Sparse coding generates efficient representations for autoassociative memories

Yazhou Zhao (yz1103@nyu.edu)
Tandon School of Engineering, New York University

Zeyu Yun (chobitstian@berkeley.edu)
Redwood Center for Theoretical Neuroscience, UC Berkeley

Bruno A. Olshausen (baolshausen@berkeley.edu)
Redwood Center for Theoretical Neuroscience, UC Berkeley

Christopher J. Kymn (cjkymn@berkeley.edu)
Redwood Center for Theoretical Neuroscience, UC Berkeley

Abstract

We propose a two-layer computational neuroscience model for storing and retrieving sensory patterns in memory. The first layer, sparse coding, generates condensed yet explicit representations adapted to the statistics of natural scenes. The second layer, a complex-valued associative memory model, can store patterns generated by the first layer and recover partial or corrupted versions of them. We demonstrate the model's collective effectiveness at denoising and recalling sensory patterns from a dataset of natural images, with both layers providing complementary contributions to improving the peak signal-to-noise ratio. In addition, the invariance of the model to pairwise phase differences allows for partial generalization to similar scenes. Collectively, these principles are consistent with prior theory and experiments in neuroscience, and lead to potential predictions about inference mechanisms in biological neural networks.

Keywords: sparse coding; associative memory; neural network; phase coding; computational neuroscience

Introduction

Deciphering the computational basis of memory in neural circuits is a fundamental problem in both neuroscience and cognitive science. An influential theoretical model in the field is the attractor neural network, which has provided useful accounts of pattern completion and working memory, among others (Chaudhuri & Fiete, 2016). Such attractor networks, also known as associative memories, have improved our understanding of specific neural circuits, such as the fly head direction system (Kim, Rouault, Druckmann, & Jayaraman, 2017; Turner-Evans et al., 2020), and its principles have proven useful to artificial intelligence (Ackley, Hinton, & Sejnowski, 1985; Ramsauer et al., 2021). An unsolved yet fundamental question concerns the relation between sensory patterns, which are typically highly structured, and the abstract, typically random, patterns that are stored in attractor neural networks. This problem is delicate to address because direct storage of correlated sensory patterns will result in crosstalk noise that interferes with memory retrieval. One proposed mechanism is that *heteroassociative synaptic weights* link pre-defined patterns to those generated by sensory codes (Kanter & Sompolinsky, 1987; Kymn, Mazelet, Thomas, et al., 2024; Sharma, Chandra, & Fiete, 2022; Chandra, Sharma, Chaudhuri, & Fiete, 2025). An implication of these accounts is that weight matrices between the sensory patterns and random patterns assist in pattern separation.

In this work, we present an alternative hypothesis for the storage of patterns, inspired by the adaptation of sensory processing to the statistics of natural scenes (Barlow et al., 1961; Zetsche, 1990; Olshausen & Field, 2004; Hyvärinen, Hurri, & Hoyer, 2009). This approach hypothesizes that neural representations are tuned to efficiently represent the statistics of sensory inputs. In particular, *sparse coding* posits the existence of sparse, dimensionality-expanding ("overcomplete") representations, which are a recurring circuit motif in areas such as primary visual cortex, cerebellum, and hippocampus (Olshausen & Field, 2004). Here we argue that these sparse, overcomplete representations provide a helpful contribution to associative memory models, by transforming highly structured sensory input into decorrelated and explicit representations of sensory patterns. Such models support the intuition that it is useful to reduce the redundancy of patterns before storing them into memory, and they lead to a mechanistic hypothesis about how biological neural networks achieve this desideratum.

Our study draws upon and unites two existing methods in computational neuroscience: sparse coding and associative memory. We demonstrate that sparse coding provides reasonable pattern separation that is a strongly desirable property for storage in an associative memory. We then show that these two stages provide complementary abilities to denoise sensory patterns. In addition, we show that a specific feature of our network, its formulation in terms of complex values, allows for some generalization to similar scenes without additional memory requirements or training data. Finally, we discuss possible implications for neuroscience and for the design of associative memories.

Methods

Mathematically, we formulate the memory retrieval problem in terms of denoising a sensory pattern that has been stored in memory. This definition is practically useful because we can formulate a quantitative metric for the model. More specifically, we can investigate the retrieval accuracy and peak signal-to-noise ratio (PSNR) following the two components:

In D. Barner, N.R. Bramley, A. Ruggeri and C.M. Walker (Eds.), *Proceedings of the 47th Annual Conference of the Cognitive Science Society*

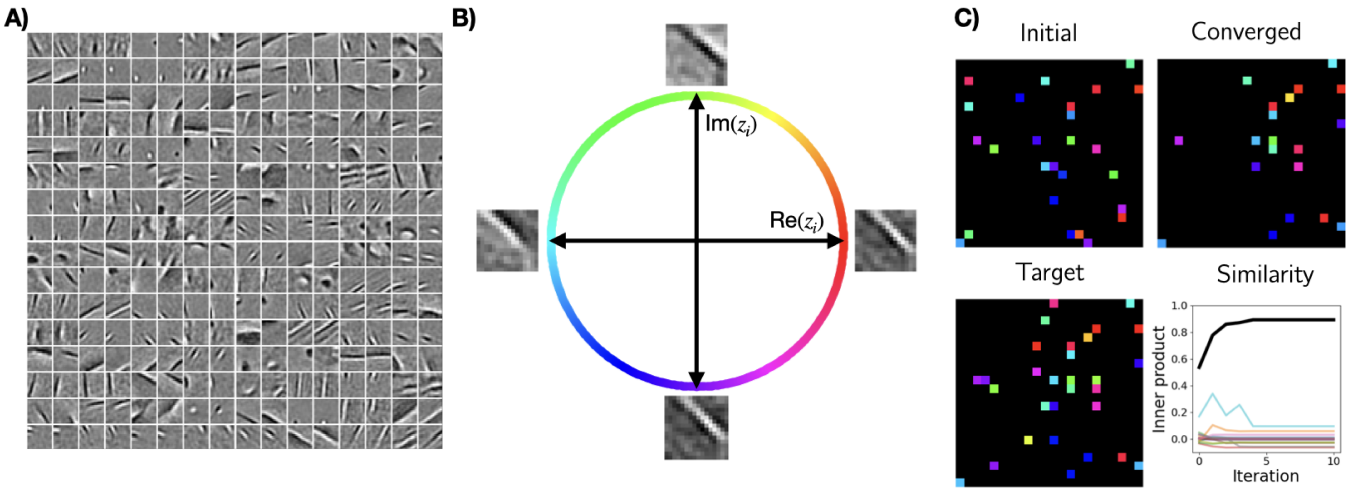

Figure 1: Visualization of main methods, subspace sparse coding and TPAM. **A)** Visualization of 256 of 1024 basis functions, which are the columns of the matrix **A**. Pairs of basis functions form two-dimensional subspaces that can be characterized by a complex number (that is, by an amplitude and phase). Only the subspace amplitudes are required to be sparse. **B)** The structure of a pair of latent variables can be mapped onto the complex plane. Changes in angle of the complex number reflect shifts in the phase of sparse code, while changes in the amplitude reflect changes in the intensity of the feature. The four images shown depict the receptive fields for four different phases. The sparsity penalty (second term of Equation 2) encourages the amplitudes to be small and does not reward or penalize specific phases. **C)** An example of the convergence of threshold phasor associative memory near a stored pattern. Most elements are zero (black), while others converge to specific phases (the hues shown in panel B indicate the phase). The similarity to each pattern for each iteration is also shown, with each line corresponding to one pattern and the thick black line corresponding to the nearest pattern.

sparse coding and associative memory dynamics.

Subspace sparse coding

Early studies of the primary visual cortex (V1) have shown that the response properties of many neurons are localized (responding only to certain parts of an image), oriented (preferentially responsive to features aligned in a specific angle), and bandpass (responsive to specific frequencies). Efficient coding models have shown how these three properties can be explained by a simple generative model that is adapted to the statistics of natural images (Olshausen & Field, 1996; Bell & Sejnowski, 1997). Subsequent work has shown that these models can also explain the properties of simple and complex cells (Hyvärinen & Hoyer, 1999; Berkes & Wiskott, 2005; Cadieu & Olshausen, 2012). Simple cells respond only to specific spatial phases (firing rates are sinusoidal in response to a moving grating), while others are invariant to the phase (firing rates are constant for such stimuli). In these models, complex cells are formed as a *quadrature pair* of simple cells, whose receptive fields are phase-shifted versions of each other (e.g., sine and cosine). Equivalently, we can think of the quadrature pair as representing a single complex number, with amplitude representing intensity and phase representing the spatial phase of the image.

In this work, we use *subspace sparse coding* (Paiton, Shepard, Chan, & Olshausen, 2020), which is a minimally sufficient model to generate latent representations with emergent quadrature pairs (examples of pairs are shown in Figure 1A).

Let $\mathbf{x} \in \mathbb{R}^D$ be an input data vector (in this case, we will consider vectorized images), $\mathbf{A} \in \mathbb{R}^{D \times N}$ is the "dictionary" of basis functions, and $\mathbf{s} \in \mathbb{R}^N$ is the latent representation. Figure 1A also demonstrates the example plot of the basis functions **A**. Sparse coding represents each image as a linear combination of basis functions:

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \nu \tag{1}$$

where ν denotes residual structure (assumed to be Gaussian) that is not explained by the basis functions.

Therefore, we must infer a code **s** for each particular image and learn a general dictionary **A** that is well-adapted to the statistics of our dataset. For all experiments presented here, we work with a dataset of whitened natural images from the American Northwest, which has been used in prior work on sparse coding (Olshausen & Field, 1996). In order to achieve a code that learns a sparse code adapted to the statistics of our dataset, we optimize **A** and **s** by minimizing the following loss function:

$$\begin{aligned} \mathcal{L} = \frac{1}{2}||\mathbf{x} - \mathbf{A}\mathbf{s}||_2^2 + \lambda \sum_{i=1}^{N/2} \sqrt{s_{2i-1}^2 + s_{2i}^2} \\ + \mu \sum_{i=1}^{N/2} |\mathbf{A}_{[2i-1:2i]}^T \mathbf{A}_{[2i-1:2i]} - \mathbf{I}_2| \end{aligned} \tag{2}$$

where the first term expresses the reconstruction error, and the second term encourages the amplitudes of each *subspace*

of two units to be small. The third term encourages the basis functions learned within a subspace to be orthogonal to each other, which encourages the learned receptive fields within each subspace to differentiate their responses. (The notation $\mathbf{A}_{[2i-1:2i]}$ is meant to emphasize that only the two columns in each group are considered for each term in the sum.) Model weights are learned by batch gradient descent. The λ and μ are hyperparameters that control the tradeoff between different training objectives (both are set to 0.1 in our experiments), and $\mathbf{I}_2$ is the 2×2 identity matrix.

We train an overcomplete sparse code, in which there are many more basis functions ($N = 1024$) than pixels ($D = 256$). Intriguingly, for natural images, these contain many dictionary elements that are not quite Gabor-like but that improve the sparsity of the representations (Olshausen, 2013). Additionally, subspace sparse coding can express richer structure than in standard sparse coding, which models all basis functions as statistically independent. Complex numbers can be decomposed into real and imaginary parts or amplitude and phase; the phase of each complex-valued component $\mathbf{z}_i$ corresponds to the spatial phases structure of natural scenes (Zetzsche, Krieger, & Wegmann, 1999). For example, slight translations of images can be expressed in phase (but not amplitude) shifts in the latent representation space. This property can be utilized by memory to store pattern more efficiently (Figure 4B).

Finally, the elements of the complex-valued vector $\mathbf{z} \in \mathbb{C}^{N/2}$ are formed based on the subspaces of basis functions:

$$z_i = s_{2i-1} + \sqrt{-1} s_{2i} \tag{3}$$

where the values s_{2i-1} and s_{2i} are the paired latent representations. That is, we set s_{2i-1} to the real part of the complex-valued vector component z_i, and s_{2i} to the imaginary part. This step allows us to store patterns in a complex-valued attractor neural network.

Threshold Phasor Associative Memory (TPAM)

Next, we store the sparse patterns into an autoassociative memory. We take as a starting point the threshold phasor associative memory model (TPAM) (Frady & Sommer, 2019), because it matches well with the sparse, complex-valued encodings learned by subspace sparse coding. As we will show, using the patterns learned by sparse coding helps reduce the cross-talk interference between patterns, increasing the signal-to-noise ratio.

First, we store a set of complex patterns $\{\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \ldots, \mathbf{z}^{(P)}\}$ using a "Hebbian" outer product learning rule:

$$\mathbf{J} = \sum_{p=1}^{P} \mathbf{z}^{(p)} \mathbf{z}^{(p)\dagger} \tag{4}$$

where $\dagger$ denotes the transpose conjugate, and P is the number of complex patterns.

Then, to recover patterns (possibly noisy) with TPAM, we first infer the initial vector $\mathbf{z}(0)$ by solving the inference algorithm for subspace sparse coding. We update the noisy image sparse patterns $\mathbf{z}(0)$ with the following rules:

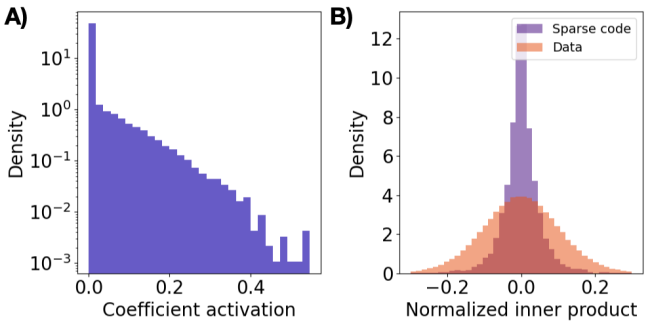

Figure 2: The inference procedure of sparse coding results in many values near zero, contributing to better pattern separation. **A)** Histogram of amplitudes of complex-valued coefficients z_i for natural image patches, with the y-axis shown on a log-scale. We observe that MAP estimation results in many values having zero or near-zero activation, which shows that the latent representations are very sparse.**B)** The inner products between distinct images is reduced with sparse, overcomplete representations compared to the original pixel-wise inner products between the same data. Inner products are normalized by the product of the two vector norms.

The dynamics can be decomposed into three steps: multiplication by the weight matrix $\mathbf{J}$, thresholding, and normalization.

$$\mathbf{u}(t) = f_\theta(\mathbf{J}\mathbf{z}(t)) \tag{5}$$

$$\mathbf{z}(t+1) = \frac{\mathbf{u}(t)}{\|\mathbf{u}(t)\|} \tag{6}$$

The term f_θ is a thresholding function applied elementwise that sets all values less than θ to 0. Like λ and β, it is a hyperparameter that should be adapted to the sparsity of the stored patterns. Empirically, we observed that setting the threshold near the L0 pattern sparsity (after applying a threshold that sets very small coefficients to 0) resulted in the highest accuracy of retrieval, although slight deviations from this value did not strongly impact these results. Finally, after a finite number of discrete-time updates, the output vector $\mathbf{z}$ is used to reconstruct the image via the basis function matrix $\mathbf{A}$.

Because the weights are Hermitian symmetric, the network dynamics are known to descend an energy function which is known to be real-valued, finite and non-increasing over the dynamics (Noest, 1987; Frady & Sommer, 2019). An example simulation of the dynamics of TPAM is shown in Figure 1C. The network is initialized near the target pattern but is not identical to it. The final state of the network converges in fewer than 10 iterations to a result that is more similar to the target state. Note, however, that the convergence is not exact due to the phenomenon of "cross-talk" interference due to a small but non-zero similarity between patterns.

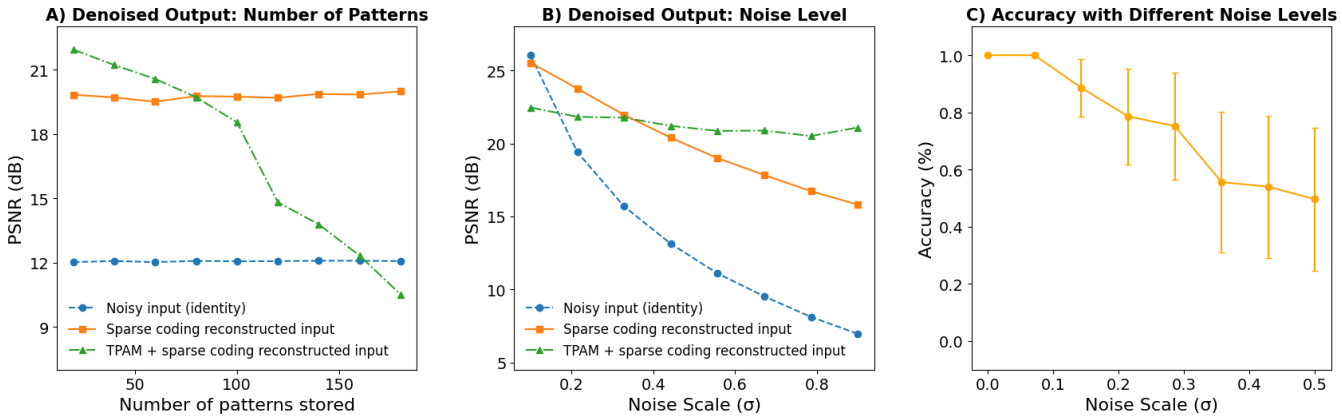

Figure 3: **A)** Peak signal-to-noise ratio (PSNR) of noisy and denoised images by sparse coding model and TPAM (the red line) with different patterns stored. Plots depict the average across 5 different trials, where each trial assesses the average PSNR for all patterns stored in memory. The noise scale is 0.5 across all trials. **B)** PSNR of noisy and denoised images by sparse coding model and TPAM with different noise levels applied. Once again, plots show averaged results from 5 trials; 50 patterns are stored per trial. **C)** The correctness of the model in retrieving patterns with different noise levels applied. Each item shows the average accuracy based on 500 different randomly selected patterns. Bars indicate the variance based on the empirical accuracy.

Results

Sparse inference effectively separates patterns

Conventional associative memories suffer when storing data such as natural images, because the patterns are correlated and contribute higher levels of cross-talk noise during retrieval. The typical solution to this problem consists of a *whitening transformation*, or a linear transformation of the data that helps to decorrelate it. While this improves over the correlated baseline, it still suffers from interference due to higher-order (that is, greater than second-order) correlations still present within the data (Kymn, Mazelet, Ng, Kleyko, & Olshausen, 2024). This whitening transformation can be performed either on the data matrix itself or implicitly with projections in a pseudoinverse learning rule.

Here, instead, we propose using sparse coding as a method for learning higher-order structure that orthogonalizes patterns. We optimize the coefficients $\mathbf{s}$ with gradient descent, which amounts to taking a maximum a posteriori (MAP) estimate of the latent variables in the coefficients. Due to competition between basis functions, points that may have partial similarity will be closer to orthogonal in the latent space (Figure 2B). This reduction in pattern similarity directly contributes to less "cross-talk" interference between the patterns being stored, allowing for a higher signal-to-noise during subsequent reconstruction. In addition, the sparsity of the inferred representations implies that many terms in the outer product learning rule are zero or near zero, reducing the explicit storage requirements per pattern.

Sparse coding and TPAM function as complementary generative models

Denoising can be interpreted as a generative process: when presented with a noisy image, the brain synthesizes an internal explanation of the visual scene. Sparse coding offers a bottom-up approach to this process. Specifically, when a noisy image is fed into the primary visual cortex (V1), it is decomposed into a sparse combination of atomic elements. Because noisy components produce only small activations, they are largely suppressed by the sparsity term in the loss function (Equation 2). This leads to effective denoising regardless of the specific image content. Indeed, as shown in Figure 3A, images reconstructed via sparse coding consistently have higher peak signal-to-noise ratios (PSNR) than the original noisy inputs, indicating that the denoising is successful. The following is the definition of the PSNR, where MAX is the maximum pixel value of the image:

$$\text{PSNR} = 10\log_{10}\left(\frac{\text{MAX}^2}{\text{MSE}}\right), \tag{7}$$

where

$$\text{MSE} = \frac{1}{D}\sum_{i=1}^{D}(x_i - \hat{x}_i)^2$$

In contrast, TPAM provides a top-down generative mechanism: a highly noisy input can be accurately recovered, provided it was previously stored in the memory. As illustrated in Figure 3B, TPAM yields a considerably greater improvement in PSNR than sparse coding alone as noise levels increase. As shown in Figure 3C, it also keeps the correctness of retrieving patterns when noise is applied. This behavior aligns with psychological findings showing that, when visual stimuli are extremely degraded, memory recall is often essential for object recognition (Bar, 2004; Oliva & Torralba, 2007). However, being a memory-based model, TPAM is limited by the number of patterns it can store. Once too many patterns are encoded, the network may converge to spurious fixed points

that do not correspond to the original stored patterns. This limitation is evident in Figure 3A, which indicates that the PSNR of TPAM reconstructions decreases as the number of stored patterns increases.

Phase-shift invariance enables generalization to new but semantically similar scenes

The dynamics of complex-valued attractor networks have a symmetry, in that the energy is invariant to constant phase shifts of the image. More concretely, if two vectors $\mathbf{y}$ and $\mathbf{z}$ are equivalent up to a constant phase shift, then they will be assigned equal energy (Figure 4A). For the case of storing sensory patterns, this carries the benefit of generalizing to new kinds of scenes with similar pairwise relationships in phase. Relationships between phase variables are highly useful to forming representations of images that are invariant to continuously varying parameters such as rotation, scale, and translation (Cohen & Welling, 2014; Chau, Qiu, Chen, & Olshausen, 2022).

In addition, we show that these continuity properties can be used for generalization to new but semantically similar patterns, such as images generated by small pixel-wise shifts of the original memory. The PSNR for a 1-pixel shift remains higher than the baseline denoising offered by sparse coding, even though this shifted image was not explicitly stored in the memory (Figure 4B). These shifted patterns are technically not minima of the energy landscape, because the spatial frequencies of different elements in the sparse code are different. However, for small shifts the patterns are still close enough to allow some generalization without additional training data or storage required. Using the intrinsic properties of the attractor manifold to take into account further structure in pairwise phase statistics remains a promising area for future study.

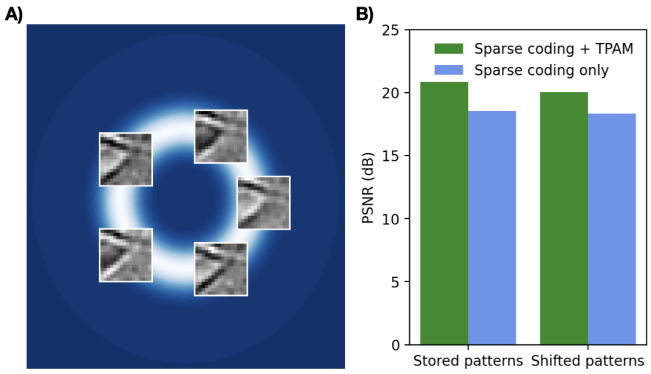

Figure 4: The equivariance of the energy function to phase shifts enables generalization to similar yet unstored scenes. **A)** Attractor points form a "ring" of equal energy, in which patterns that are phase shifted versions of each other have equal energy. The five images pictured have equal energy in the space, and the background is an analogy meant to show that there is a continuum of such points. **B)** This equivariance property helps the attractor network denoise patterns with slight shifts with favorable PSNR. Though these patterns are not of identical energy, they are close enough to still allow partial retrieval.

Discussion

We present a two-stage model of memory compression, consolidation, and retrieval. The first stage, sparse coding, learns an efficient basis for natural images that is able to compress sensory patterns into a small set of active units. The second stage, associative memory, allows further denoising to retrieve a stored memory. The formulation in terms of complex-valued coefficients allows for generalization to patterns with the same pairwise phase statistics.

Our methods bear similarities, in name and in spirit, to some previously existing approaches. Foundational work on *sparse associative memories* has shown empirically and theoretically that sparse patterns result in a higher storage capacity compared to their dense counterparts (Golomb, Rubin, & Sompolinsky, 1990; Palm, 2013; Rachkovskij, Kussul, & Baidyk, 2013; Knoblauch & Palm, 2020). These studies typically focus on randomly initialized patterns, rather than the outputs of an optimization procedure as in sparse coding. *Sparse distributed memory* (Kanerva, 1988) is a mathematical model, originally connected to the cerebellum, that uses random projection and thresholds in a high-dimensional vector space to store pattern in memory. Related models based on similar principles appear to be implemented in the *Drosophila* olfactory circuit (Dasgupta, Stevens, & Navlakha, 2017; Kleyko & Rachkovskij, 2024). Such models can be used as an alternative to auto-associative or hetero-associative memories (Keeler, 1988) and the mathematical operations bear similarities to those of transformer-based architectures in deep learning (Bricken & Pehlevan, 2021). Finally, *autoencoders* have been used to complement or implement an associative memory (Benna & Fusi, 2021; Sharma et al., 2022; Radhakrishnan, Belkin, & Uhler, 2020). Sparse autoencoders can be thought of as an approximate version of the computations we are performing here, and have been able to recover similar properties under some cases (Bricken, Schaeffer, Olshausen, & Kreiman, 2023).

A rather different approach that shares similar ambitions is to use dense associative memories to store patterns (Krotov & Hopfield, 2016). These models have higher capacity than typical Hopfield networks with the additional requirements of a greater number of parameters and computation of higher-order (polynomial or exponential) interactions. It remains an open question to what extent these networks can be distilled into smaller sparse networks, which could be easier to interpret and potentially closer to the wiring solutions instantiated by biological neural networks (Krotov & Hopfield, 2021).

Many of the parameters and design choices of our two-layer model are taken directly from neural coding motifs observed in brains. These include the prevalence of population codes with sparse neural activity, overcomplete representations (larger representation dimension relative to input), and recurrent mechanisms that may implement and improve in-

ference of latent variables.

However, the experimental predictions for this model differ from the heteroassociative memory models mentioned earlier. For example, in heteroassociative memories, the patterns in the associative memory are chosen randomly without adaptation or dependence on the data points being stored, implying that only the heteroassociative projection weights are learned. However, in the model considered here, the patterns stored in associative memory are created after sensory experiences and the resulting efficient code of it. In addition, heteroassociative models typically consist of linear projections followed by threshold functions, whereas sparse coding and associative memory both require recurrent inference procedures. These mechanistic differences could lead to testable and different predictions for regions implicated in storing episodic memories, including the hippocampus and entorhinal cortex.

The model and experiments presented here are primarily a proof of concept that sparse coding algorithms are complementary to the goals of storing patterns within an associative memory. The relatively simple structure of this two-layer model could be extended in several different ways to strengthen its connections to neuroscience and usefulness for connectionist projects. For example, one straightforward extension is to replace our one-layer sparse coding algorithm with a hierarchical sparse coding model, to better reflect the purported hierarchical processing of visual cortex. Such a scheme could yield representations with even stronger forms of sparsity or generalization. A second related direction is to investigate the scaling of network properties with different dimensions and larger images, which could be efficiently implemented with convolutional variants of sparse coding algorithms (Zeiler, Krishnan, Taylor, & Fergus, 2010). A third complementary approach is to further develop the probabilistic foundations of the connection between associative memory. For example, while our approach focuses on memorizing specific examples, others working on associative memory models have explored the border between memorizing specific examples and generalizing to a distribution of remembered patterns (Kang & Toyoizumi, 2024; Tyulmankov, Stachenfeld, Krotov, & Abbott, 2023; D'Amico, Rossi, del Bono, & Negri, 2025). It would be insightful to develop further connections with these existing lines of work, in orer to distill hypotheses and insights about how the brain solves similar kinds of problems.

Acknowledgments

We thank our reviewers for their suggestions, in addition to members of the Redwood Center for Theoretical Neuroscience for feedback. The work of CJK was supported by the Department of Defense (DoD) through the National Defense Science & Engineering Graduate (NDSEG) Fellowship Program. The work of CJK, ZY, and BAO was supported by the Center for the Co-Design of Cognitive Systems (CoCoSys), one of seven centers in JUMP 2.0, a Semiconductor Research Corporation (SRC) program sponsored by DARPA, as well as NSF awards 2147640 and 2313149.

References

Ackley, D. H., Hinton, G. E., & Sejnowski, T. J. (1985). A learning algorithm for boltzmann machines. *Cognitive science*, *9*(1), 147–169.

Bar, M. (2004). Visual objects in context. *Nature Reviews Neuroscience*, *5*(8), 617–629.

Barlow, H. B., et al. (1961). Possible principles underlying the transformation of sensory messages. *Sensory communication*, *1*(01), 217–233.

Bell, A. J., & Sejnowski, T. J. (1997). The "independent components" of natural scenes are edge filters. *Vision research*, *37*(23), 3327–3338.

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

Berkes, P., & Wiskott, L. (2005). Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of vision*, *5*(6), 9–9.

Bricken, T., & Pehlevan, C. (2021). Attention approximates sparse distributed memory. *Advances in Neural Information Processing Systems*, *34*, 15301–15315.

Bricken, T., Schaeffer, R., Olshausen, B., & Kreiman, G. (2023). Emergence of sparse representations from noise. In *Proceedings of the 40th international conference on machine learning* (pp. 3148–3191).

Cadieu, C. F., & Olshausen, B. A. (2012). Learning intermediate-level representations of form and motion from natural movies. *Neural computation*, *24*(4), 827–866.

Chandra, S., Sharma, S., Chaudhuri, R., & Fiete, I. (2025). Episodic and associative memory from spatial scaffolds in the hippocampus. *Nature*, 1–13.

Chau, H. Y., Qiu, F. Y., Chen, Y., & Olshausen, B. (2022). Disentangling images with lie group transformations and sparse coding. In *Neurips 2022 workshop on symmetry and geometry in neural representations.*

Chaudhuri, R., & Fiete, I. (2016). Computational principles of memory. *Nature neuroscience*, *19*(3), 394–403.

Cohen, T., & Welling, M. (2014). Learning the irreducible representations of commutative lie groups. In *International conference on machine learning* (pp. 1755–1763).

D'Amico, F., Rossi, S., del Bono, L. M., & Negri, M. (2025). Pseudo-likelihood produces associative memories able to generalize, even for asymmetric couplings. In *New frontiers in associative memories.*

Dasgupta, S., Stevens, C. F., & Navlakha, S. (2017). A neural algorithm for a fundamental computing problem. *Science*, *358*(6364), 793–796.

Frady, E. P., & Sommer, F. T. (2019). Robust computation with rhythmic spike patterns. *Proceedings of the National Academy of Sciences*, *116*(36), 18050–18059.

Golomb, D., Rubin, N., & Sompolinsky, H. (1990). Willshaw model: Associative memory with sparse coding and low

firing rates. *Physical Review A*, *41*(4), 1843.

Hyvärinen, A., & Hoyer, P. (1999). Emergence of complex cell properties by decomposition of natural images into independent feature subspaces. In *1999 ninth international conference on artificial neural networks icann 99.(conf. publ. no. 470)* (Vol. 1, pp. 257–262).

Hyvärinen, A., Hurri, J., & Hoyer, P. O. (2009). *Natural image statistics: A probabilistic approach to early computational vision.* (Vol. 39). Springer Science & Business Media.

Kanerva, P. (1988). *Sparse distributed memory*. MIT press.

Kang, L., & Toyoizumi, T. (2024). Distinguishing examples while building concepts in hippocampal and artificial networks. *Nature Communications*, *15*(1), 647.

Kanter, I., & Sompolinsky, H. (1987). Associative recall of memory without errors. *Physical Review A*, *35*(1), 380.

Keeler, J. D. (1988). Comparison between kanerva's sdm and hopfield-type neural networks. *Cognitive Science*, *12*(3), 299–329.

Kim, S. S., Rouault, H., Druckmann, S., & Jayaraman, V. (2017). Ring attractor dynamics in the drosophila central brain. *Science*, *356*(6340), 849–853.

Kleyko, D., & Rachkovskij, D. A. (2024). On design choices in similarity-preserving sparse randomized embeddings. In *2024 international joint conference on neural networks (ijcnn)* (pp. 1–8).

Knoblauch, A., & Palm, G. (2020). Iterative retrieval and block coding in autoassociative and heteroassociative memory. *Neural Computation*, *32*(1), 205–260.

Krotov, D., & Hopfield, J. J. (2016). Dense associative memory for pattern recognition. *Advances in neural information processing systems*, *29*.

Krotov, D., & Hopfield, J. J. (2021). Large associative memory problem in neurobiology and machine learning. In *International conference on learning representations.*

Kymn, C. J., Mazelet, S., Ng, A., Kleyko, D., & Olshausen, B. A. (2024). Compositional factorization of visual scenes with convolutional sparse coding and resonator networks. In *2024 neuro inspired computational elements conference (nice)* (pp. 1–9).

Kymn, C. J., Mazelet, S., Thomas, A., Kleyko, D., Frady, E., Sommer, F. T., & Olshausen, B. A. (2024). Binding in hippocampal-entorhinal circuits enables compositionality in cognitive maps. *Advances in Neural Information Processing Systems*, *37*, 39128–39157.

Noest, A. (1987). Phasor neural networks. In *Neural information processing systems.*

Oliva, A., & Torralba, A. (2007). The role of context in object recognition. *Trends in cognitive sciences*, *11*(12), 520–527.

Olshausen, B. A. (2013). Highly overcomplete sparse coding. In *Human vision and electronic imaging xviii* (Vol. 8651, pp. 168–176).

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

Olshausen, B. A., & Field, D. J. (2004). Sparse coding of sensory inputs. *Current opinion in neurobiology*, *14*(4), 481–487.

Paiton, D. M., Shepard, S., Chan, K. H. R., & Olshausen, B. A. (2020). Subspace locally competitive algorithms. In *Proceedings of the 2020 annual neuro-inspired computational elements workshop* (pp. 1–8).

Palm, G. (2013). Neural associative memories and sparse coding. *Neural Networks*, *37*, 165–171.

Rachkovskij, D. A., Kussul, E. M., & Baidyk, T. N. (2013). Building a world model with structure-sensitive sparse binary distributed representations. *Biologically Inspired Cognitive Architectures*, *3*, 64–86.

Radhakrishnan, A., Belkin, M., & Uhler, C. (2020). Overparameterized neural networks implement associative memory. *Proceedings of the National Academy of Sciences*, *117*(44), 27162–27170.

Ramsauer, H., Schäfl, B., Lehner, J., Seidl, P., Widrich, M., Gruber, L., ... Hochreiter, S. (2021). Hopfield networks is all you need. In *International conference on learning representations.*

Sharma, S., Chandra, S., & Fiete, I. (2022). Content addressable memory without catastrophic forgetting by heteroassociation with a fixed scaffold. In *International conference on machine learning* (pp. 19658–19682).

Turner-Evans, D. B., Jensen, K. T., Ali, S., Paterson, T., Sheridan, A., Ray, R. P., ... others (2020). The neuroanatomical ultrastructure and function of a biological ring attractor. *Neuron*, *108*(1), 145–163.

Tyulmankov, D., Stachenfeld, K., Krotov, D., & Abbott, L. (2023). Memorization and consolidation in associative memory networks. In *Associative memory & hopfield networks in 2023 (neurips workshop).*

Zeiler, M. D., Krishnan, D., Taylor, G. W., & Fergus, R. (2010). Deconvolutional networks. In *2010 ieee computer society conference on computer vision and pattern recognition* (pp. 2528–2535).

Zetsche, C. (1990). Sparse coding: the link between low level vision and associative memory. *Parallel processing in neural systems and computers*.

Zetzsche, C., Krieger, G., & Wegmann, B. (1999). The atoms of vision: Cartesian or polar? *JOSA A*, *16*(7), 1554–1565.