Inter-Layer Recurrent Hebbian Feedback Enhances Neural Network Robustness

Yan Li†, Mingyue Qin†, Shuyu Yin, Peilin Liu, Fei Wen
School of Integrated Circuits (School of Information Science and Electronic Engineering)
Shanghai Jiao Tong University
Shanghai, China
floraaa@alumni.sjtu.edu.cn

***Abstract*—The human brain exhibits extraordinary robustness and adaptability in noisy and out-of-distribution (OOD) scenarios. Prior studies suggest that this capability is largely attributable to rich feedback pathways and dynamic synaptic plasticity. Inspired by these neuro-scientific insights, in this paper we introduce ReHNet, which is a recurrent neural model featuring inter-layer feedback connections and partially plastic fast weights. Unlike traditional feedforward-only networks, ReHNet introduces inter-layer feedback in certain layers. Additionally, ReHNet is a partially plastic model, in which the synaptic weights are decomposed into a fixed component and a data-dependent short-term plastic component that is dynamically updated online via the Hebbian rule. We show that such local Hebbian plasticity, combined with feedback connections, can enhance the network's sensitivity to correlated activations and enhance resilience to distributional shifts. This mechanism is especially beneficial under noisy and OOD conditions, where feature representations are prone to degradation. On corrupted datasets like CIFAR-10C and CIFAR-100C, ReHNet significantly improves the performance and even surpasses domain adaptation methods that rely on backpropagation-based optimization. In OOD evaluations on the PACS dataset, ReHNet also demonstrates excellent robustness. Overall, our results show that incorporating biologically inspired feedback loops and online Hebbian learning can markedly enhance the robustness to noise and OOD data, which may provide insights for both neuroscience research and the development of more robust and adaptive neural networks.**

***Index Terms*—Hebbian plasticity, recurrent feedback networks, biologically inspired learning**

I. Introduction

The primate visual cortex can recognize objects rapidly and with remarkable resilience to sensory noise, occlusion, and contextual change. These capabilities still surpass those of modern artificial intelligence systems. Neuroscientific anatomical and electro-physiological studies suggest that this robustness relies on the brain's dense web of feedforward, lateral, and especially feedback connections that recursively refine cortical representations over short timescales.

The researches in neuroscience show that top-down pathways are widespread in the brain [1]–[5]. It has been well recognized that recurrent signalling is indispensable for hard cases of perception. Human psychophysics and MEG recordings show that disrupting early feedback loops impairs recognition of low-contrast or cluttered images [6], [7]. Electrophysiology further reveals that ventral-stream neurons depend on local recurrence to complete partially occluded shapes and to correct noisy inputs [8], [9]. Inspired by these findings, computer-vision models that explicitly unroll cortical-like loops have shown to close the gap to human performance. For example, shallow recurrent convolutional networks can match deep feed-forward CNNs with fewer parameters [10], whilst top-down visual-attention controllers can improve recognition in clutter [11]. However, most existing architectures retain fixed synapses during inference and lack the brain's capability for on-the-fly adaptation.

The researches in neuroscience also point to fast synaptic plasticity, Hebbian updates driven by a few spikes, as a key mechanism for contextual recalibration and working memory. Early computational models equipped networks with fast weights that were refreshed by Hebbian rules to deblur activations after interference [12]. Recent works have made such plasticity differentiable end-to-end, which allow networks to learn how much and where to adapt via gradient descent [13], [14]. Yet most of these studies update only lateral or feedforward models, leaving the interaction between inter-layer feedback and Hebbian plasticity largely unexplored.

In this work, we introduce a novel biologically-motivated model architecture, namely ReHNet, which integrates explicit inter-layer feedback pathways with data-dependent plastic components on synapse weight. Specifically, we consider a partially plastic model, in which each weight is decomposed into a fixed parameter and a rapidly updated Hebbian trace that is refreshed during the forward pass only. The update rule follows a normalized Hebbian equation that strengthens correlations between co-active pre- and post-synaptic units, which enables the network to strengthen features consistent with previously seen patterns and suppress distractors. Experiments on corrupted datasets like CIFAR-10C and CIFAR-100C, and OOD generalization benchmark PACS, demonstrate that ReHNet can significantly improve the performance under OOD conditions and even surpasses domain adaptation methods that rely on backpropagation-based optimization.

The main contributions of this paper are as follows.

- We integrate inter-layer recurrent feedback with differentiable Hebbian plasticity to design a compact biologically-plausible model, namely ReHNet. We show

This work was supported by the Science and Technology Innovation (STI) 2030–Major Project under Grant 2022ZD0208700.

† Equal contribution.

the Hebbian trace can be interpreted as an implicit energy minimizer that biases the network toward stable attractors representing previously co-occurring features.
- Extensive evaluation show that ReHNet is significantly more robust than feedforward model, and even surpasses domain adaptation methods that rely on backpropagation-based optimization.
- Practical impact: as ReHNet requires only a forward pass and local updates, it is well suited for energy-constrained edge devices and neuromorphic hardware.

Our results demonstrate that combining cortical-style feedback with fast Hebbian plasticity provides a powerful foundation for noise-tolerant and OOD perception, which points to a promising direction for next-generation brain-inspired adaptive neural networks. We will make our code available upon acceptance.

II. Related Work

a) Recurrent Feedback in Visual Recognition: Researches in neuroscience and machine learning highlight the importance of recurrent feedback for robust object recognition. The works [6] and [7] show that recurrent feedback in hierarchical visual models enables robust perception under occlusion and noise, which reflects cortical mechanisms for pattern completion and feature grouping. The works [8] and [9] provide empirical and computational evidence that recurrent circuits in the primate ventral stream are critical for recognizing challenging or degraded images. In artificial neural network, [11] show that recurrent attention models can improve efficiency and recognition in cluttered scenes. Furthermore, [10] demonstrate that shallow recurrent CNNs with feedback can match the accuracy and temporal dynamics of deeper feedforward models while more closely mirroring primate neural responses. However, these models typically employ static synaptic weights during inference, which limits their capacity for rapid adaptation to new conditions. Our work addresses this gap by combining inter-layer recurrent feedback with dynamic input-dependent synaptic plasticity.

b) Hebbian Plasticity and Fast Weight Mechanisms: Hebbian plasticity rules update synaptic weights based on locally correlated pre- and post-synaptic activity, which has long been recognized as a biologically grounded mechanism for fast adaptation and online memory in neural systems [15]–[25]. Early computational models, such as [12] fast weights with partial plasticity, allow networks to rapidly encode and recall recent patterns, enabling single-shot memory and deblurring of noisy cues. Moreover, the work [13] introduces differentiable plasticity, where each synaptic weight comprises a fixed and a Hebbian plastic component, trained jointly via backpropagation. This framework has demonstrated improved meta-learning and one-shot adaptation capabilities. [14] further incorporates neuromodulated plasticity to allow networks to learn when and where to adapt by using differentiable modulatory signals. It has shown improved effectiveness on sequential and memory-based tasks. [26] further considers differentiable plasticity in spiking neural networks. Despite these advances, most Hebbian models have been applied to simple tasks or recurrent architectures, with limited exploration in deep vision networks. Short-term plasticity models, such as [27], further extend these ideas to enable networks to learn to adapt and forget on short timescales. While these mechanisms have shown promise for rapid adaptation and continual learning, their application in conjunction with inter-layer recurrent feedback in large-scale vision tasks remains underexplored.

Unlike previous studies that focus solely on recurrence or plasticity, our approach integrates both to achieve intrinsic capability of robustness and adaptability. The recurrent Hebbian mechanism allows the network to recalibrate its internal representations for each input, without explicit test-time optimization or access to batches of target data, as required by standard domain adaptation methods. This results in inherent robustness and flexibility, allowing rapid, online adaptation to noise and distribution shifts from a single example, thus bridging biological plausibility and practical robustness for OOD generalization.

III. ReHNet: Integrating Feedback Pathways with Fast Hebbian Plasticity

A. Motivation

Biological neural systems exhibit a remarkable capacity for rapid adaptation to new inputs while retaining long-term knowledge. A key mechanism underlying this flexibility is *synaptic plasticity*—activity-dependent changes in the strength of neuronal connections [28]. Such changes are often driven by local co-activation of pre- and post-synaptic neurons, enabling fast, context-sensitive modulation without relying on global error signals.

Cognitive neuroscience further suggests the existence of two complementary learning systems: one for fast adaptation to novel experiences and another for gradual integration of knowledge over time [29]. This dual-process theory, known as Complementary Learning Systems (CLS), highlights the importance of combining short-term, input-specific plasticity with long-term, stable memory to support continual learning and generalization.

In contrast, standard deep neural networks operate with static parameters during inference, limiting their ability to cope with distribution shifts. Recent test-time adaptation (TTA) methods, such as TTT [30], TENT [31], and SITA [32], mitigate this issue by updating model parameters on-the-fly using test data. However, these methods typically require backpropagation during inference and assume access to mini-batches to compute reliable statistics or gradients. As a result, they perform poorly in online or single-sample settings, and their mechanisms deviate substantially from biological plausibility.

Our proposed model, ReHNet, is grounded in biological principles and designed for forward-only, per-sample adaptation. It introduces data-dependent Hebbian plasticity during inference, allowing the network to adapt to each input without tuning or gradient computation. This is achieved by decomposing each weight into a fixed (slow) component and

a plastic (fast) Hebbian trace, which is updated based on local correlations during the forward pass. ReHNet thus aligns with the CLS framework by combining fast adaptation with stable, slow weights.

B. Architecture of ReHNet

Convolutional neural networks (CNNs) initially drew inspiration from the neural structures of the mammalian visual system in their early efforts for object recognition, and have since been used as models for simulating feedforward computations in the ventral visual pathway of primates [33]. However, CNNs are not structurally or architecturally similar to the circuits of biological brains. First, they correspond to relatively shallow cortical layers, while advanced artificial neural networks tend to be much deeper and more complex. Second, CNNs overlook the massive presence of feedback connections in real neural circuits, which are essential for processing degraded or ambiguous stimuli.

In this work, we incorporate brain-inspired structures, such as recurrence and shallow-depth constraints, into the network architecture to improve the functional resemblance of the model to the brains of primates.

1) Model Architecture with Feedback Pathways.: ReHNet is a recurrent convolutional network in which every block receives bottom-up input and a top-down feedback signal from its own previous state. Formally, the activation of layer l at unrolled step t is

$$x_t^l = \varphi\big(W^l[\,x_t^{l-1} + x_{t-1}^l] + b^l\big), \tag{1}$$

where φ denotes the post-convolution operations (normalisation + ReLU), W^l denotes the bottom-up convolutional weights at layer l, and b^l represents the bias. The network recurrent for T steps, and a task loss (e.g., cross-entropy) is computed on the final step.

2) Partially Plastic Fast Weight with Hebbian Update Rule.: In the proposed Hebbian model, each synaptic weight is composed of a fixed component and a plastic component. Specifically, for any two neurons i and j, the synaptic connection consists of a fixed part denoted by W_{ij} and a plastic part modulated by a Hebbian trace Hebb_{ij}. The Hebbian rule provides an effective implementation mechanism for plastic weight modulation. On the one hand, the Hebbian mechanism embodies associative learning observed in the mammalian visual cortex, reflecting the biological basis of synaptic plasticity. On the other hand, Hebbian modulation enables the network to dynamically adjust connection strengths based on neural activities, capturing the associative relevance between specific stimuli and facilitating task-adaptive learning.

During inference, the Hebbian trace serves as a complete trajectory of lifetime memory, from early unsupervised experience to current task execution. The Hebbian trace is a dynamic variable that evolves during the lifespan and updates continuously based on input-output neural activity. In the simplest case, the Hebbian trace is accumulated through the correlation between presynaptic and postsynaptic activations and updated by averaging their product over time.

The strength of the plastic component is controlled by a modulation coefficient α_{ij}, which determines the contribution of Hebbian modulation to each synaptic connection. The effective weight at time t is

$$\tilde{W}_{ij}^l = W_{ij}^l + \alpha_{ij}^l\,\text{Hebb}_{ij,t}^l, \tag{2}$$

The trace is updated online by a local Hebbian rule

$$\text{Hebb}_{ij,t}^l = \text{Hebb}_{ij,t-1}^l + \eta^l\,\tilde{x}_{j,t}^l\,x_{i,t}^{l-1}, \tag{3}$$

where $\tilde{x}_t^l = \sigma\left(x_t^l\right)$ is the postsynaptic activation and σ is a nonlinear function for lateral inhibition. This formulation allows Hebbian modulation to be dynamically integrated into the total synaptic weight. The plastic learning rate η controls the Hebbian growth. In contrast, more sophisticated Hebbian rules are capable of maintaining stable memory traces in the absence of external stimulation [23], enabling long-term memory formation while mitigating the risk of catastrophic forgetting. A well-known example is the Oja rule [17], which can be expressed as a modified version of (3) as follows:

$$\text{Hebb}_{ij,t}^l = \text{Hebb}_{ij,t-1}^l + \eta^l\tilde{x}_{j,t}^l\left(x_{i,t}^{l-1} - \tilde{x}_{j,t}^l\text{Hebb}_{ij,t-1}^l\right) \tag{4}$$

The Hebbian fast trace Hebb_{ij} is initialized to zero at the beginning of each lifetime cycle and is dynamically updated according to the aforementioned rule during the cycle. It serves purely as an intra-lifetime variable. In contrast, the slow weights W, b, plasticity coefficient α and plasticity learning rate η are structural parameters of the network, which remain fixed throughout the lifetime. These parameters are optimized across multiple lifetimes via gradient descent to maximize the overall performance of the network.

During training, only parameters W, b, α and η receive gradient updates. the Hebbian traces are not backpropagated, keeping computation lightweight (see Algorithm 1). During inference, this design enables the model to adapt to each input using forward-only updates, incurring an additional per-sample memory cost $O(|W|)$ for storing Hebbian traces, without any backward computation overhead. In the training of ReHNet, fixed weights are trained in batch mode, while plastic weights are updated on a per-sample basis. From the perspective of computational efficiency, we still employ mini-batch training. However, within each mini-batch, the Hebbian traces for individual samples are updated independently. This design enables the plastic components to adaptively learn and respond to each input sample.

3) Single sample Consideration.: Most existing domain adaptation methods rely on batch normalization (BN) by recalculating mean and variance statistics on the target domain. However, BN's effectiveness significantly degrades when the batch size is small, especially in single-sample scenarios where statistical estimates become unreliable. To address this, we adopt Group Normalization (GN) [34], which computes statistics over feature channels and remains stable across varying batch sizes.

Furthermore, to ensure effective adaptation at the single-sample level, our recurrent Hebbian model incorporates data-

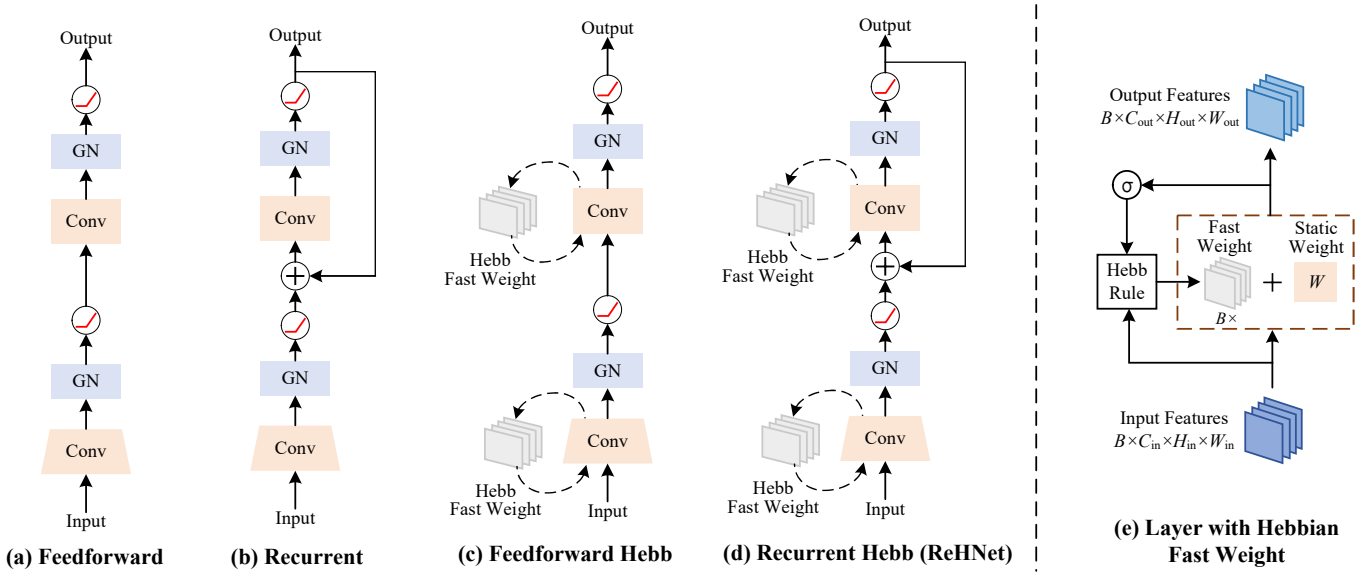

Fig. 1. Block comparison between ReHNet and feedforward, recurrent, and feedforward Hebb architectures.

Algorithm 1 Training Procedure of ReHNet

Require: dataset $\mathcal{D}$, number of layers L, unfolding steps T
Ensure: static weights W, b, plasticity coefficients α, learning rate η

Initialize Hebbian traces $\text{Hebb}_0^l \leftarrow 0$ and states $x_0^l \leftarrow 0$ for all $l = 1, \ldots, L$
for all iteration steps **do**
  Sample data from $\mathcal{D}$
  for $t = 1 \rightarrow T$ **do**
    for $l = 1 \rightarrow L$ **do**
      $x_t^{l-1} \leftarrow x_t^{l-1} + x_{t-1}^l$
      $x_t^l \leftarrow \left(W^l + \alpha^l \odot \text{Hebb}_t^l\right) x_t^{l-1} + b^l$
      $\tilde{x}_t^l \leftarrow \sigma\left(x_t^l\right)$
      $\text{Hebb}_t^l \leftarrow \text{Hebb}_{t-1}^l + \eta^l \tilde{x}_t^l \left(\left(x_t^{l-1}\right)^\top - \left(\tilde{x}_t^l\right)^\top \text{Hebb}_{t-1}^l\right)$
    end for
  end for
  Compute cross-entropy loss and backpropagate
  Update parameters W, b, α, η for all $l = 1, \ldots, L$
end for

dependent plasticity that responds specifically to individual inputs. This design contrasts with traditional mini-batch training, where gradients are averaged across samples and may obscure important input-specific correlations.

Therefore, we adopt a hybrid strategy to train the proposed ReHNet: fixed weights are trained in batch mode, while plastic weights are updated on a per-sample basis. From the perspective of computational efficiency, we still employ mini-batch training. However, within each mini-batch, the Hebbian traces for individual samples are updated independently. This design enables the plastic components to adaptively learn and respond to each input sample.

C. Recurrent Hebbian can Strengthen Informative Features

We analyze how the recurrent Hebbian computation endows ReHNet with strong robustness and adaptability. The weight increment $\Delta W_{ij,t}$ induced by (3) is proportional to the negative gradient of an implicit energy function

$$\frac{\partial \mathcal{L}_{\text{im}}}{\partial W_{ij}^l} \propto -\Delta W_{ij,t}^l \propto -x_{i,t}^{l-1} x_{j,t}^l. \tag{5}$$

The expression of the loss function $\mathcal{L}_{\text{im}}$ can be obtained through integration

$$\begin{aligned} \mathcal{L}_{\text{im}} &\approx -\sum_{t=1}^{T}\sum_{l=1}^{L} x_t^l \int x_t^{l-1} dW^l \\ &= -\sum_{t=1}^{T}\sum_{l=1}^{L} x_t^l W^l x_t^{l-1}. \end{aligned} \tag{6}$$

As discussed in [26], the form of the loss function in (6) is analogous to an energy function, and thus the model's effectiveness can be understood from the perspective of energy optimization. Concretely, end-to-end supervised training ensures that the network selectively activates in a subset of neurons, thereby generating correct responses to input patterns. Consequently, the correlated activation behavior of neuron pairs (x_t^{l-1}, x_t^l) is more likely to capture the network's response to an input. Moreover, Hebbian updates act to lower the energy landscape at each update step. By optimizing this energy surface, the local plasticity module imposes an implicit regularization on the network's architecture, accompanied by the penalty term $-\sum_{l=1}^{L} x_t^{l-1} W^l x_t^l$. This mechanism encourages the network to strengthen weights that promote co-activation of neurons, thereby providing stronger drives for similar or repeated patterns.

Overall, the Hebbian model relaxes the network's hierarchical representation towards local minima that are more likely to encode effective responses to previously encountered associative patterns. In this way, it harnesses the correlations embedded in training patterns to recognize incomplete motifs

in test data under distribution shifts. Therefore, each Hebbian update performs a step of energy minimisation that reinforces co-activated feature pairs and regularizes the network. Empirically, feature-map visualisation (Fig. 2) shows that after recurrent Hebbian computation the model activates more filters under contrast corruption and produces stronger responses under blur, which indicates enhanced edge and contour sensitivity.

Furthermore, the effectiveness of the model can also be interpreted from the perspective of constructing a memory matrix for classifying interference patterns. Given a general training set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where $y_i \in \mathbb{R}^m$ denotes the layer-l response to input x_i, suppose that we receive a test sample $\bar{x} \in \mathbb{R}^m$ that belongs to class $\mathcal{D}_k$. The local plasticity module then generates the enhancement signal $\mathbf{I}$ as

$$\begin{aligned} \mathbf{I} = W\bar{x} &= \sum_j y_j y_j^\top \bar{x} \\ &= \sum_{k \in \mathcal{D}_k} y_k \, (y_k^\top \bar{x}) + \sum_{j \notin \mathcal{D}_k} y_j (y_j^\top \bar{x}). \end{aligned} \tag{7}$$

When a test sample $\bar{x}$ arrives from a shifted distribution, the plastic weights modulate each stored mode y_j by the coefficient $y_j^T \bar{x}$, thereby amplifying the contribution of previously stored associative patterns. Because samples from the same class yield stronger inner products $y_k^T \bar{x}$, this mechanism enables the local plasticity module to exploit the correlation between the new input and past patterns, facilitating correct classification under interference.

We visualize and analyze the feature maps of both the Feedforward and ReHNet models to further interpret and discuss the findings. Fig. 2 visually illustrates the feature extraction capabilities of the two models under perturbations in contrast and blur. We select the feature map from the channel with the highest average activation in the first convolutional block of each model. When comparing under contrast corruption, ReHNet is able to activate a larger number of filters than Feedforward, indicating that the model extracts more diverse and complex features. For blur corruption, ReHNet produces brighter feature maps than Feedforward, i.e., with larger numerical values, which implies that the model enhances its response to blurry features. In summary, ReHNet can adaptively re-weight features through synaptic plasticity, allowing the model to focus more on the task-relevant features. This mechanism enables it to better detect edges and contours in corrupted images, thus achieving higher accuracy on OOD data.

IV. Experimental Results

We evaluate the robustness of ReHNet on corruption and OOD datasets. We first consider CIFAR-10C and CIFAR-100C [35], [36], which are corruption benchmarks derived from CIFAR-10 and CIFAR-100, respectively. These benchmarks are specifically constructed to evaluate the robustness of classification models under varying types and severities of corruption. CIFAR-10C and CIFAR-100C each contain 10,000 images derived from their clean counterparts. Each image in these benchmarks is subjected to one of 15 corruption types, such as Gaussian noise, defocus blur, motion blur, and zoom blur. Each corruption type is further applied at five severity levels, increasing from level 1 (mild) to level 5 (severe). This design enables comprehensive evaluation of model robustness under a wide range of environmental perturbations.

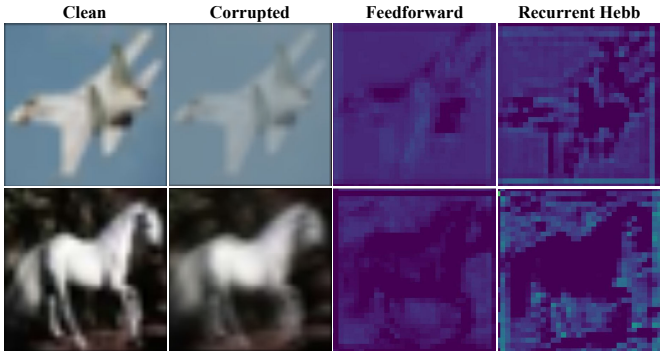

Fig. 2. ReHNet can enhance the feature representations under data corruption.

TABLE I
Difference between test-time adaptation methods and ReHNet.

	TTT	TENT	BN	SITA	ReHNet
Target Label	✗	✗	✗	✗	✗
Test Loss	$\mathcal{L}(x^t)$	$\mathcal{L}(x^t)$	✗	✗	✗
Batched	✓	✓	✓/✗	✗	✗
Backwards	✓	✓	✗	✗	✗
Data Augmentation	✗	✗	✗	✓	✗
Tuning	✓	✓	✓	✓	✗
Bio-Plausible	✗	✗	✗	✗	✓

We then consider PACS [37], which is a popular benchmark for domain generalization, designed to evaluate a model's ability to generalize to novel, unseen domains. The dataset consists of images from four distinct domains: Art (2048 images), Cartoon (2344 images), Photo (1670 images), and Sketch (3929 images). Each domain shares the same set of seven semantic categories, covering various visual concepts such as animals, humans, and objects. This setup poses a realistic and challenging testbed for studying the robustness and generalization capacity of visual recognition models across heterogeneous domains. Experiments follow the leave-one-domain-out protocol: one domain is held out as the target, the remaining domains serve as sources [37]. A model is trained on all source domains, then evaluated on the held-out target domain.

a) Plasticity Ablation Setting.: To evaluate the effect of plasticity components and different network architectures on final performance, we conduct an ablation study under consistent parameter settings. In addition to the proposed ReHNet model, we also construct a Feedforward baseline (Feedforward), a recurrent model without plasticity (Recurrent), and a Feedforward model with Hebbian plasticity (Feedforward Hebb) for comparison (Fig. 1).

b) Source Model Pretraining Setup.: For each dataset, we pretrain three model variants as described in Section 3. We use the SGD optimizer with cosine learning rate decay,

TABLE II
ARCHITECTURE SETTING FOR CIFAR AND PACS. ONLY V1 AND IT DIFFER BETWEEN DATASETS; ALL OTHER STAGES SHARE THE SAME OPERATIONS WITH DATASET-SPECIFIC OUTPUT SHAPES.

Block	Operation	Output (CIFAR / PACS)
Input	—	32×32×3 / 224×224×3
V1	Conv3×3(s=2) / Conv7×7(s=4)[†] Conv 3×3 (s=1)	32×32×64 / 56×56×64 32×32×64 / 56×56×64
V2	Conv 3×3 (s=2) Conv 3×3 (s=1)	16×16×128 / 28×28×128 16×16×128 / 28×28×128
V4	Conv 3×3 (s=2) Conv 3×3 (s=1)	8×8×256 / 14×14×256 8×8×256 / 14×14×256
IT	Conv 3×3 (s=2) Conv 3×3 (s=1)	4×4×512 / 7×7×512 4×4×512 / 7×7×512
Decoder	AvgPool Flatten Linear	1×1×512 / 1×1×512 512 / 512 10or100 / 7

[†] For first layer, CIFAR uses conv 3×3 with stride 2, and PACS uses conv 7×7 with stride 4.

and search the initial learning rate from $\{5 \times 10^{-2}, 10^{-2}, 5 \times 10^{-3}, 10^{-3}\}$. The batch size is set to 64 for CIFAR and PACS datasets.

A. Single-Sample Domain Adaptation

We first evaluate our method on single-sample domain adaptation tasks using the CIFAR10/100 datasets. Models are pre-trained on clean source domains and adapted to 15 target domains, each with corruption severity 5, following the standard CIFAR-C protocols. During adaptation, the model is only allowed to access a single test sample from the target domain at a time. We compare ReHNet with representative single-sample test-time-adaptation (TTA) baselines: the source model (Source), BN-based method [38], and SITA [32]. The source model is directly evaluated on corrupted inputs without adaptation. The BN baseline adapts batch norm statistics using individual test samples. SITA performs robust statistics estimation by aggregating multiple augmentations of each test sample. For BN and SITA, we follow prior work to use the source model's batch norm statistics as initialization, and progressively adapt them using streaming test samples. Following [38], we sweep the momentum factor in 0.6, 0.7, 0.8, 0.9 for optimal performance. Table I summarizes the differences between ReHNet and TTA methods. Notably, ReHNet uniquely supports single-sample adaptation, requires no backward computation or batch diversity, and is consistent with neurobiological evidence. These characteristics are especially critical in real-world applications such as autonomous driving and online robotic perception, where inputs arrive sequentially and distribution shifts are commonplace.

On clean CIFAR10 data, the source model with BN achieves a test accuracy of 93.51%, while the model with GN achieves 91.17%. As shown in Table III, when directly applying the source model to the corrupted data without adaptation, the average error is 44.02%. After applying batch-level statistics correction, both BN and SITA improve the model's accuracy on the target domain to some extent. However, the GN-based source model suffers from suboptimal results due to its statistical limitations. Unlike BN and SITA, which use running averages of the batch statistics to maintain historical context, Group Normalization (GN) computes mean and variance independently for each sample and each group of channels, without leveraging information across samples or over time. As a result, GN is less effective under sparse supervision or heavy noise, since it cannot benefit from accumulated statistical knowledge. In our experiments, Feedforward models with GN yield an average classification error of 38.81%, which is worse than both BN and SITA. With layer-wise feedback connections, the Recurrent model further reduces the target domain error to 35.35%.

The Feedforward Hebb model, which enables plastic adaptation based on data-dependent weighted updates, achieves an error rate of 35.39%. When combining both fast weight plasticity and recurrent connectivity, ReHNet achieves the best performance, with an average error of 30.53%. As shown in Table IV, for the more complex CIFAR100-to-CIFAR100C task, the proposed method also achieves the best performance, which demonstrates the excellent robustness and adaptability of ReHNet.

B. Mixed-Distribution Domain Adaptation

We further evaluate the robustness of our method under mixed-distribution domain shifts. Specifically, we mix corrupted images from multiple CIFAR10C or CIFAR100C corruption types at each severity level to form more challenging test batches with distribution heterogeneity. Each test batch contains samples drawn from multiple corruption types (e.g., brightness, blur, noise), making it a harder scenario than single corruption adaptation. Evaluation is based on the average error across test batches under all five severity levels. We compare our method against the following baselines designed for test-time adaptation in mixed distributions: the frozen source model (Source), PTN [39], and Tent. PTN recomputes batch statistics using exponential moving averages and combines them with historical statistics to stabilize adaptation. Tent performs entropy minimization at test time through optimizing affine parameters of batch norm layers. During test-time adaptation, we employ the Adam optimizer to update the model parameters, with the learning rate selected from $\{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$. To mitigate error accumulation and enhance robustness, both PTN and Tent reset model parameters when switching between different test batches.

The results are presented in Table V. Tent and PTN rely on batch-wise distribution normalization to adapt affine parameters. However, these statistics often reflect the distribution of a specific batch of test samples, which may not generalize well to different corruption patterns. Moreover, it is difficult to estimate multiple distributions accurately using a single set of statistics, leading to performance degradation. In contrast, our proposed ReHNet is independent of batch statistics. It focuses on the accumulated properties of single samples and is therefore more robust under both single-sample and batch distribution shift conditions. It demonstrates consistent

TABLE III
RESULTS (ERROR %) ON CIFAR10-TO-CIFAR10C SINGLE IMAGE DOMAIN ADAPTATION.

Method	Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixelate	JPEG	Mean↓
Source	71.93	64.02	70.85	48.44	51.22	38.43	38.33	25.71	36.17	29.64	11.65	77.30	23.65	49.07	23.64	44.02
BN	46.60	43.73	56.16	**24.33**	47.17	21.84	25.05	24.99	25.58	24.71	13.46	25.36	29.37	40.36	31.26	32.00
SITA	**44.89**	**40.40**	56.34	25.26	44.72	**20.88**	**24.30**	22.38	23.95	**22.53**	11.92	31.52	27.10	41.64	29.92	31.18
Feedforward	54.01	49.15	54.51	39.64	46.30	34.14	32.46	26.27	32.92	40.82	14.78	69.33	23.40	42.08	22.34	38.81
Recurrent	47.51	43.81	**51.04**	31.31	46.95	27.88	30.92	25.25	31.39	32.54	14.29	54.06	24.62	44.84	23.86	35.35
Feedforward Hebb	55.21	51.46	59.36	35.19	48.12	29.47	29.99	26.51	34.02	28.15	14.22	27.83	24.12	43.17	23.97	35.39
ReHNet	47.91	44.32	54.97	27.31	**44.27**	24.12	25.49	**22.02**	28.35	24.32	**11.64**	**23.53**	**22.48**	**35.59**	**21.63**	**30.53**

TABLE IV
RESULTS (ERROR %) OF CIFAR100-TO-CIFAR100C SINGLE IMAGE DOMAIN ADAPTATION.

Method	Gaussian	Shot	Impulse	Defocus	Glass	Motion	Zoom	Snow	Frost	Fog	Bright	Contrast	Elastic	Pixelate	JPEG	Mean↓
Source	89.66	88.95	94.89	65.37	79.46	56.77	59.49	60.20	65.79	**60.26**	44.72	81.63	52.24	70.88	59.16	68.63
BN	82.46	81.41	88.06	**56.49**	74.69	50.64	**52.66**	57.19	57.02	60.63	41.19	**62.67**	57.14	69.16	62.73	63.61
SITA	84.13	83.46	89.99	58.11	72.37	**50.07**	52.89	**53.97**	56.56	57.30	**38.05**	69.41	51.60	69.81	60.03	63.18
Feedforward	80.79	78.25	88.94	64.80	70.65	60.82	57.36	56.77	61.64	71.69	45.60	88.66	50.54	64.59	51.83	66.20
Recurrent	78.22	76.44	86.79	61.53	68.19	57.78	57.33	55.50	61.43	72.37	48.11	83.67	50.35	60.14	53.83	64.78
Feedforward Hebb	78.78	76.25	85.93	62.65	70.04	58.16	55.58	57.39	60.06	64.07	46.45	65.82	51.05	64.54	52.76	63.30
ReHNet	**67.36**	**65.50**	**79.27**	56.74	**59.31**	56.27	53.02	54.57	**55.37**	66.51	47.34	64.16	**49.17**	**54.98**	**47.88**	**58.50**

TABLE V
RESULTS ON CIFAR10C/CIFAR100C DOMAIN ADAPTATION UNDER MIXED DISTRIBUTION SHIFTS (SEVERITY 5).

Method	**CIFAR-10C** Mean Error (↓)	**CIFAR-100C** Mean Error (↓)
Source	44.02	68.63
PTN	39.32	66.73
TENT	54.93	89.16
Feedforward	38.81	66.20
Recurrent	35.35	64.78
Feedforward Hebb	35.39	63.30
ReHNet	**30.53**	**58.50**

performance under diverse test settings. Among all methods, ReHNet achieves the lowest error, showing its effectiveness in dealing with mixed domain shifts. This further illustrates that ReHNet can handle complex distribution shifts more efficiently and adaptively across a wide range of corruptions and input variations. Its generalization and adaptability benefit from its intrinsic fast plasticity.

C. Robustness on Cross-Domain Data

For the four domains in the PACS dataset, we select three domains as the source domain for pretraining, and the remaining one as the target domain. After the model is pretrained on the source domains, it is tested on the target domain. During adaptation, the pretrained source model can only access a single test sample from the target domain at each time. The model performance is evaluated by the average error across all target domains. Table VI provides the cross-domain performance of ReHNet. Our primary focus lies in domain generalization tasks with significant domain discrepancies, including concept shift. ReHNet demonstrates significantly stronger gains over the Feedforward baseline, with an average improvement of 5.78%, which demonstrates the robustness of ReHNet under severe domain shifts.

V. CONCLUSION

In this work, we have presented ReHNet, a biologically inspired architecture that integrates inter-layer feedback connections with fast, partially plastic Hebbian synapses. Our results demonstrate that this combination substantially enhances robustness to noise and OOD shifts, outperforming both standard feedforward architectures and state-of-the-art domain adaptation methods that require backpropagation-based test-time optimization. Notably, ReHNet achieves these gains using only forward-pass computations and local online updates, which makes it highly suitable for resource-constrained and real-time applications.

Our work highlights the practical and scientific value of incorporating neurobiological principles such as feedback pathways and dynamic synaptic plasticity—into artificial neural systems. Our findings suggest a promising pathway toward the development of new robust and adaptive neural networks, bridging the gap between biological plausibility and practical deployment for challenging real-world scenarios. Future work will further explore the integration of additional neuro-inspired mechanisms and broader applications in dynamic and open-world environments.

REFERENCES

[1] C. D. Gilbert and W. Li, "Top-down influences on visual processing," *Nature Reviews Neuroscience*, vol. 14, no. 5, pp. 350–363, 2013.

[2] T. J. Buschman and E. K. Miller, "Top-down versus bottom-up control of attention in the prefrontal and posterior parietal cortices," *Science*, vol. 315, no. 5820, pp. 1860–1862, 2007.

[3] J. H. Reynolds and D. J. Heeger, "The normalization model of attention," *Neuron*, vol. 61, no. 2, pp. 168–185, 2009.

[4] M. Corbetta, E. Akbudak, T. E. Conturo, A. Z. Snyder, J. M. Ollinger, H. A. Drury, M. R. Linenweber, S. E. Petersen, M. E. Raichle, D. C. Van Essen *et al.*, "A common network of functional areas for attention and eye movements," *Neuron*, vol. 21, no. 4, pp. 761–773, 1998.

[5] L. Huang, Z. Ma, L. Yu, H. Zhou, and Y. Tian, "Long-range feedback spiking network captures dynamic and static representations of the visual cortex under movie stimuli," *Advances in Neural Information Processing Systems*, vol. 37, pp. 11 432–11 455, 2024.

[6] R. C. O'Reilly, D. Wyatte, S. Herd, B. Mingus, and D. J. Jilk, "Recurrent processing during object recognition," *Frontiers in Psychology*, vol. 4, p. 124, 2013.

TABLE VI
CROSS-DOMAIN GENERALIZATION RESULTS ON PACS.

Method	**Photo**		**Art Painting**		**Cartoon**		**Sketch**	
	Pretrain Acc	Avg Acc	Pretrain Acc	Avg Acc	Pretrain Acc	Avg Acc	Pretrain Acc	Avg Acc
Feedforward	85.65	61.50	89.08	41.80	84.56	56.57	82.79	56.58
Recurrent	83.51	68.68 (**+7.18**)	86.35	44.14 (+2.34)	82.37	57.98 (+1.41)	80.03	65.29 (+6.46)
Feedforward Hebb	85.88	63.53 (+2.03)	89.21	42.87 (+1.07)	84.81	55.59 (-0.98)	82.79	50.95 (-5.63)
ReHNet	84.10	67.43 (+5.93)	88.09	46.09 (**+4.29**)	82.37	59.22 (**+2.65**)	80.84	66.84 (**+10.26**)

[7] D. Wyatte, D. J. Jilk, and R. C. O'Reilly, "Early recurrent feedback facilitates visual object recognition under challenging conditions," *Frontiers in Psychology*, vol. 5, p. 674, 2014.

[8] H. Tang, M. Schrimpf, W. Lotter, C. Moerman, A. Paredes, J. Ortega Caro, W. Hardesty, D. Cox, and G. Kreiman, "Recurrent computations for visual pattern completion," *Proceedings of the National Academy of Sciences*, vol. 115, no. 35, pp. 8835–8840, 2018.

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

[10] J. Kubilius, M. Schrimpf, K. Kar, R. Rajalingham, H. Hong, N. Majaj, E. Issa, P. Bashivan, J. Prescott-Roy, K. Schmidt *et al.*, "Brain-like object recognition with high-performing shallow recurrent anns," in *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[11] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," in *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[12] G. E. Hinton and D. C. Plaut, "Using fast weights to deblur old memories," in *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, 1987, pp. 177–186.

[13] T. Miconi, K. Stanley, and J. Clune, "Differentiable plasticity: training plastic neural networks with backpropagation," in *International Conference on Machine Learning*, 2018, pp. 3559–3568.

[14] T. Miconi, A. Rawal, J. Clune, and K. O. Stanley, "Backpropamine: training self-modifying neural networks with differentiable neuromodulated plasticity," in *International Conference on Learning Representations*, 2019.

[15] C. von der Malsburg, "Self-organization of orientation sensitive cells in the striate cortex," *Kybernetik*, vol. 14, no. 2, pp. 85–100, 1973.

[16] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological Cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.

[17] E. Oja, "Simplified neuron model as a principal component analyzer," *Journal of Mathematical Biology*, vol. 15, no. 3, pp. 267–273, 1982.

[18] S. Song and L. F. Abbott, "Cortical development and remapping through spike timing-dependent plasticity," *Neuron*, vol. 32, no. 2, pp. 339–350, 2001.

[19] H. Markram, J. Lübke, M. Frotscher, and B. Sakmann, "Regulation of synaptic efficacy by coincidence of postsynaptic aps and epsps," *Science*, vol. 275, no. 5297, pp. 213–215, 1997.

[20] G. Q. Bi and M. M. Poo, "Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type," *Journal of Neuroscience*, vol. 18, no. 24, pp. 10 464–10 472, 1998.

[21] Q. Gu, "Neuromodulatory transmitter systems in the cortex and their role in cortical plasticity," *Neuroscience*, vol. 111, no. 4, pp. 815–835, 2002.

[22] J. C. Magee and C. Grienberger, "Synaptic plasticity forms and functions," *Annual Review of Neuroscience*, vol. 43, pp. 95–117, 2020.

[23] D. Krotov and J. J. Hopfield, "Unsupervised learning by competing hidden units," *Proceedings of the National Academy of Sciences*, vol. 116, no. 16, pp. 7723–7731, 2019.

[24] T. Moraitis, D. Toichkin, A. Journé, Y. Chua, and Q. Guo, "Softhebb: Bayesian inference in unsupervised hebbian soft winner-take-all networks," *Neuromorphic Computing and Engineering*, vol. 2, no. 4, p. 044017, 2022.

[25] A. Journé, H. G. Rodriguez, Q. Guo, and T. Moraitis, "Hebbian deep learning without feedback," in *The Eleventh International Conference on Learning Representations*, 2022.

[26] Y. Wu, R. Zhao, J. Zhu, F. Chen, M. Xu, G. Li, S. Song, L. Deng, G. Wang, H. Zheng *et al.*, "Brain-inspired global-local learning incorporated with neuromorphic computing," *Nature Communications*, vol. 13, no. 1, p. 65, 2022.

[27] H. G. Rodriguez, Q. Guo, and T. Moraitis, "Short-term plasticity neurons learning to learn and forget," in *International Conference on Machine Learning*. PMLR, 2022, pp. 18 704–18 722.

[28] D. Kumaran, D. Hassabis, and J. L. McClelland, "What learning systems do intelligent agents need? complementary learning systems theory updated," *Trends in cognitive sciences*, vol. 20, no. 7, pp. 512–534, 2016.

[29] C. H. Bailey, E. R. Kandel, and K. M. Harris, "Structural components of synaptic plasticity and memory consolidation," *Cold Spring Harbor perspectives in biology*, vol. 7, no. 7, p. a021758, 2015.

[30] Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt, "Test-time training with self-supervision for generalization under distribution shifts," in *International conference on machine learning*. PMLR, 2020, pp. 9229–9248.

[31] D. Wang, E. Shelhamer, S. Liu, B. Olshausen, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," *arXiv preprint arXiv:2006.10726*, 2020.

[32] A. Khurana, S. Paul, P. Rai, S. Biswas, and G. Aggarwal, "Sita: Single image test-time adaptation," *arXiv preprint arXiv:2112.02355*, 2021.

[33] J. Kubilius, M. Schrimpf, A. Nayebi, D. Bear, D. L. Yamins, and J. J. DiCarlo, "Cornet: Modeling the neural mechanisms of core object recognition," *BioRxiv*, p. 408385, 2018.

[34] Y. Wu and K. He, "Group normalization," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[35] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[36] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," *arXiv preprint arXiv:1903.12261*, 2019.

[37] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, "Deeper, broader and artier domain generalization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5542–5550.

[38] S. Schneider, E. Rusak, L. Eck, O. Bringmann, W. Brendel, and M. Bethge, "Improving robustness against common corruptions by covariate shift adaptation," *Advances in neural information processing systems*, vol. 33, pp. 11 539–11 551, 2020.

[39] Z. Nado, S. Padhy, D. Sculley, A. D'Amour, B. Lakshminarayanan, and J. Snoek, "Evaluating prediction-time batch normalization for robustness under covariate shift," *arXiv preprint arXiv:2006.10963*, 2020.