Engram Memory Network: Brain-Inspired Prototype Explanations

Hyunjun Kim
Department of Electrical and Computer Engineering
Seoul National University
Seoul, South Korea
doctor3390@snu.ac.kr

Myoung Hoon Ha*
Center for Neuroscience-inspired AI
KAIST
Daejeon, South Korea
mh.ha.soar@gmail.com

***Abstract*—Artificial intelligence systems are increasingly being deployed in high-stakes decision-making scenarios. In response, prototype-based models in explainable AI have attracted growing attention for their ability to retrieve visually similar examples as faithful explanations. Although these models achieve competitive classification accuracy, they often exhibit degraded explanation quality across datasets with varying characteristics, thereby limiting their practical applicability. To address this limitation, we draw on insights from cognitive neuroscience, particularly the human memory system, which encodes input-relevant information into representations that support recognition. In this paper, we propose an Engram Memory Network (EMN), a model that emulates the ventral visual stream and inferior temporal cortex in the human brain. Central to this model is a Hopfield-inspired memory module that iteratively retrieves representative prototypes. These memory-refined latents are used for both reconstruction and classification, encouraging the selected prototypes to capture input-relevant information across diverse datasets. To address the lack of standardized evaluation for prototype explanations, we assess three complementary aspects: perceptual similarity (LPIPS and DISTS) for perceived resemblance, subclass alignment, measuring whether prototypes match the input's fine-grained category beyond supervised labels, and information preservation (NMI), measuring how much input information is captured by selected prototypes. Experiments show that EMN improves visual similarity by 17.06%, subclass alignment by 104.57%, and NMI by 42.42% on average across CIFAR-10/100, MNIST, HAM10000, and CUB-200, while achieving the highest average classification accuracy (87.62%) among all methods. These findings underscore the effectiveness of neuroscience-inspired explanation mechanisms in advancing the interpretability and reliability of prototype-based models.**

I. Introduction

Prototype-based explanation models [1], [2] have emerged as a central approach for explainable AI (XAI; [3]). They justify model predictions by retrieving examples that resemble the input. As deep learning extends beyond benchmark classification tasks to high-stakes domains such as medical diagnosis, autonomous driving, and financial decision-making [4], the demand for accurate and interpretable models has grown. In these applications, prototype-based methods hold particular promise, as they aim to provide explanations that are not only faithful but also intuitive and understandable to human users. While early methods suffered from limited predictive performance [5], [6], recent advances in model architecture and training strategies have substantially improved predictive accuracy [7], [8]. However, consistently achieving high-quality, human-understandable explanations across diverse datasets remains challenging [9].

The explanatory effectiveness of these models varies substantially across datasets. For instance, ProtoVAE [10] reconstructs prototypes from latent vectors and performs well on simple datasets such as MNIST [11], but often produces blurry, ambiguous outputs on more complex datasets such as CIFAR-10 [12]. In contrast, patch-based methods such as ProtoPNet [1] and ProtoViT [8] improve explainability by identifying localized regions of similarity between input and prototypes. These approaches are practical on datasets like CUB-200 [13], where object structures are well-defined and redundant across samples. However, when patch-level spatial correspondence between the input and prototypes is weak, as in CIFAR-10 [12], the selected prototypes may not align clearly with the input, ultimately undermining explanatory validity (See Figure 1).

In addition, metrics for evaluating whether prototypes are genuinely relevant to inputs remain underexplored [9]. We propose three complementary metrics: (i) visual similarity, measured by Learned Perceptual Image Patch Similarity (LPIPS [14]) and Deep Image Structure and Texture Similarity (DISTS [15]), for human-perceived resemblance; (ii) subclass alignment for fine-grained relevance, evaluated by training on superclass labels and measuring whether prototypes match withheld subclasses—critical in applications like medical diagnosis where within-class distinctions (e.g., cancer subtypes) determine explanation validity; and (iii) information preservation, measured by Normalized Mutual Information (NMI [16]) between input images and their selected prototypes, measuring how much input information prototypes capture.

To address the dataset-dependency problem, we draw on insights from the human visual system, where memory representations are believed to encode input-relevant information for recognition [17]. Accordingly, memories that contribute most to such representations may serve as valid explanations; this motivates our use of prototypes.

In this paper, we propose the Engram Memory Network (EMN), which integrates classification, reconstruction, and explanation. Specifically, EMN comprises three components:

*Corresponding author

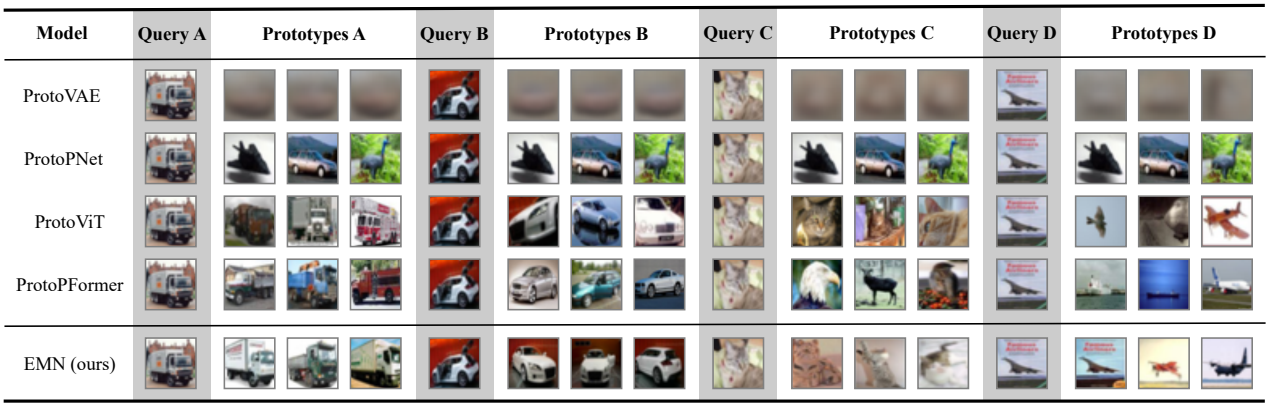

Fig. 1. Queries and selected prototypes by baselines (ProtoVAE [10], ProtoPNet [1], ProtoViT [8], ProtoPFormer [7]) and EMN (ours). Each row shows a model's top-3 retrieved prototypes for four representative queries (A–D); each Query column shows the input image and the three columns on the right shows its retrieved prototypes. In these examples, EMN retrieves prototypes that appear visually clearer and more class-consistent, consistent with the improved explanation metrics reported in Sec. IV.

(i) an encoder that extracts visual features; (ii) an Engram Memory Module (EMM) that performs memory-based association via a modified update rule inspired by a continuous modern Hopfield network [18]; and (iii) a decoder that reconstructs the input. Visual and categorical latent vectors are iteratively refined through interaction with prototype memories. After refinement of the latent vectors, the model predicts the label, reconstructs the image, and selects the top-K activated prototypes for explanation. Furthermore, we introduce regularization that promotes prototype diversity and mitigates over-reliance on a narrow subset of prototypes. We empirically evaluate EMN on diverse benchmarks including CIFAR-10/100 [12], MNIST [11], HAM10000 [19], and CUB-200 [13]. On average across datasets, EMN improves visual similarity (LPIPS [14] and DISTS [15]) by 17.06% over ProtoPFormer [7] and information preservation (NMI [16]) by 42.42% over ProtoViT [8]. EMN achieves 104.57% higher subclass alignment over ProtoPFormer [7], demonstrating its ability to capture fine-grained semantic structure. For classification, EMN achieves the highest average accuracy (87.62%) among all compared methods, notably achieving the best performance on HAM10000 [19] (88.51%). Our further studies in Sec. IV-E and IV-F reveal an explicit accuracy–explanation trade-off: relaxing the separation loss and/or increasing the latent dimensionality improves accuracy (e.g., 97.37% on CIFAR-10 [12], surpassing most baselines) while moderately reducing explanation quality, which nonetheless remains superior to baseline; we adopt a default setting that prioritizes explanation quality over classification accuracy in this paper. These results suggest that EMN produces interpretable explanations grounded in memory-based association, achieving consistent performance across diverse datasets [1].

Our contributions are threefold:

- *Brain-Inspired Architecture*: We present a model that structurally reflects biological memory recall, leveraging continuous modern Hopfield networks [18] to achieve robust explainability and maintained accuracy across diverse datasets with varying characteristics.
- *Novel Prototype Evaluation Protocol*: We propose a comprehensive evaluation protocol that assesses visual similarity (LPIPS [14] and DISTS [15]), subclass alignment, and NMI [16].
- *Cross-dataset Explanatory Robustness*: EMN yields consistently high-quality explanations across diverse datasets, at the cost of a modest decrease in classification accuracy.

[1]Code is available at https://github.com/attractorlab/engram-prototype.

II. Related Work

Post-hoc explanation methods [20], [21] are widely used in XAI. Among them, attribution-based approaches [20], [21] generate explanations after prediction by highlighting influential input regions most associated with the output using saliency maps [20], attention analysis [22], and gradient-based attributions [21]. However, these explanations are external to the model's decision process, often relying on local cues that may not faithfully reflect the model's internal decision-making process or provide sufficient guidance to users.

In contrast, self-explainable models [23] embed interpretability mechanisms directly into the model, generating explanations jointly with predictions. Prototype-based models generate explanations by associating each input with prototypical examples, providing case-based justifications grounded in visual evidence. Early CNN-based methods such as ProtoPNet [1], TesNet [24], and Deformable ProtoPNet [25] rely on patch-to-prototype similarity matching on feature maps, yielding localized explanations. Transformer-based variants like ProtoViT [8] and ProtoPFormer [7] improve classification accuracy and explanation quality, while ProtoVAE [10] learns class-specific prototypes that can be decoded back into input space for visual verification, and SENN [23] decomposes predictions into interpretable concept activations and their corresponding relevance scores.

Despite these advances, our experiments indicate that prior prototype-based methods exhibit dataset-specific limitations. Patch-based models struggle when local similarity between inputs and prototypes is weak, as in CIFAR-10 [12], often yielding poorly aligned explanations (See Figure 1). Latent-

prototype methods like ProtoVAE [10] mitigate this constraint but produce blurry reconstructions, reducing explainability in complex scenarios.

Overall, prior prototype-based methods differ in how they match inputs to prototypes (e.g., patch-level correspondence or latent reconstruction), yet their explanatory effectiveness can vary substantially with dataset characteristics, particularly when spatial correspondence is weak or when reconstructions become ambiguous.

Motivated by these limitations, we propose the EMN. Rather than relying solely on similarity-based scoring, EMN grounds prototype selection in memory-refined latent representations that jointly support reconstruction and classification, and selects prototypes with the highest contribution to these latents as explanations. By separating visual and categorical latents, EMN enables associative prototype retrieval and yields more consistent explanation quality across datasets while maintaining competitive predictive performance.

III. Engram Memory Network

This section presents EMN, which grounds prototype selection in memory-refined latent representations. The key insight is that prototypes that contribute most to the latents—optimized for both reconstruction and classification—are expected to encode input-relevant information, yielding more consistent explanations across diverse dataset characteristics. The EMN is designed to emulate the neurocognitive process by which humans recognize objects and retrieve relevant memories for interpretation. Its architecture follows a three-stage structure comprising an encoder, an engram memory module, and a decoder. We first describe the neurocognitive basis of our approach (Section III-A), then detail the encoder (Section III-B), the engram memory module (Section III-C), the decoder (Section III-D), and the learning objectives (Section III-E).

A. Neurocognitive Basis

Human object recognition integrates sensory evidence with stored visual knowledge through hierarchical processing in the visual system [26], [27]. The proposed EMN is inspired by this hierarchy and adopts a three-stage architecture, as illustrated in Figure 2(a–c).

Stage 1: Visual Feature Encoding (Figure 2(a)). Visual signals propagate from the retina to the primary visual cortex (V1). Along the ventral visual stream, representations become progressively more abstract through a hierarchy of visual areas (e.g., V1–V4) and culminate in the inferior temporal cortex (ITC), supporting object recognition [27], [28]. EMN models this process using an encoder $f(\cdot)$ that transforms images into visual latent representations (Figure 2(d) and Section III-B).

Stage 2: Associative Memory Activation (Figure 2(b)). In the brain, ITC activates associative memories—or engrams—through stimulus-driven and recurrent excitation [29]. Formed via Hebbian learning [30], engrams enable pattern completion: partial input can reactivate the full memory trace [31]. For example, seeing a cat may evoke related visual or verbal memories. EMN is motivated by this process using our modified update rule (Section III-C), where visual and categorical latent vectors iteratively interact with prototype memories. These interactions integrate three components: continuous input injection, recurrent memory feedback, and state decay, converging to stable representations that encode both perceptual and semantic information (Figure 2(e) and Section III-C).

Stage 3: Recognition and Reconstruction (Figure 2(c)). Converged activity in ITC informs object classification and memory recall, with feedback signals projecting to early visual areas such as V1. EMN captures this process via a decoder that reconstructs the input from the final latent state (Figure 2(f) and Section III-D).

Since memory-refined latents are designed to capture sufficient information for both reconstruction and classification, prototypes with high contribution to these latents are likely input-relevant, facilitating valid explanations regardless of dataset characteristics.

B. Encoder

As illustrated in Figure 2(d), the encoder in EMN transforms visual input into latent representations that serve as the initial substrate for memory interaction and subsequent inference.

a) Input Encoding: EMN encodes input images through an encoder $f(\cdot)$, mimicking early visual processing along the ventral stream (Figure 2(a)). Given an input image $\mathbf{x}$, the encoder produces a latent vector $\mathbf{h} = \sigma(f(\mathbf{x})) \in \mathbb{R}^{d_1}$, where $f(\cdot)$ denotes the encoding module. The activation function $\sigma(\cdot)$ is biologically inspired and defined piecewise as follows: it outputs ϵx for $x < 0$; returns x when $0 \leq x \leq 1$; and computes $\epsilon x + 1 - \epsilon$ for $x > 1$. This formulation ensures non-zero gradients across the domain while encouraging activations to concentrate near biologically plausible firing thresholds.

b) Prototype and Engram Initialization: To model associative memory, EMN initializes a set of N prototype-label pairs $(\mathbf{p}_{\mathbf{x},i}, p_{y,i})$ by randomly sampling from the training set, where $\mathbf{p}_{\mathbf{x},i}$ is the i-th prototype image, $p_{y,i}$ is its corresponding class label, and subscripts $\mathbf{x}$ and y denote image and label modalities, respectively. These are encoded into memory vectors $\mathbf{m}_{\mathbf{x},i} = \sigma(f(\mathbf{p}_{\mathbf{x},i})) \in \mathbb{R}^{d_1}$ and $\mathbf{m}_{y,i} = \sigma(W_y[p_{y,i}]) \in \mathbb{R}^{d_2}$, where W_y is a class embedding matrix. The resulting set $\{\mathbf{m}_{\mathbf{x},i}, \mathbf{m}_{y,i}\}_{i=1}^{N}$ forms the engram memory, which supports associative retrieval and pattern completion during inference.

C. Engram Memory Module

To model associative memory dynamics observed in the brain (Figure 2(b) and Section III-A), we introduce the EMM as a module for an iterative process that refines latent states by integrating three components: (i) input injection from the encoder, (ii) memory-driven recurrent feedback, and (iii) leaky decay of the current state (Figure 2(e)). Given an encoded input $\mathbf{h} \in \mathbb{R}^{d_1}$, the iterative process produces three outputs after T steps: a refined visual latent vector $\mathbf{z}_{\mathbf{x}}^{(T)} \in \mathbb{R}^{d_1}$ used for reconstruction, a categorical latent vector $\mathbf{z}_{y}^{(T)} \in \mathbb{R}^{d_2}$ used for classification, and an activation vector $\mathbf{c}^{(T)} \in \Delta^{N}$ used to

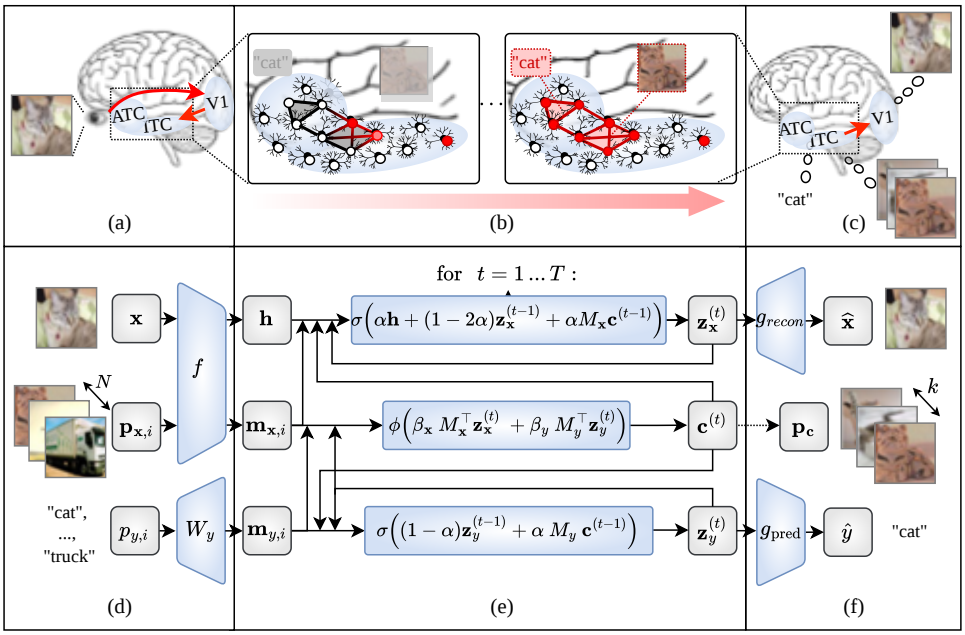

Fig. 2. (a–c) Visual pathway from ventral stream to inferior temporal cortex (ITC), enabling associative memory and categorization with feedback to early visual areas. (d–f) EMN models this process via an encoder, memory-guided refinement, and a decoder.

select explanatory prototypes. At each iteration $t = 1, \dots, T$, the update rules are given by:

$$\mathbf{z}_{\mathbf{x}}^{(0)} = \mathbf{0}, \quad \mathbf{z}_y^{(0)} = \mathbf{0}, \quad \mathbf{c}^{(0)} = \frac{\mathbf{1}}{N} \tag{1}$$

$$\mathbf{z}_{\mathbf{x}}^{(t)} = \sigma\left(\alpha\mathbf{h} + (1-2\alpha)\mathbf{z}_{\mathbf{x}}^{(t-1)} + \alpha M_{\mathbf{x}}\mathbf{c}^{(t-1)}\right) \tag{2}$$

$$\mathbf{z}_y^{(t)} = \sigma\left((1-\alpha)\mathbf{z}_y^{(t-1)} + \alpha M_y\mathbf{c}^{(t-1)}\right) \tag{3}$$

$$\mathbf{c}^{(t)} = \phi\left(\beta_{\mathbf{x}} M_{\mathbf{x}}^\top \mathbf{z}_{\mathbf{x}}^{(t)} + \beta_y M_y^\top \mathbf{z}_y^{(t)}\right), \tag{4}$$

where $\sigma(\cdot)$ is the activation function from Section III-B, $\phi(\cdot)$ denotes softmax, $\alpha \in (0, 1)$ controls the update rate, and $\beta_{\mathbf{x}}, \beta_y > 0$ are learnable coefficients. Here, $M_{\mathbf{x}} \in \mathbb{R}^{d_1 \times N}$ and $M_y \in \mathbb{R}^{d_2 \times N}$ denote the memory matrices containing latent visual and label prototypes, respectively. The vector $\mathbf{c}^{(t)}$ reflects the activation levels of the N prototypes based on the current latent state.

Each component of the update mechanism has a specific computational role. Equation 1 zero-initializes the latent states to begin refinement from a neutral point. In Equations 2 and 3, the visual latent vector $\mathbf{z}_{\mathbf{x}}$ and categorical latent vector $\mathbf{z}_y$ are updated, respectively, via: (i) $\alpha\mathbf{h}$ provides continuous input from the encoder (for $\mathbf{z}_{\mathbf{x}}$ only), coupling encoder input with memory refinement and enabling end-to-end training; (ii) $\alpha M_{\mathbf{x}}\mathbf{c}^{(t-1)}$ or $\alpha M_y\mathbf{c}^{(t-1)}$ contributes recurrent feedback weighted by memory activation; (iii) $(1-2\alpha)\mathbf{z}_{\mathbf{x}}^{(t-1)}$ or $(1-\alpha)\mathbf{z}_y^{(t-1)}$ represents state decay—the difference in decay rates reflects the presence or absence of encoder input. Components (ii), (iii), and the softmax in Equation 4 follow continuous modern Hopfield dynamics [18], while (i) is introduced to apply Hopfield dynamics in latent space.

Differences from [18]: While structurally related to continuous modern Hopfield network [18], this update mechanism differs in three key ways: latent states are zero-initialized (Eq. 1); bottom-up input $\mathbf{h}$ is injected at each step (Eq. 2); and the contribution weights $\beta_{\mathbf{x}}, \beta_y$ are learnable rather than fixed.

D. Decoder

As shown in Figure 2(f), EMN generates three task-specific outputs—image reconstruction, classification, and explanation—based on the refined latent states $\mathbf{z}_{\mathbf{x}}^{(T)}$, $\mathbf{z}_y^{(T)}$, and the engram activation vector $\mathbf{c}^{(T)}$. Image reconstruction is performed by a decoder g_{rec}, which maps $\mathbf{z}_{\mathbf{x}}^{(T)}$ to $\hat{\mathbf{x}} = g_{\text{rec}}(\mathbf{z}_{\mathbf{x}}^{(T)})$, reflecting feedback from ITC to early visual areas. The decoder architecture is flexible and task-agnostic. Classification uses a linear head g_{pred} applied to $\mathbf{z}_y^{(T)}$: $\hat{\mathbf{y}} = \beta_y\sigma(W_y)\mathbf{z}_y^{(T)}$. This design reflects the alignment between categorical activation and class prediction. Explanations are derived from $\mathbf{c}^{(T)}$ by selecting the top-K activated prototypes, emulating memory-based associative retrieval. These decoding pathways instantiate perception, recognition, and memory retrieval, reflecting distinct cognitive functional roles. Because the same memory-refined latent supports both reconstruction and classification, prototypes that contribute most to this latent are likely to encode input-relevant information, providing a principled basis for explanation validity across diverse domains.

E. Learning Objectives

EMN is trained using a multi-objective loss that jointly optimizes classification, reconstruction, and explanation quality:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{\text{div}}\mathcal{L}_{\text{div}} + \lambda_{\text{sep}}\mathcal{L}_{\text{sep}},$$

where each λ balances the contribution of the respective term. The classification loss ensures that the label latent vector predicts the correct class: $\mathcal{L}_{\text{cls}} = \text{CrossEntropy}(\hat{\mathbf{y}}, y)$, encouraging class-consistent engram activations. To retain visual fidelity, the reconstruction loss minimizes the discrepancy between

the input and reconstructed image: $\mathcal{L}_{\text{rec}} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2$. To mitigate prototype overuse and collapse (see Figure 1), EMN introduces two entropy-based losses that regulate memory utilization. The diversity loss promotes global usage of the memory space by maximizing entropy over the batch-averaged activation distribution: $\mathcal{L}_{\text{div}} = -\mathcal{H}(\bar{\mathbf{c}}) = \sum_{i=1}^{N} \bar{c}_i \log \bar{c}_i$, where $\bar{\mathbf{c}} = \frac{1}{B}\sum_{j=1}^{B} \mathbf{c}_j^{(T)}$. The separation loss encourages local selectivity by aligning each sample's prototype entropy to a target: $\mathcal{L}_{\text{sep}} = \left| \frac{1}{B}\sum_{j=1}^{B} \mathcal{H}(\mathbf{c}_j^{(T)}) - \log K \right|$. This guides each explanation to rely on a small number of relevant prototypes—typically around three—to balance richness and focus. Together, $\mathcal{L}_{\text{div}}$ and $\mathcal{L}_{\text{sep}}$ shape the softmax-driven memory dynamics to ensure explanations are diverse across inputs and selective within each instance, enabling EMN to mimic human-like associative recall.

IV. Experiments

We validate the effectiveness of the proposed EMN through comprehensive experiments.

Datasets. We evaluate on CIFAR-10 [12], CIFAR-100 [12], MNIST [11], HAM10000 [19] (medical dermatology), and CUB-200 [13] (fine-grained bird classification).

Model Architecture. Although EMN is compatible with various backbone architectures, we adopt Vision Transformer (ViT) [32] for all primary comparisons to maintain consistency with recent prototype-based models. The encoder uses DeiT-Small [33][2] and the decoder uses a modified ViT architecture.

Training Process. Training proceeds in three stages: (1) the encoder is initialized with DeiT-Small weights pretrained on ImageNet [34]; (2) the encoder and decoder are jointly trained on the target dataset using classification and reconstruction losses without the engram memory module, for 30 epochs on CIFAR and 50 epochs on other datasets; (3) the full EMN is then trained for 50 epochs on all datasets.

Hyperparameters. We use N=2000 prototypes randomly sampled from the training set with K=3 for explanation. Latent dimensions are d_1=4096 (visual) and d_2=512 (categorical). Memory refinement runs for T=24 iterations with update rate α=0.05 and ε=0.001 in $\sigma(\cdot)$. Loss weights are λ_{cls}=λ_{rec}=λ_{div}=λ_{sep}=1. Training uses AdamW [35] with lr=10^{-5}, weight decay 0.05, batch size 64. Data augmentation includes RandomHorizontalFlip and ColorJitter (brightness=0.2, contrast=0.2, saturation=0.2, and hue=0.1).

Baselines. We compare EMN against ProtoVAE [10], ProtoPNet [1], ProtoViT [8], and ProtoPFormer [7]. All baselines are trained and evaluated according to their official protocols, using the same data splits, preprocessing, and augmentation. All models are implemented in PyTorch [36].

Experimental Environment. All experiments were conducted on a single NVIDIA GPU (32GB VRAM) running Ubuntu Linux. Training EMN on CIFAR-10 takes approximately 3 hours across all training stages.

TABLE I
Visual similarity (LPIPS [14] and DISTS [15]) comparison across datasets. Results are reported as mean ± std. ($\times 10^{-1}$; lower is better; bold indicates best).

Model	LPIPS [14]		DISTS [15]	
	top1	top3	top1	top3
(CIFAR-10 [12])				
ProtoVAE [10]	7.50 ± 0.06	7.50 ± 0.07	6.75 ± 0.11	6.76 ± 0.11
ProtoPNet [1]	6.95 ± 0.19	7.00 ± 0.32	4.07 ± 0.13	4.02 ± 0.12
ProtoViT [8]	6.26 ± 0.10	6.31 ± 0.07	3.67 ± 0.03	3.69 ± 0.03
ProtoPFormer [7]	6.23 ± 0.02	6.24 ± 0.02	3.68 ± 0.01	3.69 ± 0.02
EMN (ours)	**5.38 ± 0.02**	**5.50 ± 0.02**	**3.22 ± 0.01**	**3.28 ± 0.01**
(CIFAR-100 [12])				
ProtoVAE [10]	7.27 ± 0.06	7.27 ± 0.06	6.32 ± 0.07	6.32 ± 0.07
ProtoPNet [1]	6.49 ± 0.08	6.57 ± 0.08	3.83 ± 0.04	3.88 ± 0.03
ProtoViT [8]	6.11 ± 0.02	6.22 ± 0.01	3.57 ± 0.01	3.62 ± 0.00
ProtoPFormer [7]	6.34 ± 0.04	6.35 ± 0.02	3.71 ± 0.02	3.71 ± 0.02
EMN (ours)	**5.66 ± 0.02**	**5.85 ± 0.01**	**3.38 ± 0.01**	**3.49 ± 0.01**
(MNIST [11])				
ProtoVAE [10]	2.33 ± 0.06	2.34 ± 0.06	2.38 ± 0.02	2.38 ± 0.02
ProtoPNet [1]	6.71 ± 0.05	6.82 ± 0.08	4.28 ± 0.05	4.37 ± 0.07
ProtoViT [8]	6.59 ± 0.06	6.73 ± 0.07	4.12 ± 0.02	4.20 ± 0.03
ProtoPFormer [7]	3.06 ± 0.05	3.05 ± 0.03	2.22 ± 0.01	2.22 ± 0.01
EMN (ours)	**1.70 ± 0.01**	**1.82 ± 0.01**	**1.73 ± 0.01**	**1.78 ± 0.01**
(HAM10000 [19])				
ProtoVAE [10]	–	–	–	–
ProtoPNet [1]	5.19 ± 0.23	5.35 ± 0.21	3.43 ± 0.08	3.48 ± 0.07
ProtoViT [8]	4.69 ± 0.17	4.71 ± 0.16	3.15 ± 0.03	3.20 ± 0.03
ProtoPFormer [7]	5.40 ± 0.53	5.40 ± 0.47	3.61 ± 0.44	3.64 ± 0.38
EMN (ours)	**3.50 ± 0.02**	**3.59 ± 0.01**	**2.40 ± 0.02**	**2.46 ± 0.02**
(CUB-200 [13])				
ProtoVAE [10]	–	–	–	–
ProtoPNet [1]	6.79 ± 0.05	6.81 ± 0.05	4.12 ± 0.03	4.13 ± 0.02
ProtoViT [8]	6.82 ± 0.00	6.87 ± 0.00	3.97 ± 0.00	4.00 ± 0.00
ProtoPFormer [7]	6.77 ± 0.03	6.80 ± 0.02	3.72 ± 0.02	3.75 ± 0.02
EMN (ours)	**6.44 ± 0.02**	**6.55 ± 0.01**	**3.55 ± 0.01**	**3.62 ± 0.00**

A. Visual Similarity

This section assesses the visual similarity between input queries and selected prototypes. Since explanations are intended for human users, the ideal evaluation would rely on human judgments; however, due to feasibility constraints, we employ proxy metrics—LPIPS [14] and DISTS [15]—that align with human perceptual judgments, while imperfect but scalable for large-scale evaluation.

As shown in Table I, EMN achieves the best visual similarity (lowest LPIPS [14] and DISTS [15]) across all five datasets, with an average improvement of 17.06% over the best baseline (ProtoPFormer [7]). These results indicate that EMN retrieves prototypes more perceptually aligned with input queries, providing more interpretable explanations as illustrated in Figure 1. ProtoVAE [10] is excluded from CUB-200 [13] and HAM10000 [19] as its decoded prototypes become uninformative and blurry on these datasets.

B. Subclass Alignment

Representation learning aims to preserve the latent geometry of real-world objects so that semantically similar instances are placed in close proximity. Since latent structure is unobservable, we use hierarchical labels (superclass–subclass) as

[2]timm library: https://github.com/huggingface/pytorch-image-models

TABLE II
SUBCLASS ALIGNMENT (%) ON CIFAR-100S (HIGHER IS BETTER; CHANCE LEVEL IS 1%).

Method	**top1**	**top3**
ProtoVAE [10]	-	-
ProtoPNet [1]	2.28 ± 0.91	1.95 ± 0.47
ProtoViT [8]	15.93 ± 6.55	14.03 ± 7.04
ProtoPFormer [7]	19.49 ± 1.60	19.42 ± 1.38
EMN (ours)	**39.87 ± 0.72**	**36.07 ± 0.36**

TABLE III
CLASSIFICATION ACCURACY (%) ACROSS ALL DATASETS (HIGHER IS BETTER; BOLD INDICATES BEST).

Model	Acc.	Model	Acc.
(CIFAR-10 [12])		**(MNIST [11])**	
ProtoVAE [10]	81.59 ± 0.36	ProtoVAE [10]	99.05 ± 0.02
ProtoPNet [1]	94.30 ± 0.12	ProtoPNet [1]	99.06 ± 0.10
ProtoViT [8]	97.63 ± 0.08	ProtoViT [8]	99.09 ± 0.09
ProtoPFormer [7]	**97.85 ± 0.06**	ProtoPFormer [7]	**99.41 ± 0.05**
EMN (ours)	96.39 ± 0.27	EMN (ours)	99.19 ± 0.08
(CIFAR-100 [12])		**(HAM10000 [19])**	
ProtoVAE [10]	50.37 ± 0.89	ProtoVAE [10]	–
ProtoPNet [1]	68.63 ± 0.35	ProtoPNet [1]	62.92 ± 2.84
ProtoViT [8]	86.41 ± 0.13	ProtoViT [8]	80.49 ± 0.68
ProtoPFormer [7]	**87.83 ± 0.25**	ProtoPFormer [7]	66.95 ± 0.00
EMN (ours)	85.28 ± 0.21	EMN (ours)	**88.51 ± 1.02**
(CUB-200 [13])			
ProtoVAE [10]	–		
ProtoPNet [1]	22.43 ± 2.29		
ProtoViT [8]	**74.09 ± 0.24**		
ProtoPFormer [7]	50.51 ± 1.61		
EMN (ours)	68.74 ± 0.54		

a proxy: e.g., "Pine" and "Oak" within "Tree" share more features than items from other superclasses. Models are trained using only superclass labels while subclass labels are withheld; at evaluation, we reveal subclass labels as hidden ground truth to measure prototype–query alignment. To evaluate whether models identify prototypes that are class-consistent and visually aligned with the query at a finer subclass level, we conduct a hierarchical matching experiment using CIFAR-100S, which reorganizes CIFAR-100 [12]'s 100 classes into 20 superclasses (5 subclasses each).[3] ProtoVAE [10] is excluded as its generated prototypes lack subclass labels. Results are summarized in Table II. Table II shows that EMN achieves 39.87% (top-1) and 36.07% (top-3) subclass alignment, 104.57% higher than the best baseline ProtoPFormer [7] (19.49%, 19.42%). This substantial improvement demonstrates that EMN effectively captures fine-grained visual features, enabling input-specific prototype selection rather than generic class-level matching.

C. Classification

Table III reports classification accuracy across all datasets. EMN achieves the highest average accuracy (87.62%) among all methods, outperforming the best baseline ProtoViT [8] (87.54%). Notably, on HAM10000 [19] (medical imaging), EMN achieves the highest accuracy (88.51%), outperforming the best baseline by 8 percentage points.

[3]https://github.com/BillyBSig/CIFAR-100-TFDS

TABLE IV
NMI$(X;P)$ [16] SCORES ACROSS ALL DATASETS ($\times 10^{-1}$; HIGHER IS BETTER; BOLD INDICATES BEST).

Model	top1	top3
(CIFAR-10 [12])		
ProtoVAE [10]	0.07 ± 0.02	**2.41 ± 1.37**
ProtoPNet [1]	0.07 ± 0.10	0.08 ± 0.12
ProtoViT [8]	0.60 ± 0.08	0.50 ± 0.05
ProtoPFormer [7]	0.55 ± 0.05	0.55 ± 0.04
EMN (ours)	**1.26 ± 0.02**	1.21 ± 0.02
(CIFAR-100 [12])		
ProtoVAE [10]	0.04 ± 0.01	0.06 ± 0.08
ProtoPNet [1]	0.29 ± 0.01	0.24 ± 0.01
ProtoViT [8]	0.49 ± 0.03	0.44 ± 0.01
ProtoPFormer [7]	0.44 ± 0.03	0.42 ± 0.02
EMN (ours)	**1.05 ± 0.04**	**0.97 ± 0.03**
(MNIST [11])		
ProtoVAE [10]	**3.08 ± 0.06**	**3.05 ± 0.06**
ProtoPNet [1]	–	–
ProtoViT [8]	1.84 ± 0.10	1.35 ± 0.10
ProtoPFormer [7]	1.78 ± 0.10	1.73 ± 0.08
EMN (ours)	2.62 ± 0.05	2.49 ± 0.03
(HAM10000 [19])		
ProtoVAE [10]	–	–
ProtoPNet [1]	0.68 ± 0.38	0.63 ± 0.35
ProtoViT [8]	0.97 ± 0.15	0.78 ± 0.10
ProtoPFormer [7]	0.00 ± 0.00	0.00 ± 0.00
EMN (ours)	**1.18 ± 0.04**	**1.09 ± 0.04**
(CUB-200 [13])		
ProtoVAE [10]	–	–
ProtoPNet [1]	0.70 ± 0.03	0.65 ± 0.03
ProtoViT [8]	1.38 ± 0.01	1.33 ± 0.01
ProtoPFormer [7]	1.38 ± 0.03	1.32 ± 0.02
EMN (ours)	**1.41 ± 0.03**	**1.34 ± 0.02**

D. Prototype Information Content

To evaluate whether prototypes capture meaningful information, we compute NMI$(X;P)$ [16][4], which measures how well prototypes preserve input-specific visual details. As shown in Table IV, EMN achieves the best NMI$(X;P)$ [16] on four of five datasets, with an improvement of 42.42% over the best baseline (ProtoViT [8]). The exception is MNIST [11], where ProtoVAE [10] achieves higher NMI$(X;P)$ [16] due to its reconstruction-based approach being well-suited to simple, low-variance images. Overall, these results indicate that EMN's selected prototypes reflect individual query characteristics, aligning with its design goal of retrieving visually similar examples rather than generic class representatives.

E. Sensitivity Analysis

Table V shows sensitivity analysis on CIFAR-10 [12]. Increasing the number of prototypes N (from 200 to 5000) consistently improves explanation quality: LPIPS [14] (5.75→5.34), DISTS [15] (3.39→3.18), and NMI$(X;P)$ [16] (1.06→1.28), while classification accuracy remains stable (96.63–96.69%). This indicates that larger memory provides richer retrieval candidates. The target entropy K in $\mathcal{L}_{\text{sep}}$ shows robust behavior across $K \in \{1,3,5,7\}$: accuracy 96.17–96.73%, LPIPS [14] 5.40–5.50, DISTS [15] 3.22–3.33, and

[4]Calculated using k-nearest-neighbor–based entropy estimation from the NPEET package: https://github.com/gregversteeg/NPEET

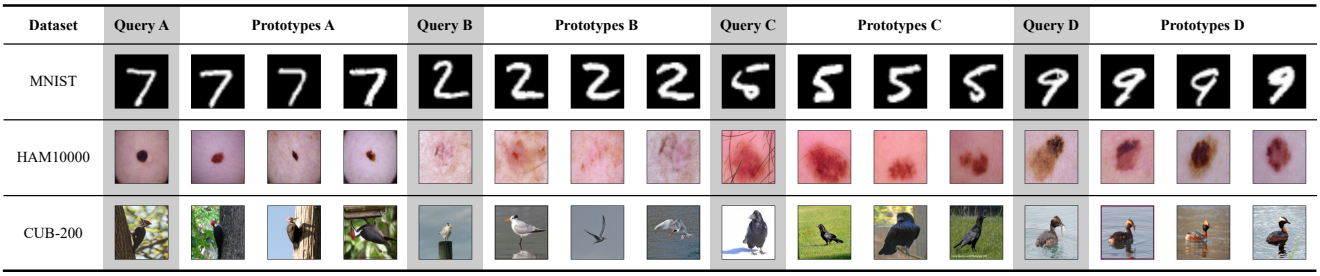

Fig. 3. Examples of prototype-based explanations across datasets: (from top to bottom) MNIST [11], HAM10000 [19], and CUB-200 [13]. The Query A-D columns show input queries, and the Prototype A-D column groups display the corresponding top-3 retrieved prototypes.

TABLE V
SENSITIVITY ANALYSIS ON CIFAR-10 [12]. N: NUMBER OF PROTOTYPES; K: TARGET ENTROPY FOR $\mathcal{L}_{\text{SEP}}$; d_2: CATEGORICAL LATENT DIMENSION. ACC (%) HIGHER IS BETTER; LPIPS [14], DISTS [15] ($\times 10^{-1}$) LOWER IS BETTER; NMI [16] ($\times 10^{-1}$) HIGHER IS BETTER.

Setting	Acc↑	LPIPS [14]↓	DISTS [15]↓	NMI$(X;P)$ [16]↑
N=200	96.63 ± 0.20	5.75 ± 0.02	3.39 ± 0.01	1.06 ± 0.04
N=500	96.63 ± 0.30	5.59 ± 0.02	3.30 ± 0.01	1.17 ± 0.02
N=2000	96.69 ± 0.25	5.40 ± 0.02	3.22 ± 0.01	1.21 ± 0.03
N=5000	96.69 ± 0.19	**5.34 ± 0.03**	**3.18 ± 0.02**	**1.28 ± 0.02**
K=1	96.17 ± 0.24	5.50 ± 0.04	3.33 ± 0.02	**1.24 ± 0.02**
K=3	96.69 ± 0.25	**5.40 ± 0.02**	**3.22 ± 0.01**	1.21 ± 0.03
K=5	96.72 ± 0.12	5.41 ± 0.03	**3.22 ± 0.01**	1.21 ± 0.04
K=7	**96.73 ± 0.25**	5.41 ± 0.01	**3.22 ± 0.01**	1.21 ± 0.03
d_2=128	96.57 ± 0.26	**5.38 ± 0.03**	**3.21 ± 0.01**	**1.25 ± 0.03**
d_2=256	96.62 ± 0.25	5.40 ± 0.01	**3.21 ± 0.01**	1.22 ± 0.02
d_2=512	96.69 ± 0.25	5.40 ± 0.02	3.22 ± 0.01	1.21 ± 0.03
d_2=1024	**96.72 ± 0.21**	5.41 ± 0.03	3.22 ± 0.01	1.24 ± 0.04

TABLE VI
ABLATION STUDY ON LOSS COMPONENTS (CIFAR-10 [12]). ACC (%) HIGHER IS BETTER; LPIPS [14], DISTS [15] ($\times 10^{-1}$) LOWER IS BETTER; NMI [16] ($\times 10^{-1}$) HIGHER IS BETTER.

Setting	Acc↑	LPIPS [14]↓	DISTS [15]↓	NMI$(X;P)$ [16]↑
EMN (ours)	96.69 ± 0.25	5.40 ± 0.02	**3.22 ± 0.01**	1.21 ± 0.03
w/o $\mathcal{L}_{\text{rec}}$	96.67 ± 0.28	**5.39 ± 0.02**	**3.22 ± 0.01**	1.24 ± 0.02
w/o $\mathcal{L}_{\text{div}}$	96.44 ± 0.33	5.43 ± 0.03	3.31 ± 0.02	1.27 ± 0.04
w/o $\mathcal{L}_{\text{sep}}$	97.37 ± 0.17	5.48 ± 0.02	3.26 ± 0.01	1.17 ± 0.01
w/o $\mathcal{L}_{\text{rec}}, \mathcal{L}_{\text{div}}$	96.36 ± 0.19	5.41 ± 0.04	3.30 ± 0.02	1.27 ± 0.05
w/o $\mathcal{L}_{\text{rec}}, \mathcal{L}_{\text{sep}}$	**97.38 ± 0.15**	5.57 ± 0.02	3.32 ± 0.00	1.16 ± 0.02
w/o $\mathcal{L}_{\text{div}}, \mathcal{L}_{\text{sep}}$	96.74 ± 0.55	5.82 ± 0.12	3.60 ± 0.13	**1.49 ± 0.20**
w/o all	97.15 ± 0.17	5.54 ± 0.04	3.43 ± 0.02	1.31 ± 0.05

NMI$(X;P)$ [16] 1.21–1.24. The categorical latent dimension d_2 reveals a trade-off: as d_2 increases from 128 to 1024, LPIPS [14] increases (5.38→5.41), DISTS [15] remains stable (3.21→3.22), and NMI$(X;P)$ [16] slightly decreases (1.25→1.24), while accuracy slightly improves (96.57→96.72%). These results confirm that EMN is robust to hyperparameter choices, with d_2 being the only source of a modest accuracy–explanation trade-off. We adopt N=2000, K=3, and d_2=512 as defaults.

F. Ablation Study

Table VI examines the contribution of each loss component. Individual removal of $\mathcal{L}_{\text{div}}$ or $\mathcal{L}_{\text{sep}}$ produces only modest changes, but removing both substantially degrades visual similarity (LPIPS [14] 5.40→5.82, DISTS [15] 3.22→3.60), demonstrating their synergistic effect. Notably, removing $\mathcal{L}_{\text{sep}}$ alone improves accuracy to 97.37%, revealing an accuracy–explanation trade-off. Nevertheless, even under this setting, EMN still outperforms the best baseline in visual similarity (e.g., LPIPS [14] 5.82 vs. 6.23 for ProtoPFormer [7]). These results reveal that $\mathcal{L}_{\text{sep}}$ controls the accuracy–explanation trade-off, while both losses together are essential for visual similarity. We adopt the full configuration to prioritize explanation quality.

G. Examples of Prototype-based Explanations

Figure 3 presents examples of EMN's prototype-based explanations across diverse datasets, including MNIST [11], HAM10000 [19], and CUB-200 [13]. For each input query (Query A–D columns; gray background), EMN retrieves the top-3 most similar prototypes from memory, displayed in the corresponding Prototype A–D columns (white background).

The selected prototypes share not only the same class label but also fine-grained visual features with the query. For MNIST [11] Query C, the retrieved prototypes match the query digit "5" in stroke style—specifically, the curvature of the upper loop and overall slant. For HAM10000 [19] Query D, the prototypes share diagnostically relevant features such as the central dark pigmentation pattern and irregular border shape. For CUB-200 [13] Query A, all three prototypes depict birds clinging vertically to a tree trunk—matching not just the species but also the pose of the query image.

These observations qualitatively support the subclass alignment results in Section IV-B: EMN selects prototypes that capture instance-specific visual details beyond class-level similarity. This suggests that the memory-refined latent preserves fine-grained information, enabling more semantically meaningful explanations across domains.

V. CONCLUSION

Existing prototype-based models often exhibit dataset-dependent explanation quality, limiting their reliability across domains. We propose the EMN, which mitigates this limitation by grounding prototype selection in memory-refined latent representations that jointly support reconstruction and classification. EMN retrieves specific training instances as explanations; this instance-based retrieval resembles episodic memory retrieval in cognitive neuroscience [37], which involves recollecting specific events along with their contextual details. The experimental results highlight the potential of incorporating brain-inspired memory dynamics into deep learning systems to improve explainability with a modest accuracy trade-off.

Limitations include the absence of formal convergence or stability analysis for the iterative update rule, reliance on proxy metrics rather than human evaluation studies, and baseline comparisons that follow each method's official protocol without unified reimplementation. A promising direction is to extend EMN toward concept-level (more semantic) prototypes that bridge instance retrieval and higher-level abstraction; we leave this as future work.

Acknowledgment

This paper was prepared with the assistance of Claude (Anthropic). The AI tool was used for writing assistance and code review.

References

[1] C. Chen, O. Li, A. Tao, A. J. Barnett, C. Rudin, and J. Su, "This looks like that: deep learning for interpretable image recognition," in *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, 2019, pp. 8930–8941.

[2] A. Di Marino, V. Bevilacqua, A. Ciaramella, I. De Falco, and G. Sannino, "Ante-hoc methods for interpretable deep models: A survey," *ACM Computing Surveys*, vol. 57, no. 6, pp. 119:1–119:39, 2025.

[3] A. Narayanan and K. J. Bergen, "Prototype-based methods in explainable ai and emerging opportunities in the geosciences," *arXiv preprint arXiv:2410.19856*, 2024.

[4] C. Rudin, "Stop explaining black-box machine learning models for high-stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[5] J. Bien and R. Tibshirani, "Prototype selection for interpretable classification," *Annals of Applied Statistics*, vol. 5, no. 4, pp. 2403–2424, 2011.

[6] O. Li, H. Liu, C. Chen, and C. Rudin, "Deep learning for case-based reasoning through prototypes," in *AAAI*, 2018.

[7] M. Xue, Q. Huang, H. Zhang, J. Hu, J. Song, M. Song, and C. Jin, "Protopformer: Concentrating on prototypical parts in vision transformers for interpretable image recognition," in *Proceedings of the 33rd Int'l Joint Conference on Artificial Intelligence (IJCAI 2024)*, 2024, pp. 1516–1524.

[8] C. Ma, J. Donnelly, W. Liu, S. Vosoughi, C. Rudin, and C. Chen, "Interpretable image classification with adaptive prototype-based vision transformers," *arXiv preprint arXiv:2410.20722*, 2024, submitted Oct 2024.

[9] M. Nauta, J. Trienes, S. Pathak, E. Nguyen, M. Peters, Y. Schmitt, J. Schlötterer, M. van Keulen, and C. Seifert, "From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI," *ACM Computing Surveys*, vol. 55, no. 13s, pp. 1–42, 2023.

[10] S. Gautam, A. Boubekki, S. Hansen, S. A. Salahuddin, R. Jenssen, M. M.-C. Höhne, and M. Kampffmeyer, "Protovae: A trustworthy self-explainable prototypical variational model," in *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, 2022, pp. 31 700–31 713.

[11] Y. LeCun, C. Cortes, and C. Burges, "Mnist handwritten digit database," *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, vol. 2, 2010.

[12] A. Krizhevsky, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009.

[13] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset," in *Technical Report CNS-TR-2011-001*, 2011.

[14] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 586–595.

[15] K. Ding, K. Ma, S. Wang, and E. P. Simoncelli, "Image quality assessment: Unifying structure and texture similarity," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 43, no. 5, pp. 1224–1236, 2021.

[16] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Wiley-Interscience, 2006.

[17] J. Liu, H. Zhang, T. Yu, L. Ren, D. Ni, Q. Yang, B. Lu, L. Zhang, N. Axmacher, and G. Xue, "Transformative neural representations support long-term episodic memory," *Science Advances*, vol. 7, no. 41, p. eabg9715, 2021.

[18] H. Ramsauer, B. Schäfl, and et al., "Hopfield networks is all you need," in *Proc. of the Int'l Conference on Learning Representations (ICLR)*, 2021, arXiv:2008.02217.

[19] P. Tschandl, C. Rosendahl, and H. Kittler, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific Data*, vol. 5, p. 180161, 2018.

[20] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," in *arXiv preprint arXiv:1312.6034*, 2014.

[21] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 618–626.

[22] S. Jain and B. C. Wallace, "Attention is not explanation," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019, pp. 3543–3556.

[23] D. Alvarez-Melis and T. S. Jaakkola, "Towards robust interpretability with self-explaining neural networks," in *Advances in Neural Information Processing Systems (NeurIPS 2018)*, 2018, pp. 7786–7795.

[24] J. Wang, H. Liu, X. Wang, and L. Jing, "Interpretable image recognition by constructing transparent embedding space," in *ICCV*, 2021, pp. 895–904.

[25] J. Donnelly, A. J. Barnett, and C. Chen, "Deformable protopnet: An interpretable image classifier using deformable prototypes," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 265–10 275.

[26] D. J. Felleman and D. C. Van Essen, "Distributed hierarchical processing in the primate cerebral cortex," *Cerebral Cortex*, vol. 1, no. 1, pp. 1–47, 1991.

[27] J. J. DiCarlo, D. Zoccolan, and N. C. Rust, "How does the brain solve object recognition?" *Neuron*, vol. 73, no. 3, pp. 415–434, 2012.

[28] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *Journal of Physiology*, vol. 160, no. 1, pp. 106–154, 1962.

[29] K. Kar, J. Kubilius, K. Schmidt, E. B. Issa, and J. J. DiCarlo, "Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior," *Nature Neuroscience*, vol. 22, no. 6, pp. 974–983, 2019.

[30] D. O. Hebb, *The Organization of Behavior: A Neuropsychological Theory*. New York, NY: John Wiley & Sons, 1949.

[31] S. A. Josselyn, S. Köhler, and P. W. Frankland, "Finding the engram," *Nature Reviews Neuroscience*, vol. 16, no. 9, pp. 521–534, 2015.

[32] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.

[33] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jegou, "Training data-efficient image transformers & distillation through attention," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 18–24 Jul 2021, pp. 10 347–10 357.

[34] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[35] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations (ICLR)*, 2019.

[36] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, 2019, pp. 8024–8035.

[37] E. Tulving, "Episodic and semantic memory," in *Organization of Memory*, E. Tulving and W. Donaldson, Eds. Academic Press, 1972, pp. 381–403.