Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Incremental and Open-Set Visual Learning with Cognitively Inspired Memory Mechanisms

Master's Thesis

2026
Dipl.-Ing. M.Arch. Jelena Epifanić

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

Incremental and Open-Set Visual Learning with Cognitively Inspired Memory Mechanisms

Master's Thesis

Study Programme:	Cognitive Science
Field of Study:	Computer Science
Department:	Department of Applied Informatics
Supervisor:	RNDr. Kristína Malinovská, PhD.
Consultant:	Prof. Dipl.-Ing. Dr.techn. Markus Vincze

Bratislava, 2026
Dipl.-Ing. M.Arch. Jelena Epifanić

Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Jelena Epifanic
Študijný program: kognitívna veda (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: informatika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: anglický
Sekundárny jazyk: slovenský

Názov: Incremental and Open-Set Visual Learning with Cognitively Inspired Memory Mechanisms
Inkrementálne a open-set vizuálne učenie s kognitívne inšpirovanými pamäťovými mechanizmami

Anotácia: Hlboké neurónové siete dosahujú vynikajúce výsledky pri close-set klasifikácii, avšak stále trpia katastrofickým zabúdaním [1]. Moderné Hopfielove siete (MHN) [2] sú známym modelom asociatívnej pamäti. Robustné reprezentácie a pamäť sú nevyhnutné pre nepretržité získavanie a aktualizáciu vedomostí. Tie možno postaviť na rôznych modelov ľudskej kategorizácie [3], na prototypovej a exemplárovej teórii.

Cieľ: Preskúmajte integráciu vektorov príznakov z foundation modelu pre videnie (napr. DINO) a MHN [2] na účely open-set klasifikácie a vyhodnoťte schopnosť inkrementálne sa učiť nové triedy objektov. Analyzujte vplyv úrovní granularity pamäte, presnosti vyhľadávania a spresňovania na úspešnosť klasifikácie a detekciu neznámych objektov ako aj robustnosť systému a porovnajte rôzne stratégie aktualizácie pamäte.

Literatúra:
[1] G. I. Parisi et al., Continual lifelong learning with neural networks: A review, Neural Networks, vol. 113, pp. 54-71, 2019.
[2] H. Ramsauer et al., Hopfield Networks is All You Need, CoRR, vol. abs/2008.02217, 2020.
[3] J. D. Smith, Prototypes, exemplars, and the natural history of categorization, Psychonomic Bulletin & Review, vol. 21, no. 2, pp. 312–331, 2013.

Vedúci: RNDr. Kristína Malinovská, PhD.
Katedra: FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry: doc. RNDr. Tatiana Jajcayová, PhD.

Dátum zadania: 12.05.2024

Dátum schválenia: 20.05.2024

prof. Ing. Igor Farkaš, Dr.
garant študijného programu

.....
študent

.....
vedúci práce

Comenius University Bratislava
Faculty of Mathematics, Physics and Informatics

THESIS ASSIGNMENT

Name and Surname:	Jelena Epifanic
Study programme:	Cognitive Science (Single degree study, master II. deg., full time form)
Field of Study:	Computer Science
Type of Thesis:	Diploma Thesis
Language of Thesis:	English
Secondary language:	Slovak

Title: Incremental and Open-Set Visual Learning with Cognitively Inspired Memory Mechanisms

Annotation: Modern deep learning models achieve strong performance in closed-set classification but they still struggle with catastrophic forgetting [1]. Modern Hopfield Networks (MHNs) [2] provide a memory-based framework with associative retrieval capabilities. In dynamic environments, robust representations and memory are essential for continuous knowledge acquisition and updating. The inspiration for building it can be drawn from competing models of human categorization [3], namely as prototype-based and exemplar-based memory regimes.

Aim: Study the integration of MHN-based [2] retrieval of embeddings from a vision foundation model (e.g. DINO) for open-set classification and evaluate its ability to incrementally learn new object classes. Analyze the impact of different memory granularities, retrieval sharpness, and refinement mechanisms on classification performance and unknown detection. Assess the robustness of the system on a benchmark and compare different memory update strategies.

Literature:
[1] G. I. Parisi et al., Continual lifelong learning with neural networks: A review, Neural Networks, vol. 113, pp. 54-71, 2019.
[2] H. Ramsauer et al., Hopfield Networks is All You Need, CoRR, vol. abs/2008.02217, 2020.
[3] J. D. Smith, Prototypes, exemplars, and the natural history of categorization, Psychonomic Bulletin & Review, vol. 21, no. 2, pp. 312–331, 2013.

Supervisor:	RNDr. Kristína Malinovská, PhD.
Department:	FMFI.KAI - Department of Applied Informatics
Head of department:	doc. RNDr. Tatiana Jajcayová, PhD.
Assigned:	12.05.2024
Approved:	20.05.2024

prof. Ing. Igor Farkaš, Dr.
Guarantor of Study Programme

Student

Supervisor

Acknowledgments: I would like to thank RNDr. Kristína Malinovská, PhD. and Dipl.-Ing. Dr.techn. Markus Vincze, for their invaluable support, patience and guidance in the writing of this thesis. Also, I am very grateful for the emotional support I received from my family and friends.

Abstract

Humans possess neurocognitive mechanisms that allow them to continuously acquire new knowledge, update existing information and preserve it over time. In dynamic environments, this ability is crucial for exploration and adaptation. Robust representations, together with memory mechanisms, support this capability. Cognitive categorization theory, through the two classical models (prototype and exemplar), attempts to explain how concepts are formed and how categorization emerges via feature-similarity comparison. This thesis combines a broad interdisciplinary theoretical overview of human memory, representation, and recognition with an experimental study of incremental visual learning. It explores Modern Hopfield Networks as associative memory models for incremental open-set visual recognition. The work introduces both a conceptual and an implemented memory-based framework that utilizes a pretrained self-supervised DINO encoder together with two complementary memory structures, prototype-based and exemplar-based. To support open-set recognition, memory is updated incrementally and MHN-based refinement can optionally be applied to query embeddings before retrieval. For classification, we compare the model performance using cosine-similarity scoring and MHN-based attention scoring. The study investigates how MHN-based retrieval influences the recognition of both known and unknown objects. The proposed method is evaluated under the Open World Object Detection COCO dataset split protocol using ground-truth object regions. The experiments analyze trade-offs between memory compactness, known-category recognition, and unknown-category rejection. Results show that prototype memories provide the strongest balance between open-set performance and memory efficiency, while exemplar memories improve known-class recognition at substantially larger memory sizes. MHN-based classification improves the known-unknown balance, most clearly for exemplar memory, mainly by increasing unknown recall, while the prototype benefit depends on the MHN sharpness setting. MHN-based query refinement reduces open-set performance because it shifts unknown-query embeddings closer to known memory representations, making unknown objects more likely to be accepted as known.

Keywords: continual learning, cognitive categorization theory, Modern Hopfield Networks, open-set recognition

Abstrakt

Ľudia disponujú neurokognitívnymi mechanizmami, ktoré im umožňujú neustále získavať nové vedomosti, aktualizovať existujúce informácie a uchovávať ich v priebehu času. V dynamických prostrediach je táto schopnosť kľúčová pre exploráciu a adaptáciu. Táto schopnosť je založená na robustných reprezentáciach a pamäťových mechanizmoch. Teória kognitívnej kategorizácie sa prostredníctvom dvoch klasických modelov (prototypový a exemplárový) má za cieľ vysvetliť, ako sa formujú koncepty a ako vzniká kategorizácia na báze porovnania podobnosti vlastností. V tejto práci skúmame moderné Hopfieldove siete (Modern Hopfield Networks, MHN) ako asociatívne pamäťové modely pre inkrementálne vizuálne rozpoznávanie v tzv. open set scenári. Navrhujeme framework, ktorý disponuje pamäťou, využívajúci predtrénovaný self-supervised model DINO spolu s dvoma komplementárnymi pamäťovými štruktúrami, založenými na prototypoch a exemplároch. Pre umožnenie rozpoznania nových objektov sa pamäť aktualizuje inkrementálne a navyše je možné aplikovať na zakódované dotazy zjemnenie od MHN ako vylepšenie. Pri klasifikácii porovnávame úspešnosť nášho modelu vuyžívajúceho MHN s klasickou metódou na báze kosínusovej podobnosti. Cieľom štúdie je preskúmať, ako dynamika pamäte podporovaná vyhľadávaním založeným na MHN ovplyvňuje rozpoznávanie známych aj neznámych objektov. Navrhovanú metódu vyhodnocujeme pomocou Open World Object Detection protokolu rozdelenia na datasete COCO s využitím referenčných (ground truth) objektových regiónov. Experimenty analyzujú kompromisy medzi kompaktnosťou pamäte, rozpoznávaním známych kategórií a odmietaním neznámych kategórií. Výsledky ukazujú, že pamäť založená na prototypoch poskytuje najlepšiu rovnováhu medzi úspešnosťou v open-set klasifikácii a efektívnosťou pamäte, zatiaľ čo exemplárová pamäť poskytuje zlepšenie rozpoznávania známych tried, ale vyžaduje podstatne väčšiu veľkosť pamäte. Klasifikácia založená na MHN zlepšuje rovnováhu medzi známymi a neznámymi objektami, najzjavnejšie v prípade exemplárovej pamäte, hlavne zvýšením miery recall u neznámych objektov, zatiaľ čo prínos prototypovej pamäte závisí od nastavenia ostrosti MHN. Vylepšenie založené na MHN znižuje úspešnosť v open-set úlohe tým, že pripodobňuje neznáme dotazy smerom k známym reprezentáciám pamäte.

Kľúčové slová: kontinuálne učenie, kognitívna teória kategorizácie, moderné Hopfieldove siete, open-set klasifikácia

Contents

List of Figures

List of Tables

Abbreviations

ANN Artificial Neural Network

CL Continual Learning

MHN Modern Hopfield Network

OWOD Open World Object Detection

CLS Complementary Learning Systems

DL Deep Learning

IOD Incremental Object Detection

CSS Continual Semantic Segmentation

CRL Continual Reinforcement Learning

ViT Vision Transformer

SOTA State-of-the-Art

LTP Long-Term Potentiation

HBM Hierarchical Bayesian model

ML Machine Learning

CNN Convolutional Neural Network

SSL Self-Supervised Learning

NLP Natural Language Processing

RNN Recurrent Neural Network

LLM Large Language Model

GAN Generative Adversarial Network

ACM Associative Memory Cell

EMA Exponential Moving Average

Introduction

Continual learning has been a subject of great interest in the field of artificial intelligence. In real-world applications, particularly robotics, it is essential for systems to adapt autonomously to dynamic environments and preserve previously learned information. Inspiration comes from biological systems, especially human cognitive mechanisms, as they are capable of learning incrementally and retaining stable representations over time despite changing inputs. However, in practice, Artificial Neural Networks (ANNs) cannot yet overcome catastrophic forgetting under shifting data distributions, where new information interferes with previously acquired knowledge. Although the underlying biological principles are still not fully understood, stable representations and memory consolidation have motivated research in incremental learning.

This thesis follows an interdisciplinary approach that combines ideas from machine learning, cognitive psychology, neuroscience, and computer vision. Theoretical overview of human memory, representation, and recognition with an experimental study allows us to connect biologically inspired memory mechanisms with modern self-supervised visual representations. The work investigates how principles of human categorization and associative memory can support continual learning in artificial systems.

It introduces both a conceptual and an implemented memory-based framework aimed at addressing open-set recognition and incremental learning through the exploration of Modern Hopfield Networks (MHNs) as associative memory models combined with large-scale self-supervised visual representations. For feature extraction, we employ a pretrained DINOv3 encoder. Our framework supports two memory regimes, prototype-based and exemplar-based representations, which are grounded in cognitive categorization models. This setup allows us to investigate different strategies for storing and updating knowledge over time.

We evaluate our framework under the Open World Object Detection (OWOD) protocol on the COCO dataset. The model is required to incrementally learn new categories while preserving the ability to recognize both known and previously unseen objects. In our experimental setup, we assume ground-truth object regions are available, therefore, the focus is more on classification and open-set recognition rather than detection. Motivated by biological learning principles, we aim to balance plasticity and

stability through explicit memory updates and retrieval mechanisms. The goal is to analyze how memory structure, update and retrieval strategies influence classification accuracy and novel class recognition. Understanding the biological underpinnings of continual visual learning and studying how different memory modules combined with MHNs influence incremental learning strategies could provide useful directions for future research.

To summarize, the main objectives of this thesis are: (1) to provide a detailed review of continual learning challenges and approaches in artificial systems; (2) to provide a comprehensive summary of the complexity of knowledge representation and memory formation from both neurocognitive and machine learning perspectives, covering historical and contemporary developments with an emphasis on visual systems; (3) to present a theoretical understanding of memory-based models, such as Hopfield networks and cognitive category learning theories, which directly served as an inspiration for the proposed thesis framework; (4) to study a memory-based framework using MHN retrieval and frozen DINOv3 representations for incremental and open-world visual recognition; (5) to investigate how different memory granularities, such as prototype-based and exemplar-based memory, affect the known–unknown trade-off under incremental updates; and (6) to study how MHN-based retrieval affects incremental open-world recognition.

In Chapter 1, we introduce the problem of continual learning in artificial systems together with the biological principles of Continual Learning (CL). We also present different research strategies that address this problem. In Chapter 2, we focus on related work that served as motivation for our framework. Memory and different knowledge representations that support knowledge acquisition and preservation are discussed in greater detail. Cognitive categorization theories, from classical models to contemporary research directions, are presented in Chapter 3. In this chapter, associative memory is presented from both, neurocognitive and computational perspective, where MHNs are introduced as an associative memory models. Chapter 4 describes the framework overview, methodology, implementation details, and evaluation protocol. The following chapters present the experimental evaluation and discuss the obtained results. In Experiment 1, we study how different memory granularities and update strategies affect open-set recognition. In Experiment 2, we investigate how MHN-based classification and refinement influence recognition performance. Finally, in Experiment 3, we analyze the effect of MHN-based retrieval under the OWOD protocol.

Chapter 1

Continual Learning: Biological and Computational Foundations

To effectively adapt to dynamic environments, AI systems should autonomously learn multiple tasks through observation and intervention, deal with uncertainty, and efficiently accumulate and update their knowledge. Much of the inspiration for developing CL methods comes from biological systems and neuroscience research (Kudithipudi et al., 2022; Hadsell et al., 2020). For instance, humans learn from a few examples, distil essential knowledge, and recall it as needed. Addressing this complex challenge, a group of researchers outlined features observed in biological systems that should be incorporated into models of "lifelong learning machines" (Kudithipudi et al., 2022). Additionally, recent work by Wang et al. (2023) highlights the current challenges in CL in artificial systems. The conceptual framework they proposed summarizes the objectives of continual learning as a trade-off between learning plasticity and memory stability on one hand, and achieving generalizability when dealing with dynamic data distributions, a common scenario in CL. The development of learning systems capable of preserving past knowledge while adapting to new information represents a central requirement addressed in the method proposed in our work.

1.1 Foundations of Lifelong Learning

Humans exhibit a remarkable ability to incrementally acquire, update, and transfer both knowledge and skills across different domains throughout their lifespan (Barnett and Ceci, 2002). Since the onset of AI, there has been a persistent goal to develop artificial systems that possess mechanisms enabling such capabilities. Continual learning, also known as lifelong or incremental learning in artificial systems is characterized by learning from dynamic data distributions, mirroring learning in real-world scenarios (Parisi et al., 2019). The idea of an autonomous agent that can independently explore

its environment and adapt to changes draws inspiration mostly from human learning abilities (Hassabis et al., 2017).

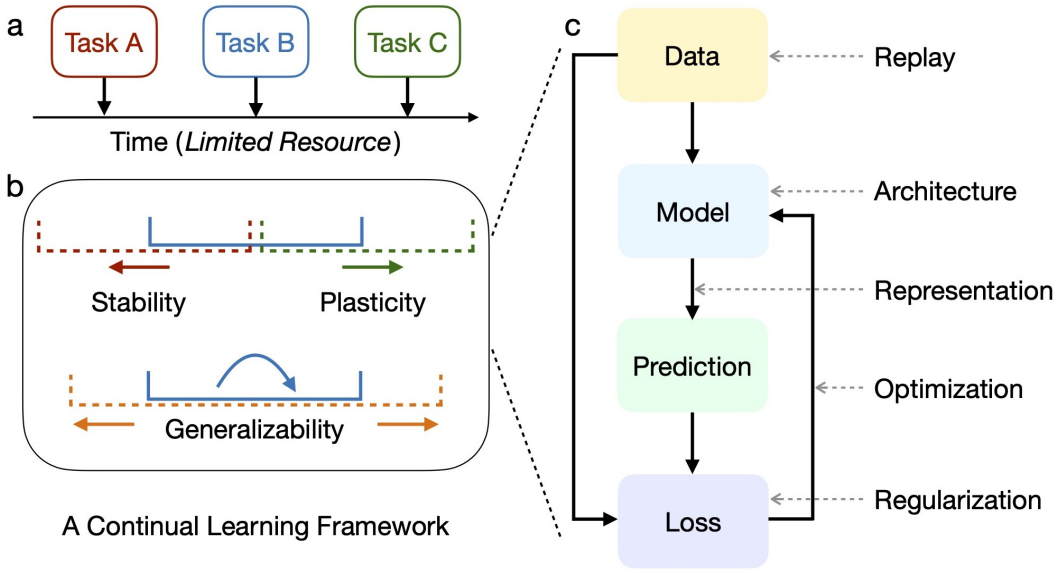


Figure 1.1: A conceptual framework of continual learning. (a) CL is viewed as the incremental adaptation to new knowledge. (b) It requires a balance between stability and plasticity while achieving generalizability. (c) Various machine learning strategies have been proposed to achieve these goals. Adapted from Wang et al. (2023).

The concept of lifelong learning for autonomous robot agents was introduced by Thrun and Mitchell (Thrun and Mitchell, 1995), where, for an array of correlated control tasks, a robot learns a selection of control policies. The authors emphasize the importance of transferring knowledge across different tasks to reduce the need for extensive real-world experiments and to increase learning efficiency. Presently, in its most basic understanding, lifelong learning focuses on the problem of catastrophic forgetting, where acquiring new knowledge interferes with previously learned knowledge, leading to performance degradation due to changes in data distributions (French, 1999). The prerequisite for overcoming the catastrophic forgetting problem is the ability to acquire new knowledge while simultaneously preserving previously acquired knowledge. The extent to which a system is capable of preserving old knowledge while still acquiring new information is termed the stability-plasticity dilemma, and it has been intensively studied in biological neural systems (Hassabis et al., 2017). The objectives of CL revolve around achieving a balance between stability and plasticity, along with ensuring adequate intra- and inter-task generalizability, all while maintaining resource efficiency (Wang et al., 2023). Such goals naturally lead to learning systems that dynamically retrieve and reinforce relevant past experiences depending on the current input, rather than storing knowledge in static parameters alone. The same group of authors of this survey systematized approaches to solving the CL problem into five groups: (i)

regularization-based approaches, (ii) replay-based approaches, (iii) optimization-based approaches, (iv) representation-based approaches, (v) architecture-based approaches (summarized in Fig. 1.1).

1.2 From Biological Inspiration to Machine Learning Systems

People learn throughout their lifespan and adapt to novel situations. Because of these fascinating, evolutionarily developed traits, humans often serve as inspiration for the development of AI solutions (Tani, 2016; Hassabis et al., 2017; Saxe et al., 2020). As shown in Fig. 1.2, Kudithipudi et al. (2022) identified a set of biological mechanisms that can serve as plausible features for the development of lifelong learning machines.

Figure 1.2: Illustration of the relationships between biological mechanisms and lifelong learning features. Adapted from Kudithipudi et al. (2022).

To overcome the catastrophic forgetting problem, biological systems possess a subtle regulatory mechanism of neurosynaptic plasticity. Neurosynaptic plasticity regulates the stability-plasticity balance in different brain areas at hierarchical levels, spanning from individual cells to circuits (Power and Schlaggar, 2016), and enables the integration of new information while preserving the consolidated one by modulating neural activity. One well-studied mechanism is Hebbian plasticity (Hebb, 1949), which strengthens the connection between neurons. On the other hand, homeostatic plasticity and metaplasticity stabilize neural activity and optimize it for learning (Vitureira et al., 2012; Li et al., 2019). Regularization approaches inspired by neuroscience models introduce strategies for constraining and penalizing the updates of neural weights

(Kirkpatrick et al., 2017). Despite a promising direction, the regularization methods still resemble a trade-off between the protection of old knowledge and acquiring of new one.

Neurogenesis, another crucial mechanism and the most extensive during early development, relates to the process of new neuron formation (Urbán and Guillemot, 2014). Dynamic architectures that mimic neurogenesis, resemble an approach where the structure of the model changes according to the need for new information acquisition by adjusting the number of neurons or layers in the network (Rusu et al., 2016). However, these solutions suffer from scalability problems and high memory demands.

Episodic replay and Complementary Learning Systems (CLS) theory (Kumaran et al., 2016) have served as an important inspiration for CL models. The brain exhibits the ability to generalize effectively by statistically extracting information from distinct episodic events. During learning, different brain areas are engaged, with one well-studied example being the neocortex and the hippocampus. According to the CLS theory (McClelland et al., 1995), the hippocampus and neocortex have interrelated functions. They mediate learning and memory processes, suggesting specialized mechanisms within the human cognitive system for protecting consolidated knowledge. The hippocampus facilitates rapid processing of streams of episodic experiences in short-term memory, which is then transferred to the neocortex for information distillation, consolidation, and long-term preservation (O'Reilly and Rudy, 2000). Models for knowledge generalization and retrieval exploit these concepts, whether in the form of memory storage or to generate old data distributions in more compact form (Shin et al., 2017; Ven et al., 2020). This perspective suggests that memory in both biological and artificial systems can be viewed as an associative process, where current inputs trigger the retrieval of structurally similar past experiences.

Transfer and adaptation are mechanisms recognized as key features in lifelong learning machines (Kudithipudi et al., 2022). Transfer learning refers to the ability to apply previously acquired knowledge from one task to different domains (Zhuang et al., 2021). Few-shot learning and meta-learning demonstrate promising potential in knowledge transfer (Hospedales et al., 2020; Wang et al., 2020).

Parisi et al. (2019) highlight the human traits of developmental and curriculum learning. Humans and other organisms have a critical period during infancy, when the brain exhibits high plasticity and is more sensitive to the consequences of experiences (Hubel and Wiesel, 1962). Achille et al. (2019) demonstrated that initial rapid learning, in the first few epochs, is a key element in defining the optimal deep neural network connections. Curriculum learning, where humans better acquire knowledge, when gradually exposed to organized information growing in complexity (Krueger and Dayan, 2009) is similar to the application of developmental learning. When transferred to ANNs, it leads to faster training performance (Elman, 1993). Similar strategies have

been applied in robotics (Sanger, 1994) and machine learning (Bengio et al., 2009).

Multisensory learning involves the integration of information from different modalities, a crucial feature of the brain (Ernst and Bülthoff, 2004). Multimodal learning in the computational domain proves useful in scenarios involving uncertainty and ambiguous sensory input. Additionally, one modality can be reconstructed using another, or a source modality can be transferred to a target one for fine-tuning (Cangea et al., 2020).

1.3 Real-world Applications in Continual Learning

Despite significant progress in the development of Deep Learning (DL) models, a big gap still remains between these models and autonomous agents that can acquire knowledge as biological systems do. This imbalance arises because current models are typically trained and evaluated on stationary data, failing to capture the complexity of real-world information and tasks. Humans learn incrementally, in an organized manner, often from few-shot examples (Wang et al., 2020), and, crucially, they transfer generalized knowledge across different tasks (Caruana, 1997).

In real-world settings, implementing these features poses an even greater challenge. As a result, solutions often target specific subsets of problems. Given the growing number of approaches , we will present only the most relevant directions. Incremental Object Detection (IOD) involves continual object detection, where new classes are introduced incrementally during new training episodes. The model must detect newly learned classes while retaining knowledge of previously learned ones (Shmelkov et al., 2017). The OWOD problem serves as the basis for our framework. In this scenario, the model identifies both known and unknown classes (Joseph et al., 2021). As an extension of IOD, Continual Semantic Segmentation (CSS) operates on a pixel level, requiring the model to preserve knowledge of old classes while learning new ones, often employing knowledge distillation and unsupervised learning techniques (Cermelli et al., 2020).

It is important to mention Continual Reinforcement Learning (CRL), where the idea is that agents learn throughout their entire lifespan (Abel et al., 2023). Another domain where continual learning has great potential and significance is in Natural Language Processing (NLP) (Li et al., 2025).

The need for AI models to efficiently update their knowledge has also shaped the future direction of lifelong learning research. A major trend in current research is the use of foundation models pre-trained on large-scale data corpora, such as BERT (Devlin et al., 2019) and CLIP (Radford et al., 2021), for downstream tasks. This approach is an effective method for knowledge transfer. The dominant choice for foundation models

is transformer-based architecture, which is used for both language and vision problems (Vaswani et al., 2017; Wang et al., 2022). Additionally, the use of multimodal models trained with contrastive learning can mitigate the problem of catastrophic forgetting, especially through the use of Large Language Models (LLMs) (Radford et al., 2021). For our study, the greatest interest lies in the vision foundation models (Siméoni et al., 2025; Ravi et al., 2024). Self-supervised models based on Vision Transformers (ViTs) architecture currently represent State-of-the-Art (SOTA) solutions in computer vision. A more detailed overview of these models will be presented in the following chapters.

1.4 Open Challenges in Continual Learning

In the last few years, significant progress has been made in addressing the CL challenge. However, key limitations remain, particularly with respect to how knowledge is preserved and maintained over time. Existing approaches primarily rely on parameter updates and replay-based methods. Even foundation models, despite their great performance, represent only a time-specific snapshot of data and typically require extensive retraining when adapting to new data distributions.

The central problem lies in how internal representations are formed, stored, and structured to enable efficient update and retrieval. Biological systems strongly rely on structured representations and memory dynamics, which support generalization through categorical and associative mechanisms. This suggests that lifelong learning is not only a problem of the stability-plasticity trade-offs and optimization, but also a problem of representation formation and memory organization.

Motivated by this perspective, in the following chapters we focus on representational learning and memory mechanisms, in both biological and artificial systems. In particular, we discuss how modern self-supervised models produce rich visual representations, how memory systems support incremental learning and how categorization principles connect representations with classification in incremental learning settings.

Chapter 2

Representations and Memory in Biological and Artificial Systems

In this chapter we discuss how we acquire information, encode it, and preserve it over time. Neuroscience, cognitive science, and philosophy provide different perspectives for explaining how mental representations of the external world, as well as abstract concepts, are formed. To better understand the multiple layers involved in these complex mechanisms, we approach the problem from the perspectives of neuroscience, cognitive science and artificial intelligence. This framing could be loosely connected to Marr's levels of analysis (Marr and Poggio, 2004), which distinguish between the computational, algorithmic, and implementation understanding of cognitive systems.

2.1 Neural Foundations of Representation

In recent decades, with advances in brain imaging techniques, neuroscience studies have provided insights into neural mechanisms underlying cognitive functions such as representations encoding, learning and memory formation.

Learning and memory formation in the brain are associated with changes in neural circuit activity. The brain encodes information as patterns of neural activity, meaning the representations correspond to physical states in neural populations. As discussed in the previous chapter, one of the first theories explaining how the brain encodes and preserves information was proposed by Hebb (1949). Hebb introduced the principle of synaptic plasticity - the co-activation of neurons strengthens the connection between them. The physiological and structural mechanisms supporting this process are associated with Long-Term Potentiation (LTP), including the changes in the number of receptors or increased neurotransmitter release (Tocco et al., 1992; Edwards, 1995). Hebbian learning provides a theoretical framework for understanding how learning emerges in neural networks through repeated activation and synchronization.

2.1.1 Neural Encoding and Representational Geometry

Early work by (Hubel and Wiesel, 1959) showed that neurons in the primary visual cortex act as feature detectors and respond selectively to specific visual stimulus properties. Some neurons in V1 are orientation-selective (e.g., respond to edges of particular orientation), whereas others are sensitive to motion. Later studies demonstrated that information is not encoded by a single neuron alone, but rather distributed across populations of neurons, explaining the robustness and generalization of the neural system. Experiments conducted by Lashley (1950) further suggested that the learned knowledge is distributed across multiple brain regions.

Visual encoding, which underlies object recognition and spatial navigation (Fig. 2.1), is a multi-step process. Information captured by visual apparatus is transmitted through the visual pathway to the primary visual cortex (V1), where the elementary visual features such as shape, orientation, and colors are processed. This information is further propagated through two higher-order streams: the dorsal "where" and the ventral "what" pathways. Research by Wilson et al. (1993) demonstrated that working memory involves these two distinct pathways for object shape composition and semantic interpretation (parietal and infero-temporal sites in the left hemisphere) and for spatial localization (occipital, parietal, and frontal regions in the right hemisphere). This suggests that object detection and recognition rely on distributed activity across multiple specialized brain regions.

Modern neuroscience interprets these distributed representations through the framework of population coding (Pouget et al., 2000). In this view, neural population activity can be represented as a vector in a high-dimensional space, where the activity of each neuron corresponds to one dimension. The collective activity of neural populations forms distributed representations of information. However, neural population activity is often constrained to a much lower-dimensional subspace. Structures that capture the topology and geometry of neural subspaces are referred to as neural manifolds (Cunningham and Yu, 2014). Constrained neural dynamics have been associated with both perceptual processing and cognitive variables (often described by a smaller set of latent dimensions). This suggests the brain does not represent variables as isolated symbolic units but encodes them as trajectories on a structured continuous manifold embedded within a high-dimensional neural space. This interpretation closely resembles modern representation learning approaches in machine learning, where information is encoded as trajectories or clusters within high-dimensional latent embedding spaces.

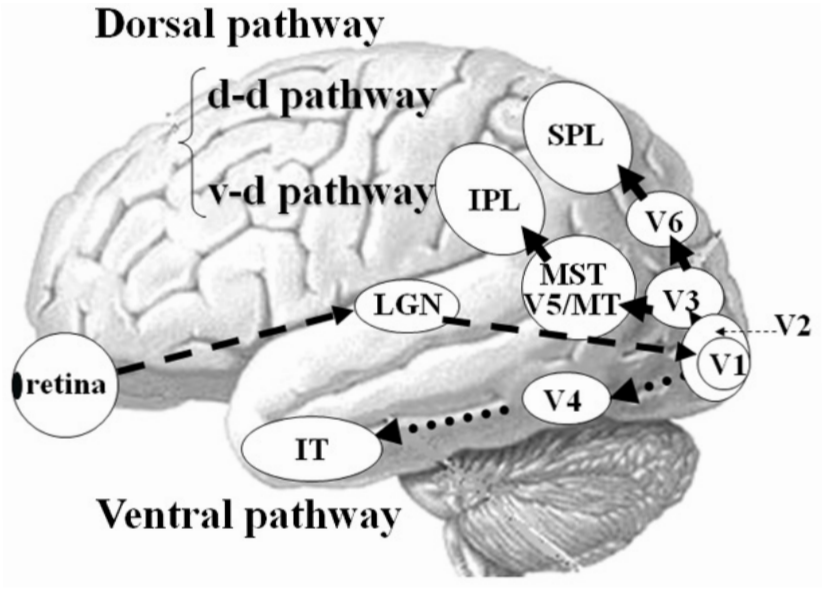
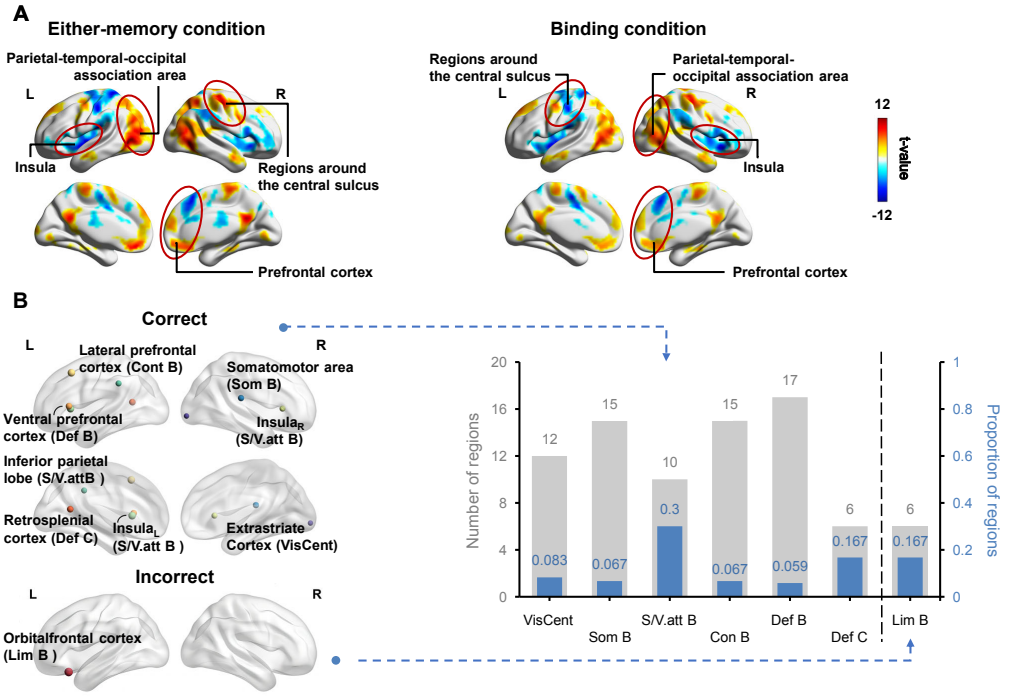


Figure 2.1: Parallel visual pathways in humans. d-d (dorso-dorsal) pathway; v-d (ventro-dorsal) pathway; LGN (lateral geniculate nucleus); V1, V2, V3, V4, and V6 denote visual cortices; V5 visual cortex /MT (middle temporal) area; MST (medial superior temporal) area; IPL (inferior parietal lobule); SPL (superior parietal lobule); and IT (inferior temporal) cortex. Adapted from Yamasaki and Tobimatsu (2011).

2.1.2 Feature Binding and the Integration Problem

How these distributed feature representations are integrated into a unified percept remains an open question known as the binding problem. Despite extensive studies, this mechanism is not completely understood. Hebb (1949) suggested the existence of two forms of neural grouping (i) cell assemblies, neurons that stimulate one another for primary feature encoding and (ii) phase sequences, composed of synchronized assemblies for higher-order concepts. Similarly, Singer (1996) proposed that neural synchrony is crucial for the object representations. Sets of neurons encode individual stimulus features. Temporal synchronization of neural activity binds them into unified object representations. He also proposed different mechanisms for selective cell assembly activation (i) neurons not participating are temporarily inhibited, (ii) response of participating cells is increased and (iii) cells synchronize their firing rate. Other approaches suggest that only subsets of neurons within the assembly dynamically synchronize their discharge rates rather than the entire population (Sakurai and Takahashi, 2008). More recent research by Cao et al. (2026) suggests that binding occurs through the simultaneous participation of neurons distributed across a larger cortical network (prefrontal cortex, insula, somatomotor areas) as illustrated in Fig. 2.2.



VisCent: Visual central network, Som B: Somatomotor B network; S/V.att B: Salience/Ventral Attention B network; Con B: Control B network; Def B: Default B network; Def C: Default C network; Lim B: Limbic B network.

Figure 2.2: BOLD-based neural responses in either-memory and binding conditions, indicating similar activation patterns across distributed cortical regions. Adapted from Cao et al. (2026).

2.1.3 Spatial and Predictive Neural Representations

Research on spatial navigation and mapping suggests that information about the environment is organized into an internal coordinate system. The evidence suggests that the hippocampal pyramidal place cells demonstrate location-specific firing indicating spatial position encoding (O'Keefe and Dostrovsky, 1971), while the dorsocaudal medial entorhinal cortex encodes metric spatial structure (Hafting et al., 2005). This suggests that, for navigation and memory, the brain constructs internal geometric models of the physical space.

Predictive coding, integrates many of the previously discussed principles into unified theory. In this view, neural representations are not passive encodings of sensory stimuli, but dynamically updated latent models used to minimize prediction error through continuous interaction with environment (Millidge et al., 2022). Closely related theories, such as the Bayesian brain hypothesis, propose that perception operates as Bayesian inference and that representations function as predictive latent models of the world. These theories place the brain in an active role during perception, where percepts of the world are continuously updated through prediction and sensory feedback.

2.2 Cognitive Theories of Representation

Mental representations allow us to relate our inner world to the environment and interact with it. These internal symbolic and/or structured representations support reasoning, memory, and cognition. How does the mind encode and manipulates information about the world?

2.2.1 Symbolic, Analog, and Propositional Representations

Thagard in his book *Mind: Introduction to Cognitive Science*, distinguishes four types of mental representations: (i) concepts, (ii) propositions, (iii) rules, and (iv) analogies (used for comparison). A concept represents an entity or a group of entities and refers to more general concrete or abstract ideas (e.g., flower vs. happiness). Relations between concepts form hierarchical semantic networks (e.g., lily vs. flower). Propositions are more complex representations that operate on concepts to form statements that can be evaluated as true or false, closely related to propositional logic. Rules operate over propositions to generate more complex structures and relations. Together, these incremental representational processes contribute to knowledge formation, whether declarative or procedural. In Thagard (2005)'s classification, analogies are particularly important for knowledge generalization and transfer.

On the other hand, Friedenberg and Silverman (2006) propose a more compact classification scheme: symbolic (digital), analog (e.g., visual imagery), and propositional representations. According to the dual-code theory (Paivio, 1979), many concrete concepts can be represented through both symbolic and analog formats. Propositional representations differ from analog and symbolic representations in that they are best described through logical relations and connections. A central property of grounded representations is the meaningful relationship between an internal representation and its referent. Symbolic representations typically refer indirectly to external objects or concepts, whereas analog representations preserve structural similarity (isomorphism) with the phenomena they represent. Another important aspect is the causal relationship between internal representations and the objects or processes to which they refer.

According to the classical cognitive theories, cognition was often modeled as formal symbolic manipulation (Newell and Simon, 1972). However, such models could not account for the representational robustness under noise, perceptual learning, graded similarity or biological plausibility.

2.2.2 Connectionist Models of Cognition

For the purposes of this study, the connectionist approach to cognition (Rumelhart and McClelland, 1986) is more relevant. In contrast to symbolic approaches, connection-

ist models encode knowledge as distributed patterns of activation in neural networks operating in a parallel distributed processing fashion. Rather than relying on explicit symbolic rules, representations emerge through learned weighted connections between processing units.

ANNs are computational abstractions loosely inspired by biological neural systems, where concepts are represented through distributed activity patterns. Learning occurs by adjusting the weights in the network to minimize prediction error. Artificial neurons receive weighted inputs, aggregate them, and apply an activation function to produce outputs. Early work by McCulloch and Pitts (1943) demonstrated that simplified neural models could perform logical operations (OR, AND, NOT). Later, Rosenblatt (1958) introduced the perceptron (Fig. 2.3), an early learning model that adjusted weights based on differences between predicted and desired outputs. Although early neural networks contained only a small number of layers, they were capable of distinguishing simple visual features such as oriented edges.

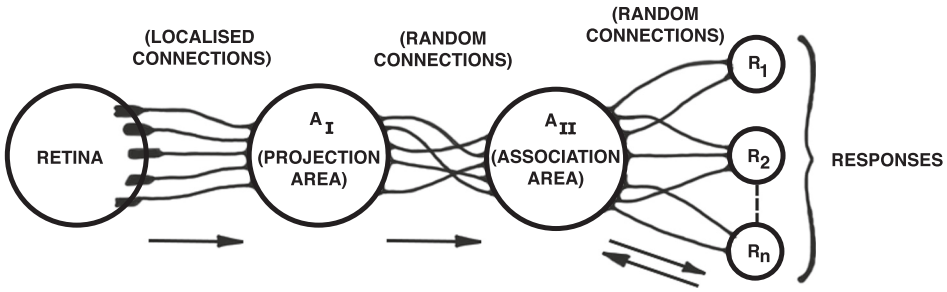


Figure 2.3: The perceptron, a simplified neural model originally proposed by Rosenblatt. Adapted from Nash et al. (2018).

In neural networks, learning generally occurs through iterative minimization of an error signal, where weights are updated to reduce the differences between predicted and target outputs. These models illustrate how learning rise through adaptation and how sematic structure can be encoded in distributed representations. A more detailed overview of artificial learning paradigms and network architectures will be presented in the next section.

2.2.3 Cognitive Models of Visual Representation and Object Recognition

From a cognitive science perspective, visual perception is the process through which organisms interpret sensory input and recognize patterns in the environment. As discussed earlier, light reflected from stimuli is projected onto the retina, and signals are transmitted through visual pathways to cortical regions responsible for visual processing. A central question is how recognition emerges from these signals.

Early Template Matching Theories proposed that stimuli are compared against stored internal templates. However, these models struggle to account for variability and transformation invariance. Feature Detection Theory instead proposed that objects are represented through combination of elementary features. The pandemonium model (Selfridge, 1958) suggests that recognition emerges through hierarchical feature competition and integration, consistent with findings by (Hubel and Wiesel, 1962), showing that neuron in the primary vision cortex respond selectively to oriented edges and simple features.

Treisman and Gelade (1980) proposed the two stage Feature Integration Theory, in which elementary visual features such as color, motion, and orientation are first processed automatically during a preattentive stage and later integrated into coherent object representations through attention mechanisms. A key open issue is how distributed features are integrated into unified percepts. The possible interpretation is that synchronization between the neurons encoding different features serves as a bonding mechanism guided by attention.

Similarly, Biederman (1987) introduced Recognition-by-Components Theory of Pattern Recognition, where objects are represented through combination of simple volumetric primitives (geons). This theory explains how object recognition remains robust under transformations, and noisy sensory conditions.

A major computational framework for understanding vision was proposed by Marr (1982), who described perception as a hierarchical transformation from simple edge detector to increasingly abstract object-centered representations. Visual processing starts with a primal sketch, capturing simple edge structures and basic intensity changes. Further precessing groups visual features according to the properties such as similarity and continuity, producing a viewer-centered 2.5D sketch. Next transformations create an object-centered 3D conscious construct which enables recognition of objects under different conditions and transformations. Fig. 2.4 illustrates MarrNet, a computational model, inspired by the Marr's theory of vision that estimates 2.5D sketches and 3D object shapes (Wu et al., 2017).

2.3 Probabilistic and Simulation-Based Cognition

2.3.1 Mental Imagery and Internal Simulation

Visual imagery represents an important cognitive phenomenon. Friedenberg and Silverman (2006) states that " a visual image is a mental representation of an object or scene that preserves metric spatial information. Visual images are therefore isomorphic to their referents. They preserve spatial characteristics of the objects they represent." This provides insight into how visual information can be internally recon-

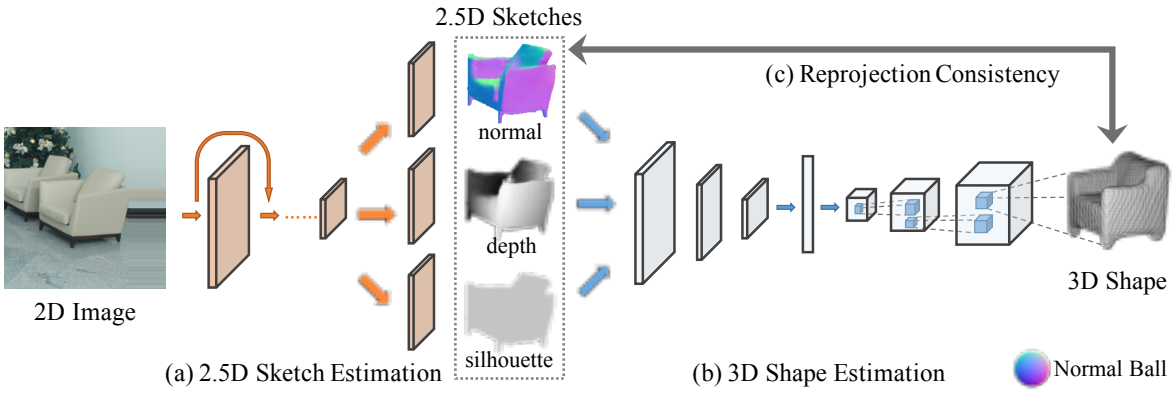


Figure 2.4: Illustration of Marr's theory of visual perception implemented in the MarrNet framework, which incrementally constructs 3D shape from 2.5D sketches. Adapted from Wu et al. (2017).

structed and manipulated without direct sensory input. Kosslyn and Shwartz (1977), hypothesizes that visual imagery relies on a visual-spatial buffer that generates surface representations from long-term memory. Although debated, this theory highlights the reconstructive nature of representational mechanisms, which is consistent with generative theories of cognition, where mental representations are used to reconstruct and simulate perceptual experience based on stored information.

More recent cognitive theories emphasize hybrid representational systems, combining compositional symbolic structure with distributed neural encoding under probabilistic inference. This shift toward probabilistic modeling supports both discrete and continuous representations (Chater et al., 2010). In this view, symbols can be interpreted as discrete latent variables, whereas distributed representations correspond to continuous latent spaces, with cognition understood as inference over structured latent models. This perspective closely aligns with modern representation learning approaches in machine learning, where neural networks learn latent embedding space that organize sensory information according to semantic similarity and statistical structure.

2.3.2 Bayesian and Hierarchical Models of Cognition

A particularly influential framework is the hierarchical Bayesian model of cognition proposed by Tenenbaum et al. (2011). The authors argue that cognition is best understood as Bayesian inference over structured generative models. According to Hierarchical Bayesian models (HBMs), learning arises from interactions between inference, hierarchical representations, and abstraction, as cognition cannot be explained solely through associative learning and memorization. Instead, cognition depends on hierarchical generative models that enable generalization from limited data.

Learning begins with prior knowledge and inductive biases that constrain sensory

inputs and experiences. Similarly, modern machine learning systems rely on architectural inductive biases, such as convolution, attention, and hierarchical structure, to efficiently learn representations from data. Probabilistic inference mechanisms, often formalized through Bayesian models, allow the brain to evaluate competing hypotheses and update beliefs under uncertainty.

Structured representations such as causal graphs, grammars, hierarchies, and relational systems organize knowledge into meaningful forms that support efficient reasoning and transfer learning (Fig. 2.5).

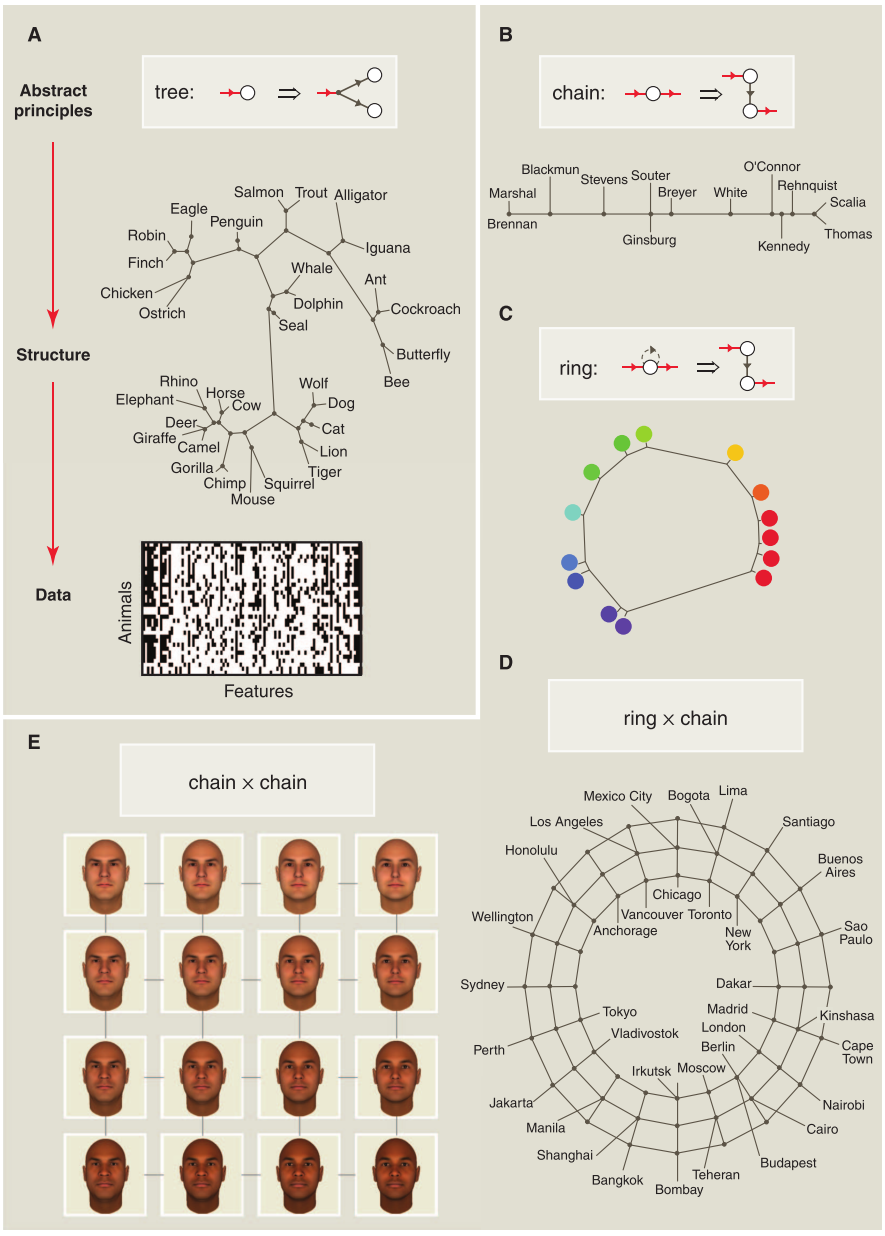


Figure 2.5: Illustration of a hierarchical Bayesian framework that learns both the general structure and the specific relationships needed to represent real-world knowledge from data. Adapted from Tenenbaum et al. (2011).

This framework explains how humans achieve generalization, abstraction and few shot learning by integrating Bayesian inference, structural analogy, and hierarchical representation learning. It suggests that cognition involves abstraction through compression of structure and prediction though the simulation of latent dynamics.

2.4 Machine Learning Perspectives on Representation Learning

The primary goal of representation learning is to transform high-dimensional sensory input into compact and informative latent spaces that preserve semantically meaningful structure while discarding irrelevant, redundant and/or noisy information. This process addresses the phenomenon known as "curse of dimensionality" (Bellman, 1957) where increasing data dimensionality leads to sparse statistical structure, reduced generalization and increased computational cost. An interesting parallel can be drawn with cognitive and neuroscience theories suggesting biological systems also compress and organize high-dimensional sensory data into lower-dimensional internal representations while preserving statistically relevant features of the environment. In machine learning, representations are understood as learned latent variables that encode relevant structural information. These representations are used for classification, prediction, reasoning and control. Throughout the history of Machine Learning (ML), different techniques, both manual and automatic, were used to extract information from raw data and make it suitable for modeling.

2.4.1 Classical Feature Engineering

Historically, ML systems relied heavily on manually engineered feature selection and feature extraction methods. Feature selection such as filtering, wrapping, and embedding involves identifying the most relevant subset of features from the original data, while feature extraction transforms the original input into new representations that preserves the most informative structure (e.g., Principal Component Analysis, autoencoder-based dimensionality reduction, Linear Discriminant Analysis).

Traditional computer vision methods relied on handcrafted feature representations specifically designed by human experts, often inspired by biological systems. Some of the most influential methods were Scale-Invariant Feature Transform (SIFT) (Lowe, 1999), Histogram of Oriented Gradients (HOG) (Dalal and Triggs, 2005) or Local Binary Patterns (LBP) (Ojala et al., 1994). SIFT extracts image features invariant to scale, rotation, and illumination changes. This enables robust object recognition under varying viewing conditions. HOG computes histograms of local gradient ori-

entations and was partly inspired by discoveries in visual neuroscience, particularly the orientation-selective receptive fields discovered by Hubel and Wiesel (1959). LBP captures local texture information by comparing intensities between neighboring pixels and provides robustness to illumination variation. In natural language processing, early techniques mostly relied on manually constructed statistical representations. Bag-of-Words (BoW) model, represents a text as collection of words and their frequencies while ignoring grammatical structure and word order. This method proved to be effective for simple classification tasks, but failed to capture semantic similarity and contextual meaning.

2.4.2 Deep Hierarchical Representation Learning

The development of DL fundamentally transformed representation learning by enabling models to automatically compute hierarchical feature representations directly from data. The prominent "Parallel Distributed Processing" group (Rumelhart and McClelland, 1986; Rumelhart et al., 1986) introduced the backpropagation algorithm used for training neural networks, significantly advanced research in this area. LeCun and Hinton (2015) further developed complex neural network architectures and advanced techniques for efficient automatic feature computation (Fig. 2.6). Unlike handcrafted approaches, deep neural networks learn distributed representations through optimization on large-scale datasets. Lower network layers typically encode local and low-level patterns, while deeper layers capture higher-level semantic information. In these hidden layers, activations form learned feature hierarchies.

Convolutional Neural Networks (CNNs) (LeCun and Hinton, 2015) became the dominant paradigm for visual representation learning. CNNs learn hierarchical image features. It is a multi-step process where through convolutional operations local features are extracted, and pooling mechanism reduces dimensionality while preserving important structural information. The CNN architectures such as AlexNet, VGGNet, ResNet (Krizhevsky et al., 2017; Simonyan and Zisserman, 2015; He et al., 2015) demonstrated that deep hierarchical representations outperform handcrafted feature extraction methods across various computer vision tasks. Importantly, pretrained CNN backbones enabled transfer learning, where representations learned on large datasets could be applied to new tasks and domains.

Embedding-based approaches such as Word2Vec, FastText (Mikolov et al., 2013; Joulin et al., 2016), represent words as dense vectors embedded in high-dimensional semantic space. These techniques preserve semantic relationships between words, allowing the model to capture the context of words. Word2Vec learns word embedding by predicting neighboring words, while FastText additionally considers subwords, capturing language morphology. This was the important conceptual shift in ML as semantic

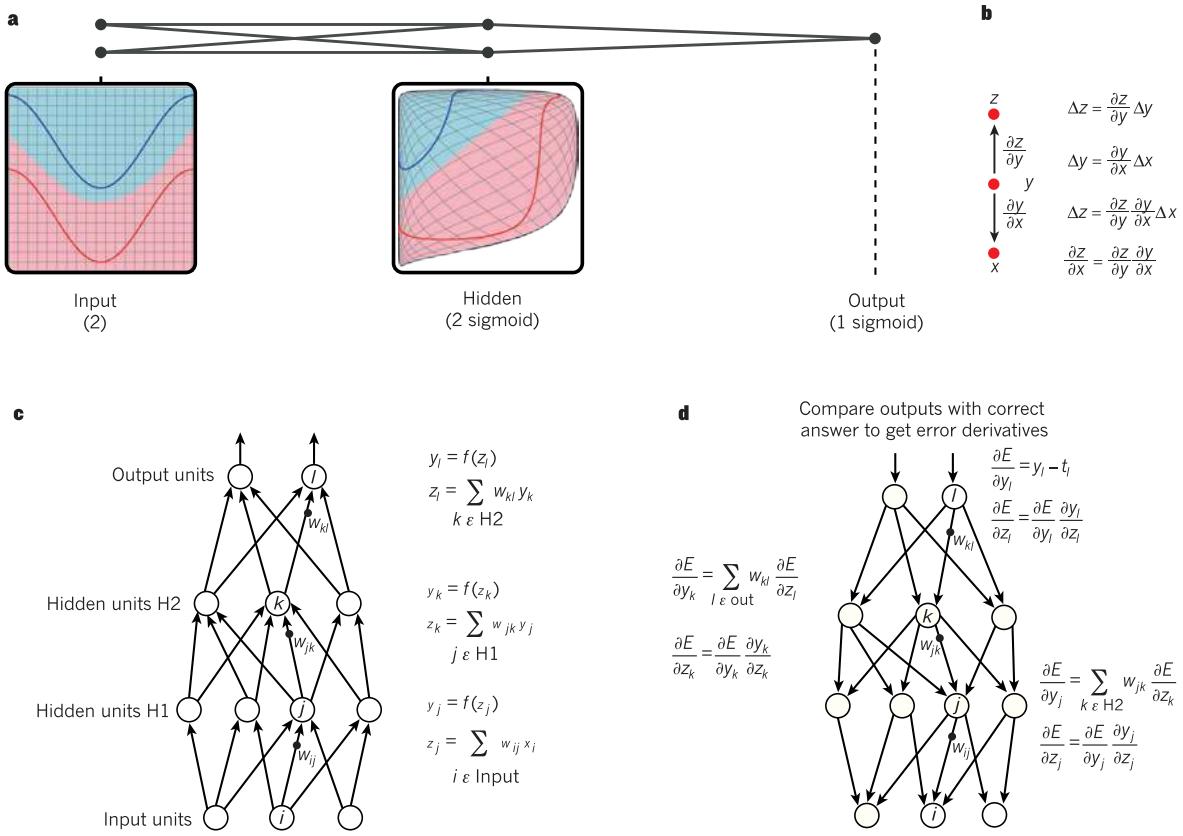


Figure 2.6: Illustration of multilayer neural network and backpropagation. (a) Neural network transforms input space into representations that make classes linearly separable. (b) The chain rule explains how gradients are propagated through composed functions. (c–d) The forward and backward passes compute activations and gradients for updating network weights. Adapted from LeCun and Hinton (2015).

meaning could be geometrically repressented thorough latent vector spaces.

2.4.3 Transformer-Based Contextual Representations

Recent advances in ML mostly rely on transformer-based architectures, including Bidirectional Encoder Representations from Transformers (BERT), ViTs, and Segment Anything Models (SAM) (Devlin et al., 2019; Dosovitskiy et al., 2021; Ravi et al., 2024). Transformers were originally introduced in NLP, through the work of Vaswani et al. (2017). The core mechanism in transformers is self-attention, which computes contextual relationships between elements in an input sequence. Instead of processing information locally, self-attention dynamically determines which parts of the input are most relevant to one another.

The self-attention mechanism is defined as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.1}$$

where Q, K, and V denote the query, key, and value matrices, respectively, and d_k is the dimensionality of the key vectors.

This mechanism enables the model to compute contextual representations in which the meaning of each element depends on its relationship to the entire input sequence. Models like BERT learn bidirectional contextual embeddings capable of capturing semantic ambiguity, syntax, and long-range dependencies.

The transformer architecture was later adapted to computer vision through ViTs (Dosovitskiy et al., 2021) Instead of using convolutional filters, ViTs divide an image into fixed-size patches that are treated as sequential tokens. Each patch is embedded into a latent space and processed through stacked self-attention layers. Unlike CNNs, which capture features locally, Vision Transformers learn globally contextualized representations. This allows ViTs to model long-range spatial dependencies, object relations and to capture scene structure. Representations learned by ViTs show semantic organization in latent embedding space, and a parallel could be drawn from human hierarchical perception.

2.4.4 Self-Supervised Representation Learning

While supervised learning relies on manually labeled datasets, biological learning systems primarily learn from unlabeled sensory experience. This limitation motivated the development of Self-Supervised Learning (SSL) where representations are learned directly from the inherent statistical structure of input data. For instance, autoencoders (Hinton and Salakhutdinov, 2006), learn compressed representations using reconstruction-based techniques. Similarly, Generative Adversarial Networks (GANs) learn useful representations by modeling the underlying data distribution via adversarial training between a generator and a discriminator networks (Goodfellow et al., 2014).

More recently, self-supervised vision foundation models such as the DINO (Distillation with No Labels) family (Caron et al., 2021; Siméoni et al., 2025) introduced representation learning frameworks based on invariance and consistency across multiple views of the same input. These models learn stable latent representations by enforcing similarity between embeddings of augmented versions of the same image. DINO employs a teacher-student architecture in which the student network is trained to match the representations produced by a momentum-updated teacher network across different augmented views. Interestingly, such models are capable of learning perceptual structures, including object-centric representations, semantic clustering, and attention maps, without direct supervision, while remaining robust to distribution shifts in the input data. These properties motivated our decision to utilize the DINO family within our framework.

2.5 Biological Foundations of Memory

To be able to process information and preserve it, a memory capacity capable of storing it is needed. This mechanism enables not only learning, but also survival in dynamic environments, allowing organisms to learn from past experiences and adapt to new situations. Without memory, organisms would be unable to accumulate experience, recognize patterns, form concepts, or predict future events. Memory and intelligence are deeply intertwined, and both are connected to the concept of representations. In order to preserve knowledge, retrieve it when needed, and update it, information needs to be distilled and efficiently encoded. Mechanisms supporting these capabilities have been intensively studied for a long time, yet the general principles are still not fully known. In neuroscience, memory is understood as persistent changes in neural connectivity and activity patterns that encode past experience. In cognitive science, memory is modeled as structured mental representations that support recall, reasoning, and prediction. However, contemporary approaches increasingly view memory not as passive storage, but as an active representational process. In machine learning, memory emerges as stored information within learned parameters, latent embeddings, and contextual representations. A unified perspective across these fields is that memory reflects the persistence and reuse of representations.

As already mentioned in previous chapters when discussing continual learning and information encoding at the cellular level, the work of (Hebb, 1949) stands as a hallmark in our understanding of neural plasticity. Gazerani (2025) identifies four key mechanisms supporting neuroplasticity: (i) synaptic plasticity, (ii) structural plasticity, (iii) neurogenesis, and (iv) functional reorganization. Synaptic plasticity, understood as the modulation of synaptic strength, is supported by two opposing mechanisms: long-term potentiation (LTP) and long-term depression (LTD) (Bliss and Cooke, 2011). This adaptive form of neural connectivity emerges during skill acquisition while also helping maintain excitatory-inhibitory balance. At the cellular level, changes involve synapses and dendrites, including chemical and metabolic modifications (Fig. 2.7). Structural plasticity, as described by Gazerani (2025), "reflects large-scale reorganization of neural circuits and connectivity patterns." For example, this can be observed during memory consolidation, when neuronal circuits in the hippocampus and prefrontal cortex exhibit coordinated oscillatory activity that supports the stabilization of newly acquired information. In this way, experience is gradually organized into structured representations. Dynamic network changes are associated with synaptic reorganization and, in some cases, longer-term structural plasticity such as axonal sprouting (Kelly and Castellanos, 2014). Functional reorganization is often studied in the context of brain injury and refers to the brain's ability to redistribute functions across different regions, with other areas taking over tasks that were previously supported elsewhere.

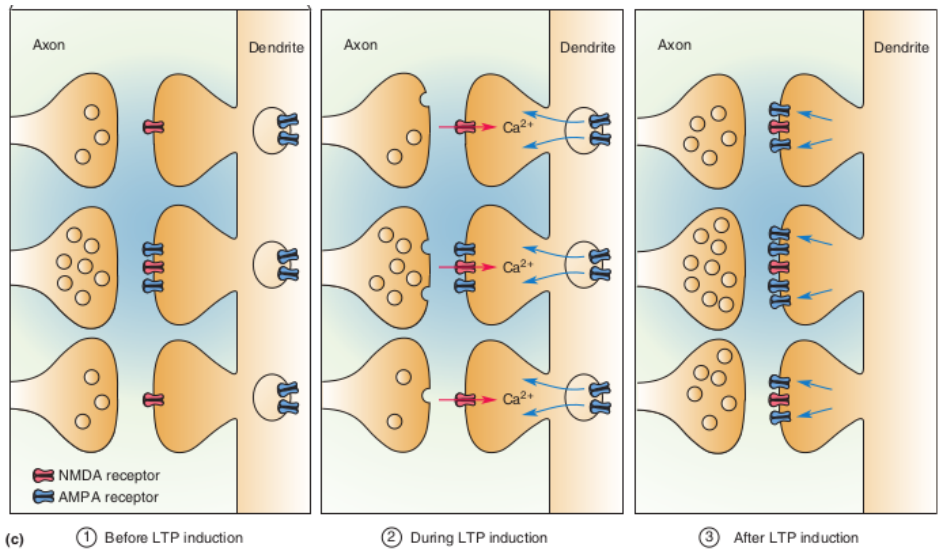


Figure 2.7: Insertion of AMPA receptors into the postsynaptic membrane during long-term potentiation (LTP). Adapted from Bear et al. (2006).

2.6 Cognitive Architectures of Memory

From a more hierarchical perspective, cognitive neuroscience has identified functionally distinct types of memory, typically differentiated by retention time, capacity, and the nature of the information being processed. Sensory memory stores raw sensory information for a very brief period of time, usually a few hundred milliseconds. Its role is to preserve information about a stimulus long enough to be encoded (Darwin et al., 1972). Fig. 2.8 illustrates different brain regions involved in memory preservation. Short-term memory (STM), also referred to as working memory, has a more limited capacity than sensory memory but retains information for a longer period, typically on the order of seconds. In contrast to sensory memory, it supports active cognitive processing. Encoding studies have shown different results depending on modality and task demands, including visual and acoustic encoding effects (Conrad, 1964; Shepard and Metzler, 1971), while some findings suggest semantic encoding as well (Wickens, 1973).

Long-term memory (LTM) refers to the ability to retain information over extended periods of time, thereby enabling learning. It can be further divided into distinct types, including procedural (implicit) memory and declarative (explicit) memory. Procedural memory, refers to knowledge of how to perform actions and operates largely unconsciously. Declarative memory, in contrast, refers to consciously accessible knowledge and is further divided into semantic memory (factual knowledge) and episodic memory (personal experiences). Research suggests that implicit memory is strongly associated with the cerebellum and related subcortical structures, whereas explicit memory is distributed across cortical regions.

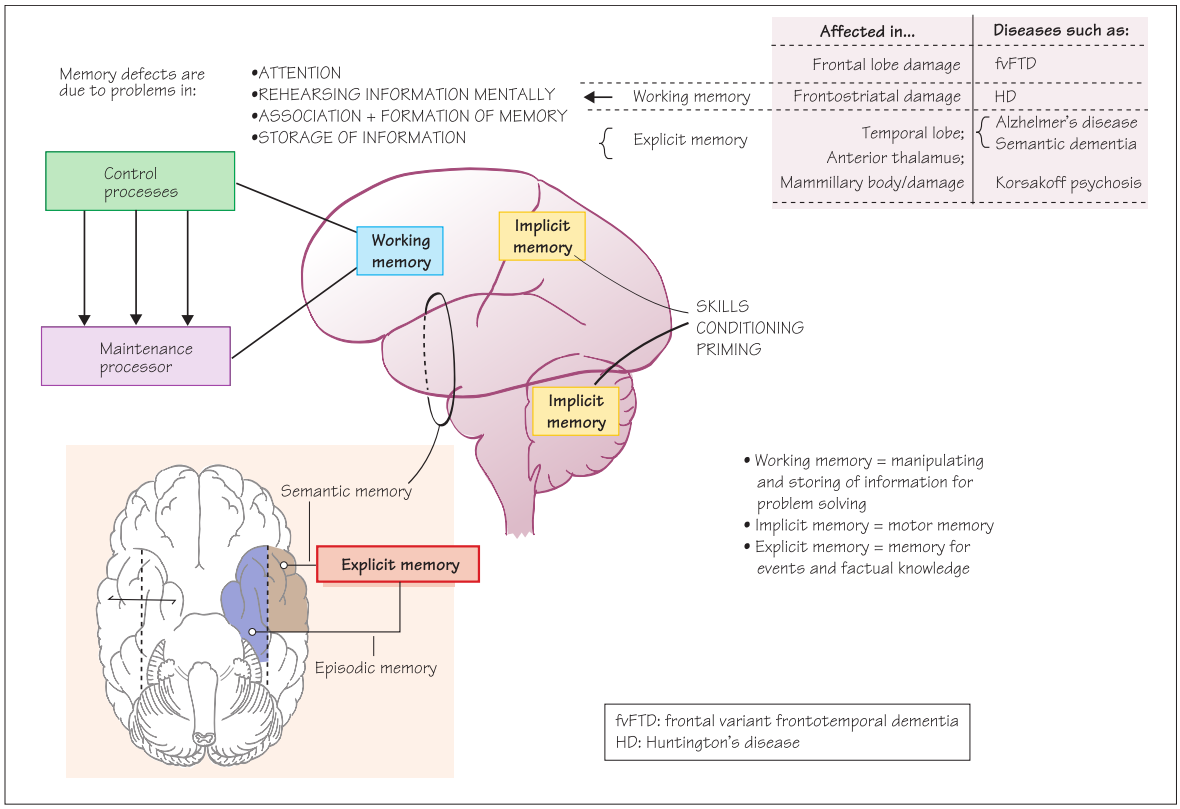


Figure 2.8: Different types of memory and the associated brain areas: working memory, implicit memory, and explicit memory. Adapted from Barker et al. (2012).

How working memory, memory consolidation work and how attention modulates memory? Reverberatory circuits can partially provide an explanation (Hebb, 1949). Specialized neuronal pathways create feedback loops that allow sustained activity in a network, even after the initial stimulus is no longer present. Such persistent activity enables temporary maintenance of information over time. This mechanism could explain how information is preserved and later recalled through ongoing network dynamics.

2.6.1 Models of Cognitive Control

Cognitive science also offered a perspective how knowledge is preserved and retrieved. Atkinson and Shiffrin (1971) proposed the modal model, in which information flows sequentially from sensory memory to short-term memory and then to long-term memory. In this framework, information is transferred between working memory and long-term memory for encoding and retrieval. Anderson (1990) proposed a more comprehensive model in which memory is described as an interaction between stored knowledge and task execution, rather than as a strictly sequential set of storage stages.

Baddeley (1992) proposed a working memory model in which visual and auditory processing are treated as separate subsystems under the control of a central executive. He extended the concept of short-term memory into a system of multiple interacting

components responsible for active information manipulation rather than passive storage. Empirical studies provide partial support for this model, as distinct brain regions are activated for verbal and visual processing. Verbal storage and executive functions are associated with parietal and frontal regions of the left hemisphere, while visual processing is linked to occipital, parietal, and frontal regions in the right hemisphere.

Schema Theory, on the other hand, suggests that memory is an active process supported by how the brain structures knowledge (Piaget, 1959; Bartlett and Burt, 1933). Schemas, as organized structures of knowledge, store both declarative and procedural information. They are considered dynamic and are continuously updated through new information acquisition. Bartlett and Burt (1933) suggests the active and reconstructive nature of memory, rather than a static reproduction of the past.

2.7 Memory Mechanisms in Artificial Systems

Memory-based approaches in ML have been greatly inspired by discoveries and theories from neuroscience and cognitive science. Among the most important are biologically inspired principles of working memory as active representation, attention-guided reasoning, and associative memory retrieval. In DL, memory is partially encoded in network weights, where the compressed statistical structure of extracted data is preserved.

The idea of reverberatory circuits served as a model for how sequential information could be captured. Recurrent Neural Networks (RNNs) were built on the principle of recurrent units, a form of memory where the previous hidden state is repeatedly passed through the computational node. These models struggled with the vanishing gradient problem, but a major breakthrough came with the Long Short-Term Memory (LSTM) architecture (Hochreiter and Schmidhuber, 1997). It introduced memory cells capable of remembering values over long time sequences, while gates regulate the information flow. This approach revolutionized speech recognition. To capture contextual information, Bidirectional Recurrent Neural Networks (BRNNs), composed of two RNNs, were introduced (Schuster and Paliwal, 1997). They consist of layers operating in opposite directions, providing information flow in both forward and backward directions.

The classical Hopfield network (Hopfield, 1982), also known as associative memory, is a type of recurrent neural network consisting of symmetrically and bidirectionally connected nodes within the same layer (Fig. 2.9). When trained using Hebbian learning principles, it represents a form of content-addressable memory. In the following chapter, this type of network and its more advanced variants will be presented.

The most significant shift in recent years came with transformer architectures, which can capture long-range dependencies by employing the attention mechanism explained in the previous section. Advances achieved through training on large-scale datasets

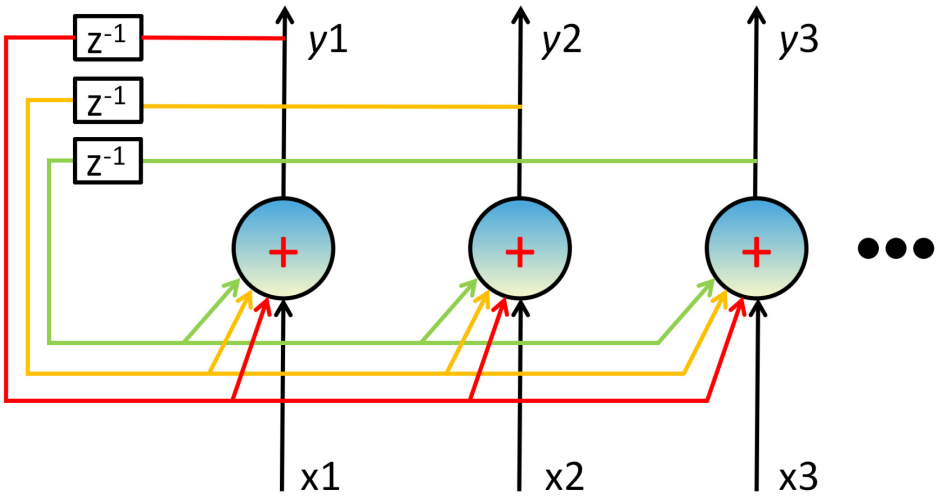


Figure 2.9: Illustration of a Hopfield neural network, a recurrent neural architecture used for associative memory and pattern retrieval. Adapted from Zheqi et al. (2022).

have initiated a new wave of progress, from LLMs to large foundation vision models, and even more advanced systems integrating different modalities. Combined with self-supervised learning, this enables the learning of stable object representations and global contextual dependencies (Siméoni et al., 2025). A conceptual parallel can be drawn between DINO and biological memory formation, where representations are compressed and structured via memory consolidation mechanisms, while view consistency is preserved. The teacher-student update mechanism can be related to consolidation-like processes, and semantic abstraction is achieved through clustering in embedding space.

Chapter 3

Associative Memory and Categorization Learning

3.1 Associative Memory Mechanisms

Associative memory represents the ability to learn and retrieve connections between unrelated items (e.g., objects, concepts) and create binding units and paths (Suzuki, 2008). This process is crucial for learning, allowing the brain to create coherent representations from distributed and noisy sensory inputs, while in same time functioning as an error-correction mechanism (Krotov et al., 2025). For example, in the binding problem mentioned in the previous chapter, simple features, context, location, and structure of visual stimuli are combined into unified representations. This enables detection and recognition of objects in parallel, but also inference and relation formation, including both episodic and semantic knowledge. How the brain integrates multi-modal representations and retrieves them through association is still part of ongoing research (Wang and Cui, 2017; Wang, 2019). From a computational perspective, associative memory is viewed as content-addressable retrieval, studied in the reconstruction of stored memories from incomplete or noisy inputs (Fig. 3.1).

3.1.1 Neurocognitive Foundations of Association

Historically, associative memory was mostly studied in the context of classical and operant conditioning. Classical conditioning (Pavlov and Anrep, 1927) describes how unconditioned and conditioned stimuli are associated, effectively binding two stimuli together, where the strength of the connections is a function of the time between the stimuli occurrence. On the other hand, operant conditioning studied how an action or response is associated with a stimulus, usually through reward mechanisms (Thorndike, 1898; Skinner, 2019).

Neural plasticity and associative memory have been closely related in many stud-

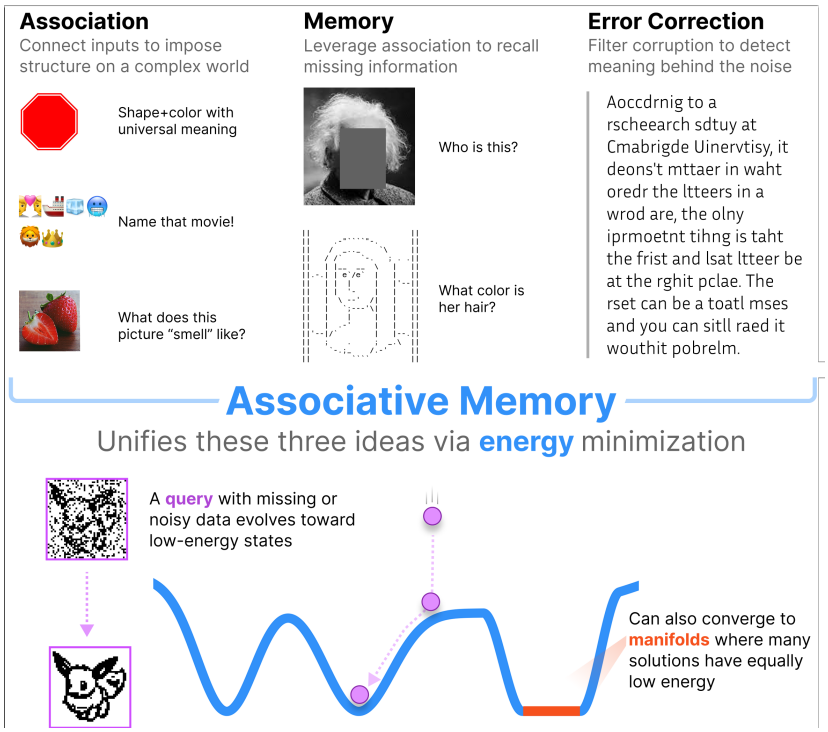


Figure 3.1: In Hopfield networks, associative memory is represented through an energy landscape in which stable low-energy states correspond to stored patterns and enable memory retrieval from partial or noisy inputs. Adapted from Krotov et al. (2025).

ies (Fanselow and Poulos, 2005). Hebb (1949) suggested that cell assemblies (neurons with highly weighted synaptic connections) could represent the building blocks of encoded concepts. Short-term memory can be viewed as evoked sequential firing within assemblies, while formed synaptic connections represent long-term associations.

Temporal synchronization theories suggest that neural assemblies representing different features may be bound through phase-locked oscillatory activity, particularly in theta (4-8 Hz) and gamma frequency bands (30 Hz and above) (Hasselmo et al., 2002; Takeuchi et al., 2015). Hasselmo et al. (2002) suggest that the hippocampal theta rhythm supports learning and reversal by alternating between phases optimized for encoding new associations and retrieving old ones. Changes in synaptic connections rely on fine-tuned timing in cell assemblies in theta frequency (Clouter et al., 2017).

These mechanisms explain how storage, consolidation, and retrieval occur within intramodal brain regions, but they cannot fully account for the interpretation of cross-modal memory formation. Wang (2019) suggests that the brain encodes associative information in specialized cell assemblies known as Associative Memory Cells (ACMs) (Fig. 3.2). ACMs can be divided into two groups: primary pAMCs responsible for encoding associated signals, and secondary sAMCs, which support cognitive processes such as imagination, associative thinking, and decision-making. ACMs encode new sensory signals together with initial ones, as synapses are modulated through co-activated

pathways (co-active neurons and formation of synaptic interconnections).

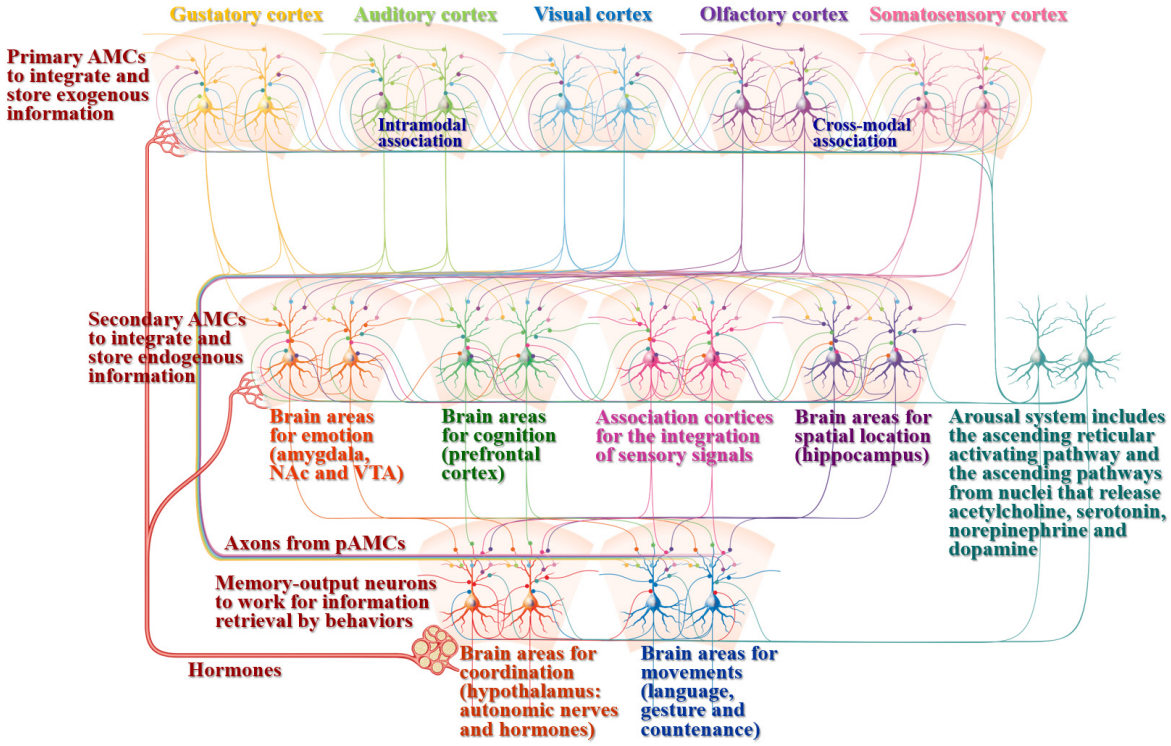


Figure 3.2: Hierarchical organization of associative memory cells (ACMs), responsible for the integration and storage of exogenous and endogenous signals, and memory retrieval. Adapted from Wang (2019).

Associative memory is supported by a distributed network including the medial temporal lobe, particularly the hippocampus and adjacent cortices, parietal-hippocampal interactions, as well as the prefrontal cortex (Wagner et al., 2005; Squire et al., 2007). Studies of "place" cells and "grid" cells, which encode visuospatial information, show that they are active during navigation and memory retrieval (Miller et al., 2013). Interestingly, as already mentioned in the previous chapter, these regions are also strongly involved in memory consolidation and formation of episodic memory (Li et al., 2010).

Contemporary cognitive neuroscience frames memory as an active, reconstructive, and predictive process, moving from classical views of memory as static storage. In this framework the hippocampus has a role in generative model formation (Pezzulo et al., 2017), supporting the Bayesian brain hypothesis (Doya et al., 2011) and perceptual inference (Rao and Ballard, 1999). Clark (2013) suggests that generative models are updated through prediction error signals. This might result in dynamic associative plasticity changes.

3.1.2 Cogntive Representation Binding

Dual coding theory (Paivio, 1979), mentioned in the previous chapter, suggests that when information is encoded in multimodal form, it is better preserved. It can be explained by different pathways supporting retrieval or visual imagery, which encode information in a more vivid form, closely connected to mnemonic effects of mental imagery.

Spatial memory, evolutionarily developed through interaction and existence in the physical environment, is highly robust and can be effectively utilized for spatio-temporal information encoding. The method of loci represents a mnemonic strategy where established visuospatial routes are used to encode new information. During retrieval, these paths are mentally retraced (Worthen and Hunt, 2011). This suggests that information is structured and interconnected. Another way to study associative memory contribution is through semantic networks. A common way to symbolically represent concepts is as a network of associations (Quillian, 1968). In this way, knowledge is organized in structurally hierarchical connected graph representations. Another way to approach associative memory formation is through chaining memory, which connects information in a story-like manner. This resembles how episodic memory and associative memory are closely connected, supporting discoveries in neuroscience. Attentional theories propose different perspective. In contrast to the bottom-up theories, feature binding emerges through top-down selection mechanisms that dynamically route feature information into integrated object representations (Treisman, 1986).

3.1.3 Mathematical Models of Associative Memory

Computational models of associative memory, focused on bioplausible learning for pattern retrieval, have been widely studied in neural networks (Amari, 1972; Cohen and Grossberg, 1983). Predictive coding networks were able to store and reconstruct images using noisy cues (Salvatori et al., 2021). This suggests that Hebbian learning can be interpreted as error minimization between inputs and network-generated predictions, while retrieval, implemented through inferential neural dynamics, may employ the same error-minimization mechanism. Additional work by Tang et al. (2023) introduced recurrent connections into predictive coding frameworks for associative memory. These covariance-learning predictive coding networks (covPCNs) increased the biological plausibility of such models by incorporating recurrent associative dynamics.

For our interest, one of the most famous mathematical models of associative memory formation and retrieval was proposed by Hopfield (Hopfield, 1982), where memories are stored as point attractors of the network dynamics. In this framework, recurrent connections are often interpreted as learning a covariance association matrix between neurons co-activated by a memory item, providing a simplified model of associative

memory in hippocampal circuits. In this framework, large assemblies of neurons receive multidimensional synaptic input through an interconnected population of neurons, where Hebbian synaptic plasticity rules govern the strengthening of associations.

3.1.4 Energy-Based Associative Memory Models

In associative memory networks, input and output representations are mapped based on stored pattern similarity, meaning that they are a content-addressable type of memory. The Hopfield model is biologically inspired, as it uses local synaptic updates, stores memory in a distributed way across weights, and retrieves patterns through network dynamics (Hopfield, 1982). It consists of one layer of interconnected nodes, as shown in Fig. 2.9. Every node is connected to all others except itself. The connections between nodes are stored in a weight matrix W, which is square and symmetric, with $w_{ii} = 0$. The symmetry condition guarantees the existence of an energy function that always decreases during updates, converging to a stable state corresponding to local minima (attractor states), which are not necessarily all valid stored patterns, since spurious attractors may also occur (e.g., combinations of stored patterns, inverted versions, or random stable states).

Following the classical Hopfield formulation (Hopfield, 1982), let I_i be the external input to neuron i. Then the total input (local field) to node i is given by

$$h_i = \sum_{j \neq i} w_{ij} \xi_j + I_i. \tag{3.1}$$

The network is auto-associative, as the same units serve as both input and output. Neuron states are updated dynamically according to

$$\xi_i \leftarrow \text{sign}(h_i), \tag{3.2}$$

and converge to low-energy states, which represent attractor states corresponding either to stored patterns or spurious stable configurations.

Learning in the network follows the Hebbian learning principle, where connections between co-activated neurons are strengthened during the learning process. A standard Hebbian rule for storing P patterns $\{\xi^\mu\}_{\mu=1}^P$ is given by

$$w_{ij} = \frac{1}{N} \sum_{\mu=1}^{P} \xi_i^\mu \xi_j^\mu, \quad i \neq j, \tag{3.3}$$

with $w_{ii} = 0$. (Normalization is taken as $1/N$, although other conventions such as $1/P$ also appear in the literature.)

In classical Hopfield networks, neurons can take only two states (typically $\xi_i \in \{-1, 1\}$), and the neuron state is updated according to the sign of the local field. Each

stored pattern is represented by a bipolar vector ξ^μ. In this case, the weights can be interpreted as correlations between pattern components, since

$$w_{ij} \propto \sum_\mu \xi_i^\mu \xi_j^\mu. \tag{3.4}$$

This means that memorized patterns are encoded in the synaptic connections between neurons.

For recall, the input to every node is computed via the local field h_i, and each neuron updates its state according to the sign of this value. In this way, the network iteratively evolves until it converges to a fixed point, corresponding to a stored pattern or an attractor state.

Under asynchronous updates (only one neuron is updated at a time), the network's energy function decreases monotonically, ensuring convergence to a stable fixed point.

Hopfield (1982) defined the memory landscape by the following function:

$$E = -\frac{1}{2}\sum_{i,j} w_{ij}\xi_i\xi_j + \sum_i \theta_i \xi_i. \tag{3.5}$$

From the structure of the weight matrix (often learned via Hebbian learning), neurons that agree with strong positive weights reduce the energy, while neurons that disagree increase the energy of the network. With the neuron update rule, each change in a unit's state either decreases or leaves unchanged the value of the function.

The limitations of classical Hopfield networks are that they have limited storage capacity (approximately $0.138N$ patterns), and as the number of stored patterns increases, the distinguishability of individual memories decreases due to interference between patterns. This limitation was later addressed by the introduction of Dense Associative Memories or Modern Hopfield Networks, with modifications to the energy function and the introduction of continuous-valued states in the network (Krotov and Hopfield, 2016; Ramsauer et al., 2021).

3.1.5 Modern Hopfield Networks

Dense Associative Memories (Krotov and Hopfield, 2016) extend classical Hopfield networks by introducing stronger nonlinearities and higher-order interactions between neurons. These modifications produce sharper attractor basins, improve the separation between stored memories, and reduce the occurrence of spurious states. The corresponding nonlinear energy function is given by

$$E = -\sum_{\mu=1}^{N_{\text{mem}}} F\left(\sum_{i=1}^{N_f} \xi_{\mu i}\xi_i\right), \tag{3.6}$$

where $\xi_{\mu i}$ denotes the i-th component of the μ-th stored memory pattern, ξ_i represents the current state of neuron i, and $F(\cdot)$ is a rapidly increasing nonlinear function. The

asynchronous neuron update rule is defined as

$$\xi_i^{(t+1)} = \text{sign}\left[\sum_{\mu=1}^{N_{\text{mem}}}\left(F\left(\xi_{\mu i} + \sum_{j\neq i}\xi_{\mu j}\xi_j^{(t)}\right) - F\left(-\xi_{\mu i} + \sum_{j\neq i}\xi_{\mu j}\xi_j^{(t)}\right)\right)\right]. \tag{3.7}$$

Compared to classical Hopfield networks, these higher-order nonlinear interactions significantly increase the memory storage capacity and improve retrieval robustness. Unlike classical Hopfield networks, modern formulations operate on continuous vector representations stored in a memory matrix (X).

In their work, Ramsauer et al. (2021) introduce the Modern Hopfield Network with a differentiable energy function (modified from Demircigil et al. (2017)):

$$E = -\text{lse}\Big(\beta, X^\top\xi\Big) + \frac{1}{2}\xi^\top\xi + \beta^{-1}\log N + \frac{1}{2}M^2. \tag{3.8}$$

The energy consists of two competing terms. The first term, based on log-sum-exp similarity, encourages the state ξ to align with stored memory patterns in X and amplifies contributions from highly similar patterns. The second term is a quadratic regularization that prevents unbounded growth of ξ and stabilizes the dynamics of the system. The additional constant terms do not affect the dynamics.

The corresponding update rule is given by

$$\xi_{\text{new}} = X\,\text{softmax}\Big(\beta X^\top\xi\Big). \tag{3.9}$$

This update can be interpreted as first computing similarities between the current state and the stored patterns via $X^\top\xi$, then converting these similarities into normalized attention weights using the softmax function, and finally reconstructing the state as a weighted combination of stored memory patterns via multiplication with X.

Thus, the output corresponds to a weighted average of stored memory vectors, biased toward the most similar patterns.

This formulation enables Modern Hopfield Networks to be incorporated into deep learning architectures as differentiable memory layers, allowing the storage and retrieval of continuous vector representations such as embeddings. The update typically converges to a fixed point in a single step under suitable conditions, with exponentially small retrieval error in high-dimensional settings.

An important result is that the storage capacity can scale exponentially with the dimensionality of the representation space under appropriate assumptions. Moreover, the update rule is mathematically equivalent to key-value attention in Transformer architectures. In this correspondence, the current state ξ plays the role of the query, X corresponds to keys and values, and the softmax weighting implements an energy-based selection mechanism. This establishes Transformers as a special case of continuous Hopfield networks.

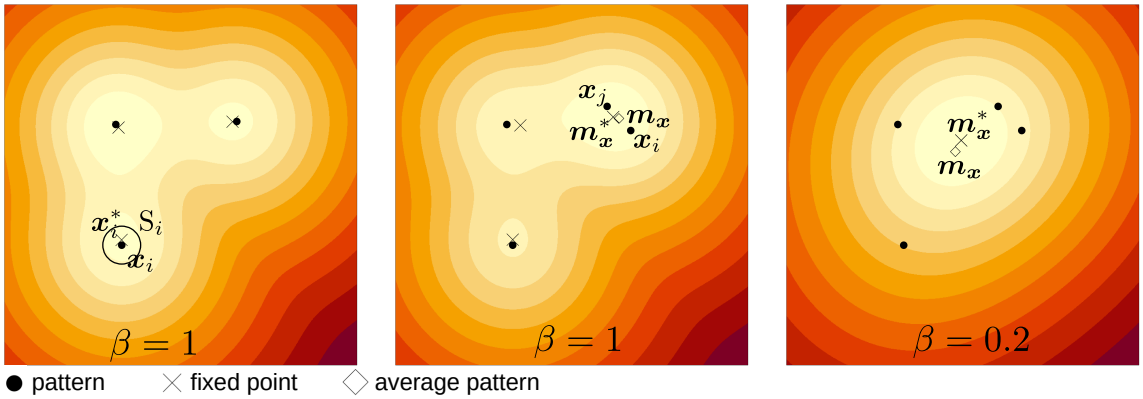


Figure 3.3: Three cases of system stable states: (a) stored patterns, where the fixed point corresponds to a single stored pattern; (b) metastable state, where the fixed point corresponds to an average of similar patterns; (c) global fixed point, where the fixed point corresponds to the average of all patterns. Adapted from Ramsauer et al. (2021).

As the system can support multiple stable states, this is reflected in its attractor dynamics (Fig. 3.3). When memories are well separated, modern Hopfield networks retrieve individual patterns; when they overlap, the system instead converges to either a global average or metastable cluster states representing groups of similar patterns.

3.2 Category learning

How do humans categorize objects and ideas, and how are concepts and mental structures formed? According to Minda et al. (2024), different strategies are employed, some are previously learned (verbal) rule-based strategies, while others rely on more unconscious decisions based on salient features of examples or previous experience. Different theories have been proposed regarding how categorization learning occurs, and this remains an ongoing debate. Some theories assume that we employ a single system (Nosofsky and Johansen, 2000), while others suggest two different systems (Ashby et al., 1998), one for language, reasoning, and executive functions and other for associations and similarities between examples

3.2.1 Single-System Models of Categorization

Cognitive psychology proposes two main classical categorization models (Bowman et al., 2020). Prototype theory argues that we categorize objects by comparing stimuli to an averaged "ideal" representation of a category in memory (Rosch, 1973; Minda and Smith, 2001), whereas exemplar theory suggests that we compare stimuli to many stored instances of each category. This is formalized in the Generalized Context Model

(GCM) and ALCOVE (Nosofsky, 2011; Kruschke, 1992).

Both models rely on similarity-based comparison processes, but they differ in how category representations are stored, accessed in memory, and in their sensitivity to variation and context. Prototypes are memory-economical representations based on the central tendency of experienced category members, whereas exemplar models represent categories as collections of individual stored instances. Exemplar models are more computationally demanding but preserve within-category variability, which allows them to better account for atypical members and variable category structures, and to generalize to natural concepts. This helps explain how diverse items, such as different flowers, can still be categorized under the same "flower" concept. Intensive research of these categorization models declines more to the exemplar view of categorization as a single-system categorization, as the prototype approach needs support from exemplars or other systems (Nosofsky and Johansen, 2000). On the other hand, the boundary-based approach assumes that categorization is based on learning and applying decision rules that separate categories in a structured feature space (Ashby and Gott, 1988). Fig. 3.4 illustrates the main approaches to single-system categorization.

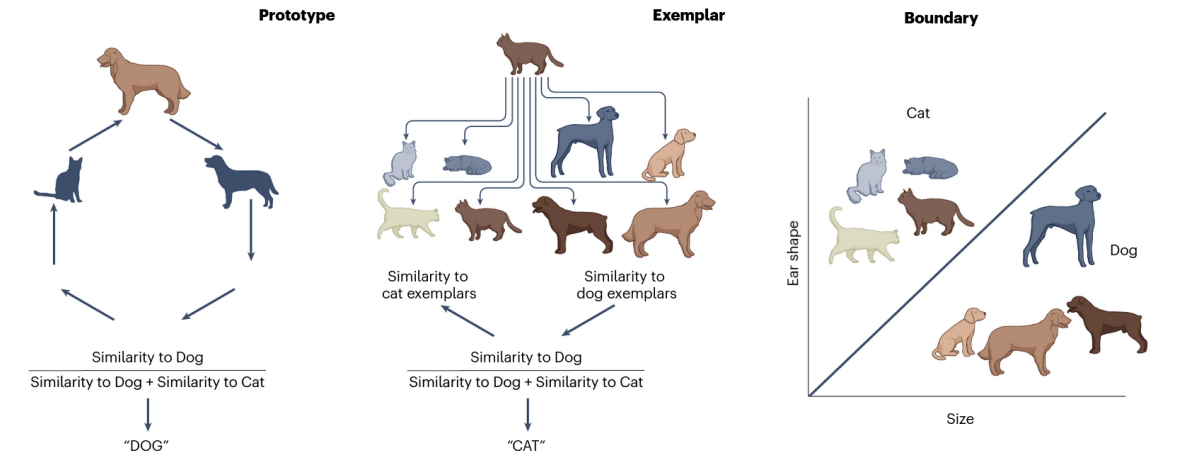


Figure 3.4: Single-system approaches to categorization: (a) prototype models, (b) exemplar models, and (c) boundary models. Adapted from Minda et al. (2024).

Some researchers suggest that both strategies might be employed: exemplars may contribute to the initial learning of categories and help preserve within-class variability, while prototype representations may become dominant with experience and support more abstract categorization. On the other hand, other accounts suggest that exemplar-based processing remains important even with increasing experience. The hybrid approach proposes that the use of these strategies depends on context, category complexity, and familiarity with the stimuli.

3.2.2 Hybrid and Multiple-System Models of Categorization

The hybrid system proposed by Demircigil et al. (2017), named the Dual Prototypes and Exemplars based Conceptual Categorization System (DUAL PECCS), improves categorization performance compared to classical single-system models. Also, some recent neuroimaging studies support a hybrid approach (Bowman et al., 2020; Ashby and Zeithamova, 2022).

Although hybrid systems describe multiple processes, they do not necessarily specify whether distinct brain systems are involved. In contrast, multiple-system theories such as the COVIS model (Competition between Verbal and Implicit Systems) (Ashby et al., 1998; Ashby and Valentin, 2005) links two neural systems for explicit (verbal) and implicit learning. The first system is rule-based and verbal, and relies on the prefrontal cortex and medial temporal lobe, whereas the second system detects statistical patterns automatically through a error-feedback mechanism.

The neural underpinnings of the COVIS system involve the prefrontal cortex, hippocampus, cingulate cortex, occipital cortex, caudate nucleus, and premotor cortex (Fig. 3.5). Learning often begins with the explicit system, which is conscious and rule-driven, whereas information integration in the implicit system develops more gradually with experience.

Some studies of fMRI brain activity have shown that feedback-based learning (implicit system) relies more on basal ganglia/striatal circuits, whereas rule-based learning seems to involve different networks such as frontal areas (Nomura et al., 2007; Cincotta and Seger, 2007). Later findings show overlap in activation patterns across shared networks (Carpenter et al., 2016).

3.2.3 Typicality, Exemplar Effects, and Behavioral Dynamics

An important question in categorization research is how category members come to represent a concept and influence categorization decisions. Smith and Medin (2002) argue that typical examples are the ones that share the most characteristics, and that categorization occurs when stimuli are similar to these typical examples. Typicality is supported by faster retrieval, whereas less typical examples require additional time for memory search (Reisberg, 2013). Their work also suggests a positive correlation between the typicality of an exemplar and its frequency of occurrence. In exemplar categorization, accuracy is strongly influenced by the recency factor (categorization of stimuli presented soon after the example was encountered) and priming of exemplars. Hampton (2015) suggests that prototypes are not the most typical exemplar, but sees them as a set of correlated features.

Rouder and Ratcliff (2006) discovered that in categorization people may use two mechanisms: one based on rules, where a set of rules is learned in the initial stage,

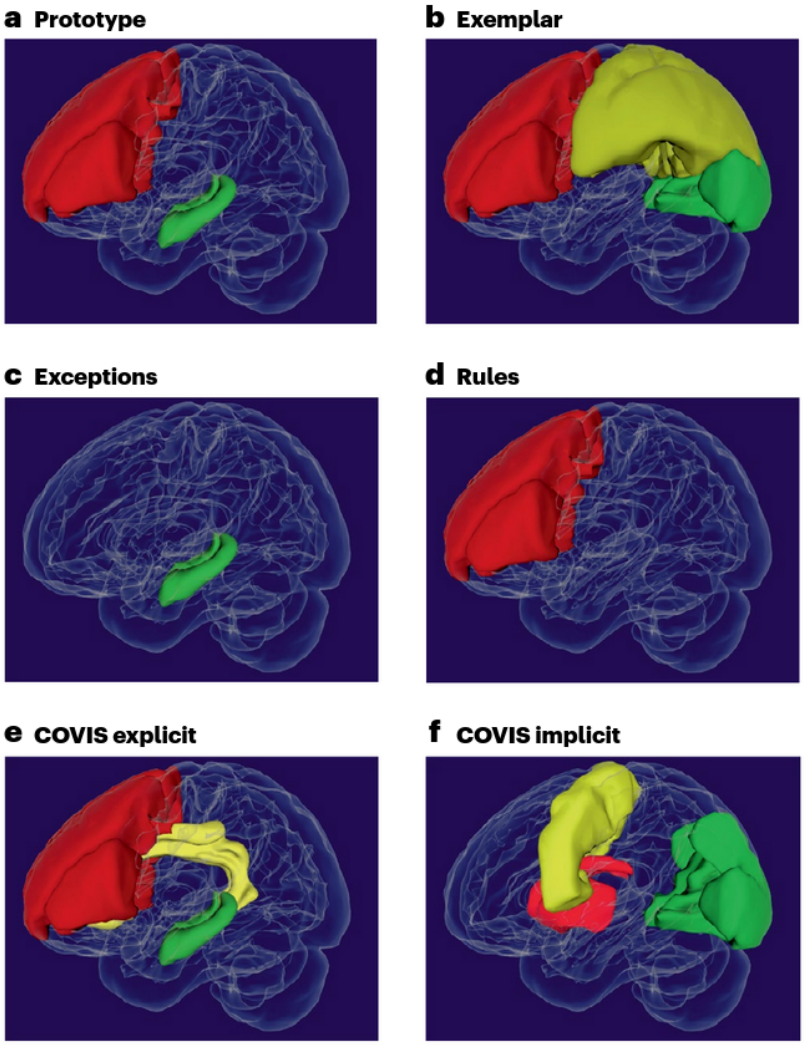


Figure 3.5: Hypothesized neural structures of different categorization models. (a) Prototype models are associated with the prefrontal cortex and hippocampus. (b) Exemplar models are associated with the prefrontal, parietal, and occipital cortices. (c) Exception learning in mixed models is associated with the hippocampus. (d) Rule learning is associated with the hippocampus and prefrontal cortex. (e) The explicit system of the COVIS model is associated with the prefrontal cortex, hippocampus, and cingulate cortex. (f) The implicit system of the COVIS model is associated with the occipital cortex, caudate nucleus, and premotor area. Adapted from Minda et al. (2024).

and another based on exemplars, which becomes more prominent as new examples are stored and distinctive features are learned. This aligns with both rule-based and exemplar-based systems.

With exemplar models, scientists can explain the typicality effect on categorization time, category concepts, and the correlation between variability and categorization time. Smith and Medin (2002) also found that people use examples both for affirming and revising decisions about categorization.

Barsalou et al. (1998) studied event exemplars and individual exemplar differences, suggesting a distinction between summarized event-based exemplar grouping and individual categorization processes.

Inspired by the presented research, in our study we investigate how the rich semantic representational power of self-supervised foundation models, together with exemplar-based and prototype-based incremental memory construction, can support associative categorization and classification. We further examine whether MHN-based retrieval and refinement mechanisms improve classification accuracy and memory organization in comparison to cosine similarity retrieval.

Chapter 4

Methodology and Experimental Setup

4.1 Our method

Motivated by biological continual learning systems that maintain both stable and adaptable representations through memory consolidation and incremental update, we propose a memory-based framework for incremental and open-set visual recognition. The proposed approach combines large-scale self-supervised visual representations with an associative memory mechanism (Ramsauer et al., 2021; Siméoni et al., 2025). We use a pretrained DINOv3 encoder for feature extraction and a memory module parameterized as either class prototypes or class exemplars inspired by cognitive category theories (Nosofsky and Johansen, 2000; Bowman et al., 2020). The pipeline consists of: (i) feature extraction from object regions, (ii) memory construction from labeled support embeddings, (iii) incremental memory updates that allow new classes to be integrated without retraining from scratch, and (iv) open-world inference with optional MHN-based query refinement and unknown recognition (Fig. 4.1).

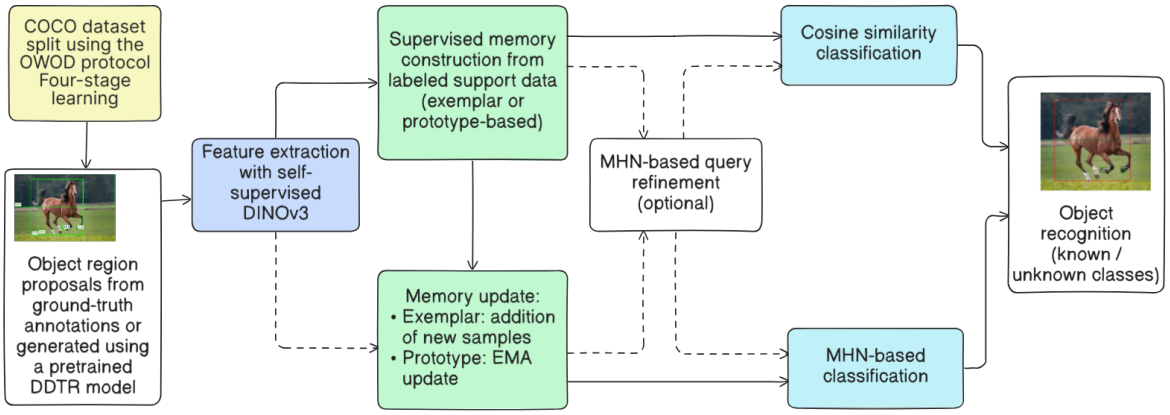


Figure 4.1: Illustration of the proposed incremental open-set learning pipeline.

4.1.1 Memory Construction

Given an input image and corresponding object regions, obtained either from ground-truth (GT) annotations or from a Deformable DETR (DDETR) detector pretrained on the COCO dataset (Lin et al., 2015; Zhu et al., 2021), each region is encoded using a pretrained self-supervised DINOv3 backbone (Siméoni et al., 2025). In our main experiments, regions are taken from GT annotations. Although the present work focuses on region-level recognition using object crops, the framework is designed to support future integration with Deformable DETR used as a class-agnostic detector within complete OWOD pipelines.

The encoder produces a feature embedding $z \in \mathbf{R}^d$, which is L2-normalized before storage:

$$\hat{z} = \frac{z}{\|z\|_2}. \tag{4.1}$$

In our experiments the embedding dimensionality is $d = 768$.

The memory is defined as

$$M = \{(p_j, c_j)\}_{j=1}^{N_p},$$

where p_j denotes a stored embedding and c_j its associated class label. We consider two memory representations:

Prototype-based memory, where each class is represented by a compact set of prototypes initialized from support embeddings, either as k-means centroids on the unit sphere or as the support embeddings themselves.

Exemplar-based memory, where support embeddings are stored directly without aggregation, preserving instance-level detail at the cost of increased memory usage.

This design provides a trade-off between compact semantic abstraction and fine-grained instance preservation.

4.1.2 Support Selection and Threshold Calibration

For each known class c, let

$$\mathcal{D}_c = \{(z_i, b_i)\}_{i=1}^{N_c}$$

denote the set of GT training crops, where $z_i \in \mathbf{R}^d$ is the L2-normalized DINO feature embedding and b_i is the corresponding bounding box.

Support crops are selected in embedding space, similarly to metric-based prototype learning approaches (Snell et al., 2017). Crops whose cosine similarity to the class mean is below a threshold τ_{supp} (default 0.15) are removed as outliers. On the selected pool, spherical k-means is applied to the L2-normalized embeddings. For each cluster, the crop whose embedding has the highest similarity to the centroid is retained, with at most one crop per source image. The intermediate support set enforces diversity

and image-level constraints before memory construction. The support set for class c is

$$\mathcal{S}_c \subset \mathcal{D}_c, \qquad |\mathcal{S}_c| \leq n_{\text{supp}}. \tag{4.2}$$

Features in $\mathcal{S}_c$ initialize the class memory items

$$\mathcal{P}_c = \{p_{c,1}, \ldots, p_{c,M_c}\},$$

using either spherical k-means centroids on the support embeddings (prototype mode, with M_c set by configuration) or direct exemplar selection (exemplar mode, $M_c = |\mathcal{S}_c|$).

For open-set recognition, global confidence thresholds are estimated on the training image set used for calibration, following confidence-based rejection strategies commonly used in open-set recognition (Bendale and Boult, 2016). For each calibration embedding $\hat{z}_j$, the model predicts the class c_j^* with the highest memory score and computes two confidence signals: the maximum similarity between the query and any memory item of the predicted class (s_{top1}), and the margin between the predicted class score and the highest competing class score (s_{margin}). When MHN classification is enabled, c_j^* is the MHN-selected class, but s_{top1} and s_{margin} are still computed from cosine similarities evaluated at that selected class.

A query is rejected as unknown unless

$$s_{\text{top1}}(\hat{z}_j) > \tau_{\text{top1}} \quad \wedge \quad s_{\text{margin}}(\hat{z}_j) \geq \tau_{\text{margin}}. \tag{4.3}$$

Under the default *known-only* calibration, τ_{top1} and τ_{margin} are set to fixed low percentiles (default percentile 5) of the corresponding confidence scores measured on calibration crops with known GT labels only. Unknown GT crops do not influence threshold estimation. For computational efficiency and fair comparison, calibration is capped at 10 000 stratified crops using a fixed random seed. This provides a practical sample size for estimating low-percentile thresholds while keeping calibration cost manageable. The same calibration cap and seed are used across prototype, exemplar, and MLP tracks. The harmonic mean of known-class accuracy and unknown recall is then computed on the calibration batch for analysis and model comparison. The resulting thresholds are stored in the memory checkpoint and reused during inference.

4.1.3 Incremental Memory Update and Online Refinement

To support continual learning, memory is updated incrementally as new labeled samples become available. When samples from a *new* known class first appear, that class is added using the static construction procedure of Sec. 4.1.1. After a class is present in memory, *online refinement* may be applied to further refine or expand its prototype set.

Let $\hat{z}$ denote the L2-normalized embedding of an incoming sample assigned to class c, and let $\mathcal{P}_c$ denote the current set of class prototypes.

In prototype-based memory, the incoming sample is first matched to the most similar prototype:

$$p^* = \arg\max_{p \in \mathcal{P}_c} p^\top \hat{z}, \tag{4.4}$$

with similarity score

$$\text{sim} = p^{*\top} \hat{z}. \tag{4.5}$$

If

$$\text{sim} > \tau_{\text{update}},$$

the matched prototype is updated using an Exponential Moving Average (EMA):

$$p^* \leftarrow \text{norm}_2 \left((1 - \alpha) p^* + \alpha \hat{z} \right), \tag{4.6}$$

where α is the update rate. Unless otherwise specified, we use

$$\alpha = 0.1, \qquad \tau_{\text{update}} = 0.7, \qquad \tau_{\text{new}} = 0.5.$$

If

$$\text{sim} < \tau_{\text{new}},$$

the embedding $\hat{z}$ is appended as a new prototype, increasing the number of prototypes associated with the class.

For intermediate similarities,

$$\tau_{\text{new}} \leq \text{sim} \leq \tau_{\text{update}},$$

the memory remains unchanged to avoid unstable updates.

During online refinement, updates are applied only to crops whose cosine similarity to the mean of the class prototypes exceeds a minimum threshold τ_{online}; unless otherwise specified, $\tau_{\text{online}} = 0.15$.

In exemplar-based memory, each accepted update appends $\hat{z}$ directly (no EMA or τ_{update} / τ_{new} logic). Online refinement may still filter crops based on cosine similarity to the class memory mean.

This update strategy enables continuous adaptation while preserving previously learned representations. For incremental open-world tasks, memory can be resumed from a prior checkpoint and updated on newly introduced classes.

4.1.4 Inference

At inference time, query embeddings may optionally be refined using a Modern Hopfield retrieval mechanism over the stored memory items (Ramsauer et al., 2021).
Given a query embedding $\hat{z}$ and memory matrix P,where $P \in \mathbf{R}^{N_p \times d}$ contains the stored memory embeddings stacked as rows, attention weights are computed as:

$$w = \text{softmax}(\beta \hat{z} P^\top), \tag{4.7}$$

where β controls retrieval sharpness.
A memory-based representation is then retrieved as:

$$r = wP. \tag{4.8}$$

The retrieved representation is combined with the original query through a single Hopfield-style update:

$$\tilde{z} = \text{norm}((1-\lambda)\hat{z} + \lambda r), \tag{4.9}$$

where λ controls the influence of retrieved memory. Unless otherwise specified, $\beta = 30$ and $\lambda = 0.5$.

If refinement is enabled, classification operates on $\tilde{z}$; otherwise, the original embedding $\hat{z}$ is used:

$$q = \begin{cases} \tilde{z}, & \text{if MHN refinement is enabled,} \\ \hat{z}, & \text{otherwise.} \end{cases} \tag{4.10}$$

For cosine-based classification, the predicted class is assigned by aggregating cosine similarities within each class:

$$s_c = \max_{p \in \mathcal{P}_c} q^\top p, \qquad \hat{y} = \arg\max_c s_c, \qquad \text{conf} = s_{\hat{y}}. \tag{4.11}$$

Cosine similarity is used as a simple baseline to isolate the contribution of memory design and MHN-based mechanisms. Similar nearest-prototype evaluation is common for DINO-style representations, but is typically reported on image classification benchmarks (e.g., ImageNet) in a closed-set setting rather than under open-world recognition protocols.

For MHN-based classification, class scores are obtained by a separate attention pass over all memory items on q:

$$w = \text{softmax}(\beta\, q P^\top), \qquad s_c = \sum_{j \in \mathcal{M}_c} w_j, \qquad \hat{y} = \arg\max_c s_c, \qquad \text{conf} = s_{\hat{y}}. \tag{4.12}$$

Since $\sum_j w_j = 1$, conf is the attention mass assigned to the predicted class. $\mathcal{M}_c = \{j \mid c_j = c\}$ denotes the indices of memory items belonging to class c.

Following the open-world setting described earlier, a predicted class $\hat{y}$ is accepted as known only if the class-aware gating signals in Eq. (4.3) are satisfied. The same open-set rule is used for cosine-based and MHN-based classification; only the assignment of $\hat{y}$ and conf differs. Predictions failing these conditions are assigned to the unknown class.

4.2 Experimental protocol

4.2.1 Dataset and Protocol

We follow the standard OWOD task-wise split setting (Gupta et al., 2022) on the COCO dataset (Lin et al., 2015), with four incremental stages ($t \in \{1, 2, 3, 4\}$). In each stage, twenty new classes are introduced. We report results on the corresponding test partitions. Using the OWOD protocol on COCO also allows future integration with existing detection-based benchmarks. Task 1 contains 450 103 train instances, while the cumulative tasks contain 896 782. Figure 4.2 illustrates 80 classes grouped in 12 super-categories.

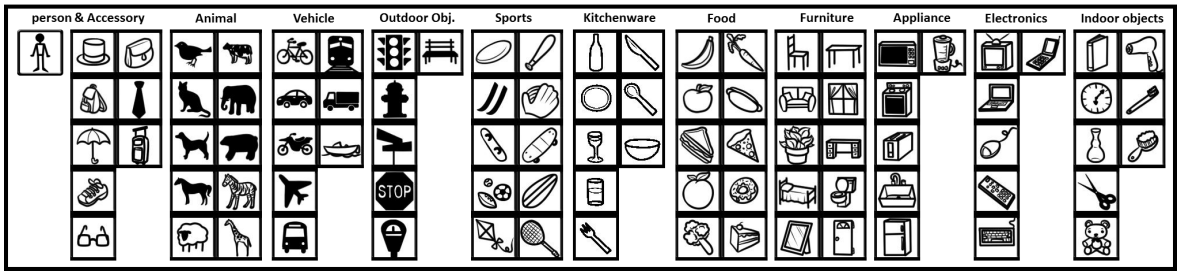


Figure 4.2: COCO dataset: 80 OWOD classes grouped into 12 super-categories.

We use ground-truth object bounding boxes as input regions and do not evaluate object proposal or detection performance. All methods are evaluated at the region level, where each ground-truth object crop is treated as an independent recognition instance.

For future experiments, we plan to use a Deformable DETR (Zhu et al., 2021) model pretrained on the COCO dataset to generate object proposals. The detector will be used in a class-agnostic manner, where only objectness scores are used for proposal filtering.

We evaluate the effect of memory representation (prototype vs. exemplar), classifier type (cosine vs. MHN), and optional MHN-based query refinement. Prototype-memory experiments further vary support size, prototype initialization (k-means centroids vs. direct support protoypes), and online-update hyperparameters (τ_{update}, τ_{new}, α). Unless stated otherwise, prototype memory uses three k-means centroids per class ($M_c = 3$).

Performance is evaluated under the OWOD incremental protocol. We report known-class accuracy, unknown recall (U-Recall), and the harmonic mean H of these two quantities. For memory builds, we optionally report memory size (total number of stored vectors) and the number of successful online updates. At Task 4, all 80 COCO categories have been introduced, so no unknown GT classes remain in the evaluation split. Unknown recall and the harmonic mean H are therefore not computed and we report only known-class accuracy for Task 4.

4.2.2 Experimental Setup

We evaluate configurations varying along three axes: (i) memory representation (prototype vs. exemplar), (ii) classification strategy (cosine similarity as a baseline vs. MHN-based classification), and (iii) optional MHN-based query refinement.

This results in a total of eight main configurations, summarized in Table 4.1. All configurations are first evaluated under the OWOD protocol on Task 1. The best-performing setup is then further evaluated across the full incremental sequence of Tasks 1-4.

Row	**Memory**	**Classifier**	**Refinement**
1	Prototype	Cosine (baseline)	-
2	Prototype	MHN	-
3	Prototype	Cosine	Yes
4	Prototype	MHN	Yes
5	Exemplar	Cosine (baseline)	-
6	Exemplar	MHN	-
7	Exemplar	Cosine	Yes
8	Exemplar	MHN	Yes

Table 4.1: Evaluated configurations across memory representation, classifier type, and optional MHN-based query refinement.

All experiments use frozen DINOv3 (Siméoni et al., 2025) features extracted from object regions. Memory is built from labeled support crops (n_{supp}) selected by embedding-space k-means, using either: (i) prototype-based aggregation (k-means centroids or direct support prototypes), or (ii) direct exemplar storage.

During memory construction, support crops are selected independently of the representation strategy. The strategy determines only how the selected supports are stored in memory.

Unless stated otherwise, we keep the encoder, data splits, and extracted features fixed across all experiments, and vary only memory and inference configurations. Key hyperparameters include the number of support crops (n_{supp}), online-update thresholds (τ_{update}, τ_{new}, α), and open-set gates (τ_{top1}, τ_{margin}) calibrated on the task train split. For MHN, the key parameter is the associative sharpness β, while refinement weight $\lambda = 0.5$ is kept constant unless stated otherwise. On the standard COCO OWOD benchmark, we report the incremental protocol across tasks T1-T4 for the selected prototype and exemplar tracks. For every experiment, a more detailed setup is presented in the result tables.

4.3 Aim

The goal of the experimental study is to analyze how memory representation, associative retrieval mechanisms, and refinement influence incremental open-set recognition using frozen self-supervised visual representations.

4.4 Implementation

The experiments were implemented in Python using the PyTorch framework. The codebase was developed as part of our OWOD memory project and includes modules for detection, encoding, memory construction, online memory updates, open-set calibration, and Modern Hopfield Network (MHN) based classification and refinement.

The data preparation and evaluation pipeline follows the OWOD protocol and was adapted from the OW-DETR framework of Gupta et al. (2022). Experiments were conducted on the COCO dataset (Lin et al., 2015) using the standard task splits and evaluation setup.

Experiment execution and evaluation were automated through dedicated experiment scripts for both single-task (Task 1) and incremental cross-task (Tasks T1-T4) evaluation settings. The final experiment configurations were documented in the separate experiment file (`OWOD_EXPERIMENTS.md`), which records the run identifiers, memory settings, calibration settings, and evaluation modes used for the reported results.

4.5 Data and Embedding Analysis

To better understand the experimental results, we first analyze dataset statistics. In the COCO-based OWOD splits, the class-wise instance distribution across incremental tasks T1 to T4 is imbalanced, mainly due to the *person* class, which dominates the remaining categories. It contains significantly more instances than all other classes, as shown in Fig. 4.3.

Most other classes form relatively stable subsets with much lower instance counts. Only *car* and *chair* show moderately higher frequencies, but both remain far below *person*. We analyze in the experiments how this imbalance is reflected in class-wise accuracy.

We also explore DINOv3 embeddings. L2-normalized crop features are projected with t-SNE and UMAP to assess class-wise separability and whether similar categories cluster together. As an illustration, Fig. 4.4 shows the 20 classes introduced in task T_2.

The projection indicates that some classes are already well separated in feature space (e.g., tennis racket, frisbee), whereas animal classes such as giraffe and zebra form closer clusters. This suggests that DINO captures both object appearance and broader

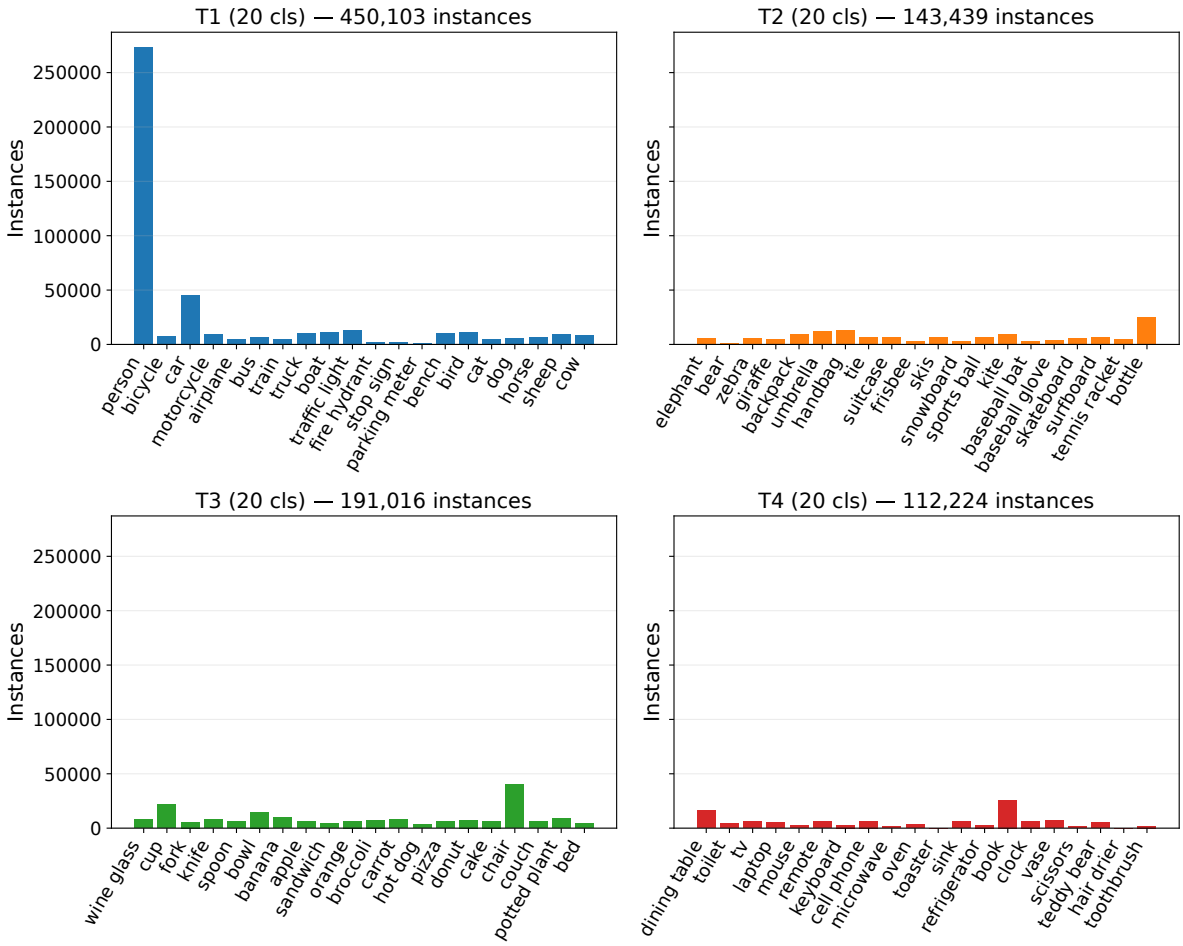


Figure 4.3: Distribution of class instances across tasks T1-T4 in the COCO dataset. We show class-wise instance distributions and imbalance ratios.

semantic structure, with animals lying closer to one another than sports-equipment categories. These properties support our use of normalized DINO features for incremental memory construction.

We further examine how language models encode COCO class names in an auxiliary analysis. We embed the 80 class labels and construct a semantic network. For visual emphasis only, node size is scaled by class instance frequency (Fig. 4.5). We use Sentence-BERT, a modification of BERT that produces semantically meaningful sentence embeddings (Reimers and Gurevych, 2019).

As expected, we observe similarity-based conceptual grouping: animals, furniture, sports, and food categories form distinct clusters. Interestingly, *person* lies near the center of the network and is connected to many other concepts.

In the following sections, we study memory construction, incremental updates, and MHN-based inference for open-set recognition. Within each experimental stage, the best configuration (selected by harmonic mean H) is carried forward as the basis for the next stage.

Figure 4.4: t-SNE projection of L2-normalized DINOv3 embeddings for the 20 COCO classes from task T 2.

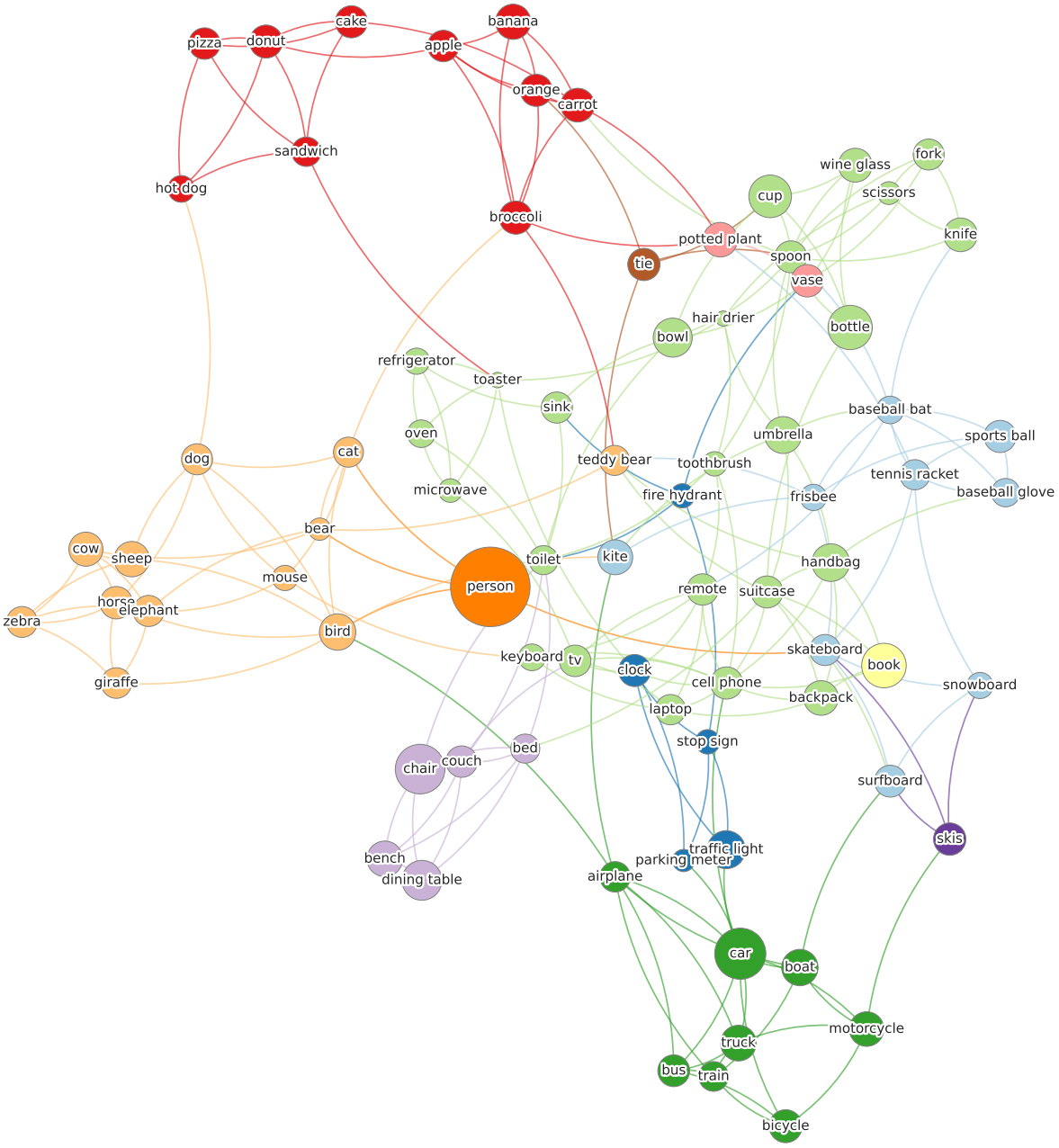


Figure 4.5: Semantic network of the 80 COCO class labels constructed using Sentence-BERT embeddings (Reimers and Gurevych, 2019). Node sizes reflect class instance frequencies and are used only for visual emphasis.

Chapter 5

Experiment 1: Memory Construction and Update Strategies

In this experiment, inspired by cognitive theories of categorization and concept formation, which investigate whether concepts are represented by clustered summaries of examples or by retaining a small set of individual instances (Minda and Smith, 2001; Hampton, 2015), we test how different memory granularities affect our model performance. We study different memory constructions and update strategies, and how these choices affect open-set object recognition on Task 1 (20 known COCO classes). All runs use cosine similarity for classification in order to isolate the effect of memory construction and update dynamics. MHN-based retrieval, refinement and threshold tuning are evaluated separately in subsequent experiments.

5.1 MLP Baseline

Before building and tuning prototype or exemplar memory, we establish the baseline P0-01 on Task 1. We use frozen DINO crops, the standard Task 1 train/test splits, and the same open-set evaluation protocol as in the memory-based runs (class-aware top-1 and margin gates calibrated on the train set). The goal is to assess how well a purely parametric model performs on open-set recognition and how it compares to our proposed framework.

P0-01 trains a two-layer MLP on frozen DINO features (256-dim hidden, 15 epochs, batch 512, lr 10^{-3}) to predict the 20 Task 1 known classes. Open-set gates are fit on the train set using the same procedure as in the memory-based setting, but there is no memory bank and no nearest-neighbor retrieval.

P0-01 reaches *known* accuracy of 91.3% and *unknown* recall of 25.4%, giving $H = 0.398$.

5.2 Experiment 1.1: Prototype Memory Construction

As a first step, we investigate prototype-based memory representations, where each class is represented by a small number of representative vectors. We examine how different prototype construction strategies affect open-world object recognition performance. Memory is constructed as described in Section 4.1.1. Each known class is represented by $K = 3$ prototype vectors built from n_{supp} support crops. At test time, classification uses cosine similarity, and open-set rejection applies the gates in Eq. 4.3.

Our goal is to quantify how (i) prototype initialization, (ii) online update budget, and (iii) gate hyperparameters affect the trade-off between known-class accuracy and unknown recall on OWOD Task 1.

All runs in this section use prototype mode only, without MHN retrieval during classification, and operate on pre-extracted embeddings from ground-truth training crops. Evaluation is performed on the standard test split. Two runs repeat the $n_{\text{supp}} = 32$, $K = 3$, k-means configuration using alternative open-set gate modes (Mahalanobis-based gating and global-prototype margin gating, respectively). These runs are reported alongside Stage 1.1 but are not part of the initialization study.

After selecting the best initialization configuration (highest harmonic mean H), we enable online refinement using the remaining training crops. Prototype updates follow the online update rules described in Section 4.1.3. Table 5.1 lists parameters held constant unless explicitly varied in a given stage. The prototype study is organized into three stages (Table 5.2), where the best-performing configuration from each stage is used as the reference configuration for the subsequent stage.

Table 5.1: Fixed hyperparameters for Experiment 1.1 (prototype memory).

Parameter	**Value**
OWOD task	Task 1 (20 known classes)
Prototypes per class K	3
Support outlier filter	$\tau_{\text{supp}} = 0.15$
Open-set gates	τ_{top1}, τ_{margin} (train-split calibration)
Default τ_{update} / τ_{new}	0.7 / 0.5
EMA coefficient α	0.1
Online filter	$\tau_{\text{online}} = 0.15$
Classifier	Cosine (no MHN)

5.2.1 Stage 1.1: Prototype Initialization

We first isolate the effect of memory initialization by disabling online refinement. Each class is represented by $K = 3$ prototypes constructed from n_{support} support samples.

Table 5.2: Stages of the prototype memory study (Experiment 1.1).

Stage	**Runs**	**Factors varied**
1.1 Initialization	P1-01-P1-06	$n_{\text{support}} \in \{24, 32, 48\}$; k-means vs. examples; no online
1.1 Gate ablation	P1-07, P1-08	$n_{\text{support}} = 32$, $K = 3$ k-means; Mahalanobis-based gating vs. global-prototype margin
1.2 Online budget	P1-09-P1-11	5k / 10k / full train stream
1.3 Update gates	P1-09, P1-12-P1-18	τ_{update}, τ_{new}, α, τ_{online}

With 20 known classes, the initial memory therefore contains 60 prototype vectors. For the initial number of prototypes per class, we evaluated values between 1 and 5 on the COCO dataset. Preliminary results, not reported here, showed that $K = 3$ yields the best trade-off.

Results and Discussion. In comparison to the parametric learning baseline, which achieves high known-class accuracy, our memory-based metric approach underperforms on known classes. In contrast, unknown-class recall is substantially better in our setting. The pattern is familiar, as flexible parametric head fits the closed training distribution well, yet many unknown crops still receive a confident known-class label. This result motivates the use of an explicit memory module, as high known-class accuracy alone does not imply robust open-world behavior.

Another substantial drawback of parametric learning is that retraining is required when new samples arrive. In contrast, our approach updates memory incrementally and online as new samples are observed or new classes are introduced, without requiring the model to be rebuilt.

As we can see in the Table 5.3, across all support sizes, prototype representations constructed with k-means outperform direct storage of support examples. This suggests that clustering provides a more stable summary of category structure than retaining a small set of individual exemplars. This observation is broadly aligned with prototype-based theories of categorization, which propose that concepts can be represented through abstracted category summaries rather than individual instances.

Increasing the support set from 32 to 48 samples improves known-class recognition, but does not improve the balance between known and unknown categories (lower H). This suggests that additional support samples increase category specificity while providing limited benefit for open-set discrimination.

As we can observe, three prototype centroids per class are sufficient to capture a substantial portion of class structure without online refinement. Based on the highest harmonic mean and the lower support requirement, we select **P1-03** as the preferred

Table 5.3: Stage 1.1 prototype initialization (no online refinement)

Run	n_{support}	Init	Known (%)	U-recall (%)	H
MLP baseline	/	/	91.3	25.4	0.398
P1-01	24	k-means	64.3	47.9	0.549
P1-02	24	examples	44.5	35.1	0.392
P1-03	32	k-means	61.8	54.3	**0.578**
P1-04	32	examples	43.6	32.3	0.371
P1-05	48	k-means	65.6	48.7	0.559
P1-06	48	examples	37.3	29.2	0.328

initialization configuration.

To compare alternative open-set gating strategies at a fixed support size, we additionally report Mahalanobis-based gating replacing cosine+margin (P1-07) and global-prototype margin replacing class-aware margin (P1-08), both constructed using $n_{\text{support}} = 32$ support samples and $K = 3$ k-means prototypes.

Table 5.4: Prototype open-set gate ablations on the Task 1 test split, with 32 support embeddings per class ($n_{\text{support}} = 32$) and 3 prototypes per class ($K = 3$).

Run	**Gate mode**	Known (%)	U-recall (%)	H
P1-03 (ref.)	class-aware cosine+margin	61.8	54.3	**0.578**
P1-07	Mahalanobis gating	64.0	30.8	0.415
P1-08	global-prototype margin	62.4	50.3	0.556

Replacing cosine+margin with Mahalanobis-only gating substantially reduces unknown recall and consequently lowers the harmonic mean, despite comparable known-class recognition. The global-prototype margin variant performs similarly to the reference configuration but does not improve the trade-off between known and unknown categories.

Therefore, we retain the class-aware cosine+margin gating strategy used in **P1-03** for the remainder of the experiments.

5.2.2 Stage 1.2: Online refinement and budget

In this set of experiments, our aim is to evaluate how *prototype growth* (adding new prototype vectors per class) and *prototype refinement* (EMA updates to existing vectors) affect open-world performance. We expect high-variance classes, such as *person* or *car*, to accumulate more prototype vectors under this procedure, whereas classes with more stable appearance may retain comparatively fewer vectors under the same budget.

Table 5.5: Online prototype refinement budgets. Capped runs subsample known-class train crops proportionally.

Run	Known (%)	U-recall (%)	H	Online cap	Vectors
MLP baseline	91.3	25.4	0.398	–	–
P1-09	75.8	37.7	0.503	5000	2 003
P1-10	73.7	37.3	0.496	10000	3 675
P1-11	75.0	39.4	0.516	50000	13 167
P1-all	**63.1**	**65.5**	**0.643**	all train	**61 159**

Starting from the P1-03 memory build ($n_{\text{support}} = 32$, $K = 3$ k-means prototypes per class), we enable refinement on labeled known-class training crops, excluding the support set, under three budgets: (i) at most 5000 training crops; (ii) at most 10000 training crops; (iii) at most 50000 training crops; (iv) the full known-class training stream.

For runs with capped budgets, we use a fixed proportion per-class sample selection with seed 153. For the full-budget run, we use all known-class crops, shuffled and processed once.

Each run is evaluated on the test split with class-aware open-set gates calibrated on the training split.

Results and Discussion. Online refinement increases known-class accuracy at small budgets relative to the static prototype build (Table 5.5), although it still does not reach the known-class accuracy of the MLP baseline. As shown in Figure 5.1, the learning trend on the full-stream build exhibits the expected trade-off: known accuracy reaches an early peak of approximately 75% between 5k and 25k updates, while unknown recall remains low in that regime (∼38-39%). Only after substantially more online updates does unknown recall rise, and the harmonic mean H reaches its maximum near the end of full-stream memory update ($H \approx 0.643$).

Full-stream refinement starts from the same static initialization as P1-03, then adapts memory over the entire training stream. Although known-class accuracy decreases relative to the capped budgets (63.1% vs. 74-76%), unknown recall increases substantially (65.5% vs. 37-39%), yielding the highest harmonic mean ($H = 0.643$ vs. 0.578 for static P1-03).

Inspection of prediction statistics shows that for capped runs, about 60% of unknown ground-truth crops are assigned to a known label, with *person* the dominant false-positive class, whereas P1-all reduces this open-set error to 34.5% of unknowns assigned to any known label, at the cost of rejecting a larger number of known crops as *Unknown*. Figure 5.2 illustrates this trend.

Prototype-vector EMA updates and class-wise growth of the memory bank (through

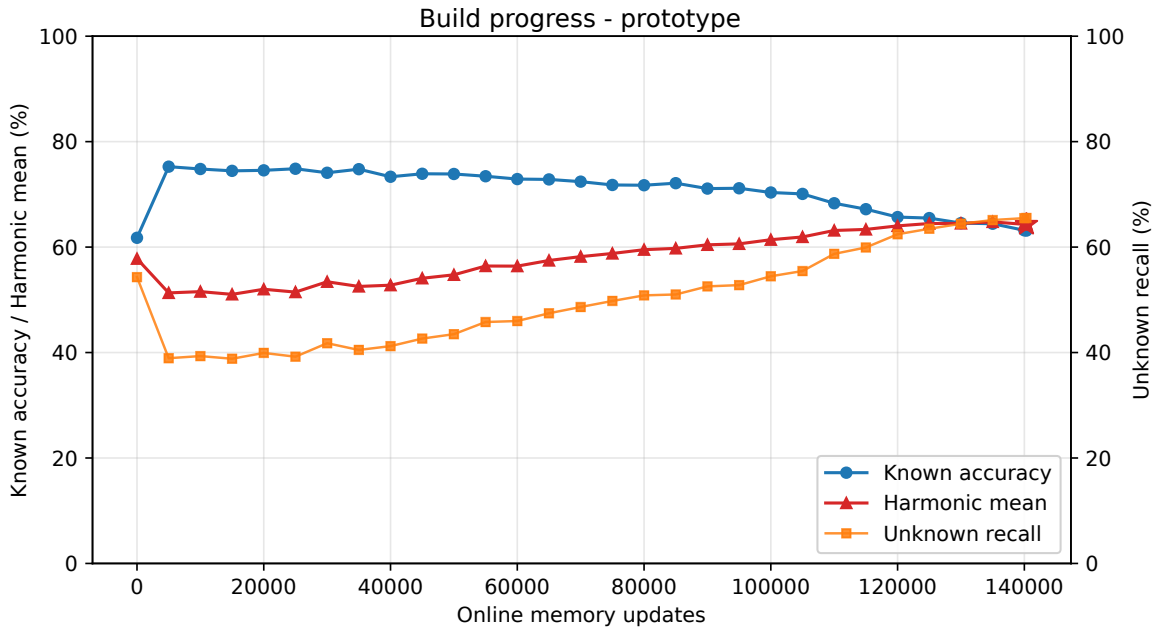


Figure 5.1: Learning curve during prototype-memory build with online refinement on the full training stream

the addition of new prototype vectors per class, e.g. for *person* and *car*) improve OWOD performance under the appropriate update budget. Under the 5k-crop budget, the number of stored prototypes increases by approximately 33× relative to the initial memory. High-variance classes dominate this growth: *person* increases by approximately 410×, whereas low-variance classes remain comparatively small (e.g. *stop sign* grows to 4 prototypes and *fire hydrant* to only 5 prototypes).

Processing the full training stream yields approximately 1020× growth, with *person* increasing by nearly 11000×. This behavior is consistent with the expectation that high-variance classes accumulate substantially more prototypes during online refinement. This increased accumulation improves H mainly by strengthening rejection of unknowns that were previously confused with over-represented classes (especially *person*), at a moderate cost to known accuracy. A potential drawback of this growth is that the prototype representation gradually approaches an exemplar-based memory, reducing the memory-efficiency advantages of the prototype formulation.

Figure 5.3 compares per-class precision and recall for P1-03 (static memory, no online refinement) and P1-all (full-stream online refinement). For each known class, recall is the fraction of ground-truth crops of that class assigned the correct label, while precision is the fraction of predictions of that class that are correct, including false positives from unknown ground truth (open-set errors) and confusions with other known classes.

Compared with P1-03, P1-all improves global open-world balance (unknown recall

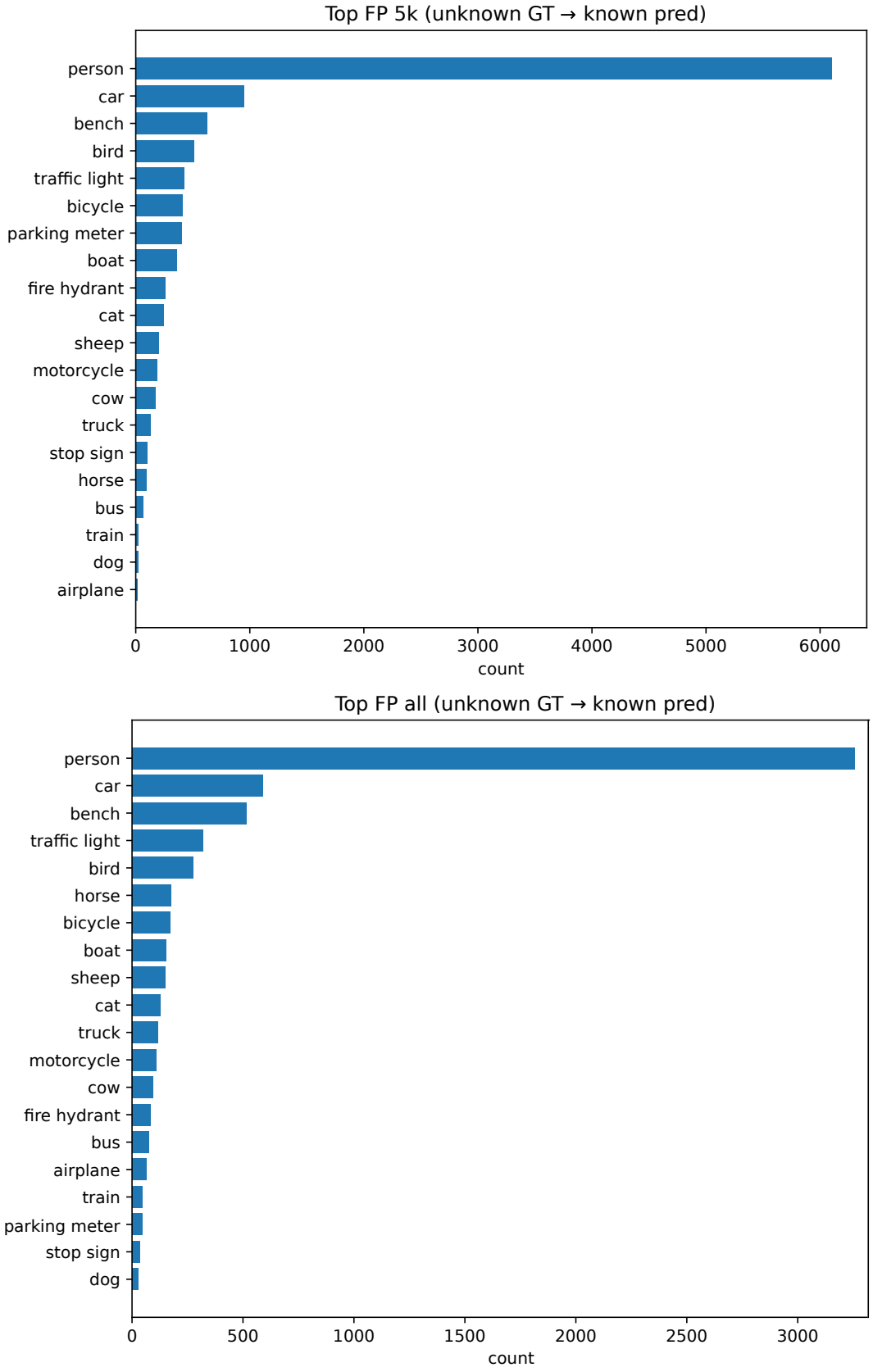


Figure 5.2: Open-set false positives: for unknown ground-truth crops, the number of predictions assigned to each known class (P1-09, 5k cap vs. P1-all, full stream).

54.3% to 65.5%, harmonic mean 0.578 to 0.643). P1-03 already exhibited high-recall, low-precision asymmetry on several classes (e.g. bench, bird, fire hydrant) due to over-prediction of those labels. After online refinement, that pattern persists on some classes but shifts on others. Some classes (e.g.fire hydrant bench) remain precision-limited on P1-all, whereas *dog* becomes recall-limited (approximately 60% to 35% recall), indicating that many dog crops are rejected as *Unknown* or assigned another known class. The bigest gain is the balace of *person* which moves toward a more balanced precision-recall trade-off (approximately 80%/55% on P1-03 vs. 70%/70% on P1-all). This indicates that high-varianced clases and biased dataset distribution benefits from this trade-off.

Online refinement improves OWOD performance by continuously adapting the prototype memory to the training stream. However, these gains come at the cost of substantial memory growth, particularly for high-variance classes such as *person*. This motivates the following update-gate sensitivity analysis and memory-bank optimisation experiments.

5.2.3 Stage 1.3: Sensitivity to update gates

As shown in the previous section, online refinement improves open-world performance but leads to substantial memory growth, particularly for high-variance classes. As the number of stored prototypes increases, the representation gradually shifts from a compact prototype memory toward a more exemplar-like memory. The aim of this experiment is therefore to investigate how different update-gating hyperparameters affect memory growth and model performance, and whether stricter update policies can preserve the benefits of online refinement while maintaining a compact memory representation.

We use P1-09 as the baseline configuration, with a 5000-crop update budget, τ_{update}=0.7, τ_{new}=0.5, α=0.1, and τ_{online}=0.15. Starting from a stricter configuration (τ_{update}=0.85, τ_{new}=0.40, α=0.03, τ_{online}=0.25), we vary one hyperparameter at a time while keeping the others fixed (Table 5.6).

The update gates control different aspects of memory adaptation. The parameter τ_{update} determines when an existing prototype is refined, τ_{new} controls the creation of additional prototypes, α regulates the strength of EMA updates, and τ_{online} filters incoming samples before they can modify memory. Together, these parameters determine the balance between memory compactness, adaptation to intra-class variation, and open-world recognition performance.

Results and Discussion. Relative to P1-09, the stricter grid trades known accuracy for unknown recall: known changes approximately 69-72% and U-recall approximately 41-46%. The best harmonic mean in this stage is P1-18 ($H = 0.554$), combining

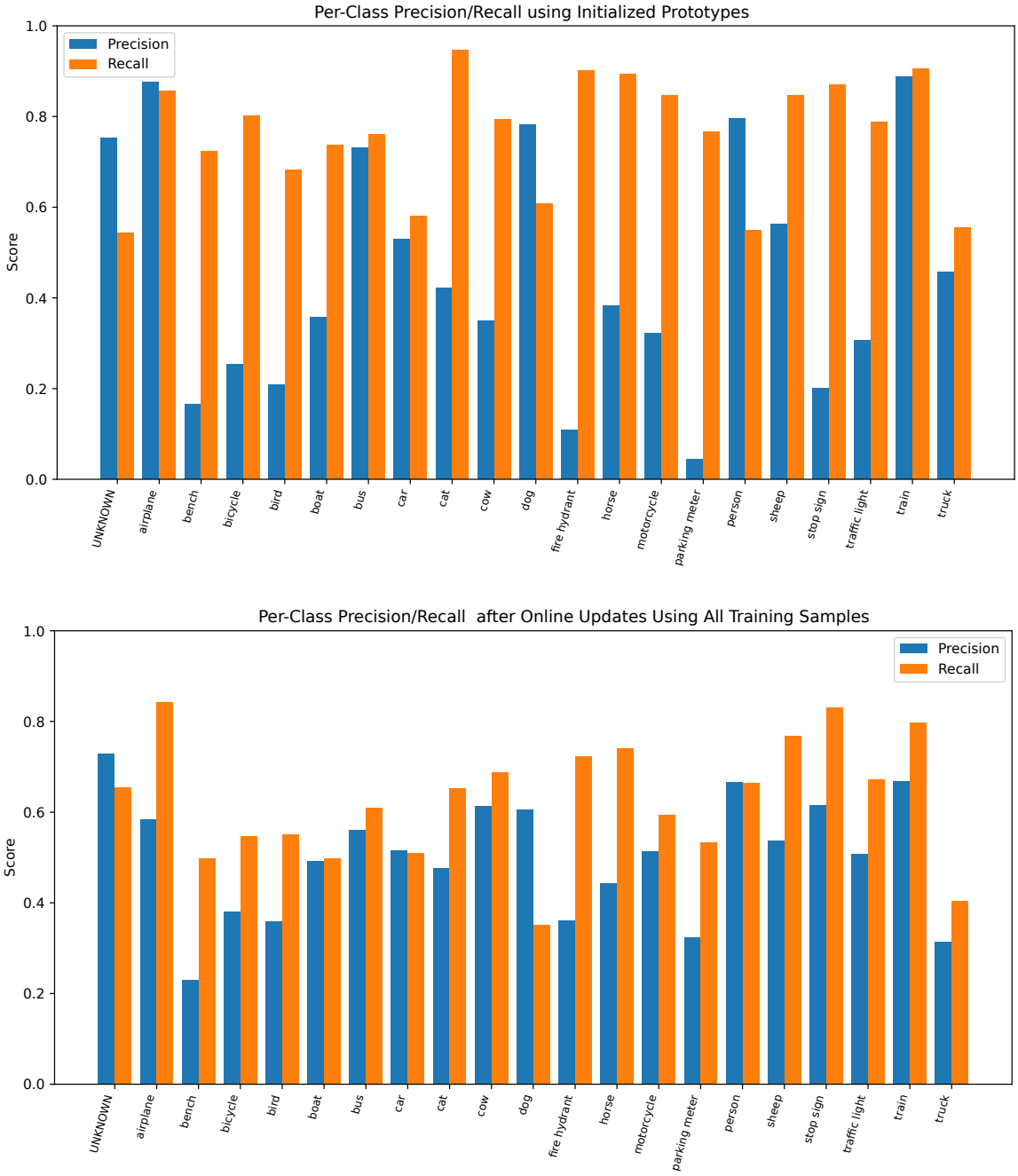


Figure 5.3: Per-class precision and recall on the Task 1 test split: P1-03 (static) vs. P1-all (full-stream online refinement).

τ_{update}=0.85, τ_{new}=0.40, and a stricter online filter τ_{online}0.25). This improves U-recall from 37.7% (P1-09) to 46.1% at the cost of lower known accuracy (69.3% vs. 75.8%). All gate variants remain below full-stream refinement ($H = 0.643$).

The results indicate that stricter update policies consistently favor unknown recall over known-class accuracy what is expected. Increasing τ_{update} and reducing τ_{new} make memory adaptation more selective, while a stricter online filter ($\tau_{\text{online}} = 0.25$) yields the largest individual improvement in H (P1-16). This suggests that filtering noisy or weakly matching updates is more important than modifying the EMA update strength (α), which has only a minor effect on performance.

Combining stricter update and prototype-creation thresholds with a stronger online filter (P1-18) produces the best balance within the 5k-crop budget, increasing unknown recall from 37.7% to 46.1%. However, the gain remains modest compared with the improvement obtained through full-stream refinement. This indicates that update-gate tuning primarily controls how memory evolves under a fixed budget, whereas the dominant factor remains the amount of observed training data.

Table 5.6: Update-gate grid at maximum 5000 samples

Run	Known (%)	U-recall (%)	H	Vectors	Main change
P1-09 (E2)	75.8	37.7	0.503	2 003	default gates
P1-12	72.3	41.6	0.528	1 257	τ_{update}=0.85
P1-13	71.0	41.5	0.524	863	τ_{new}=0.40
P1-14	71.0	41.8	0.526	863	τ_{update}=0.85, τ_{new}=0.40
P1-15	72.3	41.3	0.526	1 258	α=0.03
P1-16	71.1	44.9	0.551	1 016	τ_{online}=0.25
P1-17	70.9	41.9	0.527	863	P1-14 + α=0.03
P1-18	69.3	46.1	**0.554**	**655**	P1-14 + τ_{online}=0.25

All Stage 1.3 runs process the same 5000 training crops but differ in how many prototypes are appended versus merged. The permissive baseline P1-09 grows to 2 003 vectors (approximately 33× the initial memory size of 60 prototypes), with *person* accounting for 1 238 vectors (62% of the memory bank). Under the stricter gate configurations, memory remains substantially smaller, ranging from 655 to 1 258 vectors (approximately 11× to 21× growth).

The best-performing configuration, P1-18, achieves the highest harmonic mean while using the smallest memory bank (655 vectors, 600 updates). This suggests that improved open-world balance under a fixed update budget is achieved primarily through controlled memory growth rather than through aggressive prototype accumulation. Although threshold tuning alone cannot match the gains obtained through full-stream online refinement, it substantially improves memory efficiency while maintaining competitive OWOD performance. This result supports the idea that selective memory

updates can provide a more compact representation without severely degrading recognition performance. This observation is consistent with the broader motivation for compact memory representations, where selective storage can preserve useful information while limiting memory growth.

5.2.4 Conclusion

Experiment 1.1 leads to the following conclusions:

1. Prototype representations constructed using k-means consistently outperform direct storage of support examples. Across all support sizes, k-means prototypes achieve substantially higher known-class accuracy, unknown recall, and harmonic mean, supporting the use of clustered category summaries over individual exemplars.

2. Online refinement is the primary driver of open-world performance improvements. While capped update budgets improve known-class accuracy, processing the full training stream produces the best balance between known-class recognition and unknown rejection (P1-all, known 63.1%, U-recall 65.5%, $H = 0.643$).

3. Online refinement causes substantial memory growth, particularly for high-variance classes such as *person*. Although this growth improves open-world performance, it gradually shifts the representation toward an exemplar-like memory and reduces the memory-efficiency advantages of the prototype formulation.

4. Update-gate tuning provides a mechanism for controlling memory growth. Stricter update policies improve memory efficiency while maintaining competitive performance, with P1-18 achieving the best balance within the fixed 5k-crop budget ($H = 0.554$) using the smallest memory bank among the evaluated configurations.

5. Prototype memory benefits from both compact category summaries and online adaptation. The strongest gains arise from continued refinement on additional data rather than from threshold tuning alone.

Based on the highest harmonic mean and substantially reduced memory growth, we select **P1-18** as the reference configuration for the subsequent experiments.

5.3 Experiment 1.2: Exemplar Memory Construction

Following the prototype study (Section 5.2), we evaluate an *exemplar memory* representation, in which each known class is represented by a set of raw unit-norm embedding vectors rather than a small compressed prototype set. From a cognitive perspective, exemplar memory represents the opposite extreme of the prototype formulation, retaining individual category instances rather than abstract category summaries.

Initial exemplars are selected using k-means diversification on the per-class training pool (M crops per class, one representative per cluster) and stored as raw embedding vectors. Classification uses maximum cosine similarity per class together with the same default open-set rejection protocol as in Experiment 1.1.

The objective of this experiment is to determine whether retaining individual examples, rather than compressed prototype summaries, improves known-class recognition and unknown rejection on OWOD Task 1. We investigate how exemplar capacity M, open-set gate design, and online update budget affect the trade-off between known-class accuracy and unknown recall.

All runs in this section use exemplar mode, cosine scoring, and no MHN-based refinement or classification. Features are the same pre-extracted L2-normalised DINO embeddings computed from ground-truth training crops and evaluated on the standard Task 1 test split.

Table 5.7: Fixed hyperparameters for Experiment 1.2 (exemplar memory).

Parameter	**Value**
OWOD task	Task 1 (20 known classes)
Initial selection	M raw crops per class
Storage	Raw unit-norm vectors
Calib. exclusion	All stored initial exemplars (M per class)
Open-set gates (default)	Class-aware top-1 + margin
Update mode	Append only
Online filter	τ_{online}=0.15
Classifier	Cosine (no MHN)

Unlike the prototype track, we do not process the full training stream during online refinement. As demonstrated in Section 5.2.2, memory growth can become substantial during online updates. Since exemplar memory stores raw vectors without compression, unrestricted refinement would result in significantly larger memory banks than in the prototype setting. We therefore limit online refinement to a maximum of 50 000 training crops.

Crops that pass the online update threshold (τ_{online}) are appended as new exemplar vectors, whereas the merge and prototype-creation thresholds τ_{update} and τ_{new} used in

prototype EMA mode are not applied. The parameter M controls only the size of the *initial* memory bank ($20M$ vectors at build time). During online refinement, additional exemplars may be appended, allowing the total memory size to grow to the specified budget.

Table 5.8: Stages of the exemplar memory study (Experiment 1.2).

Stage	**Runs**	**Factors varied**
2.1 Capacity	E2-01–E2-03	$M \in \{20, 50, 100\}$; no online update
2.2 Open-set gates	E2-04, E2-05	$M = 50$; Mahalanobis-based gating vs. global-prototype margin
2.3 Online budget	E2-06–E2-09	$M = 50$; max online cap $\in \{5000, 10000, 50000\}$.

5.3.1 Stage 2.1: Exemplar Capacity Without Online Refinement

In this experiment stage, we evaluate the effect of exemplar memory capacity without online refinement (E2-01–E2-03).

Results and Discussion. Static exemplar memory achieves moderate known-class accuracy (63.5-67.4%) and unknown recall (50.4-53.0%), resulting in harmonic means between 0.564 and 0.590 (Table 5.9). Increasing the initial capacity from $M = 20$ to $M = 50$ improves both known-class recognition and the known-unknown balance (H increases from 0.564 to 0.590). Further increasing capacity to $M = 100$ reaches the highest known-class accuracy (67.4%), but slightly reduces unknown recall, resulting in a lower harmonic mean than the $M = 50$ configuration (0.576 vs. 0.590).

Memory size grows linearly with capacity, increasing from 400 vectors at $M = 20$ to 2 000 vectors at $M = 100$. The improvement beyond $M = 50$ suggests that simply storing more exemplars does not necessarily improve open-world performance. Additional exemplars appear to increase category specificity, but provide limited benefit for rejecting unknown objects.

Table 5.9: Exemplar capacity ablation (no online refinement).

Run	M	Known (%)	Unknown (%)	H	Vectors
E2-01	20	63.5	50.8	0.564	400
E2-02	50	66.6	53.0	0.590	1000
E2-03	100	67.4	50.4	0.576	2000

Compared with prototype memory, exemplar representations operate in a similar performance regime despite substantially larger storage requirements. For example, the best static prototype configuration (P1-03) achieves 61.8% known accuracy, 54.3% unknown recall, and $H = 0.578$ using only 60 stored vectors. In contrast, exemplar memory requires between 400 and 2000 vectors to obtain comparable performance. This suggests that prototype representations provide a considerably more memory-efficient summary of category structure, while exemplar memory offers only modest gains in known-class recognition at build time.

Based on the highest harmonic mean, we select E2-02 ($M = 50$) as the reference exemplar configuration for the subsequent experiments.

5.3.2 Stage 2.2: Open-Set Gate Ablations

At a fixed capacity of $M = 50$ and without online refinement, we evaluate the same open-set gate variants as in the prototype study. E2-04 replaces the default cosine+margin scheme with Mahalanobis-based gating, while E2-05 retains cosine similarity but replaces the class-aware margin with a global margin signal. The default cosine+margin configuration (E2-02) serves as the reference.

Results and Discussion. The default-gate baseline (E2-02) achieves 66.6% known-class accuracy, 53.0% unknown recall, and $H = 0.590$. Replacing cosine+margin with Mahalanobis-based gating substantially reduces unknown recall (53.0% to 37.1%) while producing only a minor change in known-class accuracy, resulting in a considerably lower harmonic mean ($H = 0.472$). This mirrors the results observed in the prototype study (P1-07), suggesting that Mahalanobis-based gating is less effective at separating known and unknown samples in this feature space (Table 5.10).

The global-margin variant (E2-05) achieves the highest known-class accuracy (67.4%), but this improvement is accompanied by lower unknown recall (49.5%), resulting in a slightly worse trade-off than the reference configuration ($H = 0.571$ vs. 0.590). As in the prototype setting, the gain in category accuracy does not translate into improved open-world performance.

The results indicate that the class-aware cosine+margin setup provides the most balanced open-set behavior for exemplar memory. We therefore retain E2-02 as the reference exemplar configuration for the subsequent online-refinement experiments.

5.3.3 Stage 2.3: Online Refinement and Budget

We fix $M = 50$ and online *append only* refinement under crop budgets $\in \{5000, 10000, 50000\}$ (E2-06–E2-08). Each budget caps how many training crops are scanned, but appended updates are lower because of τ_{online} filtering. As in the prototype-based experiments,

Table 5.10: Exemplar open-set gate ablations on the Task 1 test split with 50 exemplars per class ($M = 50$).

Run	Gate mode	Known (%)	Unknown (%)	H
E2-02 (baseline)	class-aware cosine+margin	66.6	53.0	0.590
E2-04	Mahalanobis-based gating	64.7	37.1	0.472
E2-05	global-prototype margin	67.4	49.5	0.571

crops are sampled using a class-proportional random subsampling strategy with a fixed seed. Unlike the prototype track, we do not run a full-training-stream exemplar fit because of memory budget.

Results and Discussion. Online refinement substantially increases known-class accuracy. At 5k online cap (E2-06), known accuracy reaches 76.8%, unknown recall 42.7%, and $H = 0.549$, with memory growing after 4 481 append updates. Known accuracy further increases, at 50k (E2-08), known accuracy reaches 79.3% and unknown recall recovers slightly to 43.0% ($H = 0.558$) in comparison to 10k cap, with 45 793 stored vectors. Despite the increase in known-class recognition, none of the online exemplar configurations surpass the static E2-02 baseline in terms of known-unknown balance ($H = 0.590$) (Table 5.11).

Table 5.11: Online exemplar refinement and prototype comparison.

Configuration	Known (%)	Unknown (%)	H	Vectors	Updates
Exemplar (this section)					
E2-02, no online (M=50)	66.6	53.0	0.590	1 000	0
E2-06, online 5k	76.8	42.7	0.549	5 481	4 481
E2-07, online 10k	78.1	40.2	0.531	9 978	8 978
E2-08, online 50k	79.3	43.0	0.558	45 793	44 793
E2-09, strict gates, online 5k	75.5	48.1	0.588	4 777	3 777
Prototype (Section 5.2)					
P1-09, online 5k	75.8	37.7	0.503	2 003	2 354
P1-10, online 10k	73.7	37.3	0.496	3 675	4 520
P1-11, online 50k	75.0	39.4	0.516	13 167	19 013
P1-all, full stream	63.1	65.5	0.643	61 159	140 330
P1-18, strict gates, online 5k	69.3	46.1	0.554	655	600

In exemplar memory, appending additional vectors mainly improves category specificity and known-class recognition, while providing limited benefit for unknown rejection. As a result, memory grows substantially without a corresponding improvement in H.

Under the stricter gate configuration shared with prototype P1-18, exemplar run

E2-09 improves unknown recall to 48.1% and reaches $H = 0.588$ with a smaller memory bank (4 777 vectors) than E2-06. This suggests that controlling memory growth is beneficial also in exemplar mode, although the improvement remains modest.

Compared with prototype memory, exemplar memory consistently achieves higher known-class accuracy under matched online budgets, but requires substantially larger memory banks. At the 5k budget, E2-06 improves both known accuracy and unknown recall relative to P1-09, but uses approximately 2.7× more stored vectors. Similar trends are observed at 10k and 50k budgets. Full-stream prototype refinement P1-all remains the strongest open-world configuration ($H = 0.643$), indicating that memory structure and update strategy are more important than simply storing additional examples, but at the cost of memory growth.

Additionally, we investigated class-wise performance for P1-18 and E2-09, as these configurations are selected as the reference prototype and exemplar setups for the subsequent experiments. Figure 5.4 presents the corresponding confusion matrices.

Both configurations show a very similar distribution of correctly classified and misclassified instances, suggesting that prototype and exemplar memory introduce comparable class-specific biases. Interestingly, the dominant source of known-class errors is the *dog* category, which is frequently assigned the *Unknown* label. Among unknown objects, the most common open-set error is assigning them to the *person* class.

The confusion matrices suggest that the differences between prototype and exemplar memory are mainly reflected in the global performance metrics and memory growth, rather than in substantially different class-level error patterns. A more detailed analysis of these biases and error modes will be analyzed later. At this stage, our goal is simply to verify whether prototype and exemplar representations are differently biased toward specific classes, which does not appear to be the case.

5.3.4 Conclusion

Experiment 1.2 leads to the following conclusions:

1. Static exemplar memory achieves competitive open-world performance without online refinement. Increasing capacity from $M = 20$ to $M = 50$ improves both known-class accuracy and harmonic mean, while larger capacities provide only marginal benefit. The best static configuration is E2-02 ($M = 50$, $H = 0.590$).

2. Consistent with the prototype study, alternative open-set gate formulations do not improve the known-unknown balance. The default class-aware cosine+margin configuration remains the strongest static exemplar setting.

3. Online append-only refinement substantially improves known-class accuracy, reaching nearly 80% at the 50k crop budget. However, unlike prototype memory, these

gains do not translate into improved open-world balance, and none of the online exemplar configurations surpass the static E2-02 baseline in harmonic mean.

4. At matched online budgets, exemplar memory generally achieves higher known-class accuracy and slightly higher harmonic mean than the corresponding prototype configurations, but requires substantially larger memory banks.

5. Exemplar memory benefits from retaining individual category instances and achieves strong known-class recognition. However, these gains come at the cost of substantial memory growth, while improvements in open-world performance remain limited.

Based on its strong open-world performance and controlled memory growth, we select **E2-09** as the reference exemplar configuration for the subsequent experiments.

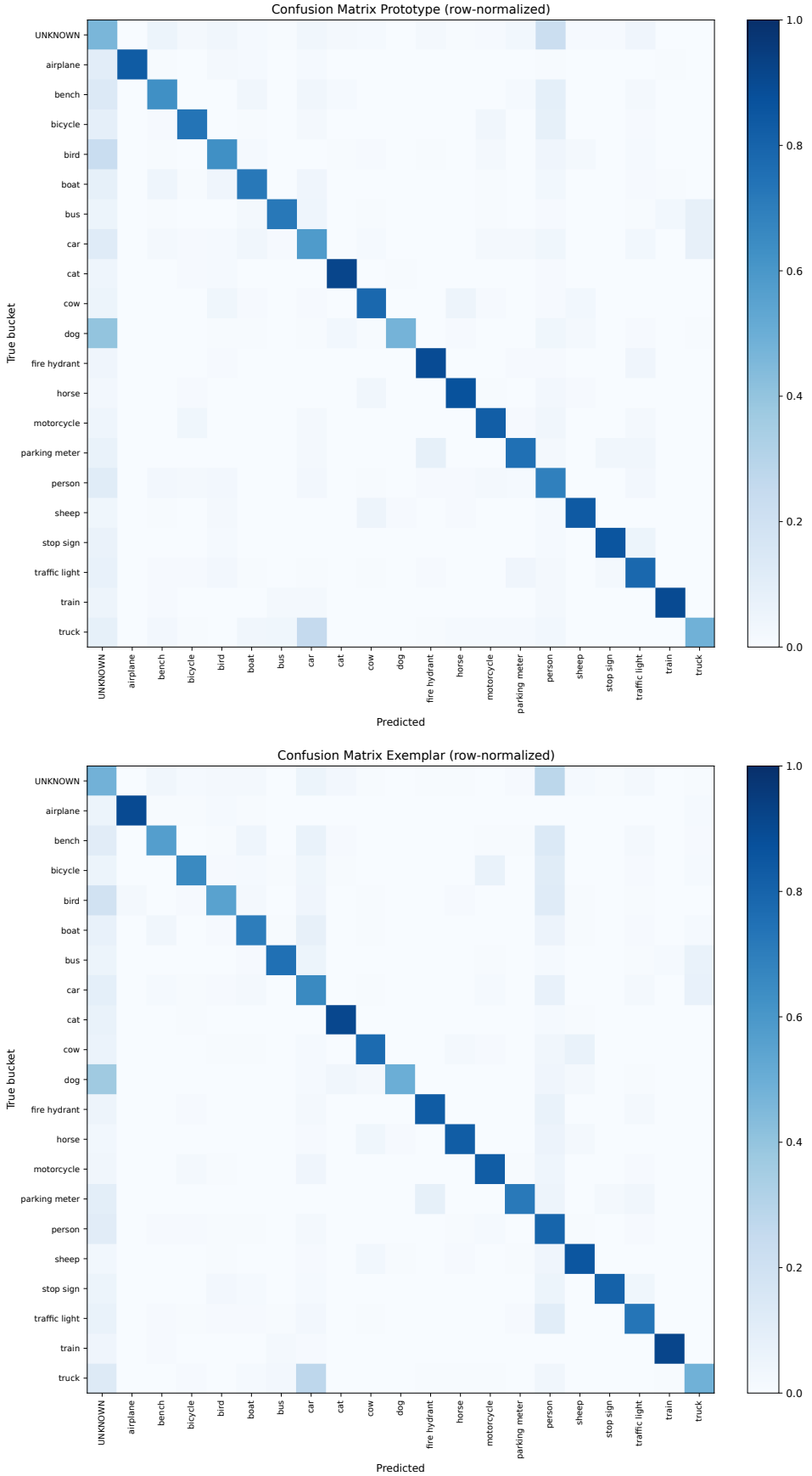


Figure 5.4: Confusion matrices for the selected prototype (P1-18) and exemplar (E2-09) configurations on the Task 1 test split. Both memory representations exhibit similar class-level error patterns, with *car* frequently assigned to *Unknown* and unknown objects often misclassified as *person*.

Chapter 6

Experiment 2: MHN-based Retrieval

Experiment 1 focused on how memories should be represented. In this experiment, we investigate a complementary question: how should stored memories be retrieved? Until now, classification relied on direct cosine matching between the query and stored memory vectors. Here, we replace this retrieval mechanism with MHN-based associative retrieval and study whether memory-guided refinement alters the known-unknown trade-off in open-set recognition.

As explained in Section 3.1.4 and Section 3.1.5, MHN-based retrieval can be interpreted as a memory-guided refinement mechanism that projects query representations onto a manifold of stored prototypes or exemplars. By aligning a query with weighted memory items, the model performs attractor-like dynamics that can reduce embedding noise and alter class-score distributions in high-dimensional feature space.

Experiment 1 showed that both prototype and exemplar memory can support open-world recognition, but also revealed clear limitations. Prototype memory provided the best trade-off between known-class recognition and unknown rejection, while exemplar memory achieved higher known-class accuracy at substantially higher memory cost. In both cases, classification relied on direct cosine matching between the query and stored memory vectors.

The aim of this experiment is to evaluate whether MHN-based retrieval can improve memory utilisation beyond direct cosine matching. Here, we investigate whether memory-guided refinement improves known-class recognition, strengthens open-set rejection, and changes the trade-off between prototype and exemplar memory representations. For these experiments, we use the selected configurations from Experiment 1 (P1-18 and E2-09) as reference memory configurations and compare standard cosine retrieval with MHN-based retrieval under the same evaluation protocol.

6.1 Experiment 2.1: Cosine vs. Modern Hopfield Classification

Experiments 1.1 and 1.2 fixed the *memory content* (prototype vs. exemplar construction and online updates) while using cosine similarity-based classification and the same open-set gates. In this section we isolate the *classifier head*: we load fixed memory banks and compare cosine scoring with MHN-based class aggregation. MHN refinement is disabled here and will be studied later.

We evaluate the strict-gate 5k memories selected at the end of Stages 1.3 and 2.3, for prototype P1-18 and exemplar E2-09. We use the same open-set protocol as for memory experiments (class-aware top-1 and margin gates calibrated at build time).

Class scores are produced by aggregating over all stored vectors of each class. The predicted label is the argmax of the Hopfield energy-based class scores. For this experiment stage, Hopfield inverse temperature is $\beta = 30$.

Results and Discussion. The effect of MHN-based classification differs between prototype-based and exemplar-based memory. On the exemplar memory bank, MHN classification substantially improves unknown recall and increases the harmonic mean (H), while preserving similar known-class accuracy (Table 6.1). In contrast, only minor changes are observed for prototype memory, where MHN slightly decreases both unknown recall and harmonic mean.

At fixed memory and shared open-set gates, MHN classification improves H on the exemplar bank from (0.588 to 0.649). Unknown recall increases, while known-class accuracy remains nearly unchanged. Prototype memory shows a smaller and slightly negative effect, suggesting that MHN aggregation is more beneficial when memory contains a larger and more diverse set of stored vectors.

Table 6.1: Cosine vs. MHN classification on fixed Task 1 memories. The MHN inverse-temperature parameter is 30 (β=30)

Memory	Classifier	Known (%)	Unknown (%)	H
Prototype (P1-18)	Cosine (S3-01)	69.3	46.1	0.554
Prototype	MHN (S3-02)	70.6	44.8	0.548
Exemplar (E2-09)	Cosine (S3-03)	75.5	48.1	0.588
Exemplar	MHN (S3-04)	75.7	56.8	**0.649**

The results indicate that associative retrieval improves memory utilization beyond direct cosine matching, for exemplar memory. MHN aggregates similarity information across all stored vectors of a class, rather than relying only on the single nearest memory item. This changes the class-score distribution and appears to produce more

robust class predictions. In the open-world setting, this mainly manifests as improved unknown recall, while preserving similar known-class accuracy. The effect, visible for exemplar memory, where the larger number of stored vectors provides more information for MHN aggregation, suggests that the benefits of MHN classification increase with memory size and diversity.

6.2 Experiment 2.2: Sensitivity to MHN Inverse Temperature

After observing how MHN classification affects open-world performance (Section 6.1), we ablate the Hopfield inverse temperature β, which controls how sharply class scores concentrate on the strongest matching memories. We reload the same prototype bank (P1-18, 655 vectors) and exemplar bank (E2-09, 4 777 vectors), keep the open-set gates fixed from memory build, and sweep $\beta \in \{5, 10, 20\}$ on prototype and exemplar memory, respectively for MHN-based classification.

Table 6.2: Effect of the MHN inverse-temperature parameter β on fixed Task 1 memories. The memory contents and open-set gates are kept fixed from the memory build, while only the MHN inverse-temperature parameter β is varied.

Memory	Classifier	Known (%)	Unknown (%)	H
Prototype (P1-18)	Cosine (S3-01)	69.3	46.1	0.554
Prototype	MHN, β=5 (S3-05)	45.6	57.7	0.509
Prototype	MHN, β=10 (S3-06)	53.0	57.8	0.553
Prototype	MHN, β=20 (S3-07)	71.2	48.6	**0.578**
Prototype	MHN, β=30 (S3-02)	70.6	44.8	0.548
Exemplar (E2-09)	Cosine (S3-03)	75.5	48.1	0.588
Exemplar	MHN, β=5 (S3-08)	52.4	50.0	0.512
Exemplar	MHN, β=10 (S3-09)	63.0	50.3	0.559
Exemplar	MHN, β=20 (S3-10)	74.9	58.1	**0.654**
Exemplar	MHN, β=30 (S3-04)	75.7	56.8	0.649

Results and Discussion. Table 6.2 shows that MHN is sensitive to β, but the benefit is not limited to a single setting. Very low β (5) smooths aggregation across many vectors per class: unknown recall increases while known-class accuracy drops substantially, leaving H near or below the cosine baseline (0.509 vs. 0.554 on prototype; 0.512 vs. 0.588 on exemplar). For $\beta = 20$, MHN exceeds cosine H on both memory banks.

On exemplar memory, H increases from 0.512 at β=5 to a sweep maximum of 0.654 at β=20 (S3-10), then decreases slightly at the default β=30 (S3-04, H=0.649), while still outperforming cosine (0.588).

These results indicate that the improvement from MHN classification is robust across a broad mid-to-high β range. Very low values over-smooth the retrieval process, strongly favoring unknown recall at the expense of known-class accuracy, whereas larger values produce a more balanced trade-off. Across both memory representations, β=20 yields the best harmonic mean, suggesting that moderately sharp retrieval is preferable to both highly diffuse and highly concentrated aggregation.

We also analyzed the normalized confusion matrix and per-class precision/recall for the best MHN configuration (S3-10, MHN on exemplar memory E2-09) on Task 1 (Figure 6.1).

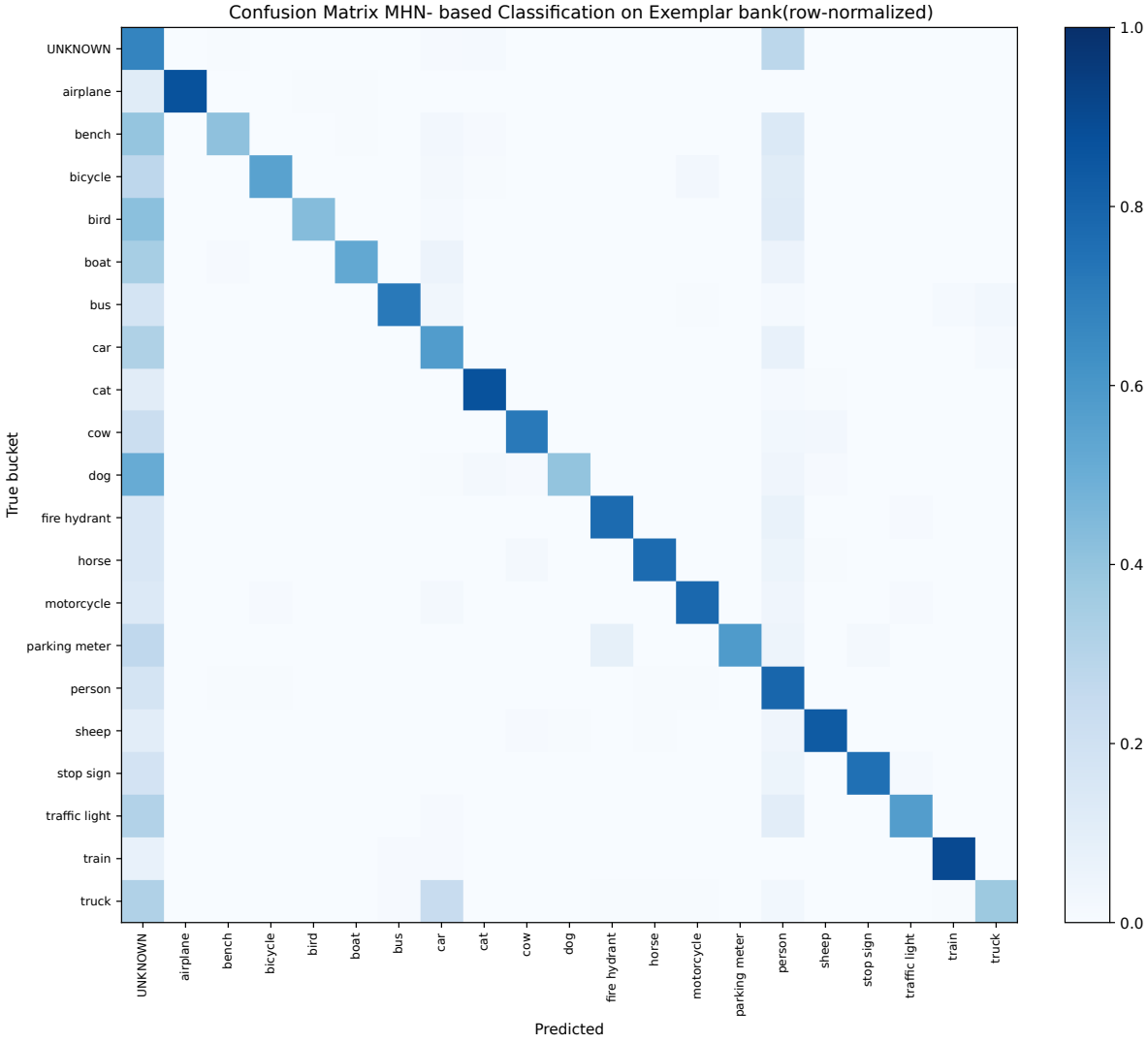


Figure 6.1: Confusion Matrix on the Task 1 test split for MHN-based classification on Exemplar memory bank (E2-09).

Compared with cosine classification on P1-18 and E2-09 (Figure 5.4), the *Unknown*

category shows improved recognition, with a larger proportion of unknown objects correctly assigned to the *Unknown* label. However, this improvement is accompanied by a larger number of known-class instances being assigned to *Unknown*. Interestingly, the confusion between *Unknown* and *person* remains present, indicating that *person* continues to act as the dominant attractor class for open-set errors. We also observe confusions between *truck* and *car*, which is expected given the visual similarity between these categories, even for human annotators.

The per-class precision and recall analysis shows that, for most known classes relative to E2-09 (Figure 5.4), precision increases at the cost of a reduction in recall. This suggests that MHN classification produces more conservative predictions, reducing false-positive assignments while improving the balance between known-class recognition and unknown-object rejection.

For the main experimental pipeline, subsequent experiments use the β=20 configuration. The current results suggest that additional gains may be possible through further tuning of the retrieval sharpness.

6.3 Experiment 2.3: MHN-based Refinement

With MHN-based refinement, each query embedding is updated according to Eq. 4.9 using attention over all memory vectors, *before* classification. We evaluate refinement with both *cosine* similarity classification and *MHN* class aggregation (β=20), using the strict-gate 5k memory banks P1-18 (prototype) and E2-09 (exemplar). Open-set gates are fixed from memory build.

Results and Discussion. Table 6.3 compares the four inference modes on each memory bank. For both classifiers, adding MHN-based refinement lowers unknown recall and consequently reduces the harmonic mean H, while producing only small changes in known-class accuracy.

On prototype memory, refinement reduces H from 0.578 (S3-07) to 0.518 (S4-02) when combined with MHN classification. A similar effect is observed for exemplar memory, where H decreases from 0.654 (S3-10) to 0.642 (S4-04).

This behavior suggests that refinement pulls query embeddings toward stored memory representations, making unknown samples appear more similar to known classes. While such attractor dynamics may be beneficial in closed-set recognition, they reduce the effectiveness of open-set rejection by increasing the tendency to associate unknown objects with existing memory items.

MHN-based refinement does not improve open-world recognition on either prototype or exemplar memory. Since both cosine + refine (S4-01, S4-03) and MHN + refine (S4-02, S4-04) perform worse than their corresponding classify-only variants,

Table 6.3: Classification with and without MHN refinement on fixed Task 1 memories. MHN is evaluated with inverse-temperature parameter $\beta = 20$, and the open-set gates are kept fixed from the memory build.

Memory	Mode	Known (%)	Unknown (%)	H
Prototype (P1-18)	Cosine (S3-01)	69.3	46.1	0.554
Prototype	Cosine + refine (S4-01)	70.1	42.1	0.526
Prototype	MHN classify (S3-07)	71.2	48.6	0.578
Prototype	MHN + refine (S4-02)	70.9	40.8	0.518
Exemplar (E2-09)	Cosine (S3-03)	75.5	48.1	0.588
Exemplar	Cosine + refine (S4-03)	75.6	40.6	0.528
Exemplar	MHN classify (S3-10)	74.9	58.1	0.654
Exemplar	MHN + refine (S4-04)	74.1	56.6	0.642

refinement is not adopted in the remainder of the experiments. Instead, MHN classification without refinement at β=20 (S3-07 and S3-10) is retained as the preferred retrieval strategy.

6.4 Conclusion

Experiment 2 leads to the following conclusions:

1. MHN-based classification generally improves open-world performance compared with direct cosine classification on the same memory banks. The improvement is primarily driven by higher unknown recall, while known-class accuracy remains largely unchanged.

2. The benefits of MHN classification are more pronounced for exemplar memory than for prototype memory. The largest improvement is observed for E2-09, where MHN increases the harmonic mean from 0.588 to 0.654 using the same stored memory.

3. MHN performance is sensitive to the inverse temperature parameter β. Very low values ($\beta = 5$) over-smooth retrieval, substantially increasing unknown recall but causing a strong reduction in known-class accuracy. Intermediate values produce a more balanced known-unknown trade-off.

4. Across both prototype and exemplar memory, $\beta = 20$ yields the highest harmonic mean, outperforming both the default $\beta = 30$ and direct cosine classification. This suggests that moderately sharp retrieval provides the best balance between associative aggregation and class specificity.

5. MHN-based test-time refinement does not improve open-world recognition. Although refinement slightly increases known-class accuracy in some cases, it consistently reduces unknown recall and lowers the harmonic mean. This indicates that attractor-style refinement pulls unknown queries toward known memory representations, reducing the effectiveness of open-set rejection.

MHN classification improves memory utilization beyond direct cosine matching and produces the strongest open-world performance observed in this work. The results suggest that associative retrieval is particularly effective when combined with large and diverse exemplar memory banks. Based on the highest harmonic mean, we select **S3-07** and **S3-10** (P1-18 and E2-09 + MHN classification, $\beta = 20$) as the reference configuration for the subsequent experiments. MHN-based refinement is not adopted in the following study.

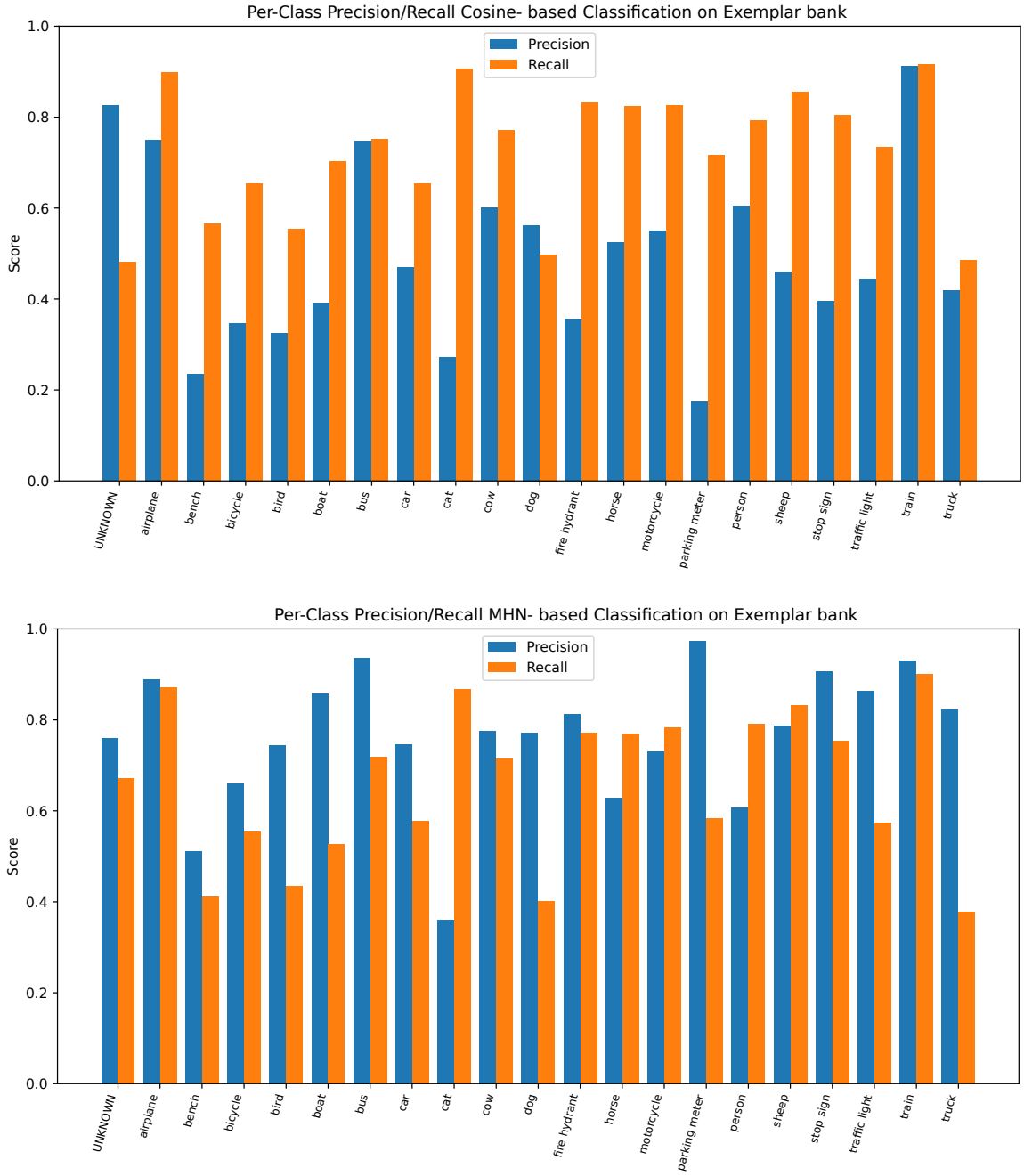


Figure 6.2: Per-class precision and recall on the Task 1 test split for Cosine-based and MHN-based classification on Exemplar memory bank (E2-09).

Chapter 7

Experiment 3: Incremental Memory Construction Across OWOD Tasks

After the Task 1 memory study (Chapters 5 and 6), we investigate how the selected recognition pipelines generalise across the full OWOD curriculum. The OWOD protocol consists of four sequential tasks, each introducing additional COCO categories while requiring recognition of previously learned classes.

We compare three learning strategies using the same pre-extracted L2-normalised DINO embeddings:

1. MLP baseline: a parametric classifier trained from scratch on the cumulative training set available at each task.

2. Prototype memory: incremental memory construction using the selected prototype configuration P1-18 together with MHN-based classification ($\beta = 20$).

3. Exemplar memory: incremental memory construction using the selected exemplar configuration E2-09 together with MHN-based classification ($\beta = 20$).

For the memory-based approaches, memory is updated incrementally as new classes and training samples become available. Prototype memory follows the P1-18 configuration selected in Experiment 1.1, while exemplar memory follows the E2-09 configuration selected in Experiment 1.2. Both approaches use the MHN classifier with $\beta = 20$, identified in Experiment 2 as the best-performing retrieval setting.

All methods use the same class-aware open-set calibration procedure described in Eq. 4.3. Performance is evaluated after each OWOD task using known-class accuracy, unknown recall, and harmonic mean H. At Task 4 all classes in the evaluation split are known, and therefore unknown recall and H are no longer applicable. In this case we report known-class accuracy only.

The objective of this experiment is to determine whether memory-based representations can accumulate knowledge across tasks without retraining, and how they compare

to a conventional parametric classifier trained from scratch on the cumulative data at each stage of the OWOD curriculum.

Results and Discussion. The results of all three pipelines are presented in Tables 7.1,7.2,7.3.

The MLP baseline achieves the highest known-class accuracy at every task, reaching 91.3% on Task 1 and 80.8% on Task 4 (Table 7.1). However, unknown recall remains very low throughout the curriculum (25-28%), resulting in harmonic means close to 0.40. This indicates that the parametric classifier learns the known classes effectively but continues to assign many unknown objects to known categories.

Table 7.1: MLP baseline across OWOD tasks.

Task	Known (%)	Unknown (%)	H
1	91.3	25.1	0.394
2	86.6	27.9	0.422
3	82.3	28.2	0.420
4	80.8	–	–

For the prototype memory track, memory grows gradually from 655 vectors at Task 1 to 5 879 vectors at Task 4 (Table 7.2). Known-class accuracy decreases as the number of known categories increases (71.2% to 55.3%), which is expected given the growing classification difficulty. Nevertheless, unknown recall remains relatively stable between 49% and 59%, producing harmonic means between 0.53 and 0.589. This suggests that the prototype representation retains its ability to separate known and unknown categories even as new classes are incrementally added.

Table 7.2: Prototype memory using the P1-18 configuration with MHN classification across OWOD tasks. MHN is evaluated with inverse-temperature parameter $\beta = 20$.

Task	Known (%)	Unknown (%)	H	Vectors
1	71.2	48.6	0.578	655
2	61.7	56.3	0.589	1 805
3	55.3	50.8	0.530	3 379
4	63.8	–	–	5 879

The exemplar track exhibits a similar trend but consistently achieves higher unknown recall and higher harmonic mean than the prototype track (Table 7.3). Unknown recall reaches 71.3% at Task 3 and remains above 58% across all tasks containing unknown categories. As a result, exemplar memory achieves the highest open-world performance in this chapter, with the highest harmonic mean of 0.679 on Tasks 2.

Table 7.3: Exemplar memory using the E2-09 configuration with MHN classification across OWOD tasks. MHN is evaluated with inverse-temperature parameter $\beta = 20$.

Task	Known (%)	Unknown (%)	H	Vectors
1	74.9	58.1	0.654	4 777
2	65.4	70.5	0.679	14 918
3	57.6	71.3	0.637	29 973
4	65.5	–	–	50 343

The improved open-world performance comes at a substantial memory cost. While prototype memory grows to only 5 879 vectors by Task 4, exemplar memory expands to 50 343 stored vectors. This confirms the trade-off observed in previous chapters: retaining individual examples improves recognition performance, but requires considerably more storage than compact prototype representations.

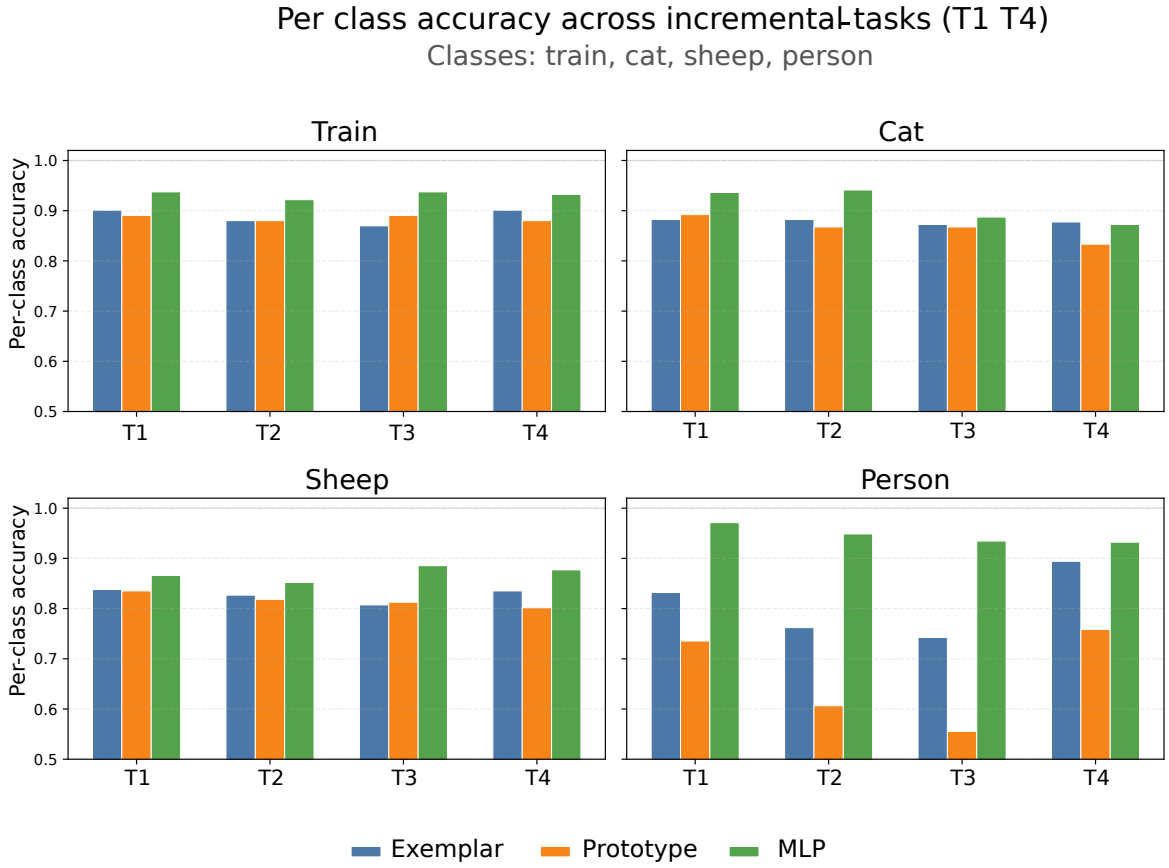


Figure 7.1: Per-class accuracy on the OWOD test split across four incremental episodes (T1-T4) for exemplar, prototype, and MLP pipelines.

A further observation is that both memory-based approaches exhibit a gradual decline in known-class accuracy as new classes are introduced until Task 4, which becomes a closed-set classification setting, since no unknown instances remain in the

evaluation split. However, unlike the MLP baseline, they maintain substantially higher unknown recall throughout the OWOD curriculum. Consequently, the memory-based approaches achieve a more balanced trade-off between known-class recognition and unknown-object rejection.

Additionally, we analyzed how per-class accuracy changed across incremental training episodes (Figure 7.1). We tracked the four classes with highest recognition accuracy at Task 1 in the exemplar memory bank: *train*, *cat*, *sheep*, and *person*.

For *train*, *cat*, and *sheep*, both memory-based pipelines and the parametric MLP maintained relatively stable accuracy across T1-T4, with the MLP consistently achieving the highest scores (as expected). Importantly, the parametric model was retrained from scratch at every episode, whereas the exemplar and prototype pipelines incrementally incorporated new class instances into memory.

The most unstable class was *person*, which accounted for most of the accuracy fluctuation in both memory models and was especially unstable in the prototype pipeline dropping from 73.5% at T1 to 55.5% at T3, then recovering to 75.8% at T4.

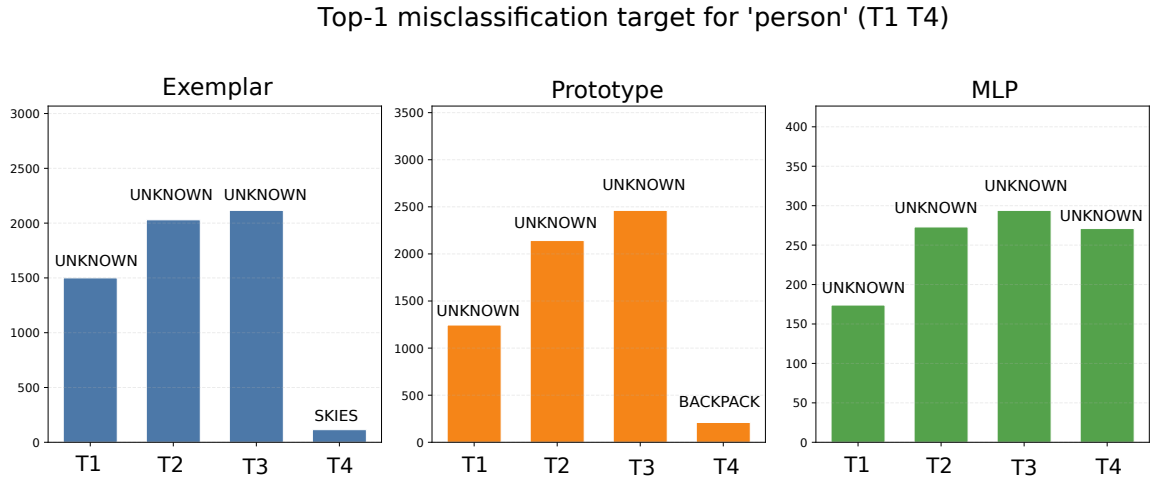


Figure 7.2: Most frequent wrong prediction for ground-truth *person* at each incremental episode (T1-T4), for exemplar, prototype, and MLP pipelines. Bars show the count and share of all person misclassifications.

Misclassification analysis further showed that *person* was predominantly predicted as *UNKNOWN* (up to 81% of exemplar errors and 50% of prototype errors at T3), suggesting that person instances lie in a sparse region of the DINO feature space, unlike more densely clustered classes such as *train* and *cat* (Figure 7.2).

7.1 Conclusion

The results indicate that incremental memory construction remains effective across multiple OWOD tasks. Prototype memory provides the most memory-efficient solution, whereas exemplar memory achieves the strongest open-world performance. The MLP baseline remains superior when known-class accuracy is the primary objective, but performs substantially worse at identifying previously unseen categories.

The improved open-world performance of exemplar memory comes at a substantial increase in memory size. Exemplar-based representations maintain known-class accuracy comparable to prototype memory while achieving consistently stronger unknown-object recognition across OWOD tasks.

Most importantly, both memory-based approaches preserve competitive recognition performance while supporting incremental updates without retraining from scratch. The current retrieval mechanism remains fixed across tasks and may be further improved through adaptive MHN inverse-temperature scheduling or task-specific refinement strategies.

Chapter 8

General Discussion

This thesis explores incremental open-set object recognition using frozen DINO features, prototype/exemplar memory, and open-set thresholding. The results show that this problem is challenging not only because unknown objects must be rejected, but also because known classes must remain separable under realistic object crops while still supporting the incremental addition of new classes and examples.

As observed in our experiments, although the MLP baseline performs well on known-category classification, its relatively low unknown recall suggests limited robustness under open-set conditions. This indicates that purely parametric classification tends to overfit toward known categories, which motivates the use of explicit memory-based retrieval and similarity-based inference in the proposed framework.

We are aware that our study has several limitations and that some aspects remain insufficiently developed, requiring additional future investigation. This is reflected in multiple components of the framework, ranging from dataset selection and feature representation to the open-set threshold calibration strategy.

8.1 Dataset and Feature Representation Limitations

Due to limited computational resources and the complexity of the problem, several research directions remain outside the scope of the present work. While this thesis primarily focuses on memory construction and associative retrieval mechanisms, future work should further investigate adaptive open-set calibration, detector integration, and end-to-end representation adaptation.

Starting from the dataset itself, as discussed in previous chapters, the COCO dataset selection was primarily guided by the OWOD problem setup, since our final goal is to evaluate the proposed framework integrated with a class-agnostic detector within a complete pipeline. Existing OWOD approaches commonly rely on the COCO benchmark as a standard for result comparison. This is a challenging benchmark as ob-

jects appear at many scales, often partially occluded, and frequently occur in crowded scenes. This makes the extracted crop features noisy, since a crop may contain not only the target object, but also background and contextual information, or parts of nearby objects. As a result, the feature embedding does not always represent a clean class identity.

This becomes especially problematic in the open-set setting because unknown rejection depends on whether known and unknown samples remain separable in feature space. If known classes already overlap, then an unknown object visually similar to a known class may be incorrectly accepted as known. At the same time, difficult known objects may be rejected as unknown. These effects become even more pronounced under incremental learning, where memory representations continuously evolve as new classes and samples are introduced. Additionally, some classes induce strong biases, not only because of large variations in appearance and pose, but also because of instance imbalance. Some classes are heavily overrepresented, while others remain underrepresented, which further affects the known-unknown recognition trade-off. Since we strictly followed the OWOD task and data-split protocol, we were aware of these limitations. Alternative balancing strategies or additional data filtering could potentially improve the results, but such approaches would require additional human-in-the-loop intervention, which we intentionally avoided in the present setting.

Regarding feature representations, DINO provides strong general-purpose visual features, but it is not trained specifically to separate the target OWOD classes. Therefore, visually similar categories can remain close in the embedding space. For example, classes such as *bus* and *truck*, *sofa* and *chair*, or *cat* and *dog* may produce similar feature vectors. This reduces the margin between classes and makes open-set thresholding more difficult.

A stronger visual backbone may partially improve this issue by producing more compact class clusters and larger separation between known categories. However, stronger features alone may not fully solve the problem, because some unknown classes are naturally close to known classes, especially under noisy and context-dependent COCO crops.

8.2 Open-Set Threshold Calibration

The current open-set thresholding strategy uses known-only calibration. A threshold is computed from known-class calibration on 10 000 samples using the 5th percentile of known scores. This represents a practical non-parametric approach that does not assume any specific distribution, and it allows the estimation of known-class confidence. We assume that known samples produce higher memory scores than truly unknown

samples and that a low known-score percentile can act as an open-set rejection boundary.

The 5th percentile is intentionally permissive. It keeps approximately 95% of known calibration samples above the threshold, reducing the risk of rejecting known objects as unknown. The calibration subset of 10 000 samples was selected as a practical compromise between computational cost and threshold stability. This setup is also more realistic for open-world scenarios, since the system does not have access to all future class instances during threshold calibration. Nevertheless, additional experiments and deeper analysis of calibration strategies should be conducted in future work.

At the same time, the global threshold has several limitations. A single threshold is applied across all classes, even though different classes may have different feature distributions. Some classes are visually compact and well represented, while others are diverse, rare, or strongly overlapping with other categories. As a result, a global threshold may become too strict for difficult classes and too permissive for easier classes. A natural extension of the current approach would be adaptive or per-class calibration. Instead of using one global threshold, each class could have its own threshold based on its score distribution. Classes with many examples and compact prototype clusters could use stricter thresholds, while classes with fewer examples or larger intra-class variation could use more permissive thresholds. This would allow the open-set gate to better reflect class-specific uncertainty and improve the flexibility of the system compared to a fixed global 5th percentile threshold. Future studies should investigate this problem in more depth.

Another limitation comes from margin scores, which measure how much the predicted class exceeds the strongest competing class. This can be useful because a low margin often indicates ambiguity. However, class overlap may produce small margins even for valid known samples. In such cases, the margin gate may incorrectly reject difficult known examples as unknown. Future work could investigate whether the margin should be adaptive, class-specific, or even omitted if it negatively affects known-class accuracy. Another possibility would be replacing the hard margin with a softer uncertainty measure, such as confidence score based on all competing classes rather than only the strongest rival.

8.3 MHN Retrieval and Score Consistency

Although MHN is expected to improve classification by aggregating memory information, in the current setting it can reduce known-class accuracy because the feature space contains overlapping classes and noisy crop embeddings. In such cases, MHN may redistribute confidence toward visually similar categories and select a class that

is not the strongest class under cosine similarity, while the open-set gate still relies on cosine-based scores. Consequently, when MHN and cosine rankings disagree, the margin may become small or negative, increasing misclassification and unknown rejection.

This MHN-versus-cosine mismatch represents one of the central limitations of the current memory-gating design. Although thresholds are recalibrated under the MHN setting, the classifier and open-set gates are still not fully score-consistent, since the gate variables are computed from cosine similarities evaluated at the MHN-selected class. Therefore, recalibration may reduce rejection mismatch, but it cannot fully prevent MHN-induced classification errors.

Another important direction for future work would be investigating the MHN sharpness parameter, which also affects classification behavior. Lower temperatures produce softer attention over many prototypes, while higher temperatures make the model focus more strongly on the closest prototypes. A single global temperature may therefore not be optimal for all classes. A promising direction could be the exploration of class-adaptive MHN sharpness. Classes with many examples or compact prototype clusters may benefit from sharper attention, while classes with fewer examples or larger intra-class variation may require softer aggregation. This could make memory retrieval more robust and better aligned with class-specific data availability.

8.4 Future Directions for Incremental OWOD

As for the incremental open-world setting, the current framework already provides a mechanism for memory construction and class representation building. Therefore, future extensions could apply the same approach to newly introduced classes using human-in-the-loop labeling. Future work could also investigate memory pruning, active sample selection, and online feature adaptation in order to improve long-term scalability and reduce memory growth during continual learning.

Conclusion

In this work, we aimed to integrate a theoretical overview of human memory and recognition with an experimental study of incremental visual learning. We examined how associative memory principles and human categorization mechanisms can contribute to incremental open-set visual recognition. The proposed conceptual and implemented memory-based framework combines MHNs with large-scale self-supervised visual representations extracted using a pretrained DINO encoder, providing insights into how memory structure and retrieval mechanisms affect open-set recognition.

Open-set recognition in our setting requires assigning each test embedding either to a known class or to an *unknown* category. Direct cosine retrieval relies on a single highest-similarity match, which becomes increasingly sensitive to spurious exemplars in large memory banks. We therefore evaluated a Modern Hopfield associative layer over the same frozen memory to test whether class-level aggregation of retrieved similarities improves robustness in OWOD tasks.

First, we studied different memory granularities and refinement techniques for building and updating memory banks, while connecting them to cognitive category theories and similarity-based retrieval of stored representations. We also treated memory size and compactness as important benchmarks. Prototype memory provides the best open-world trade-off under cosine retrieval. Online refinement is the main source of performance gains, but unconstrained refinement causes substantial memory growth. Exemplar memory improves known-class recognition but requires much larger memory banks and does not improve open-world performance proportionally.

Prototype memory with full-stream online EMA refinement provides the strongest open-world trade-off in our Task 1 cosine pipeline (P1-all, $H = 0.643$). Online refinement is the dominant source of improvement over static builds, but large crop budgets cause rapid and class-imbalanced memory growth, especially for the *person* class. Exemplar memory improves known-class accuracy at matched online budgets and can match or slightly exceed prototype H at the same crop cap, yet it requires much larger banks and does not reach full-stream prototype performance under our capped setting.

We also integrated a MHN-based classifier to aggregate multi-vector class information and optionally refine queries before open-set classification. We evaluated it against cosine max-pooling on the same fitted memories, tuned the retrieval sharpness

parameter β, and tested when it improves or harms the known-unknown harmonic mean.

Under DINO GT features on Task 1, MHN-based classification improves open-world performance mainly by increasing *unknown recall* while keeping known accuracy similar. The strongest setup is exemplar memory E2-09 with MHN classification at $\beta = 20$ (S3-10), which gives the best harmonic mean in our Task 1 experiments. In contrast, MHN-based refinement does not improve open-world recognition. It tends to pull unknown queries toward known memory vectors, reducing *unknown recall* and lowering H.

The experiments show that methods achieving the highest known-class accuracy do not necessarily achieve the best open-world performance. The MLP baseline achieves strong closed-set accuracy, but its *unknown recall* remains low under the open-set protocol. In contrast, memory-based models, especially with MHN-based classification, achieve a better known-unknown balance even when their *known* accuracy is lower. This supports the central motivation of the experiments: open-world recognition requires not only recognizing known classes, but also controlling false assignment of unknown objects to known labels.

In conclusion, results suggest that high-dimensional representations, combined with associative memory retrieval, provide a promising and interpretable foundation for incremental and open-set visual recognition based on non-parametric memory mechanisms.

Training MHN projection or scoring parameters may further improve retrieval quality, but this would introduce an additional parametric component and require retraining when the memory distribution changes. We therefore leave trained MHN variants for future work and focus here on non-parametric MHN retrieval. Since the final goal is to evaluate the pipeline within a complete OWOD setup, which also motivated the dataset selection, future work will integrate the memory-based classification pipeline with a pretrained Deformable DETR utilized as a class-agnostic detector. This would enable direct comparison with existing OWOD approaches under full detection settings.

Bibliography

Abel, D., Barreto, A., Roy, B., Precup, D., van Hasselt, H., and Singh, S. (2023). A definition of continual reinforcement learning.

Achille, A., Rovere, M., and Soatto, S. (2019). Critical learning periods in deep neural networks.

Amari, S.-I. (1972). Learning patterns and pattern sequences by self-organizing nets of threshold elements. *IEEE Transactions on Computers*, C-21(11):1197–1206.

Anderson, J. R. (1990). *The Adaptive Character of Thought*. Psychology Press, 1 edition.

Ashby, F. G., Alfonso-Reese, L. A., Turken, A. U., and Waldron, E. M. (1998). A neuropsychological theory of multiple systems in category learning. *Psychological Review*, 105(3):442–481.

Ashby, F. G. and Gott, R. E. (1988). Decision rules in the perception and categorization of multidimensional stimuli. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 14(1):33–53.

Ashby, F. G. and Valentin, V. V. (2005). *Multiple Systems of Perceptual Category Learning: Theory and Cognitive Tests*, pages 547–572. Elsevier Science Ltd.

Ashby, S. R. and Zeithamova, D. (2022). Category-biased neural representations form spontaneously during learning that emphasizes memory for specific instances. *Journal of Neuroscience*, 42(5):865–876.

Atkinson, R. C. and Shiffrin, R. M. (1971). The control of short-term memory. *Scientific American*, 225(2):82–91.

Baddeley, A. (1992). Working memory. *Science*, 255:556–559.

Barker, R. A., Cicchetti, F., and Neal, M. J. (2012). *Neuroanatomy and Neuroscience at a Glance*. John Wiley & Sons, 4 edition.

Barnett, S. M. and Ceci, S. J. (2002). When and where do we apply what we learn?: A taxonomy for far transfer. *Psychological Bulletin*, 128(4):612–637.

Barsalou, L. W., Huttenlocher, J., and Lamberts, K. (1998). Basing categorization on individuals and events. *Cognitive Psychology*, 36(3):203–272.

Bartlett, F. C. and Burt, C. (1933). Remembering: A study in experimental and social psychology. *British Journal of Educational Psychology*, 3(2):187–192.

Bear, M. F., Connors, B. W., and Paradiso, M. A. (2006). *Neuroscience*. Lippincott Williams and Wilkins, 3 edition.

Bellman, R. (1957). *Dynamic Programming*. Rand Corporation research study. Princeton University Press.

Bendale, A. and Boult, T. (2016). Towards open set deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Bengio, Y., Louradour, J., Collobert, R., and Weston, J. (2009). Curriculum learning. *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, pages 41–48.

Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94(2):115–147.

Bliss, T. V. P. and Cooke, S. F. (2011). Long-term potentiation and long-term depression: a clinical perspective. *Clinics*, 66(Supplement 1):3–17.

Bowman, C. R., Iwashita, T., and Zeithamova, D. (2020). Tracking prototype and exemplar representations in the brain across learning. *eLife*, 9:e59360.

Cangea, C., Veličković, P., and Liò, P. (2020). Xflow: Cross-modal deep neural networks for audiovisual classification. *IEEE Transactions on Neural Networks and Learning Systems*, 31(9):3711–3720.

Cao, Y., Chen, F., Wang, H., et al. (2026). Neural mechanisms of feature binding in working memory. *Communications Biology*, 9:270.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. (2021). Emerging properties in self-supervised vision transformers.

Carpenter, K. L., Wills, A. J., Benattayallah, A., and Milton, F. (2016). A comparison of the neural correlates that underlie rule-based and information-integration category learning. *Human Brain Mapping*, 37(10):3557–3574.

Caruana, R. (1997). Multitask learning. *Machine Learning*, 28(1):41–75.

Cermelli, F., Mancini, M., Bulò, S. R., Ricci, E., and Caputo, B. (2020). Modeling the background for incremental learning in semantic segmentation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9230–9239. IEEE.

Chater, N., Oaksford, M., Hahn, U., and Heit, E. (2010). Bayesian models of cognition. *Wiley Interdisciplinary Reviews: Cognitive Science*, 1(6):811–823.

Cincotta, M. and Seger, C. A. (2007). Dissociation between striatal regions while learning to categorize via feedback and via observation. *Journal of Cognitive Neuroscience*, 19(2):249–265.

Clark, A. (2013). The many faces of precision (replies to commentaries on "whatever next? neural prediction, situated agents, and the future of cognitive science"). *Frontiers in Psychology*, 4:270.

Clouter, A., Shapiro, K. L., and Hanslmayr, S. (2017). Theta phase synchronization is the glue that binds human associative memory. *Current Biology*, 27(20):3143–3148.e6.

Cohen, M. A. and Grossberg, S. (1983). Absolute stability of global pattern formation and parallel memory storage by competitive neural networks. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(5):815–826.

Conrad, R. (1964). Acoustic confusions in immediate memory. *British Journal of Psychology*, 55:75–84.

Cunningham, J. and Yu, B. (2014). Dimensionality reduction for large-scale neural recordings. *Nature Neuroscience*, 17:1500–1509.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Darwin, C. J., Turvey, M. T., and Crowder, R. G. (1972). An auditory analogue of the sperling partial report procedure: Evidence for brief auditory storage. *Cognitive Psychology*, 3(2):255–267.

Demircigil, M., Heusel, J., Löwe, M., Upgang, S., and Vermet, F. (2017). On a model of associative memory with huge storage capacity. *Journal of Statistical Physics*, 168(2):288–299.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale.

Doya, K., Ishii, S., Rao, R. P. N., and Pouget, A. (2011). *Bayesian Brain: Probabilistic Approaches to Neural Coding*. Computational neuroscience. MIT Press.

Edwards, F. A. (1995). Ltp–a structural model to explain the inconsistencies. *Trends in Neurosciences*, 18(6):250–255.

Elman, J. L. (1993). Learning and development in neural networks: the importance of starting small. *Cognition*, 48(1):71–99.

Ernst, M. O. and Bülthoff, H. H. (2004). Merging the senses into a robust percept. *Trends in Cognitive Sciences*, 8(4):162–169.

Fanselow, M. S. and Poulos, A. M. (2005). The neuroscience of mammalian associative learning. *Annual Review of Psychology*, 56:207–234.

French, R. (1999). Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135.

Friedenberg, J. and Silverman, G. (2006). *Cognitive Science: An Introduction to the Study of the Mind*. Sage Publications Ltd.

Gazerani, P. (2025). The neuroplastic brain: Current breakthroughs and emerging frontiers. *Brain Research*, 1858:1–15.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks.

Gupta, A., Narayan, S., Joseph, K. J., Khan, S., Khan, F. S., and Shah, M. (2022). Ow-detr: Open-world detection transformer.

Hadsell, R., Rao, D., Rusu, A. A., and Pascanu, R. (2020). Embracing change: Continual learning in deep neural networks. *Trends in Cognitive Sciences*, 24(12):1028–1040.

Hafting, T., Fyhn, M., Molden, S., et al. (2005). Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436:801–806.

Hampton, J. (2015). *Categories, prototypes and exemplars*, pages 124–141. Routledge.

Hassabis, D., Kumaran, D., Summerfield, C., and Botvinick, M. (2017). Neuroscience-inspired artificial intelligence. *Neuron*, 95(2):245–258.

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

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition.

Hebb, D. O. (1949). *The Organization of Behavior: A Neuropsychological Theory*. John Wiley and Sons, New York.

Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *Science*, 313:504–507.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558.

Hospedales, T., Antoniou, A., Micaelli, P., and Storkey, A. (2020). Meta-learning in neural networks: A survey.

Hubel, D. H. and Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex. *The Journal of Physiology*, 148(3):574–591.

Hubel, D. H. and Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1):106–154.

Joseph, K. J., Khan, S., Khan, F. S., and Balasubramanian, V. N. (2021). Towards open world object detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5826–5836. IEEE.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016). Bag of tricks for efficient text classification.

Kelly, C. and Castellanos, F. X. (2014). Strengthening connections: Functional connectivity and brain plasticity. *Neuropsychology Review*, 24:63–76.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., and Hadsell, R. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.

Kosslyn, S. M. and Shwartz, S. P. (1977). A simulation of visual imagery. *Cognitive Science*, 1(3):265–295.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90.

Krotov, D., Hoover, B., Ram, P., and Pham, B. (2025). Modern methods in associative memory.

Krotov, D. and Hopfield, J. J. (2016). Dense associative memory for pattern recognition.

Krueger, K. A. and Dayan, P. (2009). Flexible shaping: How learning in small steps helps. *Cognition*, 110(3):380–394.

Kruschke, J. K. (1992). Alcove: an exemplar-based connectionist model of category learning. *Psychological Review*.

Kudithipudi, D., Aguilar-Simon, M., Babb, J., Bazhenov, M., Blackiston, D., Bongard, J., Brna, A. P., Raja, S. C., Cheney, N., Clune, J., Daram, A., Fusi, S., Helfer, P., Kay, L., Ketz, N., Kira, Z., Kolouri, S., Krichmar, J. L., Kriegman, S., and Levin, M. (2022). Biological underpinnings for lifelong learning machines. *Nature Machine Intelligence*, 4(3):196–210.

Kumaran, D., Hassabis, D., and McClelland, J. L. (2016). What learning systems do intelligent agents need? complementary learning systems theory updated. *Trends in Cognitive Sciences*, 20(7):512–534.

Lashley, K. S. (1950). In search of the engram. In *Physiological mechanisms in animal behavior. Society's Symposium IV*, pages 454–482. Academic Press.

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

Li, J., Park, E., Zhong, L. R., and Chen, L. (2019). Homeostatic synaptic plasticity as a metaplasticity mechanism — a molecular and cellular perspective. *Current Opinion in Neurobiology*, 54:44–53.

Li, M., Lu, S., Li, J., and Zhong, N. (2010). The role of the parahippocampal cortex in memory encoding and retrieval: An fmri study. In *Brain Informatics*.

Li, S., Su, T., Zhang, X.-Y., and Wang, Z. (2025). Continual learning with knowledge distillation: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 36(6):9798–9818.

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., and Dollár, P. (2015). Microsoft coco: Common objects in context.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision (ICCV)*, pages 1150–1157. IEEE.

Marr, D. (1982). *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. MIT Press, Cambridge, MA.

Marr, D. C. and Poggio, T. (2004). From understanding computation to understanding neural circuitry. *Neurosciences Research Program Bulletin*, 15.

McClelland, J. L., McNaughton, B. L., and O'Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457.

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space.

Miller, J. F. et al. (2013). Neural activity in human hippocampal formation reveals the spatial context of retrieved memories. *Science*, 342:1111–1114.

Millidge, B., Seth, A., and Buckley, C. L. (2022). Predictive coding: a theoretical and experimental review.

Minda, J. and Smith, J. (2001). Prototypes in category learning: The effects of category size, category structure, and stimulus complexity. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 27(3):775–799.

Minda, J. P., Roark, C. L., Kalra, P., et al. (2024). Single and multiple systems in categorization and category learning. *Nature Reviews Psychology*, 3:536–551.

Nash, W., Drummond, T., and Birbilis, N. (2018). A review of deep learning in the study of materials degradation. *npj Materials Degradation*, 2.

Newell, A. and Simon, H. A. (1972). *Human Problem Solving*. Prentice-Hall.

Nomura, E., Maddox, W. T., Ing, A., Gitelman, D., Parrish, T., Mesulam, M.-M., and Reber, P. A. (2007). Neural correlates of rule-based and information-integration visual category learning. *Cerebral Cortex*, 17(1):37–43.

Nosofsky, R. M. (2011). *The generalized context model: an exemplar model of classification*, pages 18–39. Cambridge University Press.

Nosofsky, R. M. and Johansen, M. K. (2000). Exemplar-based accounts of "multiple-system" phenomena in perceptual categorization. *Psychonomic Bulletin & Review*, 7:375–402.

Ojala, T., Pietikäinen, M., and Harwood, D. (1994). Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In *Proceedings of the 12th International Conference on Pattern Recognition (ICPR)*, pages 582–585.

O'Keefe, J. and Dostrovsky, J. (1971). The hippocampus as a spatial map: Preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, 34:171–175.

O'Reilly, R. C. and Rudy, J. W. (2000). Computational principles of learning in the neocortex and hippocampus. *Hippocampus*, 10(4):389–397.

Paivio, A. (1979). *Imagery and Verbal Processes*. Psychology Press, 1 edition.

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

Pavlov, I. and Anrep, G. (1927). *Conditioned Reflexes: An Investigation of the Physiological Activity of the Cerebral Cortex*. Oxford University Press: Humphrey Milford.

Pezzulo, G., Kemere, C., and van der Meer, M. A. A. (2017). Internally generated hippocampal sequences as a vantage point to probe future-oriented cognition. *Annals of the New York Academy of Sciences*, 1396(1):144–165.

Piaget, J. (1959). *The Language and Thought of the Child*. International Library of Psychology, Philosophy and Scientific Methods. Routledge.

Pouget, A., Dayan, P., and Zemel, R. (2000). Information processing with population codes. *Nature Reviews Neuroscience*, 1:125–132.

Power, J. D. and Schlaggar, B. L. (2016). Neural plasticity across the lifespan. *Wiley Interdisciplinary Reviews: Developmental Biology*, 6(1):e216.

Quillian, M. R. (1968). *Semantic Networks*, page 17. MIT Press, Cambridge, MA.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning transferable visual models from natural language supervision.

Ramsauer, H., Schäfl, B., Lehner, J., Seidl, P., Widrich, M., Adler, T., Gruber, L., Holzleitner, M., Pavlović, M., Sandve, G. K., Greiff, V., Kreil, D., Kopp, M., Klambauer, G., Brandstetter, J., and Hochreiter, S. (2021). Hopfield networks is all you need.

Rao, R. P. N. and Ballard, D. H. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2:79–87.

Ravi, N., Gabeur, V., Hu, Y.-T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., Mintun, E., Pan, J., Alwala, K. V., Carion, N., Wu, C.-Y., Girshick, R., Dollár, P., and Feichtenhofer, C. (2024). Sam 2: Segment anything in images and videos.

Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks.

Reisberg, D. (2013). *Cognition: Exploring the Science of the Mind*. W. W. Norton & Company, New York, 5 edition.

Rosch, E. H. (1973). Natural categories. *Cognitive Psychology*, 4(3):328–350.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.

Rouder, J. N. and Ratcliff, R. (2006). Comparing exemplar- and rule-based theories of categorization. *Current Directions in Psychological Science*, 15(1):9–13.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323:533–536.

Rumelhart, D. E. and McClelland, J. L. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1. Foundations*. MIT Press.

Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., and Hadsell, R. (2016). Progressive neural networks. *arXiv preprint arXiv:1606.04671*.

Sakurai, Y. and Takahashi, S. (2008). Dynamic synchrony of local cell assembly. *Reviews in the Neurosciences*, 19(6):425–440.

Salvatori, T., Song, Y., Hong, Y., Frieder, S., Sha, L., Xu, Z., Bogacz, R., and Lukasiewicz, T. (2021). Associative memories via predictive coding.

Sanger, T. D. (1994). Neural network learning control of robot manipulators using gradually increasing task difficulty. *IEEE Transactions on Robotics and Automation*, 10(3):323–333.

Saxe, A., Nelli, S., and Summerfield, C. (2020). If deep learning is the answer, what is the question? *Nature Reviews Neuroscience*, 22.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45:2673–2681.

Selfridge, O. G. (1958). Pandemonium: a paradigm for learning. In *Mechanisation of Thought Processes: Proceedings of a Symposium Held at the National Physical Laboratory, November 1958*, pages 513–526. HMSO.

Shepard, R. N. and Metzler, J. (1971). Mental rotation of three-dimensional objects. *Science*, 171:701–703.

Shin, H., Lee, J. K., Kim, J., and Kim, J. (2017). Continual learning with deep generative replay.

Shmelkov, K., Schmid, C., and Alahari, K. (2017). Incremental learning of object detectors without catastrophic forgetting. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3420–3429. IEEE.

Siméoni, O., Vo, H. V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., Massa, F., Haziza, D., Wehrstedt, L., Wang, J., Darcet, T., Moutakanni, T., Sentana, L., Roberts, C., Vedaldi, A., Tolan, J., Brandt, J., Couprie, C., Mairal, J., Jégou, H., Labatut, P., and Bojanowski, P. (2025). Dinov3.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition.

Singer, W. (1996). *Neuronal synchronization: A solution to the binding problem?*, pages 100–130. MIT Press.

Skinner, B. F. (2019). *The Behavior of Organisms: An Experimental Analysis*. B. F. Skinner Foundation.

Smith, E. E. and Medin, D. L. (2002). *The exemplar view*, pages 277–292. MIT Press.

Snell, J., Swersky, K., and Zemel, R. S. (2017). Prototypical networks for few-shot learning.

Squire, L. R., Wixted, J. T., and Clark, R. E. (2007). Recognition memory and the medial temporal lobe: a new perspective. *Nature Reviews Neuroscience*, 8(11):872–883.

Suzuki, W. A. (2008). *Associative learning signals in the brain*, volume 169, pages 305–320. Elsevier.

Takeuchi, S., Mima, T., Murai, R., Shimazu, H., Isomura, Y., and Tsujimoto, T. (2015). Gamma oscillations and their cross-frequency coupling in the primate hippocampus during sleep. *Sleep*, 38(7):1085–1091.

Tang, M., Salvatori, T., Millidge, B., Song, Y., Lukasiewicz, T., and Bogacz, R. (2023). Recurrent predictive coding models for associative memory employing covariance learning. *PLOS Computational Biology*, 19(4):e1010719.

Tani, J. (2016). *Exploring Robotic Minds: Actions, Symbols, and Consciousness as Self-Organizing Dynamic Phenomena*. Oxford University Press.

Tenenbaum, J. B. et al. (2011). How to grow a mind: Statistics, structure, and abstraction. *Science*, 331:1279–1285.

Thagard, P. (2005). *Mind: Introduction to Cognitive Science*. Boston Review, 2 edition.

Thorndike, E. L. (1898). Animal intelligence: An experimental study of the associative processes in animals. *The Psychological Review: Monograph Supplements*, 2(4):i–109.

Thrun, S. and Mitchell, T. M. (1995). Lifelong robot learning. *Robotics and Autonomous Systems*, 15(1-2):25–46.

Tocco, G., Maren, S., Shors, T. J., Baudry, M., and Thompson, R. F. (1992). Long-term potentiation is associated with increased [3h]ampa binding in rat hippocampus. *Brain Research*, 573(2):228–234.

Treisman, A. (1986). Features and objects in visual processing. *Scientific American*, 255(5):114–125.

Treisman, A. M. and Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136.

Urbán, N. and Guillemot, F. (2014). Neurogenesis in the embryonic and adult brain: same regulators, different roles. *Frontiers in Cellular Neuroscience*, 8:396.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, pages 6000–6010. Curran Associates Inc.

Ven, G. M., Siegelmann, H. T., and Tolias, A. S. (2020). Brain-inspired replay for continual learning with artificial neural networks. *Nature Communications*, 11(1):1–14.

Vitureira, N., Letellier, M., and Goda, Y. (2012). Homeostatic synaptic plasticity: from single synapses to neural circuits. *Current Opinion in Neurobiology*, 22(3):516–521.

Wagner, A. D., Shannon, B. J., Kahn, I., and Buckner, R. L. (2005). Parietal lobe contributions to episodic memory retrieval. *Trends in Cognitive Sciences*, 9(9):445–453.

Wang, J. H. (2019). Searching basic units in memory traces: associative memory cells. *F1000Research*, 8:457.

Wang, J. H. and Cui, S. (2017). Associative memory cells: Formation, function and perspective. *F1000Research*, 6:283.

Wang, L., Zhang, X., Su, H., and Zhu, J. (2023). A comprehensive survey of continual learning: Theory, method and application. *arXiv*.

Wang, Y., Yao, Q., Kwok, J., and Ni, L. M. (2020). Generalizing from a few examples: A survey on few-shot learning.

Wang, Z., Liu, L., Duan, Y., Kong, Y., and Tao, D. (2022). Continual learning with lifelong vision transformer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 171–181. IEEE.

Wickens, D. D. (1973). Some characteristics of word encoding. *Memory & Cognition*, 1:485–490.

Wilson, F. A., Scalaidhe, S. P., and Goldman-Rakic, P. S. (1993). Dissociation of object and spatial processing domains in primate prefrontal cortex. *Science*, 260(5116):1955–1958.

Worthen, J. B. and Hunt, R. R. (2011). *Mnemonology: Mnemonics for the 21st Century*. Essays in Cognitive Psychology. Taylor & Francis.

Wu, J., Wang, Y., Xue, T., Sun, X., Freeman, W. T., and Tenenbaum, J. B. (2017). Marrnet: 3d shape reconstruction via 2.5d sketches.

Yamasaki, T. and Tobimatsu, S. (2011). *Motion Perception in Healthy Humans and Cognitive Disorders*, pages 156–161. IGI Global.

Zheqi, Y., Zahid, A., Taha, A., Taylor, W., Kernec, J. L., Heidari, H., Imran, M., and Abbasi, Q. (2022). An intelligent implementation of multi-sensing data fusion with neuromorphic computing for human activity recognition. *IEEE Internet of Things Journal*, PP:1–1.

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., and Dai, J. (2021). Deformable detr: Deformable transformers for end-to-end object detection.

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., and He, Q. (2021). A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76.

Appendix A: Source code

The source code is available at `https://github.com/JexEpx/JE_MT.git`.