also allows biological networks to access both less and more correlated representations by changing the level of inhibition. Instead, we can directly manipulate correlation through the DeCorr and HalfCorr loss functions. Under some conditions, the decorrelated half of the final hidden layer in HalfCorr networks indeed exhibits sparser activation than the correlated half (Fig. S7F). It is possible that directly diversifying sparsity can also improve machine learning performance, especially since sparse coding is known to offer certain computational advantages as well as greater energy efficiency (Olshausen and Field, 1996, 2004; Sze et al., 2017).

Discussion

The hippocampus is widely known to produce our ability to recall specific vignettes as episodic memories. This process has been described as indexing every sensory experience with a unique neural barcode so that separate memories can be independently recovered (Teyler and DiScenna, 1986; Teyler and Rudy, 2007). Recently, research has shown that the hippocampus is also important in perceiving commonalities and regularities across individual experiences, which contribute to cognitive functions such as statistical learning (Schapiro et al., 2014; Covington et al., 2018), category learning (Knowlton and Squire, 1993; Zeithamova et al., 2008; Mack et al., 2016; Bowman and Zeithamova, 2018), and semantic memory (Manns et al., 2003; Duff et al., 2020). Evidence for this has been obtained largely through human studies, which can present and probe memories in controlled settings. However, the detailed circuit mechanisms used by the hippocampus to generalize across experiences while also indexing them separately are not known.

Our analysis of rodent place cell recordings reveals that single CA3 neurons alternate between finer,

example-like representations and broader, concept-like representations of space across the theta cycle (Figs. 4, 5, and 6). These single-neuron results extend to the network level, which alternatively encodes more specific and more general spatial features in a corresponding manner (Fig. 6). If we accept that place cells store these features as spatial memories, then our experimental analysis reveals that CA3 can access memories of different scales at different theta phases. We propose that the computational mechanism underlying these observations is the multiplexed encoding of each memory at different levels of correlation (Figs. 1 and 7). We show that this mechanism can be biologically implemented through the storage of both sparse, decorrelated MF and dense, correlated PP inputs to CA3 and their alternating retrieval by the theta oscillation, which acts as an activity threshold (Figs. 1, 2, and 3). Our model performs successful pattern completion for both types of encodings, suggesting that patterns across the theta cycle can truly function as memories that can be recovered from partial cues.

Alone, our experimental findings contribute to a large set of observations on how coding properties vary with theta phase in the hippocampus. Of note is phase precession, in which different phases preferentially encode different segments within a firing field as it is traversed, with later phases tuned to earlier segments (O'Keefe and Recce, 1993). Phase precession is most widely reported for place cells and traversals of physical space, but it also appears during the experience of other sequences, such as images and tasks (Terada et al., 2017; Qasim et al., 2021; Reddy et al., 2021). Our analysis implies that the sharpness of tuning is not constant throughout traversals. In particular, CA3 neurons are more sharply tuned at early positions in place fields, while CA1 neurons are more sharply tuned at late positions (Fig. S4I, J). Transforming these conclusions about position into those about time through the concept of theta sequences, CA3 represents the future more precisely, while CA1 represents the future more broadly. The latter is consistent with the idea that CA1 may participate in the exploration of multiple possible future scenarios (Kay et al., 2020). Furthermore, our W-maze analysis reveals that certain hippocampal neurons which do not obviously encode an external modality across all theta phases, such as turn direction, may do so only during sparse phases (Fig. S6D). This observation adds to the subtleties with which the hippocampus represents the external world.

Other groups have investigated the variation of place field sharpness with theta phase in CA1, not CA3, and their results are largely in agreement with our CA1 analyses. Skaggs et al. (1996) partitioned theta phases into halves, one of which with higher activity than the other, and find more information per spike during the less active half. We observe no difference at the single-neuron level, though our W-maze results are only slightly non-significant (Figs. S4C and S6E). Their partitions differ from ours by 30° and they employ a different binning technique, both of which can influence the results. The more informative phases in their work correspond to earlier field positions, which we also observe (Fig. S4J). Note that their computation of sparsity is performed along a different axis compared to ours; using terms from Willmore and Tolhurst (2001), they use the *lifetime* sparsity while we compute the *population* sparsity, which fundamentally differ. Ujfalussy and Orbán (2022) also found that phases with smaller field sizes correspond to earlier field positions. Mehta et al. (2002) considered phase-dependent tuning within CA1 place fields, but they calculate field width over theta phase as a function of field progress, whereas we do the opposite. Their results appear to be compatible with ours, but a direct comparison cannot be made. Overall, our work offers original insights into hippocampal phase coding not only by focusing on CA3, which behaves differently from CA1, but also by elucidating a connection between tuning width and network sparsity.

One major simplification in our model is that we separately simulate memory storage and retrieval. These two operating regimes can represent different tones of a neuromodulator such as acetylcholine, which is

thought to bias the network towards storage (Hasselmo, 2006). Another proposal is that storage and retrieval preferentially occur at different theta phases, motivated by the variation in long-term potentiation (LTP) strength at CA1 synapses across the theta cycle (Hasselmo et al., 2002; Kunec et al., 2005; Siegle and Wilson, 2014). Although this idea focuses on plasticity in CA1, it is possible that storage and retrieval also occur at different phases in CA3. Note that our experimental analysis reveals a sharp dip in position information around the sparsest theta phase in both CA3 and CA1 (Fig. S4E, G). This phase may coincide with the storage of new inputs, during which the representation of existing memories is momentarily disrupted; the rest of the theta cycle may correspond to retrieval. This interpretation could motivate excluding the sparsest theta phase from further analysis, since our model predictions only regard memory retrieval. However, we take a conservative approach and include all phases. Intriguingly, recent work reported that the strength of LTP in CA1 peaks twice per theta cycle (Leung and Law, 2020), suggesting for our model that MF and PP patterns could have their own storage and retrieval intervals during each theta cycle. Another simplification in our model is that we do not consider the theta oscillation in EC and DG, whose encodings may also vary with inhibitory tone. Since these regions are not known to receive multiple inputs with substantially different sparsities, we focus on the theta rhythm in CA3.

Our work connects hippocampal anatomy and physiology with foundational attractor theory. Among others, Tsodyks and Feigel'man (1988) observed that sparse, decorrelated patterns can be stored at high capacity, and Fontanari (1990) found that dense, correlated patterns can merge into representations of common features. We demonstrate that both types of representations can be stored and retrieved in the same network, using a threshold to select between them. This capability can be given solid theoretical underpinnings using techniques from statistical mechanics (Kang and Toyoizumi, 2023). The convergence of MF and PP pathways in CA3 has also been the subject of previous computational investigations (Treves and Rolls, 1992; McClelland and Goddard, 1996; Kaifosh and Losonczy, 2016). In these models, CA3 stores and retrieves one encoding per memory, while our model asserts that multiple encodings for the same memory alternate across the theta cycle. Another series of models proposes, like we do, that the hippocampus can simultaneously maintain both decorrelated, example-like encodings and correlated, concept-like encodings (Schapiro et al., 2017; Sučević and Schapiro, 2022). These encodings converge at CA1 and each type is not independently retrieved there, which differs from our model. EC has also been hypothesized to differentially encode inputs upstream of CA3, with specific sensory information conveyed by lateral EC and common structural representations by medial EC (Whittington et al., 2020). Further experimental investigation into the contributions of various subregions would help to clarify how the hippocampus participates in memory generalization.

Finally, we show that complementary encodings similar to those found in CA3 can aid neural networks in solving complex tasks. We introduce a novel HalfCorr loss function that diversifies hidden layer representations to include both correlated and decorrelated components (Fig. 7). HalfCorr networks can better learn tasks that involve both distinction between similar inputs and generalization across them. They are simultaneously capable of pattern separation and categorization even based on small datasets, demonstrating a possible advantage of brain computation over conventional deep learning. Yet, we deliberately chose a neural architecture that differs from that of the CA3 network to test the scope over which complementary encodings can improve learning. Instead of a recurrent neural network storing patterns of different sparsities through unsupervised Hopfield learning rules, we implemented a feedforward multilayer perceptron, a workhorse of supervised machine learning. The success of HalfCorr networks in this scenario supports the possibility that HalfCorr can be broadly applied as a plug-and-play loss function to improve computational flexibility.

Functional heterogeneity is commonly invoked in the design of modern neural networks. It can be implemented in the form of deep or modular neural networks in which different subnetworks perform different computations in series or parallel, respectively (LeCun et al., 2015; Amer and Maul, 2019). Here, inspired by biology, we propose a different paradigm in which a loss function applied differentially across neurons promotes heterogeneity within a single layer. This idea can be extended from the two components of HalfCorr networks, correlated and decorrelated, by assigning a different decorrelation strength to each neuron and thereby producing a true spectrum of representations. Furthermore, heterogeneity in other encoding properties such as mean activation, variance, and sparsity may also improve performance in tasks with varying or unclear computational requirements. Such tasks are not limited to multitask learning, but also include continual learning (Parisi et al., 2019), reinforcement learning (Arulkumaran et al., 2017), and natural learning by biological brains.

Methods

Transformation of memories along hippocampal pathways

Binary autoencoder from images to EC

Our memories are 256 images from each of the *sneaker*, *trouser*, and *coat* classes in the FashionMNIST dataset (Xiao et al., 2017). We train a fully connected linear autoencoder on these images with three hidden layers of sizes 128, 1024, and 128. Batch normalization is applied to each hidden layer, followed by a rectified linear unit (ReLU) nonlinearity for the first and third hidden layers and a sigmoid nonlinearity for the output layer. Activations in the middle hidden layer are binarized by a Heaviside step function with gradients backpropagated by the straight-through estimator (Bengio et al., 2013). The loss function is

$$\mathcal{L} = \sum_{\substack{\mu\nu\in\\ \text{batch}}} ||\hat{\mathbf{i}}_{\mu\nu} - \mathbf{i}_{\mu\nu}||^2 + \lambda \sum_{\substack{\mu\nu\in\\ \text{batch}}} \mathrm{KL}\Big(\tfrac{1}{N_{\mathrm{EC}}} \textstyle\sum_i x^{\mathrm{EC}}_{\mu\nu i} \,\Big\|\, a_{\mathrm{EC}}\Big), \tag{5}$$

where $\mathbf{i}_{\mu\nu}$ is the image with pixel values between 0 and 1, $\hat{\mathbf{i}}_{\mu\nu}$ is its reconstruction, $\mathbf{x}^{\mathrm{EC}}_{\mu\nu}$ represents the binary activations of the middle hidden layer with $N_{\mathrm{EC}} = 1024$ units indexed by i, and $a_{\mathrm{EC}} = 0.1$ is its desired sparsity (Fig. 1E). Sparsification with strength $\lambda = 10$ is achieved by computing the Kullback-Leibler (KL) divergence between the hidden layer sparsity and a_{EC} (Le et al., 2011). Training is performed over 150 epochs with batch size 64 using the Adam optimizer with learning rate 10^{-3} and weight decay 10^{-5}.

Binary feedforward networks from EC to CA3

To propagate patterns from EC to DG, from DG to MF inputs, and from EC to PP inputs, we compute

$$x^{\mathrm{post}}_{\mu\nu i} = \Theta\Big[\sum_j W_{ij} x^{\mathrm{pre}}_{\mu\nu j} - \theta\Big], \tag{6}$$

where $\mathbf{x}^{\mathrm{pre}}_{\mu\nu}$ and $\mathbf{x}^{\mathrm{post}}_{\mu\nu}$ are presynaptic and postsynaptic patterns, W_{ij} is the connectivity matrix, and θ is a threshold. Each postsynaptic neuron receives l excitatory synapses of equal strength from randomly chosen presynaptic neurons. θ is implicitly set through a winners-take-all (WTA) process that enforces a desired postsynaptic sparsity a_{post}. Θ is the Heaviside step function, and N is the network size.

EC patterns have $N_{\mathrm{EC}} = 1024$ and $a_{\mathrm{EC}} = 0.1$. To determine N, a, and l for each subsequent region, we turn to estimated biological values and loosely follow their trends. Rodents have approximately 5-10 times more DG granule cells and 2-3 times more CA3 pyramidal cells compared to medial EC layer II principal neurons (Amaral et al., 1990; Murakami et al., 2018; Attili et al., 2019). Thus, we choose $N_{\mathrm{DG}} = 8192$ and $N_{\mathrm{CA3}} = 2048$. During locomotion, DG place cells are approximately 10 times less active than medial EC grid cells (Mizuseki and Buzsáki, 2013), and MF inputs are expected to be much sparser than PP inputs (Treves and Rolls, 1992). Thus, we choose $a_{\mathrm{DG}} = 0.005$, $a_{\mathrm{MF}} = 0.02$, and $a_{\mathrm{PP}} = 0.2$. Each DG neuron receives approximately 4000 synapses from EC and each CA3 neuron receives approximately 50 MF and 4000 PP synapses (Amaral et al., 1990). Thus, we choose $l_{\mathrm{DG}} = 205$, $l_{\mathrm{MF}} = 8$, and $l_{\mathrm{PP}} = 205$. We do not directly enforce correlation within concepts, which take values $\rho_{\mathrm{EC}} = 0.15$, $\rho_{\mathrm{DG}} = 0.02$, $\rho_{\mathrm{MF}} = 0.01$, and $\rho_{\mathrm{PP}} = 0.09$.

In Fig. 1G, for the case of $\mathbf{x}_{\mu\nu}^{\text{pre}} = \mathbf{x}_{\mu\nu}^{\text{EC}}$, we use $N_{\text{post}} = 2048$ and $l = 205$. ρ_{post} is obtained by computing correlations between examples within the same concept and averaging over 3 concepts and 8 connectivity matrices. For the case of randomly generated $\mathbf{x}_{\mu\nu}^{\text{pre}}$, we use $N_{\text{pre}} = N_{\text{post}} = 10\,000$, a single concept, and a single connectivity matrix with $l = 2000$. See Supplementary Methods for further details, including the derivation of Eq. 1.

Visualization pathway from CA3 to EC

We train a fully connected linear feedforward network with one hidden layer of size 4096 to map inputs $\mathbf{x}_{\mu\nu}^{\text{MF}}$ to targets $\mathbf{x}_{\mu\nu}^{\text{EC}}$ and inputs $\mathbf{x}_{\mu\nu}^{\text{PP}}$ also to targets $\mathbf{x}_{\mu\nu}^{\text{EC}}$. Batch normalization and a ReLU nonlinearity is applied to the hidden layer and a sigmoid nonlinearity is applied to the output layer. The loss function is

$$\mathcal{L} = \sum_{\substack{\mu\nu\in \\ \text{batch}}} ||\hat{\mathbf{x}}_{\mu\nu}^{\text{EC}} - \mathbf{x}_{\mu\nu}^{\text{EC}}||^2. \tag{7}$$

Training is performed over 100 epochs with batch size 128 using the Adam optimizer with learning rate 10^{-4} and weight decay 10^{-5}.

Hopfield-like model for CA3

Pattern storage

Our Hopfield-like model for CA3 stores linear combinations $\mathbf{q}_{\mu\nu}$ of MF and PP patterns:

$$q_{\mu\nu i} = (1-\zeta)\cdot(x_{\mu\nu i}^{\text{MF}} - a_{\text{MF}}) + \zeta\cdot(x_{\mu\nu i}^{\text{PP}} - a_{\text{PP}}), \tag{8}$$

where $\zeta = 0.1$ is the relative strength of the PP patterns (Fig. 2A). The subtraction of sparsities from each pattern is typical of Hopfield networks with neural states 0 and 1 (Tsodyks and Feigel'man, 1988). The synaptic connectivity matrix is

$$W_{ij} = \frac{1}{N_{\text{CA3}}}\sum_{\mu\nu} q_{\mu\nu i} q_{\mu\nu j}. \tag{9}$$

Pattern retrieval

Cues are formed from target patterns by randomly flipping the activity of a fraction 0.01 of all neurons (Fig. 2B). During retrieval, neurons are asynchronously updated in cycles during which every neuron is updated once in random order (Fig. 2C). The total synaptic input to neuron i at time t is

$$g_i(t) = \sum_j W_{ij} S_j(t) + h_i(t), \tag{10}$$

where $S_j(t)$ is the activity of presynaptic neuron j and $h_i(t)$ is an external input. The external input is zero expect for the cue-throughout condition in Fig. 3, in which $\mathbf{h}(t) = \sigma\mathbf{x}$ for noisy MF cue $\mathbf{x}$ and strength $\sigma = 0.2$.

The activity of neuron i at time t is probabilistically updated via the Glauber dynamics

$$P[S_i(t+1) = 1] = \frac{1}{1 + \mathrm{e}^{-\beta[g_i(t) - \theta(t)]}}, \tag{11}$$

where θ is the threshold and β is inverse temperature, with higher β implying a harder threshold. Motivated by theoretical arguments, we define rescaled variables θ' and β' such that $\theta = \theta'\cdot(1-\zeta)^2 a_{\text{MF}}$ and $\beta = \beta'/(1-\zeta)^2 a_{\text{MF}}$ (Kang and Toyoizumi, 2023). Unless otherwise indicated, we run simulations for 10 update cycles, use $\beta' = 100$, and use $\theta' = 0.5$ to retrieve MF patterns and $\theta' = 0.1$ to retrieve PP patterns. The rescaled θ' is the threshold value illustrated in Fig. 3A and Fig. S3A.

Retrieval evaluation

The overlap between the network activity $\mathbf{S}$ and a target pattern $\mathbf{x}$ is

$$m = \frac{1}{N_{\text{CA3}} a(1-a)}\sum_i S_i(x_i - a), \tag{12}$$

where a is the sparsity of the target pattern. This definition is also motivated by theory (Kang and Toyoizumi, 2023). The target pattern $\bar{\mathbf{x}}_\mu^{\mathrm{PP}}$ for PP concept μ is

$$\bar{x}_{\mu i}^{\mathrm{PP}} = \Theta\left[\sum_\nu x_{\mu\nu i}^{\mathrm{PP}} - \phi\right], \tag{13}$$

where ϕ is a threshold implicitly set by using winners-take-all to enforce that $\bar{\mathbf{x}}_\mu^{\mathrm{PP}}$ has sparsity a_{PP}. The theoretical maximum overlap between the network and $\bar{\mathbf{x}}_\mu^{\mathrm{PP}}$ is the square root of the correlation $\sqrt{\rho_{\mathrm{PP}}}$ (Kang and Toyoizumi, 2023).

See Supplementary Methods for the determination of network capacity with random binary patterns (Fig. 2H, I and Fig. S2F, G) and the definition of oscillation behaviors (Fig. 3C and Fig. S3B).

Experimental data analysis

General considerations

To calculate activity, we tabulate spike counts $c(r, \phi)$ over spatial bins r (position or turn direction) and theta phase bins ϕ, and we tabulate trajectory occupancy $u(r, \phi)$ over the same r and distribute them evenly across ϕ. Activity as a function of theta phase, the spatial variable, and both variables are respectively

$$f(\phi) = \frac{\sum_r c(r,\phi)}{\sum_r u(r,\phi)}, \quad f(r) = \frac{\sum_\phi c(r,\phi)}{\sum_\phi u(r,\phi)}, \quad \text{and} \quad f(r,\phi) = \frac{c(r,\phi)}{u(r,\phi)}. \tag{14}$$

Information per spike as a function of theta phase is calculated by

$$I(\phi) = \sum_r \frac{c(r,\phi)}{c(\phi)} \log_2 \frac{f(r,\phi)}{f(\phi)}, \tag{15}$$

where $c(\phi) = \sum_r c(r, \phi)$ (Skaggs et al., 1993). To perform sparsity correction for each neuron, we generate 100 null-matched neurons in which the spatial bin of each spike is replaced by a random value uniformly distributed across spatial bins. We subtract the mean $I(\phi)$ over the null matches from the $I(\phi)$ for the true data. To calculate the average difference in information between sparse and dense phases, we first $f(\phi)$ to partition ϕ into sparse and dense halves. We then average the sparsity-corrected $I(\phi)$ over each half, apply a ReLU function to each half to prevent negative information values, and compute the difference between halves.

See Supplementary Methods for dataset preprocessing details.

Model prediction

For the example prediction in Fig. 4A, we choose one concept from Fig. 1C and find 50 neurons that are active in at least one MF example and one PP example within it. For each neuron, we convert each active response to one spike and assign equal occupancies across all examples. We calculate the information per spike across MF examples and across PP examples using example identity ν as the spatial bin r. These values are sparsity-corrected with 50 null-matched neurons, and their difference becomes our example prediction, associating MF encodings with sparse phases and PP with dense.

For the concept prediction in Fig. 5A, we find 50 neurons that are active in at least one MF example and one PP example within any concept. For each neuron, we convert each active response to one spike and collect MF and PP concept responses by summing spikes within each concept. We assign equal occupancies across all concepts. We calculate the information per spike across MF concepts and across PP concepts using concept identity μ as the spatial bin r. We then proceed as in the example case to produce our concept prediction.

Linear track data

Single neurons in Fig. 5 are preprocessed from the CRCNS hc-3 dataset as described in Supplementary Methods (Mizuseki et al., 2013). To identify place cells, we compute the phase-independent position information per spike using 1 cm-bins across all theta phases, and we select neurons with values greater than 0.5. For each place cell, we bin spikes into various position bins as illustrated and phase bins of width 30°. Since our prediction compares sparse and dense information conveyed by the same neurons, we require at least 8 spikes within each phase value to allow for accurate estimation of position information across all theta phases. To ensure theta modulation, we also require the most active phase to contain at least twice the number of spikes as the least active phase. In Fig. 5E, these constraints yield 47, 49, and 56 valid CA3 neurons respectively for track scales 1/16, 1/8, and 1/4, and in Fig. S5B, they yield 122, 137, and 144 valid CA1 neurons.

Place fields in Fig. 4 are extracted as described in Supplementary Methods. Processing occurs similarly to the whole-track case above, except we do not enforce a phase-independent information constraint, we use 5 progress bins, and we require at least 5 spikes within each phase value. These constraints yield 35 valid CA3 fields and 47 valid CA1 fields. Phase precession is detected by performing circular–linear regression between spike progresses and phases (Kempter et al., 2012; Kang and DeWeese, 2019). The precession score and precession slope are respectively defined to be the mean resultant length and regression slope. Precessing neurons have score greater than 0.3 and negative slope steeper than $-72\,^\circ$/field.

W-maze data

Single neurons in Fig. 6A–H are preprocessed from the CRCNS hc-6 dataset as described in Supplementary Methods (Karlsson et al., 2015). For each neuron, we bin spikes into 2 turn directions and phase bins of width 45°. Since our prediction compares sparse and dense information conveyed by the same neurons, we require at least 5 spikes within each phase value to allow for accurate estimation of position information across all theta phases. To ensure theta modulation, we also require the most active phase to contain at least twice the number of spikes as the least active phase. These constraints yield 99 valid CA3 neurons and 187 valid CA1 neurons.

Bayesian population decoding in Fig. 6I–L involves the same binning as in the single-neuron case above, and we enforce a minimum spike count of 30 across all phases instead of a minimum for each phase value. We do not ensure theta modulation on a single-neuron basis. We consider all sessions in which at least 5 neurons are simultaneously recorded; there are 8 valid CA3 sessions and 25 valid CA1 sessions. For each session, we compute the total activity across neurons and turn directions as a function of theta phase to determine the sparsest and densest half of phases (similarly to Eq. 14). We then compute activities $f_i(r, \psi)$ over each half, indexed by $\psi \in \{\text{sparse}, \text{dense}\}$, for neurons i and turn directions r. For each neuron, we rectify all activity values below 0.02 times its maximum.

We decode turn direction during runs along the center arm using sliding windows of width $\Delta t = 0.5\,\text{s}$ and stride 0.25 s. In each window at time t, we tabulate the population spike count $\mathbf{c}(t, \psi)$ over sparse and dense phases ψ. The likelihood that it arose from turn direction r is

$$p(\mathbf{c}(t,\psi)|r) = \prod_i p(c_i(t,\psi)|r) \propto \left(\prod_i f_i(r,\psi)^{c_i(t,\psi)}\right) \exp\left(-\Delta t \sum_i f_i(r,\psi)\right). \tag{16}$$

This formula assumes that spikes are independent across neurons and time and obey Poisson statistics (Zhang et al., 1998). We only decode with at least 2 spikes. By Bayes's formula and assuming a uniform prior, the likelihood is proportional to the posterior probability $p(r|\mathbf{c}(t,\psi))$ of turn direction r decoded from spikes $\mathbf{c}(t,\psi)$. Consider one decoding that yields $p(R)$ as the probability of a right turn. Its confidence is $|2p(R)-1|$. Its accuracy is 1 if $p(R) > 0.5$ and the true turn direction is right or if $p(R) < 0.5$ and the true direction is left; otherwise, its accuracy is 0.

Machine learning with multilayer perceptrons

Dataset

We use the MNIST dataset of handwritten digits (LeCun et al., 1998). Each image $\mathbf{i}_\alpha$ is normalized by subtracting the mean value and dividing by the standard deviation across all images and pixels. In addition to its digit class label, we randomly assign a set number. We train networks on a subset of images from the train dataset. To test concept learning through digit classification, we use all held-out images from the test dataset. To test example learning through set identification, we use all train images corrupted by randomly setting 20% of normalized pixel values to 0.

Single-task learning

We train a fully-connected two-layer perceptron with a hyperbolic tangent (tanh) activation function applied to each hidden layer and a softmax activation function applied to the output layer. Each hidden layer contains 50 neurons, and the output layer contains 10 neurons for digit classification and as many neurons as sets for set identification.

Let $\mathbf{s}_\alpha$ be the activations of the final hidden layer for image α. The loss is composed of a cross-entropy loss function between reconstructed labels $\hat{\mathbf{y}}_\alpha$ and true labels $\mathbf{y}_\alpha$, which are one-hot encodings of either digit class or set number, and the DeCorr loss function:

$$\mathcal{L} = -\sum_{\substack{\alpha\in\\ \text{batch}}} \sum_{i=0}^{N-1} \left[y_{\alpha i} \log \hat{y}_{\alpha i} + (1 - y_{\alpha i}) \log(1 - \hat{y}_{\alpha i})\right] + \lambda \mathcal{L}_{\text{DeCorr}}, \tag{17}$$

where

$$\mathcal{L}_{\text{DeCorr}} = \frac{1}{2} \sum_{\substack{\alpha \neq \beta \in \\ \text{batch}}} \frac{\left[\sum_{i=0}^{N-1} (s_{\alpha i} - \bar{s}_\alpha)(s_{\beta i} - \bar{s}_\beta)\right]^2}{\left[\sum_{i=0}^{N-1} (s_{\alpha i} - \bar{s}_\alpha)^2 + N\epsilon\right] \left[\sum_{i=0}^{N-1} (s_{\beta i} - \bar{s}_\beta)^2 + N\epsilon\right]}. \tag{18}$$

We introduce $\epsilon = 0.001$, which is scaled by the number of hidden layer neurons N, to aid numerical convergence. Mean activations are $\bar{s}_\alpha = (1/N) \sum_{i=0}^{N-1} s_{\alpha i}$. The DeCorr strength is λ; except for Fig. S7A, B, we use $\lambda = 0$ for the baseline case and $\lambda = 1$ for the DeCorr case.

We train the network using stochastic gradient descent (SGD) with batch size 50 and learning rate 10^{-4}. In general, we train until the network reaches >99.9% accuracy with the train dataset. For example, we use 40, 100, and 200 epochs respectively for digit classification and set identification with 10 and 50 sets.

In contrast to DeCorr, the DeCov loss function formulated to reduce overfitting is

$$\mathcal{L}_{\text{DeCov}} = \frac{1}{2} \sum_{i \neq j = 0}^{N-1} \left[\sum_{\substack{\alpha \in \\ \text{batch}}} (s_{\alpha i} - \bar{s}_i)(s_{\alpha j} - \bar{s}_j) \right]^2, \tag{19}$$

where mean activations are now taken over batch items: $\bar{s}_i = (1/N_{\text{batch}}) \sum_{\alpha \in \text{batch}} s_{\alpha i}$ (Cogswell et al., 2015).

Multitask learning

We train a fully-connected two-layer perceptron with a hyperbolic tangent (tanh) activation function applied to each hidden layer. In Fig. S7E, F, we also consider applying a ReLU activation function to each hidden layer, or a ReLU to the first hidden layer and no nonlinearity to the second. The final hidden layer is fully connected to two output layers, one for digit classification and the other for set identification. A softmax activation function applied to each layer. Each hidden layer contains 100 neurons, the concept output layer contains 10 neurons, and the example output layer contains as many neurons as sets.

The loss is composed of a cross-entropy loss function between reconstructed $\hat{\mathbf{y}}_\alpha$ and true $\mathbf{y}_\alpha$ digit labels, a cross-entropy loss function between reconstructed $\hat{\mathbf{z}}_\alpha$ and true $\mathbf{z}_\alpha$ set labels, and either the DeCorr or HalfCorr loss function:

$$\mathcal{L} = -\sum_{\substack{\alpha \in \\ \text{batch}}} \sum_{i=0}^{N-1} \left[y_{\alpha i} \log \hat{y}_{\alpha i} + (1 - y_{\alpha i}) \log(1 - \hat{y}_{\alpha i}) \right] - \sum_{\substack{\alpha \in \\ \text{batch}}} \sum_{i=0}^{N-1} \left[z_{\alpha i} \log \hat{z}_{\alpha i} + (1 - z_{\alpha i}) \log(1 - \hat{z}_{\alpha i}) \right] + \lambda \mathcal{L}_{\text{DeCorr/HalfCorr}}, \tag{20}$$

where

$$\mathcal{L}_{\text{HalfCorr}} = \frac{1}{2} \sum_{\substack{\alpha \neq \beta \in \\ \text{batch}}} \frac{\left[\sum_{i=N/2}^{N-1} (s_{\alpha i} - \bar{s}_\alpha)(s_{\beta i} - \bar{s}_\beta)\right]^2}{\left[\sum_{i=N/2}^{N-1} (s_{\alpha i} - \bar{s}_\alpha)^2 + N\epsilon/2\right] \left[\sum_{i=N/2}^{N-1} (s_{\beta i} - \bar{s}_\beta)^2 + N\epsilon/2\right]}. \tag{21}$$

Mean activations are $\bar{s}_\alpha = (2/N) \sum_{i=N/2}^{N-1} s_{\alpha i}$. The DeCorr/HalfCorr strength is λ; we use $\lambda = 1$ with a tanh activation function, $\lambda = 0.04$ with a ReLU, $\lambda = 2$ with no nonlinearity, and $\lambda = 0$ for the baseline case with any nonlinearity.

We train the network using stochastic gradient descent (SGD) with batch size 50 and learning rate 10^{-4}. In general, we train until the network reaches >99.9% accuracy in both tasks with the train dataset. For example, we use 100 epochs for the results in Fig. 7I, J.

Code availability

All network training and simulation code will be made available at `https://louiskang.group/repo`.

Acknowledgments

We thank Tom McHugh and Łukasz Kuśmierz for helpful ideas. LK is supported by JSPS KAKENHI for Early-Career Scientists (22K15209) and has been supported by the Miller Institute for Basic Research in Science and a Burroughs Wellcome Fund Collaborative Research Travel Grant. TT is supported by Brain/MINDS from AMED (JP19dm0207001) and JSPS KAKENHI (JP18H05432).

References

J. B. Aimone, W. Deng, and F. H. Gage. Resolving new memories: A critical look at the dentate gyrus, adult neurogenesis, and pattern separation. *Neuron*, 70(4):589–596, 2011.

D. Amaral and L. Pierre. Hippocampal neuroanatomy. In P. Andersen, R. Morris, D. Amaral, T. Bliss, and J. O'Keefe, editors, *The Hippocampus Book*, The Hippocampus Book, pages 37–114. Oxford University Press, 2006.

D. G. Amaral, N. Ishizuka, and B. Claiborne. Neurons, numbers and the hippocampal network. *Prog. Brain Res.*, 83:1–11, 1990.

M. Amer and T. Maul. A review of modularization techniques in artificial neural networks. *Artif. Intell. Rev.*, 52(1): 527–561, 2019.

D. J. Amit, H. Gutfreund, and H. Sompolinsky. Spin-glass models of neural networks. *Phys. Rev. A*, 32(2):1007–1018, 1985.

K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath. Deep reinforcement learning. *IEEE Signal Process. Mag.*, 34(6):26–38, 2017.

S. M. Attili, M. F. M. Silva, T.-v. Nguyen, and G. A. Ascoli. Cell numbers, distribution, shape, and regional variation throughout the murine hippocampal formation from the adult brain Allen Reference Atlas. *Brain Struct. Funct.*, 224(8):2883–2897, 2019.

Y. Bengio, N. Léonard, and A. Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv* 1308.3432, 2013.

G.-q. Bi and M.-m. Poo. Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neurosci.*, 18(24):10464–10472, 1998.

C. R. Bowman and D. Zeithamova. Abstract memory representations in the ventromedial prefrontal cortex and hippocampus support concept generalization. *J. Neurosci.*, 38(10):2605–2614, 2018.

N. A. Cayco-Gajic, C. Clopath, and R. A. Silver. Sparse synaptic connectivity is required for decorrelation and pattern separation in feedforward networks. *Nat. Commun.*, 8(1):1116, 2017.

M. Cogswell, F. Ahmed, R. Girshick, L. Zitnick, and D. Batra. Reducing overfitting in deep networks by decorrelating representations. *arXiv* 1511.06068, 2015.

N. V. Covington, S. Brown-Schmidt, and M. C. Duff. The necessity of the hippocampus for statistical learning. *J. Cognit. Neurosci.*, 30(5):680–697, 2018.

N. M. Dotson and M. M. Yartsev. Nonlocal spatiotemporal representation in the hippocampus of freely flying bats. *Science*, 373(6551):242–247, 2021.

M. C. Duff, N. V. Covington, C. Hilverman, and N. J. Cohen. Semantic memory and the hippocampus: Revisiting, reaffirming, and extending the reach of their critical relationship. *Front. Hum. Neurosci.*, 13:471, 2020.

É. Duvelle, R. M. Grieves, and M. A. van der Meer. Temporal context and latent state inference in the hippocampal splitter signal. *eLife*, 12:e82357, 2023.

E. Engin, E. D. Zarnowska, D. Benke, E. Tsvetkov, M. Sigal, R. Keist, V. Y. Bolshakov, R. A. Pearce, and U. Rudolph. Tonic inhibitory control of dentate gyrus granule cells by α5-containing GABAA receptors reduces memory interference. *The Journal of Neuroscience*, 35(40):13698–13712, 2015.

J. F. Fontanari. Generalization in a Hopfield network. *J. Phys.*, 51(21):2421–2430, 1990.

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

M. E. Hasselmo. The role of acetylcholine in learning and memory. *Curr. Opin. Neurobiol.*, 16(6):710–715, 2006.

M. E. Hasselmo, C. Bodeln, and B. P. Wyble. A proposed function for hippocampal theta rhythm: Separate phases of encoding and retrieval enhance reversal of prior learning. *Neural Comput.*, 14(4):793–817, 2002.

D. A. Henze, L. Wittner, and G. Buzsáki. Single granule cells reliably discharge targets in the hippocampal CA3 network in vivo. *Nat. Neurosci.*, 5(8):790–795, 2002.

J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. U.S.A.*, 79(8):2554–2558, 1982.

P. Kaifosh and A. Losonczy. Mnemonic functions for nonlinear dendritic integration in hippocampal pyramidal circuits. *Neuron*, 90(3):622–634, 2016.

L. Kang and M. R. DeWeese. Replay as wavefronts and theta sequences as bump oscillations in a grid cell attractor network. *eLife*, 8:e46351, 2019.

L. Kang and T. Toyoizumi. A Hopfield-like model with complementary encodings of memories. *arXiv* 2302.04481, 2023.

M. Karlsson, M. Carr, and L. M. Frank. Simultaneous extracellular recordings from hippocampal areas CA1 and CA3 (or MEC and CA1) from rats performing an alternation task in two W-shapped tracks that are geometrically identically but visually distinct. *CRCNS.org*, 2015.

K. Kay, J. E. Chung, M. Sosa, J. S. Schor, M. P. Karlsson, M. C. Larkin, D. F. Liu, and L. M. Frank. Constant sub-second cycling between representations of possible futures in the hippocampus. *Cell*, 180(3):552–567, 2020.

R. Kempter, C. Leibold, G. Buzsáki, K. Diba, and R. Schmidt. Quantifying circular–linear associations: Hippocampal phase precession. *J. Neurosci. Methods*, 207(1):113–124, 2012.

B. J. Knowlton and L. R. Squire. The learning of categories: Parallel brain systems for item memory and category knowledge. *Science*, 262(5140):1747–1749, 1993.

S. Kunec, M. E. Hasselmo, and N. Kopell. Encoding and retrieval in the CA3 region of the hippocampus: A model of theta-phase separation. *J. Neurophysiol.*, 94(1):70–82, 2005.

Q. V. Le, A. Karpenko, J. Ngiam, and A. Y. Ng. ICA with reconstruction cost for efficient overcomplete feature learning. *Adv. Neural Inf. Process. Syst.* 1017–1025, 2011.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

L. S. Leung and C. S. H. Law. Phasic modulation of hippocampal synaptic plasticity by theta rhythm. *Behav. Neurosci.*, 134(6):595–612, 2020.

J. K. Leutgeb, S. Leutgeb, M.-B. Moser, and E. I. Moser. Pattern separation in the dentate gyrus and CA3 of the hippocampus. *Science*, 315(5814):961–966, 2007.

M. L. Mack, B. C. Love, and A. R. Preston. Dynamic updating of hippocampal object representations reflects new conceptual knowledge. *Proc. Natl. Acad. Sci. U.S.A.*, 113(46):13203–13208, 2016.

J. R. Manns, R. O. Hopkins, and L. R. Squire. Semantic memory and the human hippocampus. *Neuron*, 38(1): 127–133, 2003.

D. Marr. Simple memory: a theory for archicortex. *Philos. Trans. R. Soc. B*, 262(841):23–81, 1971.

C. J. McAdams and J. H. R. Maunsell. Effects of attention on orientation-tuning functions of single neurons in Macaque cortical area V4. *J. Neurosci.*, 19(1):431–441, 1999.

J. L. McClelland and N. H. Goddard. Considerations arising from a complementary learning systems perspective on hippocampus and neocortex. *Hippocampus*, 6(6):654–665, 1996.

B. McNaughton and R. Morris. Hippocampal synaptic enhancement and information storage within a distributed memory system. *Trends Neurosci.*, 10(10):408–415, 1987.

M. R. Mehta, A. K. Lee, and M. A. Wilson. Role of experience and oscillations in transforming a rate code into a temporal code. *Nature*, 417(6890):741–746, 2002.

R. K. Mishra, S. Kim, S. J. Guzman, and P. Jonas. Symmetric spike timing-dependent plasticity at CA3–CA3 synapses optimizes storage and recall in autoassociative networks. *Nat. Commun.*, 7:11552, 2016.

K. Mizuseki and G. Buzsáki. Preconfigured, skewed distribution of firing rates in the hippocampus and entorhinal cortex. *Cell Rep.*, 4(5):1010–1021, 2013.

K. Mizuseki, A. Sirota, E. Pastalkova, K. Diba, and G. Buzsáki. Multiple single unit recordings from different rat hippocampal and entorhinal regions while the animals were performing multiple behavioral tasks. *CRCNS.org*, 2013.

K. Mizuseki, K. Diba, E. Pastalkova, J. Teeters, A. Sirota, and G. Buzsáki. Neurosharing: large-scale data sets (spike, LFP) recorded from the hippocampal-entorhinal system in behaving rats. *F1000Research*, 3:98, 2014.

T. C. Murakami, T. Mano, S. Saikawa, S. A. Horiguchi, D. Shigeta, K. Baba, H. Sekiya, Y. Shimizu, K. F. Tanaka, H. Kiyonari, M. Iino, H. Mochizuki, K. Tainaka, and H. R. Ueda. A three-dimensional single-cell-resolution whole-brain atlas using CUBIC-X expansion microscopy and tissue clearing. *Nat. Neurosci.*, 21(4):625–637, 2018.

J. O'Keefe and M. L. Recce. Phase relationship between hippocampal place units and the EEG theta rhythm. *Hippocampus*, 3(3):317–330, 1993.

B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996.

B. A. Olshausen and D. J. Field. Sparse coding of sensory inputs. *Curr. Opin. Neurobiol.*, 14(4):481–487, 2004.

R. C. O'Reilly and J. L. McClelland. Hippocampal conjunctive encoding, storage, and recall: Avoiding a trade-off. *Hippocampus*, 4(6):661–682, 1994.

R. C. O'Reilly and J. W. Rudy. Conjunctive representations in learning and memory: Principles of cortical and hippocampal function. *Psychol. Rev.*, 108(2):311–345, 2001.

G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019.

X. Pitkow and M. Meister. Decorrelation and efficient coding by retinal ganglion cells. *Nat. Neurosci.*, 15(4):628–635, 2012.

S. E. Qasim, I. Fried, and J. Jacobs. Phase precession in the human hippocampus and entorhinal cortex. *Cell*, 184 (12):3242–3255, 2021.

R. Q. Quiroga, L. Reddy, G. Kreiman, C. Koch, and I. Fried. Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102–1107, 2005.

R. Q. Quiroga, A. Kraskov, C. Koch, and I. Fried. Explicit encoding of multimodal percepts by single neurons in the human brain. *Curr. Biol.*, 19(15):1308–1313, 2009.

L. Reddy, M. W. Self, B. Zoefel, M. Poncet, J. K. Possel, J. C. Peters, J. C. Baayen, S. Idema, R. VanRullen, and P. R. Roelfsema. Theta-phase dependent neuronal coding during sequence learning in human single neurons. *Nat. Commun.*, 12(1):4839, 2021.

E. T. Rolls and R. P. Kesner. A computational theory of hippocampal function, and empirical tests of the theory. *Prog. Neurobiol.*, 79(1):1–48, 2006.

G. Rosen, A. Williams, J. Capra, M. Connolly, B. Cruz, L. Lu, D. Airey, K. Kulkarni, and R. Williams. The Mouse Brain Library @ www.mbl.org. *Int Mouse Genome Conference*, 14:166, 2000.

A. C. Schapiro, E. Gregory, B. Landau, M. McCloskey, and N. B. Turk-Browne. The necessity of the medial temporal lobe for statistical learning. *J. Cognit. Neurosci.*, 26(8):1736–1747, 2014.

A. C. Schapiro, N. B. Turk-Browne, M. M. Botvinick, and K. A. Norman. Complementary learning systems within the hippocampus: a neural network modelling approach to reconciling episodic memory with statistical learning. *Philos. Trans. R. Soc. B*, 372(1711):20160049, 2017.

W. B. Scoville and B. Milner. Loss of recent memory after bilateral hippocampal lesions. *J. Neurol. Neurosurg. Psychiatry*, 20(1):11, 1957.

J. H. Siegle and M. A. Wilson. Enhancement of encoding and retrieval functions through theta phase-specific manipulation of hippocampus. *eLife*, 3:e03061, 2014.

W. E. Skaggs, B. L. McNaughton, K. M. Gothard, and E. J. Markus. An information-theoretic approach to deciphering the hippocampal code. *Adv. Neural Inf. Process. Syst.*, 1993.

W. E. Skaggs, B. L. McNaughton, M. A. Wilson, and C. A. Barnes. Theta phase precession in hippocampal neuronal populations and the compression of temporal sequences. *Hippocampus*, 6(2):149–172, 1996.

J. Sučević and A. C. Schapiro. A neural network model of hippocampal contributions to category learning. *bioRxiv* 2022.01.12.476051, 2022.

V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer. Efficient processing of deep neural networks: A tutorial and survey. *Proc. IEEE*, 105(12):2295–2329, 2017.

S. Terada, Y. Sakurai, H. Nakahara, and S. Fujisawa. Temporal and rate coding for discrete event sequences in the hippocampus. *Neuron*, 94(6):1248–1262, 2017.

T. J. Teyler and P. DiScenna. The hippocampal memory indexing theory. *Behav. Neurosci.*, 100(2):147–154, 1986.

T. J. Teyler and J. W. Rudy. The hippocampal indexing theory and episodic memory: Updating the index. *Hippocampus*, 17(12):1158–1169, 2007.

A. Treves and E. T. Rolls. Computational constraints suggest the need for two distinct input systems to the hippocampal CA3 network. *Hippocampus*, 2(2):189–199, 1992.

M. V. Tsodyks and M. V. Feigel'man. The enhanced storage capacity in neural networks with low activity level. *Europhys. Lett.*, 6(2):101–105, 1988.

B. B. Ujfalussy and G. Orbán. Sampling motion trajectories during hippocampal theta sequences. *eLife*, 11:e74058, 2022.

W. E. Vinje and J. L. Gallant. Sparse coding and decorrelation in primary visual cortex during natural vision. *Science*, 287(5456):1273–1276, 2000.

N. P. Vyleta, C. Borges-Merjane, and P. Jonas. Plasticity-dependent, full detonation at hippocampal mossy fiber–CA3 pyramidal neuron synapses. *eLife*, 5:3386, 2016.

J. C. Whittington, T. H. Muller, S. Mark, G. Chen, C. Barry, N. Burgess, and T. E. Behrens. The Tolman-Eichenbaum Machine: Unifying space and relational memory through generalization in the hippocampal formation. *Cell*, 183 (5):1249–1263, 2020.

B. Willmore and D. J. Tolhurst. Characterizing the sparseness of neural codes. *Netw. Comput. Neural Syst.*, 12(3): 255–270, 2001.

E. R. Wood, P. A. Dudchenko, R. Robitsek, and H. Eichenbaum. Hippocampal neurons encode information about different types of memory episodes occurring in the same location. *Neuron*, 27(3):623–633, 2000.

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv* 1708.07747, 2017.

D. Zeithamova, W. T. Maddox, and D. M. Schnyer. Dissociable prototype learning systems: Evidence from brain imaging and behavior. *J. Neurosci.*, 28(49):13194–13201, 2008.

K. Zhang, I. Ginzburg, B. L. McNaughton, and T. J. Sejnowski. Interpreting neuronal population activity by reconstruction: Unified framework with application to hippocampal place cells. *J. Neurophysiol.*, 79(2):1017–1044, 1998.