
Learning Sparse Functional Representations with Neural Operators

Bahareh Tolooshams*

University of Alberta
Alberta Machine Intelligence Institute (Amii)
Canada CIFAR AI Chair

Ailsa Shen*

California Institute of Technology (Caltech)

Anima Anandkumar

California Institute of Technology (Caltech)

Abstract

Physical fields – vorticity, pressure, temperature – are continuous functions over spatial domains, often observed at varying resolutions and on irregular grids. Sparse Autoencoders (SAEs) have shown promise for learning structured representations, but their fixed-dimensional Euclidean parameterization ties concepts to specific discretizations and encodes only concept presence, making them poorly suited for this setting. We introduce sparse autoencoder neural operators (SAE-NOs), operating in function spaces instead. We formalize the functional representation hypothesis, modeling data as sparse compositions of structured functions, and instantiate it as Fourier neural operators with joint concept and domain sparsity. Concept sparsity selects which concepts are active; domain sparsity governs where and how they are expressed. SAE-FNO learns localized concepts with stable structure across sparsity levels, reuses them across independently processed translated frames, and represents data with fewer effective concepts. Crucially, SAE-FNO generalizes to unseen resolutions where standard SAEs fail.

1 Introduction

Physical fields such as vorticity, pressure, and temperature are continuous functions over spatial domains, but the discretizations at which they are sampled routinely vary within a dataset and change between training and deployment. Building on sparse representation learning [1], sparse autoencoders (SAEs) [2] learn structured, interpretable representations [3] and are widely used in mechanistic interpretability [4–7]. Most existing SAEs rely on the linear representation hypothesis (LRH), which models data as sparse combinations of linearly accessible, approximately orthogonal concepts [4, 8, 9] – though neural representations may exhibit richer organization [7, 10].

The dominant SAE formulation, which we refer to as SAE-MLP, has shallow MLP encoders and linear decoders [5, 6, 11, 12]. This design restricts what a concept can represent: i) concept utilization is summarized by a scalar activation, encoding only presence and strength; ii) concepts are vectors in the input domain; and iii) learning and inference occur at a fixed Euclidean discretization. Consequently, codes encode whether concepts are present, but not *how* or *where* they are expressed across the domain, limiting their ability to capture localized or domain-dependent structure. These limitations are particularly important in scientific modeling, computer vision, and physical-signal analysis, where data often take the form of structured fields or spatiotemporal processes [13, 14]. Hence, we focus on capturing continuous structure within the representation.

*equal contribution

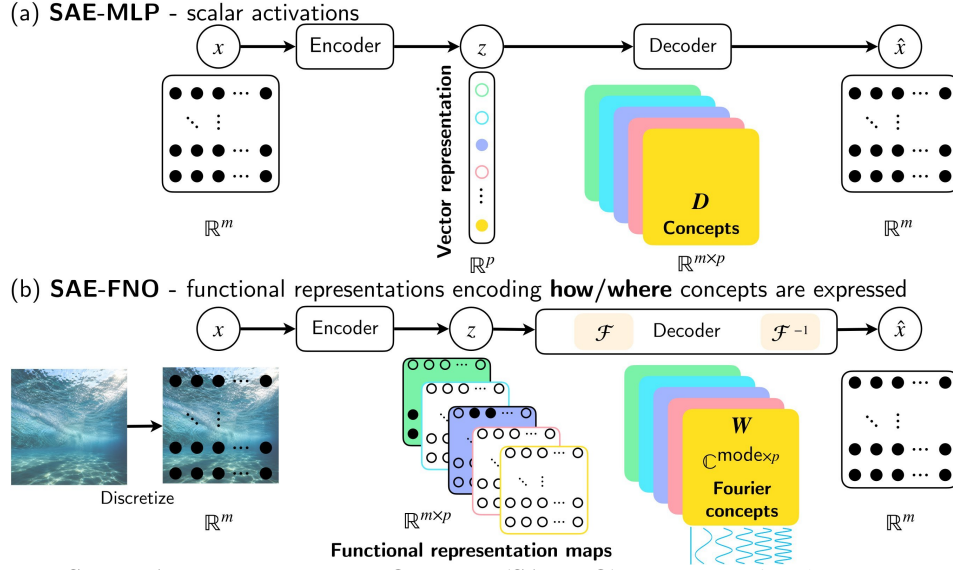


Figure 1: **Sparse Autoencoder Neural Operator (SAE-NO)**. (a) Conventional SAEs (SAE-MLP) parameterize concepts D in the Euclidean domain, assigning each concept a scalar activation indicating presence and strength. (b) SAE-FNOs use functional concept parameterization via Fourier integral operators, encoding *concept sparsity* and *domain sparsity* to capture how and where concepts are expressed. SAE-FNO treats inputs as samples from an underlying continuous function and supports querying of representations, reconstructions, and concepts at varying resolutions.

We therefore turn to neural operators (NOs) [13, 15–17], which learn mappings between function spaces and naturally generalize across discretizations. Neural operators are central to data-driven scientific modeling [18] and have shown promise in neuroscience [19]. However, while their predictive capabilities and extension to function-space autoencoders are established [20–22], their representational properties and potential for sparse concept learning remain largely unexplored.

We introduce sparse autoencoder neural operators (SAE-NOs), which learn concepts as functions rather than vectors, enabling resolution-invariant representations. We formalize the functional representation hypothesis (FRH) and instantiate it using Fourier [17] neural operators (SAE-FNOs).

We replace scalar codes with feature maps and introduce *concept sparsity* to select active concepts and *domain sparsity* to determine *how* and *where* they are expressed (Figure 1). SAE-FNO reuses concepts across spatial translations over time by shifting their domain activations, whereas SAE-MLP switches among distinct concepts. This consistency emerges without temporal priors or sequence-level training (Figure 2, see also Figures 5, 6). Moreover, SAE-FNO learns localized, frequency-structured concepts whose characteristics remain stable across concept sparsity levels. This decouples structure, controlled through domain sparsity, from expressivity, controlled through concept sparsity. SAE-FNO represents the data population using fewer effective concepts and achieves greater expressivity per concept, particularly as the domain grows. Finally, unlike fixed-resolution SAEs, SAE-FNO generalizes across discretizations and operates at resolutions beyond those seen during training.

2 Sparse Autoencoder Neural Operators

We introduce sparse autoencoder neural operators (SAE-NOs). Let $\mathcal{Z}$ and $\mathcal{X}$ be Banach spaces of functions on bounded domains, and let $\mathcal{G}_{D^*} : \mathcal{Z} \rightarrow \mathcal{X}$ be a generative operator. A sample $x \in \mathcal{X}$ follows the FRH if $x = \mathcal{G}_{D^*}(z)$, $z \sim \mu$, where μ is supported on sparse functions. We formalize SAE-NO with an encoder approximating the inverse generative operator and a decoder that captures the underlying operator and concepts from FRH. We instantiate it using a Fourier parameterization, defining Sparse Autoencoder Fourier Neural Operators (SAE-FNOs) as $\mathcal{G}_D(z) := \mathcal{F}^{-1}(\sum_{c=1}^p \mathbf{W}_c \cdot (\mathcal{F}z_c))$, where $\mathbf{W}_c = \mathcal{F}D_c$ is the Fourier transform of D_c . SAE-FNO implements

$$\begin{aligned}
 & \min_{\mathbf{W} \in \mathcal{W}} \frac{1}{2} \|x - \mathcal{F}^{-1}(\mathbf{W} \cdot \mathcal{F}z)\|_2^2 + \lambda \mathcal{R}(z) \\
 & \text{s.t. } \tilde{v} = \mathcal{F}^{-1}(\mathbf{E} \cdot (\mathcal{F}x)) \\
 & \text{(concept sparsity)} \quad v = \mathbf{1}_a \cdot \tilde{v}, \quad \text{where } a = \Pi([\|\tilde{v}_1\|_2, \|\tilde{v}_2\|_2, \dots, \|\tilde{v}_p\|_2]^\top) \\
 & \text{(domain-sparsity)} \quad z_c = \mathcal{P}(v_c), \quad \text{for } c = 1, \dots, p, \quad \text{where } \|v_c\|_2 > 0,
 \end{aligned} \tag{1}$$

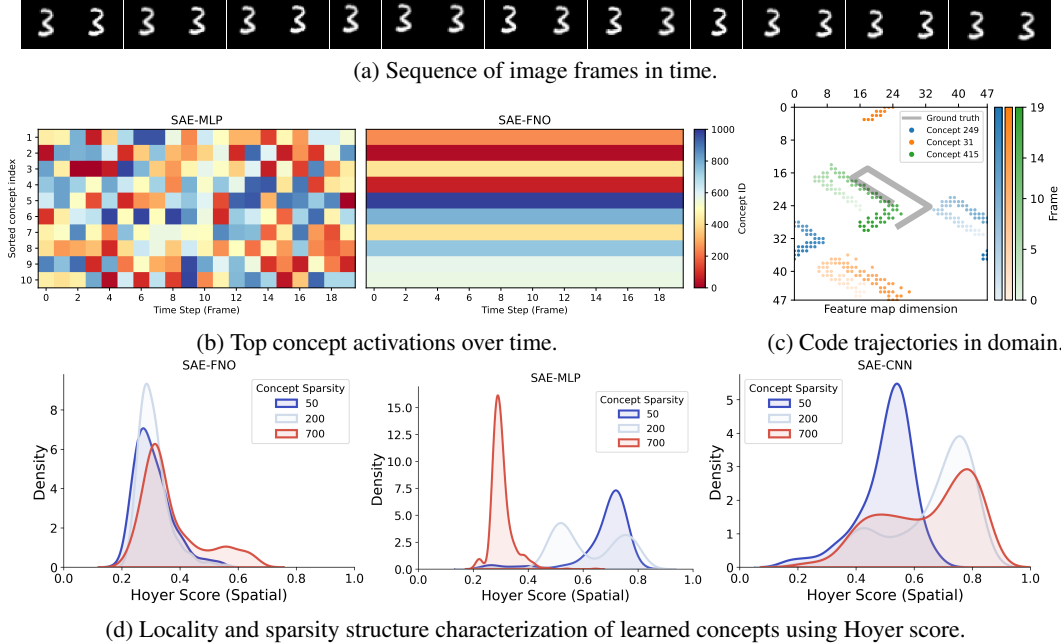


Figure 2: **Encoding where and how concepts are expressed.** (a) A digit translates across the domain over frames. (b) Top-10 concepts over time: SAE-MLP switches concepts as the digit moves, whereas SAE-FNO maintains stable concepts whose spatial activations shift. (c) Spatial feature-map trajectories of the top-3 SAE-FNO concepts follow the ground-truth motion; color denotes concept and brightness denotes time. (d) SAE-FNO maintains stable concept characteristics across sparsity levels. In contrast, SAE-MLP and SAE-CNN show significant changes in concept structure.

where $\mathcal{R}(z)$ is a sparse regularizer, $\tilde{v}$ is the dense pre-activation map, $\mathbf{1}_a$ zeros out entire concepts with low energy, and Π and $\mathcal{P}$ are the sparsifying projections for concept and domain, respectively.

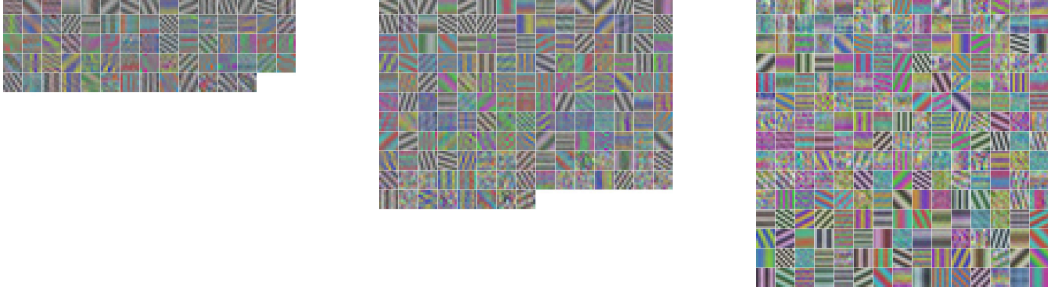
SAE-FNO captures how and where concepts are expressed, going beyond concept presence. We introduce two distinct notions of sparsity: *concept* sparsity and *domain* sparsity. Concept sparsity extends the standard notion of sparsity in SAEs by enforcing that only a small number of concepts are used to explain each data sample. Domain sparsity constrains the activation pattern within the feature map associated with each concept. Concept sparsity induces a form of *group sparsity* (where all entries of a concept’s feature map constitute one group), and domain sparsity enforces element-wise sparsity within that feature map. Intuitively, concept sparsity decides *which* concepts are active, and domain sparsity determines *where* each active concept is expressed across the input domain.

As demonstrated in the results section, this joint sparsity structure enables SAE-FNO to learn concepts that are robust to variations in concept sparsity and shared across data examples, while being expressed differently through example-specific domain sparsity patterns. For example, SAE-FNO can represent translated instances of the same pattern using a stable set of concepts whose spatial activations shift across the domain, whereas SAE-MLPs require switching between multiple distinct concepts to capture such variations (Figure 2). This ability to reuse concepts through domain sparsity leads to more efficient representations, requiring fewer concepts to explain the data.

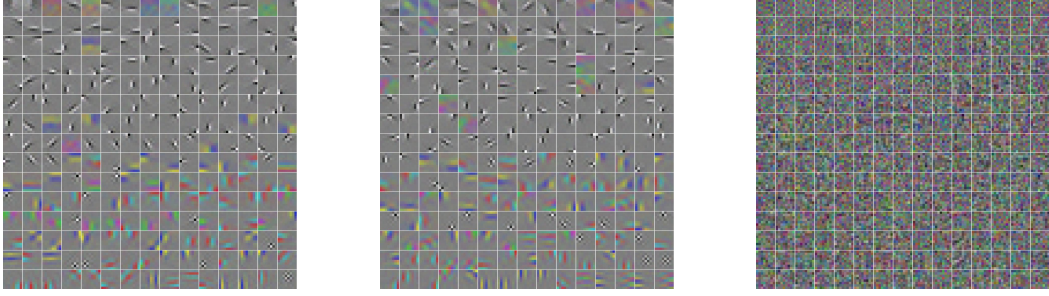
Importantly, this notion of concept stability across time emerges naturally from the Fourier parameterization and domain sparsity of SAE-FNO. In Figure 2, frames are treated as i.i.d. training examples, without imposing temporal smoothness constraints, sequence-level objectives, or temporal priors during training. This contrasts with recent works that explicitly model temporal consistency or dense/sparse structure across sequences through coordinated training over time [23, 24].

3 Results

Following work that studies SAEs on interpretable outputs rather than internal activations [25, 26], we train SAE-FNOs directly on spatially structured data. Images are scalar fields sampled on a regular spatial grid, making them a natural testbed for the discretization behavior we study. All comparisons match the sparsification mechanism and training objective, and SAE-FNO concepts are mapped



(a) SAE-FNO as concept sparsity level changes (left) $k = 25$ (middle) $k = 50$ (right) $k = 200$.



(b) SAE-MLP as concept sparsity level changes (left) $k = 25$ (middle) $k = 50$ (right) $k = 200$.

Figure 3: **Learned concepts across the dataset.** Each panel visualizes the most utilized concepts (each square is one concept) learned on CIFAR-10 8×8 images with TopK and $p = 1000$, for varying k . Fewer displayed concepts indicate more efficient concept reuse across the data population. (a) SAE-FNO exhibits stable concept characteristics across sparsity levels due to domain sparsity. As sparsity decreases, more concepts become active while preserving similar structure. (b) SAE-MLP concepts vary significantly with sparsity, losing structure and failing to capture locality (see Figure 12).

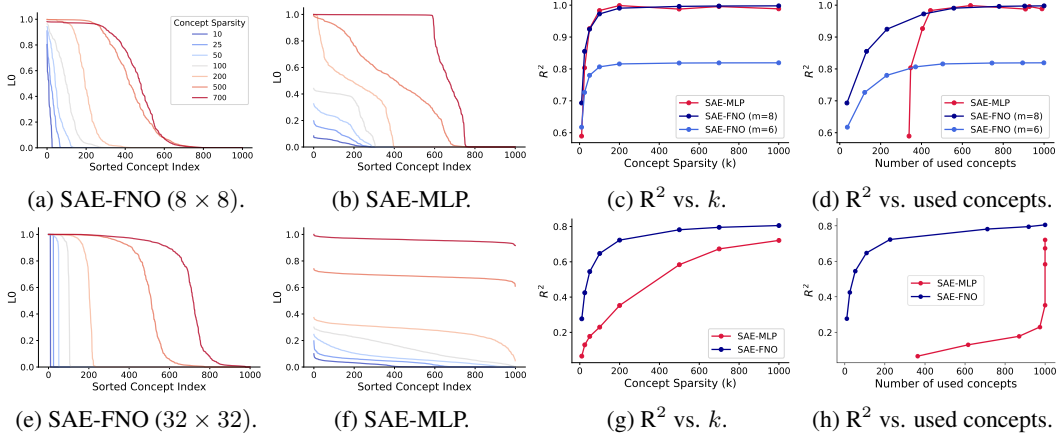


Figure 4: **Concept utilization and expressivity.** First row: 8×8 ; second row: 32×32 for CIFAR10 and TopK. (a,b,e,f) ℓ_0 code statistics (y-axis) across the data population, plotted against concept indices sorted by decreasing utilization (x-axis). SAE-FNO shows a sharp decay, indicating efficient concept usage, while SAE-MLP exhibits a flatter distribution, requiring more concepts, an effect that worsens for larger patches. (c,g) R^2 vs. k . (d,h) R^2 vs. the number of used concepts; SAE-FNO achieves higher expressivity with fewer concepts, whereas SAE-MLP fails to scale to larger domains.

from their Fourier parameterization to the data domain for analysis. We therefore attribute observed differences to functional parameterization and refer to matched models as SAE-MLP and SAE-FNO.

Structured and Stable Concepts We characterize concept structure using the Hoyer score [27] (Section B), where values near 1 indicate localized structure and values near 0 indicate dense concepts. Across concept sparsity levels k , SAE-FNO maintains stable structural characteristics, whereas SAE-MLP and SAE-CNN change substantially (Figure 2d, see also Table 2). This does not imply that concepts remain identical: SAE-FNO preserves a stable subset while adding new concepts as k increases, as further evaluated through identity consistency in Figure 9.

SAE-NO Concept Correlations SAE-FNOs permit a broader range of concept correlations. This contrasts with shallow SAEs, which tend to produce approximately orthogonal features [4, 7]. Figures 10, 11 show that SAE-FNO exhibits a broader distribution of pairwise concept correlations, whereas SAE-MLPs are sharply concentrated near zero. This flexibility arises from the functional parameterization of SAE-FNO, where concepts are represented as feature maps with concept and domain sparsity. Consequently, concepts can remain moderately correlated while being expressed differently across the domain, reducing effective interference between them.

Concept Visualization When mapped from their Fourier parameterization to the data domain, SAE-FNO concepts form sinusoidal filters on local patches and localized frequency patterns on larger images, whereas SAE-MLP struggles to capture spatial locality (Figure 3, see also Figures 12 to 22 highlighting the stable behaviour of SAE-FNO compared to SAE-MLP and SAE-CNN).

Efficient Concept Usage and Expressivity The structural differences between SAE-FNO and SAE-MLP lead to more efficient concept usage. SAE-FNO exhibits a sharper decay in population-level concept utilization and achieves higher R^2 with fewer used concepts, with the advantage increasing as the input domain grows (Figure 4). Figure 25 also shows expressivity as a function of dictionary size. These gains persist across TopK, BatchTopK, and Matryoshka (Figures 23, 24).

Generalization Across Discretizations Conventional SAEs learn concepts tied to a fixed Euclidean grid. In contrast, SAE-FNO inherits the discretization flexibility of neural operators, allowing the same learned representation to operate at resolutions unseen during training. As shown in Figure 27, SAE-FNO maintains accurate reconstructions at higher resolutions, whereas SAE-MLP reconstruction error increases sharply beyond the training discretization (see also Figure 28).

4 Discussion and Conclusion

Ablating Functional Sparse Coding and Domain Sparsity The utility of SAE-FNO stems from four properties: i) operator-based parameterization provides discretization convergence, ii) Fourier parameterization introduces a spectral bias toward smooth global structures, iii) functional sparse coding represents each concept as a function over the domain, and iv) domain sparsity encodes where and how a concept is expressed. We argue that SAE-FNO’s distinct advantages arise from the latter two: ablating concept and domain sparsity prevents the model from learning the structured and stable concepts observed in SAE-FNO (Figure 29).

When is SAE-FNO Advantageous? SAE-FNO excels when data exhibit spatial structure that varies in scale or resolution. Its functional parametrization adapts the receptive field, an attribute absent in fixed-vector SAE-MLPs. This enables efficient scaling to larger domains ($8 \times 8 \rightarrow 32 \times 32$, Figure 4d,h), in addition to the resolution shift and cross-discretization generalization.

Furthermore, SAE-FNO is highly effective when the underlying concepts are sparsely structured in the frequency domain. We demonstrate this on a synthetic time-series dataset comprising 10 single-frequency concepts, generated with a concept sparsity of 3 and a domain sparsity of 2 (i.e., each concept appears locally twice). Under additive Gaussian noise, SAE-FNO outperforms SAE-MLP in recovering the underlying concepts (Table 1). Restricting SAE-FNO’s frequency modes further improves performance. SAE-FNO’s spectral inductive bias naturally filters noise and favors smooth, low-frequency concepts, offering a powerful practical advantage when target concepts are known not to be concentrated in high frequencies (Table 1 last column).

Limitations SAE-FNO introduces two additional hyperparameters: number of modes (controlling smoothness) and domain sparsity (controlling spatial structure). Our ablations (modes: Figures 7, 26, domain sparsity: Figures 8, 13) demonstrate consistent behavior across modes, with expressivity increasing alongside more modes. Finally, we focus our empirical study on vision datasets. While not explored here, these experiments naturally open the door to future applications of SAE-NOs in mechanistic interpretability of neural operator foundation models and scientific machine learning.

Conclusion We introduced sparse autoencoder neural operators (SAE-NOs), a functional framework for sparse representation learning in structured domains. By replacing scalar activations with Fourier-based representations and domain sparsity, SAE-FNO captures not only which concepts are active, but also how and where they are expressed. Across multiple SAE variants and datasets, SAE-FNO learns more structured, stable, and efficient concepts, while naturally supporting spatial and temporal consistency, discretization generalization, and controllable expressivity. Our results highlight the central role of functional parameterization for learning representations.

Acknowledgments and Disclosure of Funding

B.T. acknowledges support from the Canada CIFAR AI Chairs Program. A.S. acknowledges support from Caltech’s Summer Undergraduate Research Fellowship (SURF) and the Mellon Mays Undergraduate Fellowship. A.A. acknowledges support from the Bren Endowed Chair and the AI2050 Senior Fellow program.

References

- [1] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.
- [2] Alireza Makhzani and Brendan Frey. k-sparse autoencoders, 2014.
- [3] Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2023.
- [4] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv:2209.10652*, 2022.
- [5] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2, 2023.
- [6] Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [7] Valérie Costa, Thomas Fel, Ekdeep Singh Lubana, Bahareh Tolooshams, and Demba Ba. From flat to hierarchical: Extracting sparse representations with matching pursuit. In *Advances in Neural Information Processing Systems*, 2025.
- [8] Chris Olah. Distributed representations: Composition & superposition. *Transformer Circuits Thread*, 27, 2023.
- [9] Carolina Zheng, Nicolas Beltran-Velez, Sweta Karlekar, Claudia Shi, Achille Nazaret, Asif Mallik, Amir Feder, and David M Blei. Model directions, not words: Mechanistic topic models using sparse autoencoders. *arXiv preprint arXiv:2507.23220*, 2025.
- [10] Usha Bhalla, Thomas Fel, Can Rager, Sheridan Feucht, Tal Haklay, Daniel Wurgaft, Siddharth Boppana, Matthew Kowal, Vasudev Shyam, Jack Merullo, Atticus Geiger, and Ekdeep Singh Lubana. Do sparse autoencoders capture concept manifolds? 2026.
- [11] Bart Bussmann, Patrick Leask, and Neel Nanda. Batchtopk sparse autoencoders. In *NeurIPS 2024 Workshop on Scientific Methods for Understanding Deep Learning*.
- [12] Senthoran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders. *preprint arXiv:2407.14435*, 2024.
- [13] Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023.
- [14] Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999.
- [15] Lu Lu, Pengzhan Jin, and George Em Karniadakis. Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *arXiv:1910.03193*, 2019.

- [16] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *preprint arXiv:2003.03485*, 2020.
- [17] Zongyi Li, Nikola Borislavov Kovachki, Kamyar Azizzadenesheli, Burigede liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021.
- [18] Kamyar Azizzadenesheli, Nikola Kovachki, Zongyi Li, Miguel Liu-Schiaffini, Jean Kossaifi, and Anima Anandkumar. Neural operators for accelerating scientific simulations and design. *Nature Reviews Physics*, pages 1–9, 2024.
- [19] L. Ghafourpour, Valentin Duruisseaux, B. Tolooshams, Philip H Wong, Costas A Anastassiou, and Anima Anandkumar. Noble–neural operator with biologically-informed latent embeddings to capture experimental variability in biological neuron models. *arXiv:2506.04536*, 2025.
- [20] Zongyi Li, Nikola Kovachki, Chris Choy, Boyi Li, Jean Kossaifi, Shourya Otta, Mohammad Amin Nabian, Maximilian Stadler, Christian Hundt, Kamyar Azizzadenesheli, et al. Geometry-informed neural operator for large-scale 3d pdes. *Advances in Neural Information Processing Systems*, 36, 2024.
- [21] Armeet Singh Jatyani, Jiayun Wang, Zihui Wu, Miguel Liu-Schiaffini, Bahareh Tolooshams, and Anima Anandkumar. Unifying subsampling pattern variations for compressed sensing mri with neural operators. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
- [22] Justin Bunker, Mark Girolami, Hefin Lambley, Andrew M. Stuart, and T. J. Sullivan. Autoencoders in function space. *preprint arXiv:2408.01362*, 2024.
- [23] Usha Bhalla, Alex Oesterling, Claudio Mayrink Verdun, Himabindu Lakkaraju, and Flavio P Calmon. Temporal sparse autoencoders: Leveraging the sequential nature of language for interpretability. *arXiv:2511.05541*, 2025.
- [24] Ekdeep Singh Lubana, Can Rager, Sai Sumedh R Hindupur, Valerie Costa, Greta Tuckute, Oam Patel, Sonia Krishna Murthy, Thomas Fel, Daniel Wurgajt, Eric J Bigelow, et al. Priors in time: Missing inductive biases for language model interpretability. *arXiv:2511.01836*, 2025.
- [25] Matyas Bohacek, Thomas Fel, Maneesh Agrawala, and Ekdeep Singh Lubana. Uncovering conceptual blindspots in generative image models using sparse autoencoders. *arXiv:2506.19708*, 2025.
- [26] Nick Jiang, Xiaoqing Sun, Lisa Dunlap, Lewis Smith, and Neel Nanda. Interpretable embeddings with sparse autoencoders: A data analysis toolkit. *arXiv:2512.10092*, 2025.
- [27] Thomas Fel, Binxu Wang, Michael A. Lepori, Matthew Kowal, Andrew Lee, Randall Balestriero, Sonia Joseph, Ekdeep Singh Lubana, Talia Konkle, Demba E. Ba, and Martin Wattenberg. Into the rabbit hull: From task-relevant concepts in DINO to minkowski geometry. In *The Fourteenth International Conference on Learning Representations*, 2026.
- [28] Patrick Leask, Bart Bussmann, Michael T Pearce, Joseph Isaac Bloom, Curt Tigges, Noura Al Moubayed, Lee Sharkey, and Neel Nanda. Sparse autoencoders do not find canonical units of analysis. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [29] Bruno A Olshausen and David J Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325, 1997.

A Appendix - Experiment Setup

Datasets and Preprocessing We evaluate our SAE-FNO and SAE-MLP (SAE-CNN) across MNIST, CIFAR-10, and ImageNet datasets. For MNIST, the images are inherently 28×28 grayscale and are passed directly into the SAEs without additional preprocessing. For CIFAR-10 and ImageNet, we first extract random patches to match the model’s expected receptive field (e.g., 8×8 for CIFAR, 16×16 or 32×32 for ImageNet images initially resized to 256×256). Additionally, Global Contrast Normalization (GCN) and ZCA Whitening ($\epsilon = 0.1$) are computed and applied after patch extraction.

Architecture and Training Unless specified otherwise, all models are trained with a dictionary size of $p = 1000$ concepts. For MNIST and CIFAR-10 experiments, we enforce concept and domain sparsity via the standard Top- K mechanism. For ImageNet, we evaluate Batch Top- K and Matryoshka SAE variants in addition to Top- K SAE.

All models use the AdamW optimizer with a base learning rate of 10^{-3} . The primary objective is the Mean Squared Error (MSE) reconstruction loss. Throughout training, we apply unit normalization along the channel dimension. After taking the L_2 norm over the spatial dimensions, we scale the weights by $1/\sqrt{\text{ch}}$ (where ch is the number of channels). Additionally, for SAE-FNOs, we scale the encoder and decoder weights of the SAE-FNO by a factor of $\sqrt{\text{modes}}$ for training stability. The learning rate is also scaled accordingly.

2D Discretization Generalization (Upsampling) For spatial image data, we apply bilinear interpolation directly to the test images to scale them to the target resolutions. To evaluate SAE-MLP on these higher-dimensional inputs, its fixed-size dense parameter matrices must be proportionally scaled. We reshape the encoder and decoder weight matrices (as well as the learned spatial encoder bias tensor when applicable) from flat vectors into their original spatial configurations (e.g., mapping from $\mathbb{R}^{d \times 784}$ to $\mathbb{R}^{d \times 28 \times 28}$). These tensors are then bilinearly interpolated to the new target resolution (e.g., $\mathbb{R}^{d \times 56 \times 56}$) and flattened back into dense matrices.

Ablation of Concept and Domain Sparsity To isolate the specific contributions of our sparsity formulation, we compare the full SAE-FNO against two restricted variants (Figure 29): 1) SAE-FNO without domain sparsity. This network retains the concept-level sparsity to select active dictionary atoms, but leaves their corresponding feature map activations dense across the spatial domain (applying only spatial ReLU); and 2) AE-FNO (no sparsity). This is a baseline model where both concept and domain sparsity mechanisms are removed. The projection onto the p dictionary atoms is followed only by a ReLU nonlinearity, allowing all concepts to activate simultaneously across the entire domain.

MNIST Translated Digits To evaluate the stability of learned concepts under continuous translation, we construct a custom moving MNIST dataset. Standard MNIST digits are placed onto an expanded 48×48 spatial canvas and linearly translated across the domain over a fixed temporal window ($T = 20$ frames). Crucially, both the SAE-MLP and SAE-FNO process each frame independently: the Top- K sparsity mechanism is computed and applied independently to each individual frame without any constraints enforcing temporal consistency. We then encode the translating digit frame-by-frame and identify the top active concepts at each time step, ranked by their L_2 activation magnitude (Figure 2).

B Appendix - Metrics

Hoyer Score For a flattened representation $\mathbf{v}_c \in \mathbb{R}^d$ of the c -th concept, the Hoyer score is given by:

$$\text{Hoyer}(\mathbf{v}_c) = \frac{\sqrt{d} - \frac{\|\mathbf{v}_c\|_1}{\|\mathbf{v}_c\|_2}}{\sqrt{d} - 1} \quad (2)$$

Identity Consistency To quantify whether SAEs learn a stable and consistent set of features regardless of the specific sparsity constraint, we introduce the Identity Consistency metric. This measures the fraction of learned concepts in an anchor model ($k_{\text{ref}} = 50$ in Figure 9) that are consistently recovered in models trained at more relaxed sparsity levels ($k > 50$).

To ensure the metric evaluates meaningful representations, we restrict the matching process to active concepts. A concept is considered active if it generates a non-zero activation for at least 1% of the

evaluation images. Let $\mathcal{D}_{\text{ref}}$ be the set of active, unit-normalized spatial dictionary atoms from the anchor model, and $\mathcal{D}_{(k)}$ be the active, unit-normalized atoms from the comparison model.

To accurately reflect the capacity of each model, we tailor the similarity metric to their respective inductive biases. Because SAE-MLPs use scalar activations, their dictionary atoms are fixed to their specific position. A structural feature learned at one spatial location cannot be shifted and applied to another. Therefore, we evaluate SAE-MLP using cosine similarity:

$$\text{sim}_{\text{MLP}}(d_i, d_j) = d_i^\top d_j \quad (3)$$

Conversely, SAE-FNO concepts are parameterized functionally in the Fourier domain, allowing the model to represent the same structural feature translated across the spatial domain. This leads us to compute similarity via maximum spatial cross-correlation:

$$\text{sim}_{\text{FNO}}(d_i, d_j) = \max_{\Delta x, \Delta y} (\mathcal{F}^{-1}(\mathcal{F}(d_i) \odot \overline{\mathcal{F}(d_j)})) \quad (4)$$

where $\odot$ denotes element-wise multiplication.

For every concept in the anchor model, we independently find the best-matching concept in the comparison model. For each $d_i \in \mathcal{D}_{\text{ref}}$, the maximum similarity score is defined as

$$s(d_i) = \max_{d_j \in \mathcal{D}_{(k)}} \text{sim}(d_i, d_j) \quad (5)$$

From this, we derive the identity consistency metric, defined as the proportion of anchor concepts successfully recovered above a similarity threshold τ (typically $\tau \in \{0.5, 0.7\}$) [28]:

$$\text{Identity Consistency} = \frac{1}{|\mathcal{D}_{\text{ref}}|} \sum_i \mathbf{1}(s(d_i) > \tau) \quad (6)$$

Concept Utilization and Expressivity To evaluate representation efficiency, we quantify how effectively the architectures use their dictionary capacity and how accurately they reconstruct the test population distribution (Figure 4 and 23).

Let $z_p^{(i)}$ be the feature map activations of concept p for image i after enforcing sparsity. For a feature map with spatial size S (where $S = 1$ for scalar SAE-MLPs and $S = H \times W$ for SAE-FNOs), the normalized concept utilization $L_0(p)$ across a validation set of N images is defined as the total count of non-zero spatial activations relative to the total possible activations:

$$L_0(p) = \frac{1}{N \cdot S} \sum_{i=1}^N \|z_p^{(i)}\|_0 \quad (7)$$

This normalization naturally bounds $L_0(p) \in [0, 1]$.

Additionally, we measure the proportion of data variance explained by the model’s reconstructions via goodness-of-fit (R^2), defined as:

$$R^2 = 1 - \frac{\text{MSE}}{\text{Var}(X)} \quad (8)$$

where X represents the ground-truth images. We analyze R^2 against concept sparsity level and the number of globally active concepts (Figure 4). A concept p is considered as "globally active" if it produces at least one non-zero spatial activation ($\|z_p^{(i)}\|_0 > 0$) on $\geq 1\%$ of the validation set. Models that achieve a higher R^2 using fewer globally active concepts show better disentanglement and representational efficiency.

C Appendix - Additional Figures and Experiments

C.1 Temporal Concept Consistency (Translating MNIST)

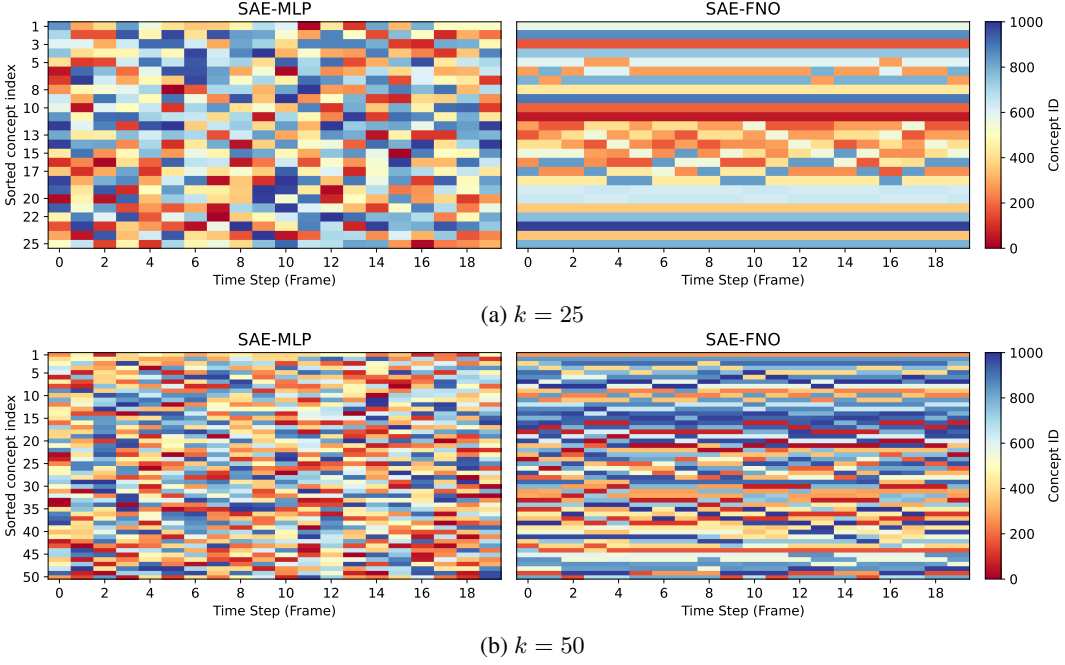


Figure 5: **Top concept activations over time for higher concept sparsity levels.** Extended visualizations of Figure 2b for (a) $k = 25$ and (b) $k = 50$. Each cell’s color represents the concept ID active at a given time step. Consistent with lower sparsity levels, SAE-FNO maintains stable concept identities over time (horizontal bands), using domain sparsity to handle spatial translations. In contrast, SAE-MLP constantly activates new, distinct concepts to represent the moving digit.

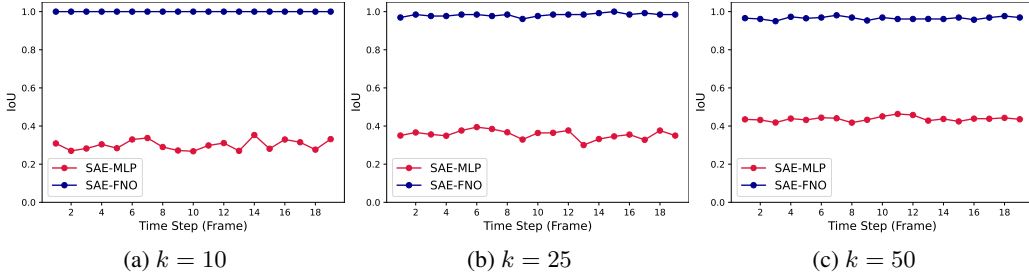


Figure 6: **SAE-FNO concept stability across sparsity levels.** Intersection over Union (IoU) of the active concept sets between consecutive frames in translating MNIST digits (see Figure 2abc). At each time step t , we compute the IoU between the top k concepts (ranked by their spatial L_2 norm) activated in frame t and those activated in frame $t + 1$, averaged across batch. Higher IoU indicates stronger concept stability and consistency, meaning the model tracks the moving digit using a consistent subset of concepts. While SAE-MLP (red) rapidly swaps its active concepts between frames, SAE-FNO (blue) maintains a highly stable set of active concepts (IoU ≈ 1).

C.2 Synthetic Recovery and Identifiability

Table 1: **Useful inductive bias of SAE-FNO in recovering frequency-oriented concepts.** Synthetic experiment recovering 10 single-frequency concepts ($k = 3$, domain sparsity 2). Values represent cross-correlation recovery scores, reported as (mean, standard deviation) across 5 runs.

	Init	SAE-MLP	SAE-FNO ($m = \text{full}$)	SAE-FNO ($m = 151$)
Clean	(0.329, 0.210)	(0.868, 0.303)	(0.863, 0.324)	(0.882, 0.612)
Noisy	(0.329, 0.210)	(0.798, 0.534)	(0.853, 0.627)	(0.890, 0.658)

C.3 Concept Structural Stability

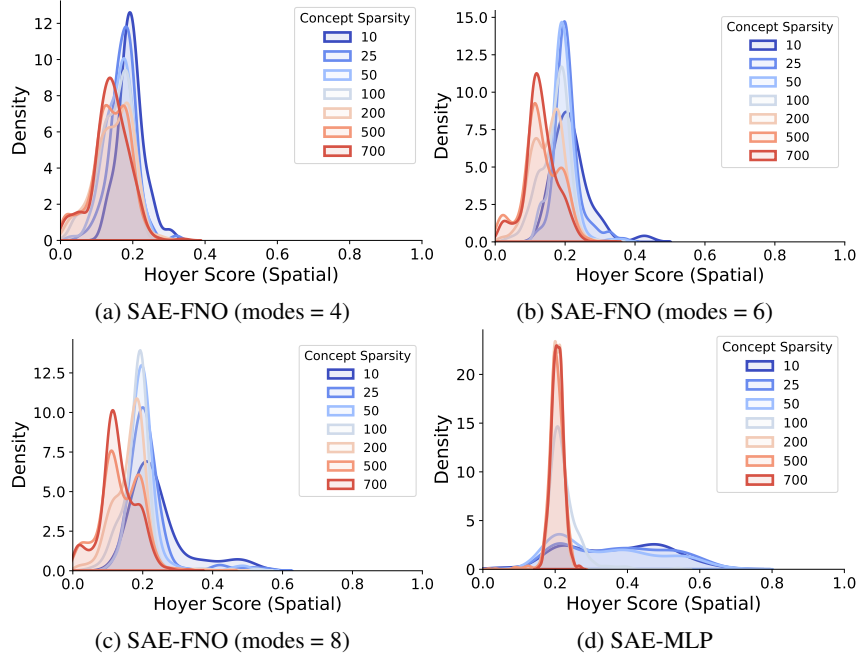


Figure 7: **Stability of concept characteristics across concept sparsity levels (CIFAR).** CIFAR Hoyer scores for SAE-MLP and SAE-FNO ($ss = 0.005$, 3-sparse)

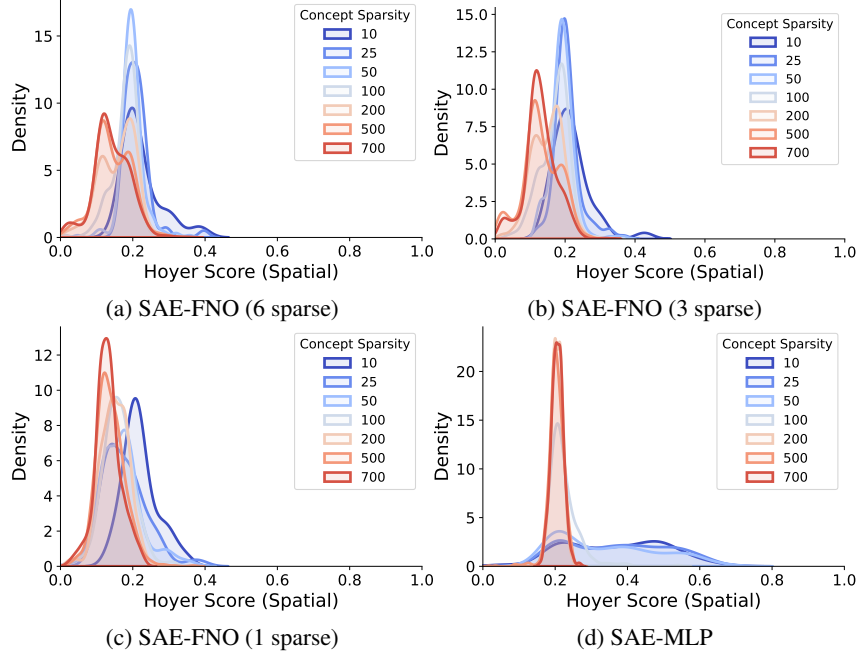


Figure 8: **Stability of concept characteristics across domain sparsity levels (CIFAR).** Hoyer scores for SAE-MLP and SAE-FNO (modes = 6).

Table 2: **CNN structural stability across kernel sizes.** Variance of the mean, median, and mode of the Hoyer-score distribution ($\times 10^{-4}$), together with the best R^2 across k , on CIFAR 32×32 (domain sparsity of 5 pixels). Lower variance indicates concept structure that is more stable as k varies.

Model	Var(mean)	Var(median)	Var(mode)	R^2 (max)
SAE-FNO (24 modes)	6.40	1.04	0.43	0.795
SAE-FNO (full, 32 modes)	1.61	0.03	0.03	0.945
SAE-CNN (5×5)	7.11	6.66	4.04	0.602
SAE-CNN (10×10)	18.9	35.5	58.8	0.678
SAE-CNN (15×15)	26.3	69.7	91.5	0.710
SAE-CNN (20×20)	6.55	71.0	64.6	0.845

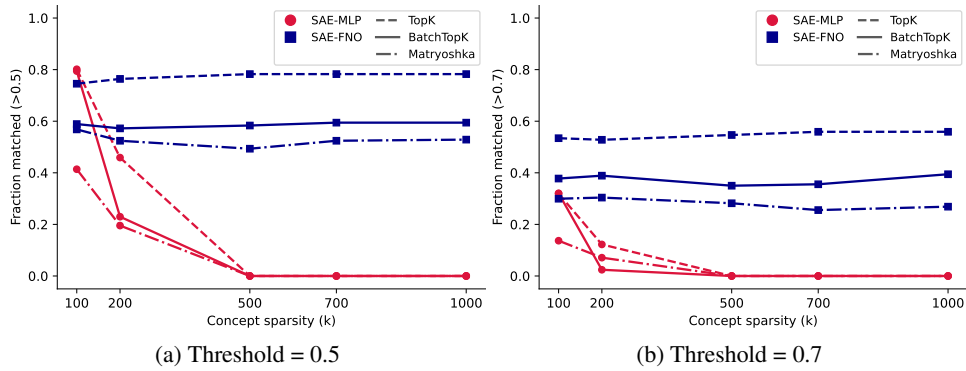


Figure 9: **Identity consistency.** New metric for consistency of concepts across k . Fraction of matched features across runs of increasing concept sparsity using $k=50$ as anchor.

C.4 Concept Correlations

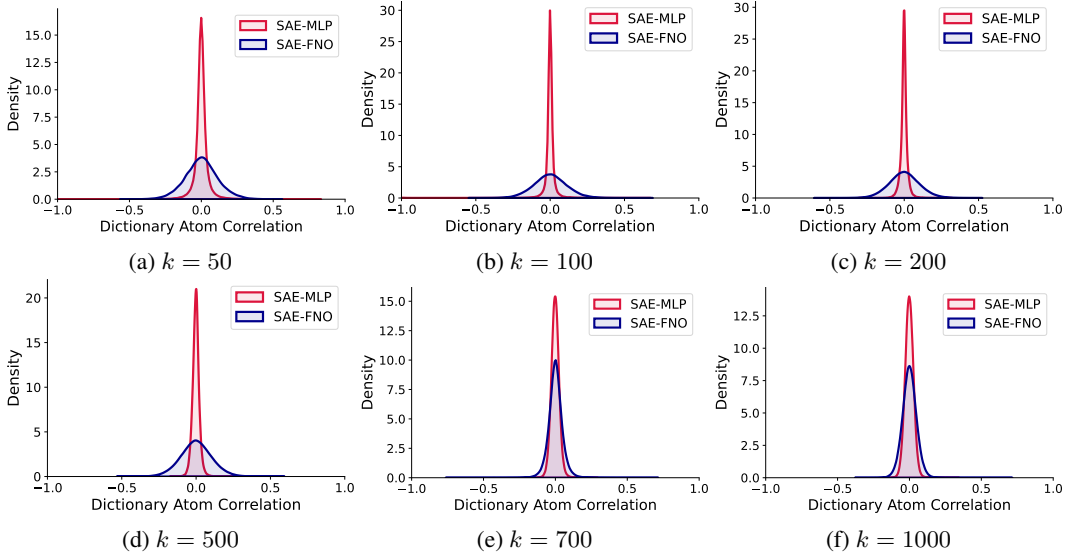


Figure 10: **SAE-FNO learns a broader range of correlated concepts in MNIST.** We compare the distribution of pairwise cosine similarities (correlations) between learned dictionary atoms for SAE-FNO (blue) and SAE-MLP (red) across varying concept sparsity levels (k). SAE-FNOs display a broader distribution with heavier tails, indicating a better ability to recover non-orthogonal, correlated concepts.

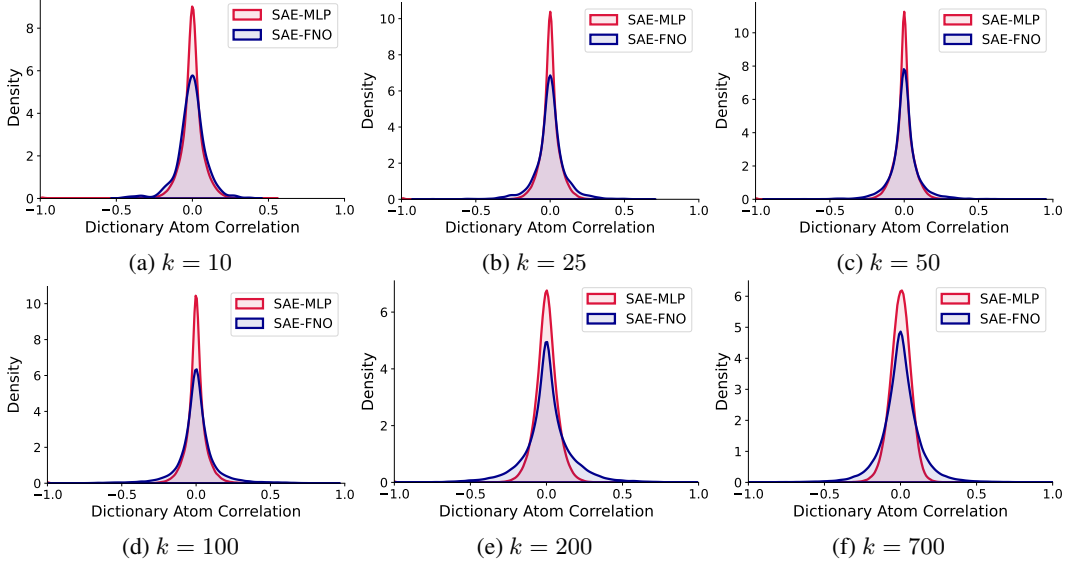
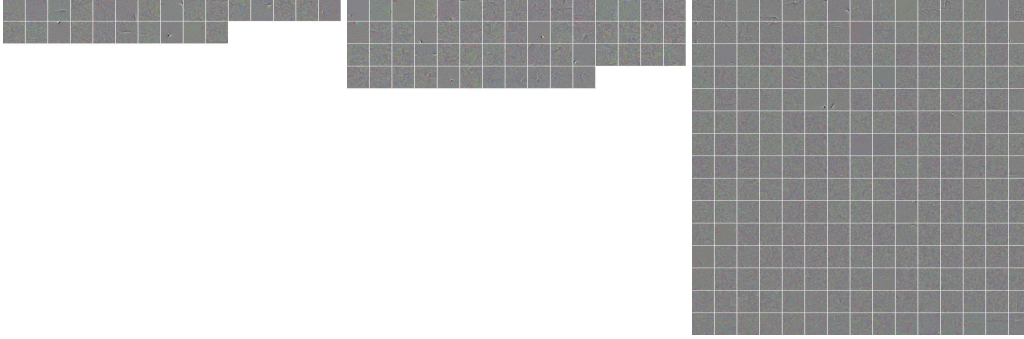
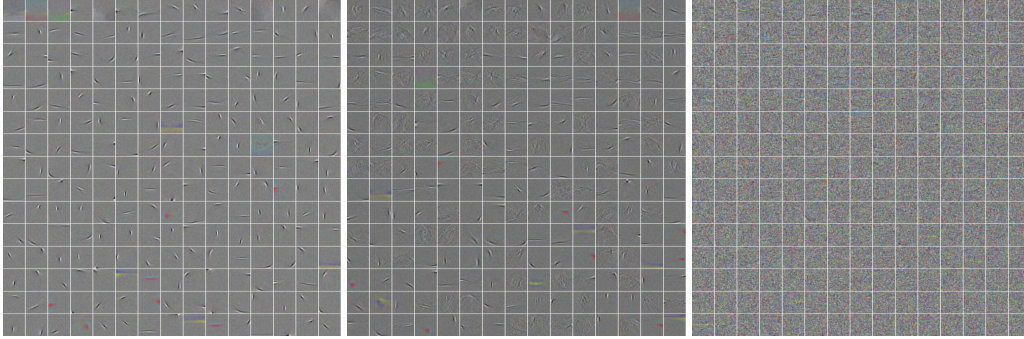


Figure 11: **SAE-FNO learns a broader range of correlated concepts in CIFAR.** Similar to the MNIST results, we compare the distribution of pairwise cosine similarities between learned dictionary atoms for SAE-FNO (blue) and SAE-MLP (red) across varying concept sparsity levels (k). The SAE-FNO models use 6 Fourier modes and a spatial sparsity of 0.02. SAE-FNO consistently displays a broader correlation distribution with heavier tails compared to SAE-MLP.

C.5 Concept Visualization

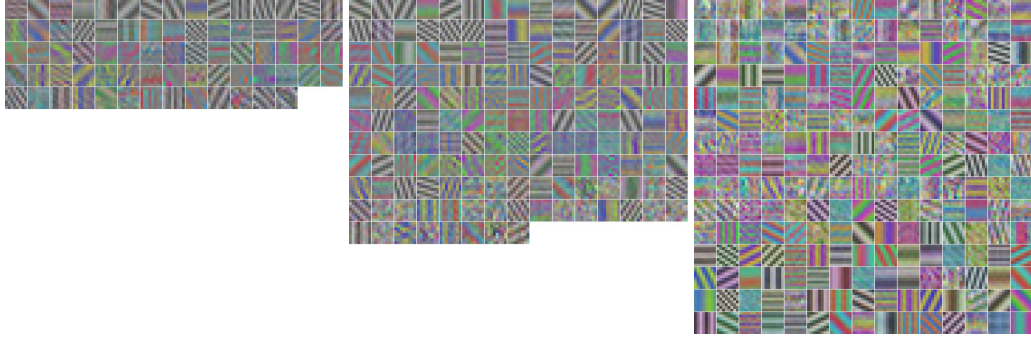


(a) SAE-FNO.

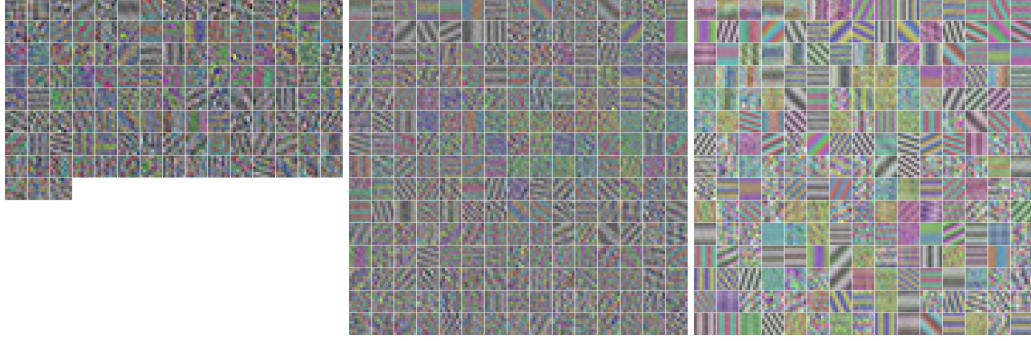


(b) SAE-MLP.

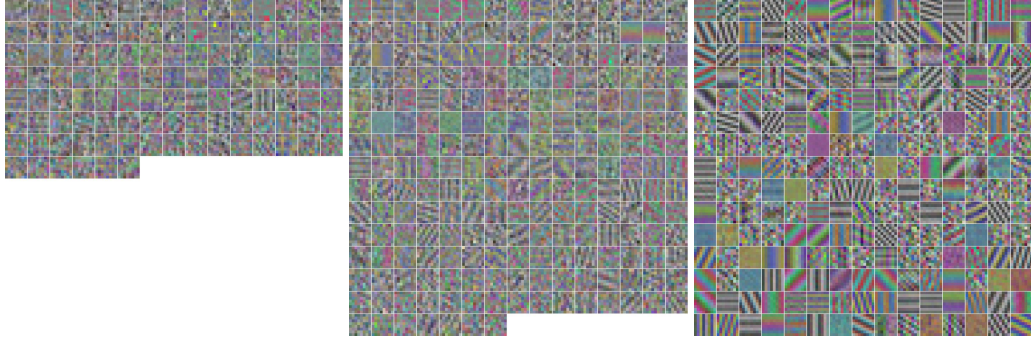
Figure 12: **Learned concepts across the dataset for CIFAR 32×32 .** SAEs are trained on CIFAR-10 image patches for SAE-MLP-TopK vs. SAE-FNO-TopK. Each square represents one concept. All models learn $p = 1000$ concepts; we only visualize the most frequently used ones, where fewer displayed concepts indicate more efficient usage (see R^2 vs. used atoms in Figure 4). From left to right, concept sparsity is $k = 25, 50, 200$. (a) SAE-FNO exhibits stable concept characteristics across sparsity levels, enabled by domain sparsity and adaptability to input size. As sparsity decreases, more concepts are activated. SAE-FNO learns localized frequency structures such as edge detectors, consistent with natural image statistics [29]. (b) SAE-MLP concepts vary significantly with sparsity, losing structure and failing to capture locality. SAE-FNO requires fewer effective concepts than SAE-MLP, indicating superior expressivity for structured, large-domain inputs.



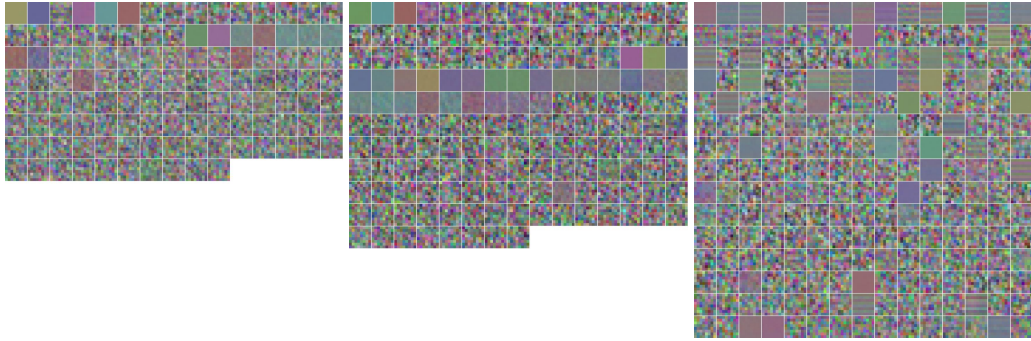
(a) 1-sparse.



(b) 3-sparse.



(c) 6-sparse.



(d) 32-sparse.

Figure 13: **Learned concepts across the dataset for CIFAR 8×8 patches for SAE-FNO (modes = 6) as spatial sparsity changes.** From left to right, concept sparsity is $k = 25, 50, 200$. Same visualization setup as Figure 12. Corresponding concept utilization and efficiency plots are shown in Figure 24a.

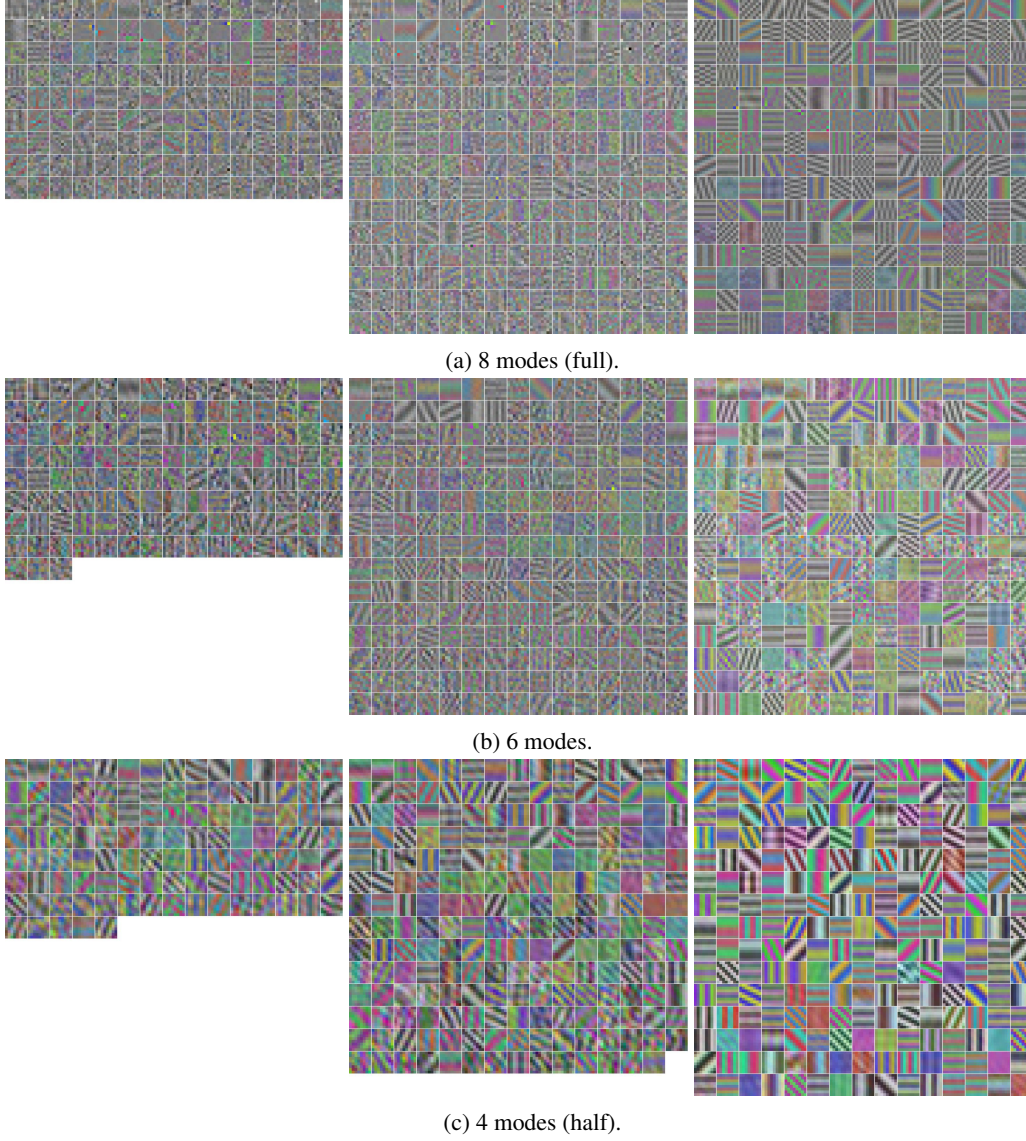


Figure 14: **Learned concepts across the dataset for CIFAR 8×8 patches for SAE-FNO (spatial sparsity = 0.05) as the number of Fourier modes changes.** From left to right, concept sparsity is $k = 25, 50, 200$. Same visualization setup as Figure 12. Corresponding concept utilization and efficiency plots are shown in Figure 24b.

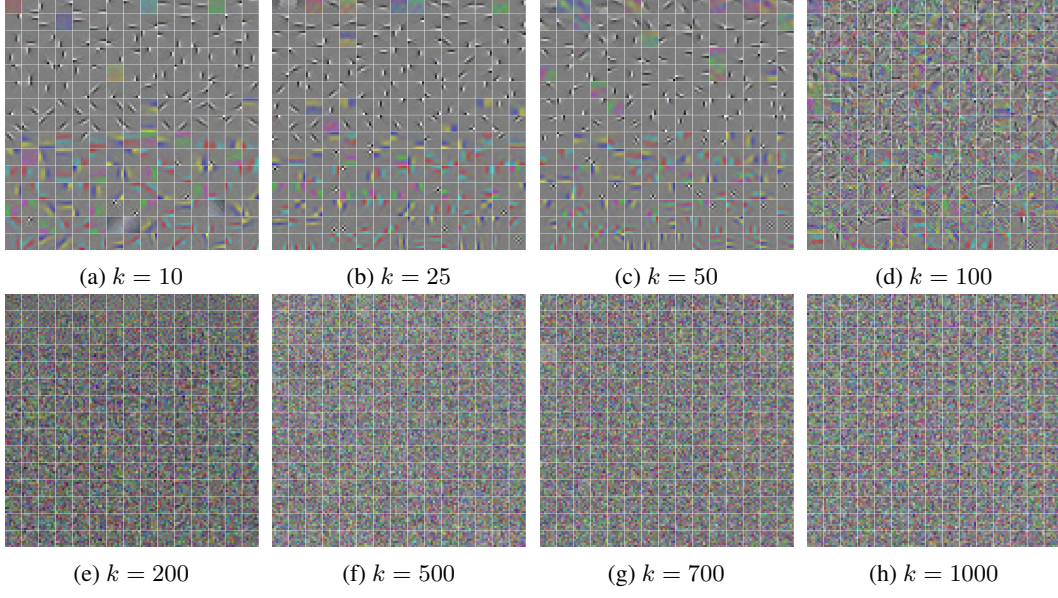


Figure 15: **Additional visualizations.** SAE-MLP learned concepts on CIFAR 8×8 patches.

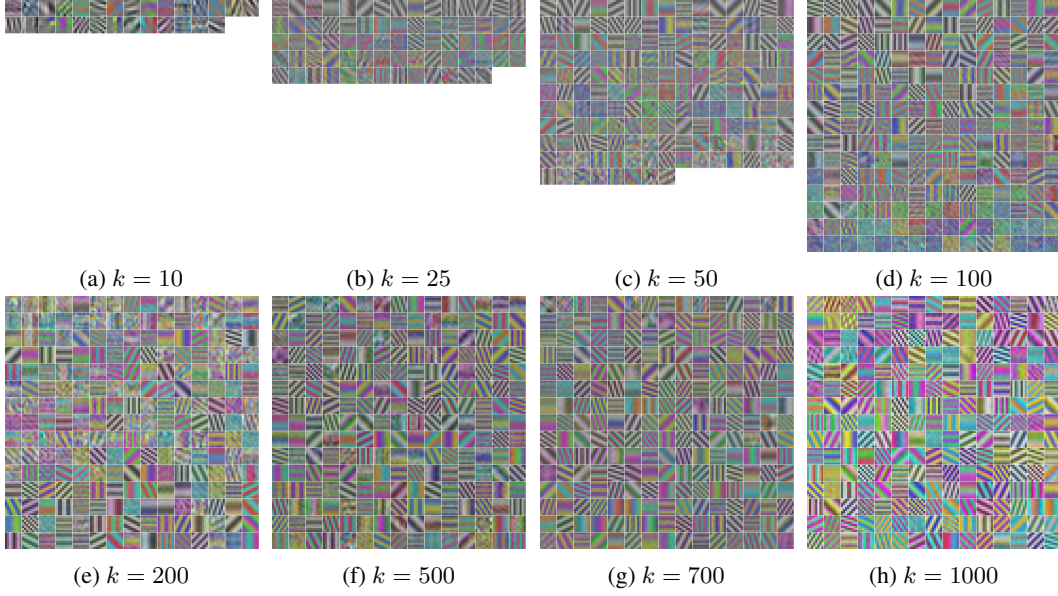


Figure 16: **Additional visualizations.** SAE-FNO (modes = 6, spatially 1-sparse) learned concepts on CIFAR 8×8 patches.

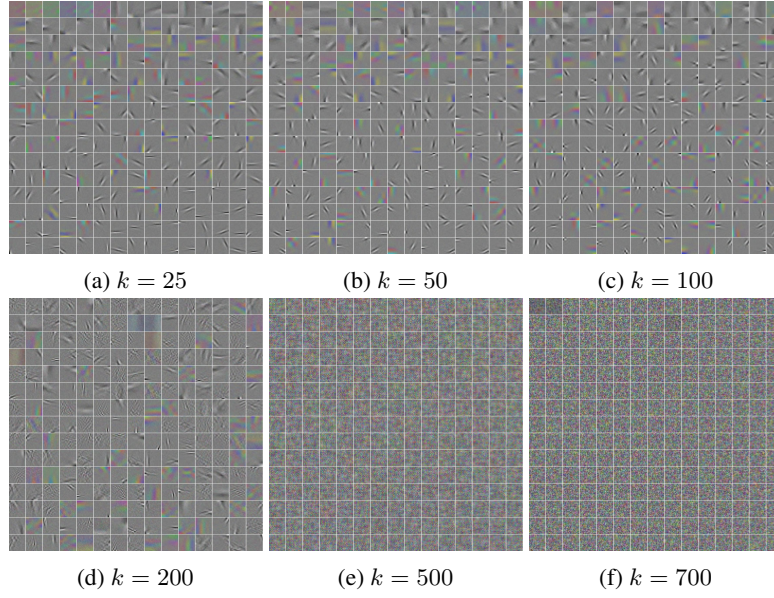


Figure 17: **Additional visualizations.** SAE-MLP-TopK learned concepts on ImageNet patches.

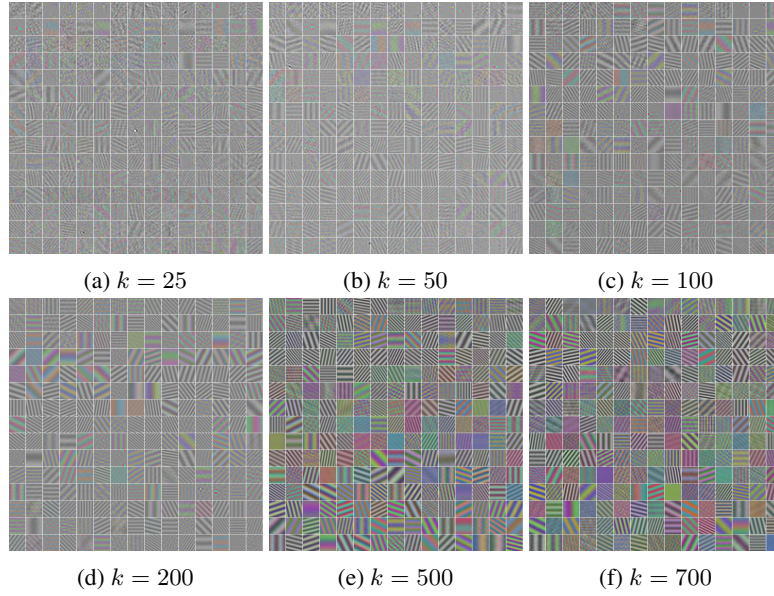


Figure 18: **Additional visualizations.** SAE-FNO-TopK (modes = 12, spatially 5-sparse) learned concepts on ImageNet patches.

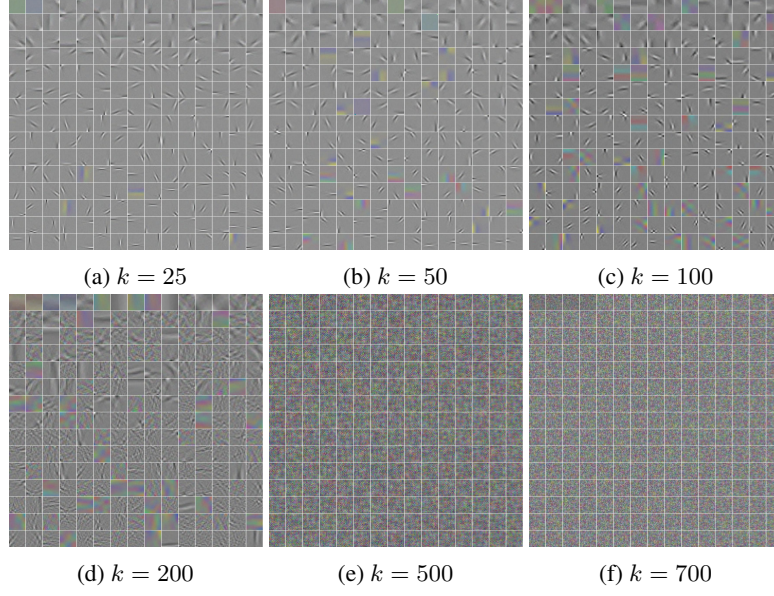


Figure 19: **Additional visualizations.** SAE-MLP-BatchTopK learned concepts on ImageNet patches.

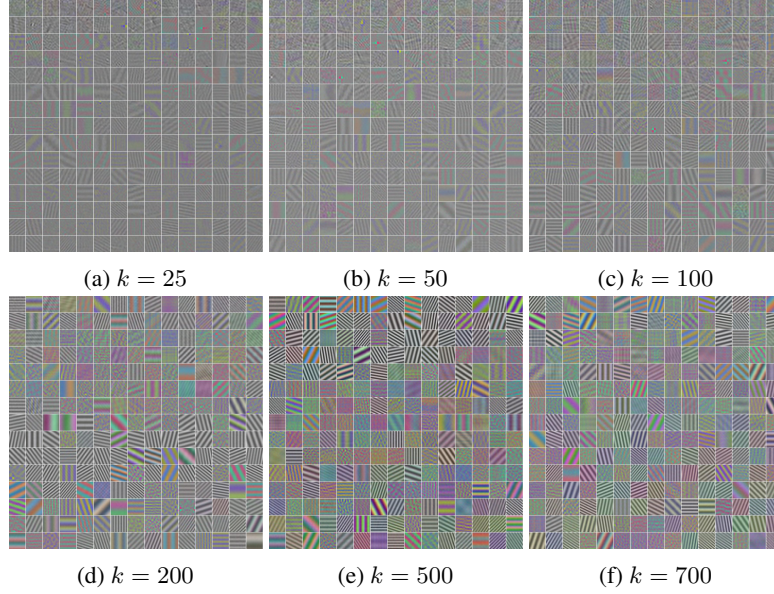


Figure 20: **Additional visualizations.** SAE-FNO-BatchTopK (modes = 12, spatially 5-sparse) learned concepts on ImageNet patches.

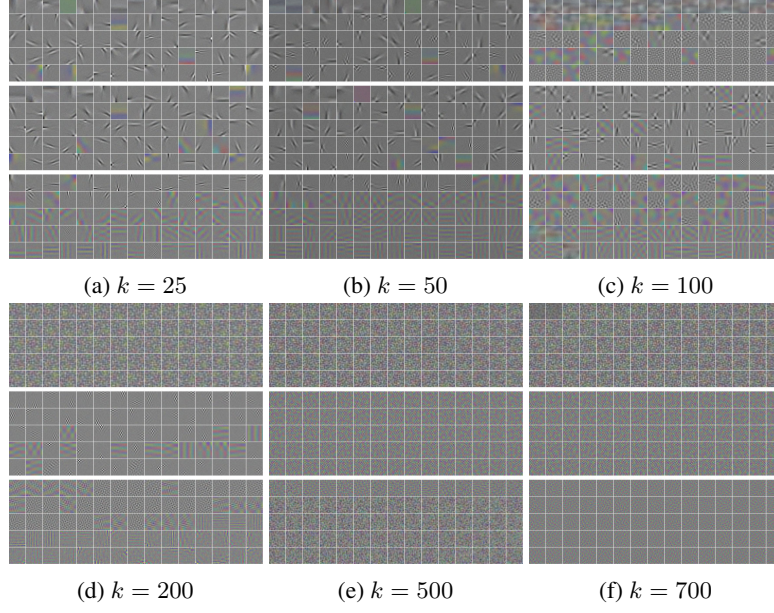


Figure 21: **Additional visualizations.** Matryoshka-SAE-MLP learned concepts on ImageNet patches.

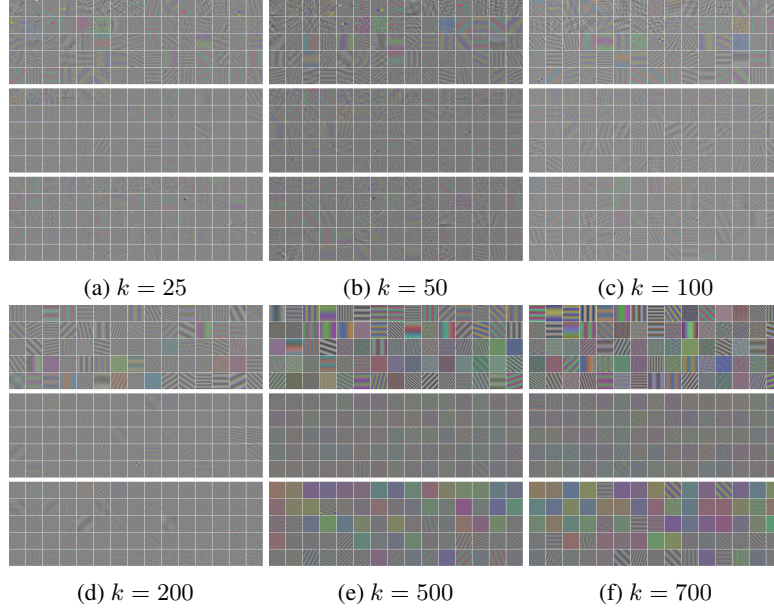


Figure 22: **Additional visualizations.** Matryoshka-SAE-FNO (modes = 12, spatially 5-sparse) learned concepts on ImageNet patches.

C.6 Concept Utilization and Expressivity

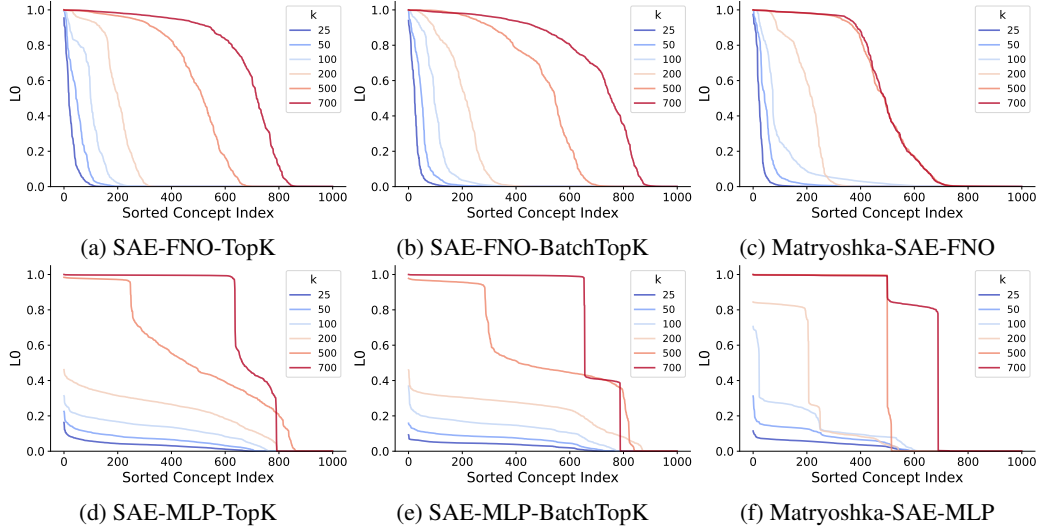


Figure 23: **Concept utilization and efficiency on ImageNet patches (16×16).** Y-axis quantifies the normalized l0-norm of codes corresponding to each concept used across the dataset. Reaching zero corresponds to the effective total number of concepts used to explain the dataset. Across TopK, BatchTopK, and Matryoshka variants, SAE-FNO requires a much smaller active dictionary to represent the data compared to SAE-MLP.

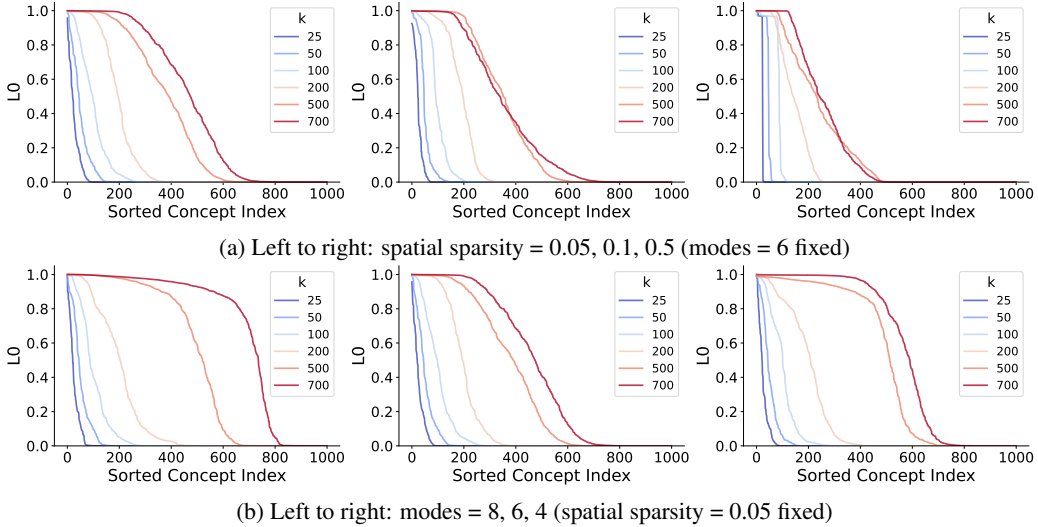


Figure 24: **Concept utilization and efficiency on CIFAR 8×8 patches for SAE-FNO-TopK across changing a) spatial sparsity and b) number of Fourier modes.** Same visualization setup as Figure 23. Visualization of learned concepts is shown in Figure 13 (change spatial sparsity) and Figure 14 (change modes).

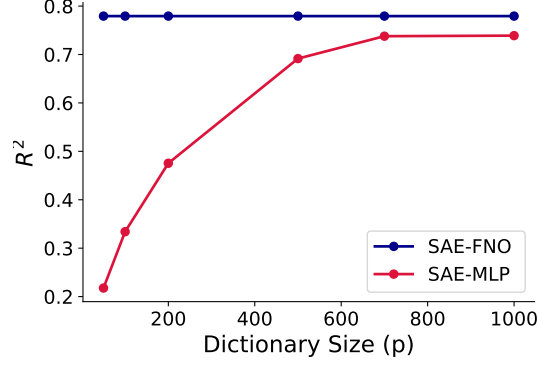


Figure 25: **Expressivity of SAE as a function of dictionary size (p).** Evaluating R^2 as the total dictionary size p increases (at fixed $k = 50$ on ImageNet 16×16 patches). SAE-FNO with 12 modes is used.

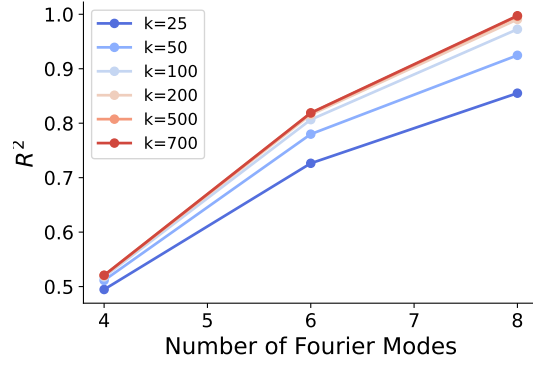


Figure 26: **Expressivity of SAE-FNO as a function of Fourier modes.** The R^2 of SAE-FNO on CIFAR patches steadily increases with the number of Fourier modes. The frequency bandwidth directly controls the resolution and detail of the learned concepts.

C.7 Discretization Generalization

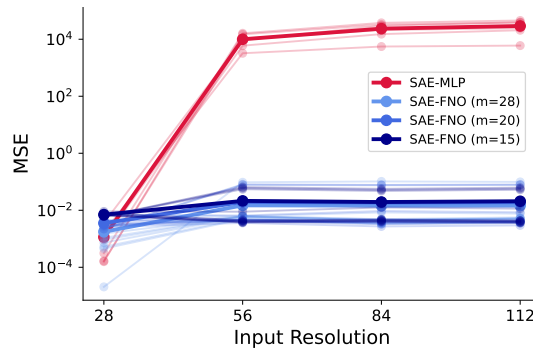


Figure 27: **SAE-FNO vs. SAE as a function of input discretization.** SAE-FNO can successfully infer the underlying representations that allow it to reconstruct data at varying resolution beyond the original trained resolution of 28. The figure shows MSE on the reconstruction. Transparent lines are individual runs, solid lines show their average.

C.8 Ablation

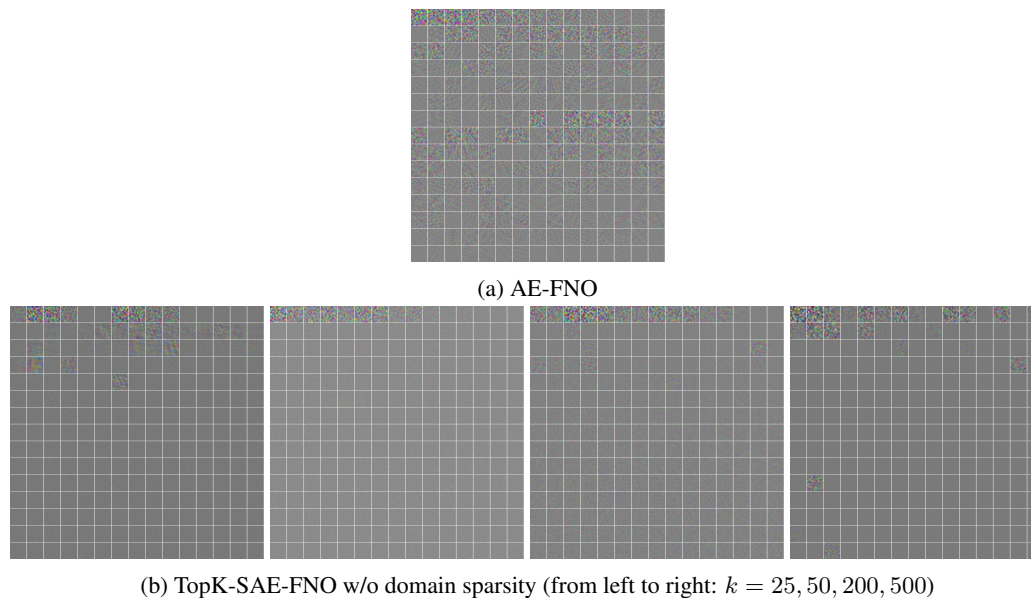


Figure 29: **Ablation of domain sparsity.** Concepts learned by a standard AE-FNO (a) or an SAE-FNO lacking domain sparsity (b) fail to produce localized, structured features. This confirms that the observed gains in mechanistic interpretability are specifically from the combination of functional sparse coding and domain sparsity, rather than the FNO architecture alone.

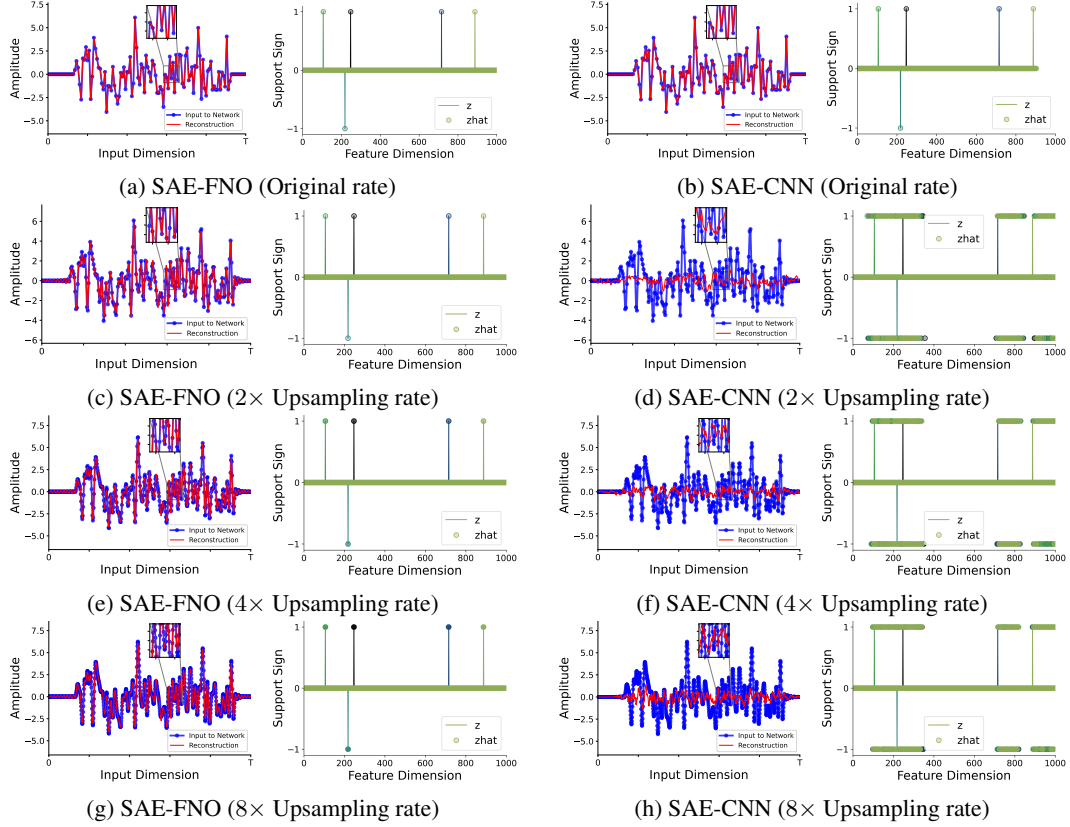


Figure 28: Discretization Generalization of SAE-FNO. The underlying data structure has five convolutional concepts along with a 5-sparse code (i.e., 1-sparse representation for each concept). Each subplot shows (left) the input data and its reconstruction, and (right) the support sign (active sparse codes). We evaluate the ability of SAE-FNO (left column) and SAE-CNN (right column) to infer sparse representations and reconstruct data across varying discretization levels. Each row represents a different upsampling rate ($1\times$ to $8\times$ applied to the input signal). For each panel, the left plot shows the spatial-domain signal reconstruction (red) against the input to the network (blue), and the right plot shows the inference of 1-sparse code supports across 5 kernels. (a-b) Original resolution ($1\times$): Baseline performance at training resolution. Both models accurately reconstruct the signal and recover the correct support. (c-h) Higher upsampling rates ($2\times$, $4\times$, $8\times$): As spatial resolution increases, SAE-FNO maintains accurate reconstruction and code support inference. In contrast, SAE-CNN fails to generalize; its support inference collapses, leading to poor reconstruction. SAE-FNO’s function space parameterization enables this generalization across discretizations.