Incremental Deep Image Prior for Continual Adversarial Defense

Dun-Ming Huang

Abstract—Deep Image Prior (DIP) has emerged as an effective single-image regularizer capable of removing noise or corruption without external training data. Recent work demonstrates that DIP can serve as a lightweight adversarial defense by reconstructing clean images before classification. However, existing DIP-based defenses operate independently on each input and cannot adapt across a sequence of attacks, limiting their applicability in continual or streaming settings. In this work, we propose **Incremental Deep Image Prior** (IDIP), which extends DIP to sequential tasks by training a single network across multiple adversarial examples while preserving previously learned defenses. Building on IDIP, we develop **Incremental DIPDefend**, a continual adversarial defense method that incorporates anchoring-based memory preservation. Experiments on ImageNet-Small under PGD attacks show that Incremental DIPDefend maintains performance comparable to single-image DIPDefend while requiring only one network, outperforming continual-learning baselines and repeated from-scratch retraining. Our results demonstrate that data-free reconstruction priors can be adapted for continual adversarial defense, enabling practical robustness in low-resource online environments.

Index Terms—Computational Imaging, Adversarial Defense, Continual Learning, Deep Image Prior

1 Introduction

Adversarial attacks perturb an image I with small, difficult-to-detect noise ϵ, causing a classifier h that originally predicts $h(I) = \hat{C}$ to instead misclassify the perturbed image $I + \epsilon$ [1–4]. As image-based deep learning systems are deployed in increasingly safety-critical and human-centric applications [5–7], developing methods that protect models from adversarial perturbations has become essential. However, many practitioners lack the computational resources and training data required by most state-of-the-art adversarial defenses [8–10]. To support such low-resource settings, we aim to develop a lightweight continual adversarial defense system that satisfies two desiderata:

$\mathcal{D}_1$. The defense requires with little-to-no training data.
$\mathcal{D}_2$. The defender can continually adapt to an incoming sequence of adversarial attacks.

From the perspective of single-example adversarial defense, Deep Image Prior (DIP) [11] provides an appealing lightweight model. DIP regularizes inverse imaging problems such as reconstruction or super-resolution by optimizing a randomly initialized, high-capacity convolutional network (eg, UNets [12]) whose architectural biases encourage convergence toward natural, low-noise images. With the insight that denoising an adversarial attack cleans its adversarial noise, several applications of DIP achieve adversarial defense [13–15]. However, DIP operates strictly on a per-image basis, and a network optimized for one example does not generalize to another; thus, existing DIP defenses satisfy $\mathcal{D}_1$ but fundamentally fail $\mathcal{D}_2$. Notably, while Deep Denoising Priors [16] can serve as general reconstruction regularizers for multiple examples, acting like a multi-example DIP, it requires large training data and fails $\mathcal{D}_1$.

Moreover, prior works on DIP have not examined a continual adversarial defense setup– whether a single lightweight model can defend against multiple attacks over time. In realistic deployments, defenders may encounter a continuous stream of diverse adversarial examples. Yet a DIP trained on one adversarial example does not transfer to subsequent examples. Retraining a new DIP for every incoming attack is computationally expensive and discards useful information carried by previous defenses, violating the continual adaptation criterion ($\mathcal{D}_2$).

To address this limitation, we propose **Incremental Deep Image Prior** (IDIP)– an extension of DIP designed to operate across sequential inputs. Given a sequence of T inputs, IDIP uses a single network that is updated on each new example while retaining its reconstructions of all previously seen images. Incremental DIP satisfies both the data-free constraint ($\mathcal{D}_1$) and adaptability towards a sequence of attacks ($\mathcal{D}_2$), making it an appropriate choice for continual adversarial defense. We integrate IDIP into a continual adversarial defense framework and show that incremental training preserves both defense effectiveness and reconstruction quality.

Our contributions are summarized as follows:

- We propose **Incremental Deep Image Prior** (IDIP), where we train DIP on consecutive images that can regularize inverse tasks for a sequence of images. The only training data are the images to be regularized.
- We introduce **Incremental DIPDefend**, a continual adversarial defense built on IDIP. It matches the performance of single-image DIPDefend while significantly outperforming continual-learning baselines.
- IDIP provides a general framework for adapting parametrized single-image models to continual adversarial defense settings.
- We provide an open-source implementation for reproducibility and evaluation. [1]

1. Our implementation for grading is linked at the following GitHub repository: `github.com/Bransthre/csc2529_final_project`

2 Related Works

2.1 Deep Image Prior

Deep Image Prior (DIP) [11] provides noise-minimal solutions to inverse imaging problems (e.g., denoising, resolution, and super-resolution). It does so by leveraging the inductive bias of high-capacity convolutional neural networks. To solve the inverse problem from some corrupted image I, DIP fits a randomly initialized CNN to I on some objective that minimizes the difference between a random network input and I. Ulyanov et al. [11] observe that the architecture of high-capacity CNNs inherently biases optimization toward natural image statistics, suppressing noise and artifacts early in training. As optimization progresses, the network first captures large-scale smooth structures before overfitting high-frequency corruption. This implicit architectural regularization makes DIP a strong baseline for a range of inverse imaging tasks.

2.2 Deep Image Prior as Adversarial Denoiser

Adversarial attacks can be interpreted as noising an image. Then, training a denoiser that cleans an adversarially attacked image would provide a functional adversarial defense. Because Deep Image Prior is a single-image denoiser, it naturally extends to a per-example adversarial defense. DIPDefend [13] operationalizes this idea and demonstrates that DIP-based reconstruction can mitigate adversarial perturbations. It is supported by two crucial discoveries. One, DIP reconstructs images from low-level details, where its avoidance of noise can reconstruct the adversarial example into an example of the ground truth class. Two, DIP's reconstruction occurs by learning a detection measure for when to stop the overfitting of DIP, such that the adversarial noise is not reconstructed. To construct a defense, one trains a DIP to denoise the adversarial attack. Following DIPDefend, the defended output is chosen from an early training checkpoint where PSNR improvement begins to plateau. Notably, several other DIP applications on defense exist: [14] and [15] operate on similar premises, while [17] uses DIP-induced features and decision boundaries to purify adversarial noise.

2.3 Continual Learning

Continual learning concerns learning a sequence of tasks without forgetting previously acquired knowledge [18], aligning closely with the incremental demands of continual adversarial defense. Regularization-based approaches constrain parameter updates based on their estimated importance to prior tasks. For instance, Elastic Weight Consolidation (EWC) [19] combats catastrophic forgetting by adding a penalty term to the loss function, which constrains the update of weights deemed important for prior tasks by calculating their importance via the Fisher Information Matrix. Continual learning methods are diverse: some may store or synthesize data to "rehearse" old tasks alongside new learning [20–22], while others may adjust the network capacity to isolate knowledge representation for new tasks [23–25]. Our work intersects with these ideas by proposing an incremental, data-free defense mechanism that adapts a single model across a sequence of adversarial inputs.

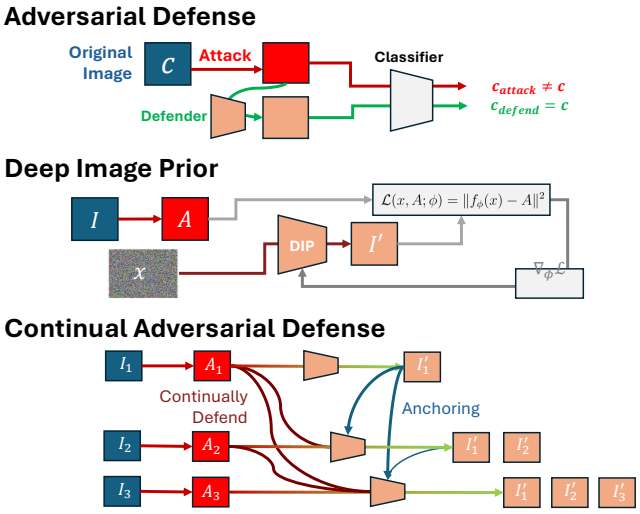

Fig. 1: A visual overview of relevant topics in this paper. **Top**: Standard adversarial defense with a defender mitigating attacks before classification, allowing the defended image to have the same prediction as the unattacked image. **Middle**: Deep Image Prior (DIP) defense denoises an adversarial example A into a cleaned image I' via optimization. **Bottom**: Continual adversarial defense, where a defender guards the classifier from a sequence of attacks. This is enabled by a composite objective that defends the current attack and memorizes the previous defenses (anchoring).

3 Method

We propose **Incremental DIPDefend**, an adversarial defense based on Deep Image Prior (DIP) that can adapt to a sequence of unseen adversarial attacks with one single network and no training data. Incremental DIPDefend is powered by an incrementally trained Deep Image Prior that solves an inverse problem on a sequence of input images. To preserve the brevity of this discussion, for all unmentioned implementational details (eg, network architecture, training hyperparameters), we refer the readers to the implementation posted at the introduction. We mirror our training setting from that of DIP and DIPDefend [11, 13].

3.1 Incremental Deep Image Prior

In the original DIP setting, a network f_ϕ is optimized for a single image A to solve an inverse problem. The network takes as input a random noise $x \sim \mathcal{N}(0.5, 0.5I)$ and is optimized to minimize a reconstruction objective $\mathcal{E}$ between $f_\phi(x)$ and A.

In Incremental DIP, the network instead receives a sequence of images $\{A_i\}_{i=1}^{T}$ over time. At each timestep $t = j$, we introduce the example A_j and its corresponding network input $x_j \sim \mathcal{N}(0.5, 0.5I)$. The input is clipped between 0 and 1. To maintain performance on previously encountered images, f_ϕ is optimized jointly for two objectives. First, at timestep $t = j$, the network minimizes a reconstruction loss $\mathcal{E}(f_{\phi_{t=j}}(x_j), A_j)$. Second, $f_{\phi_{t=j}}$ must preserve its outputs on all previously seen inputs $\{A_t\}_{t=0}^{j-1}$ by enforcing similarity between past defenses and current outputs. We refer to the choice of the dissimilarity function g as the anchoring

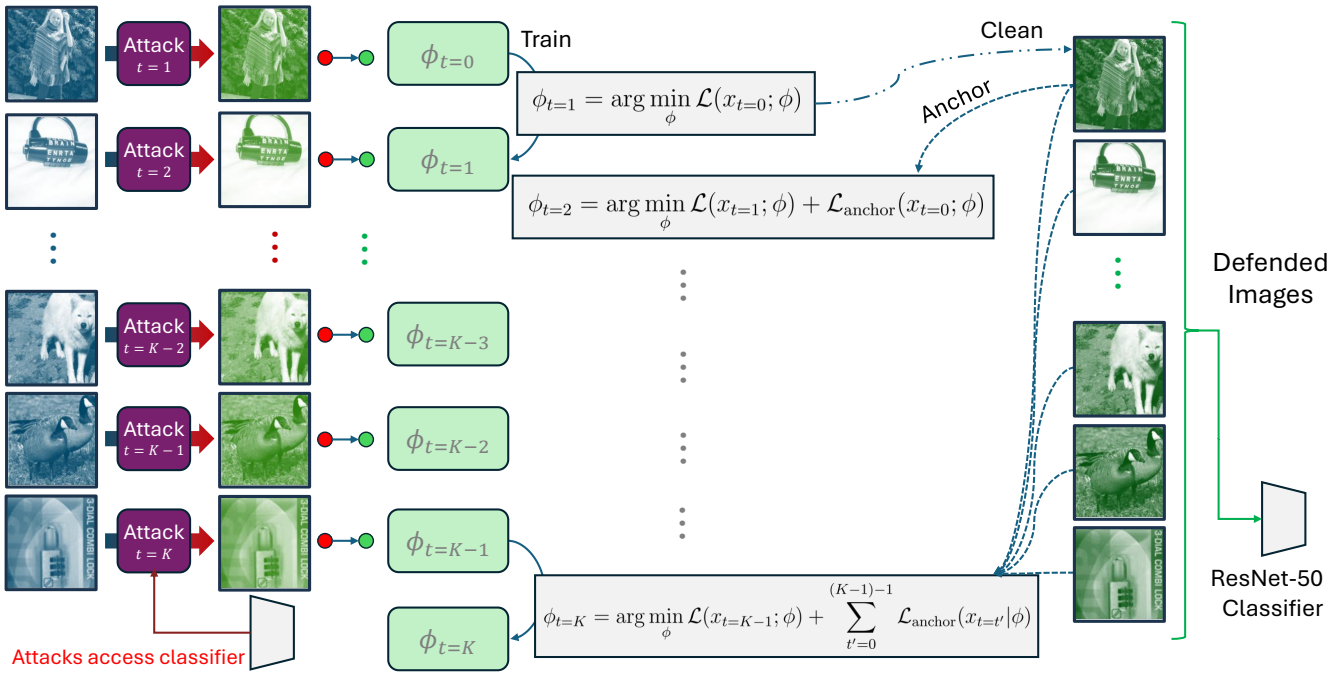

Fig. 2: A schematic overview of Incremental DIPDefend. The Incremental DIP training routine trains the same DIP network (a UNet [12] with residual connections) on a different composite objective for each incoming image. The composite objective contains a term for optimizing the inverse problem on the currently seen image, and another term to preserve the network output for previously seen images.

method, since it constrains the network to remain close to its earlier defenses.

Our work employs an intuitive training routine to optimize for both intentions– at each timestep $t = j$ within the input sequence $\{A_i\}$, we train our network on a composite objective of the two tasks:

$$\mathcal{L}_j(\phi_{t=j}) = \mathcal{E}(f_{\phi_{t=j}}(x_j), A_j) + \lambda \sum_{i=0}^{j-1} g(f_{\phi_{t=i}}(x_i), f_{\phi_{t=j}}(x_i))$$

We refer to the choice of function g as an **anchoring setting**, as it anchors the network to not stray from its original defense. The hyperparameter λ can be used to control the importance of past defenses. In practice, we normalize $\mathcal{L}_j$ by $\frac{1}{1+\lambda \times j}$ to keep gradient magnitudes comparable across timesteps. Following DIP [11], we use an hourglass-like architecture for f_ϕ, leveraging its multi-scale receptive fields and strong spatial inductive biases, and $\mathcal{E}$ is pixel-wise MSE.

3.2 Anchoring Methods

Here, anchoring methods refer to the choice of dissimilarity metric $g(f_{\phi_{t=i}}(x_i), f_{\phi_{t=j}}(x_i))$ between (1) a network's output of the same input x_i at the original timestep of introduction $t = i$, and (2) the ongoing timestep of attack sequence $t' = j$. We consider and experiment with three anchoring methods.

MSE Anchoring. Here, g is a pixel-wise mean squared error (MSE) between the first defense $f_{\phi_{t=i}}(x_i)$ and the network's current output $f_{\phi_{t=j}}(x_i)$.

Spectral MSE Anchoring. We first compute the FFT-transformed defense and the network's current output, then compute the MSE between those spectrums. We hypothesize that the robust features emphasized in DIPDefend [13] may correspond primarily to low-frequency image components. To examine this guess, we weight the MSE of the spectrum to be higher on low-frequency counterparts, and vice versa. This is implemented by constructing an inverse-exponential mask M where $M_{i,j} = \exp(-\alpha\sqrt{f_x(i)^2 + f_y(j)^2})$, where f_x and f_y are the corresponding horizontal and vertical frequency of the point (i, j) on the two-dimensional FFT spectrum.

Elastic Weight Consolidation. As introduced in Section 2.3, EWC penalizes changes to parameters deemed important for earlier tasks. In this anchoring variant, we apply EWC by treating each image's defense as a separate task. Naturally, the metric $g(f_{\phi_{t=i}}(x_i), f_{\phi_{t=j}}(x_i))$ becomes the Fisher-weighted MSE of network parameters $\phi_{t=i}$ and $\phi_{t=j}$ [19, 26].

3.3 Training Incremental DIPDefend

We present the training routine of Incremental DIPDefend by providing a generalized description of defending against an attack at timestep $t = j \in [0, K]$ on a sequence of K attacks. We initialize our network with parameter $\phi_{t=0}$, and expect networks $f_{\phi_{t=i}}$ to defend attacks $\{A_1, A_2, \cdots, A_i\}$.

First, we make an important note regarding the selection of defense. Each attack A_j is optimized on the objective $\mathcal{L}_j$ for n gradient updates. After sufficient optimization, the network may begin to reproduce adversarial high-frequency components. Thus, following DIPDefend, we select the defended image D_j at an early checkpoint $k \leq n$ to avoid

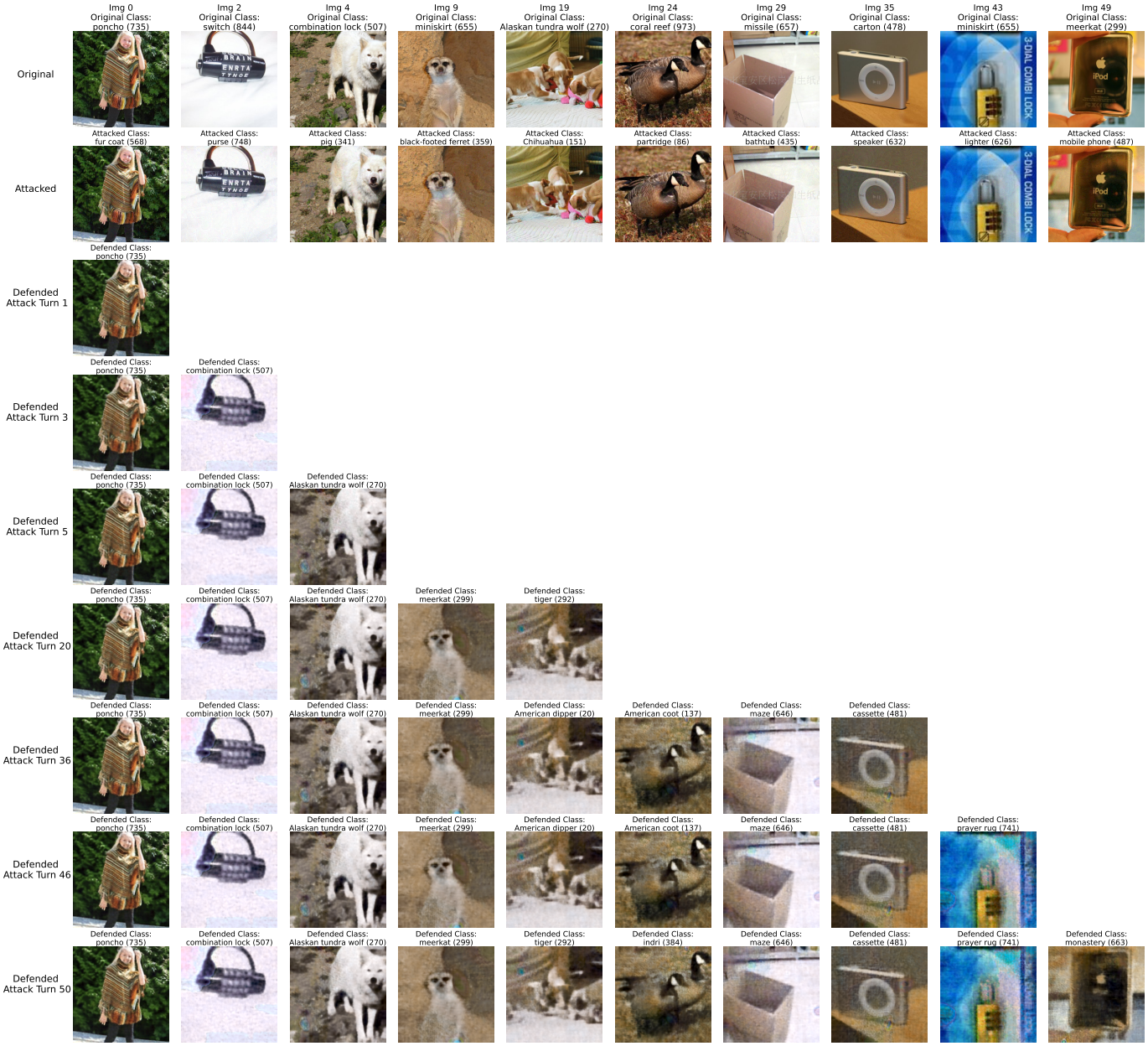

Fig. 3: Examples of adversarial attacks and defenses in our method. Each image is annotated with its original **predicted** class, its mistakenly predicted class after the adversarial attack, and its corrected predicted class on the defended image. We exhibit one of the attack sequences' defense outcomes from using Incremental DIPDefend with MSE Anchoring. As implied from Figures 5, the adversarial attacks and defenses rarely degrade over time except for Image 24.

overfitting to adversarial noise. In the context of continual adversarial defense, the composite objective at timestep $t = j$ is precisely:

$$\mathcal{L}_j(\phi_{t=j}) = \mathcal{E}(f_{\phi_{t=j}}(x_j), A_j) + \lambda \sum_{i=0}^{j-1} g(D_i, f_{\phi_{t=j}}(x_i))$$

The training routine for incremental DIPDefend becomes such. For each attack A_j, we update the network for n steps using the composite objective $\mathcal{L}_j$. We store the current defense D_j such that, for the next timestep $j+1$, it can be used in the anchoring term $\lambda \sum_{i=0}^{j} g(D_i, f_{\phi_{t=j+1}}(x_i))$. This process continues until all attacks in the sequence have been processed.

4 Experiments

We evaluate the performance of Incremental DIPDefend under the following criteria:

- **C1.** Does Incremental DIPDefend successfully denoise incoming adversarial attacks and consistently preserve its earlier defense outputs across timesteps?
- **C2.** How does Incremental DIPDefend compare to non-continual DIPDefend and standard continual learning baselines?

We evaluate Incremental DIPDefend in a continual adversarial defense setting. Our target task is image classification on the ImageNet-Small dataset [27]. We generate adversarial examples using PGD [3] against a pretrained

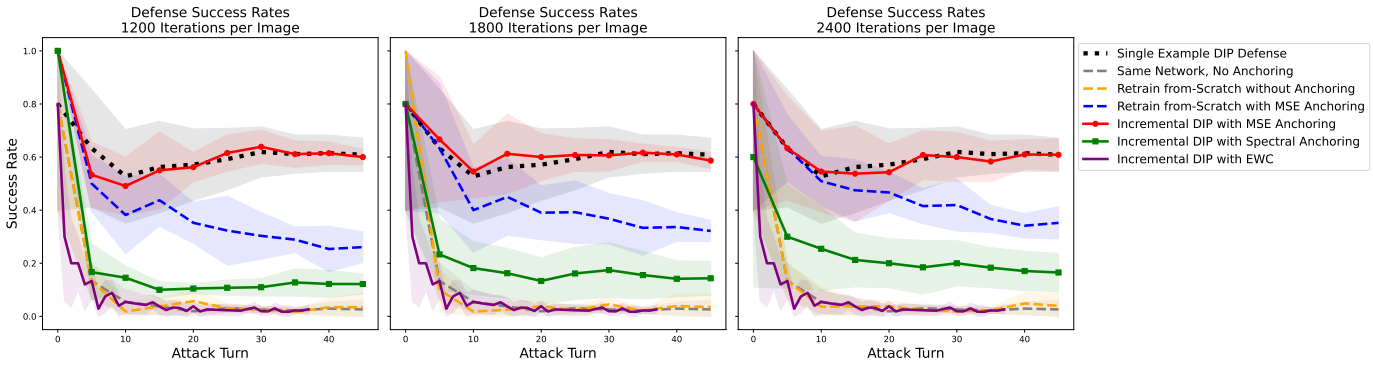

Fig. 4: The defense success rates for all adversarial attacks the network receives up to its current timestep, across five distinct sequences of attacks. To defend k attacks, Incremental DIPDefend with MSE Anchoring (red) maintains the same success rate in single-image DIPDefend with one network, but single-image DIPDefend (black) requires k networks. Furthermore, our method (red) outperforms existing continual learning approaches like Elastic Weight Consolidation, as well as other naive baselines like retraining the entire network. We leave the rest of the analyses to Section 5.

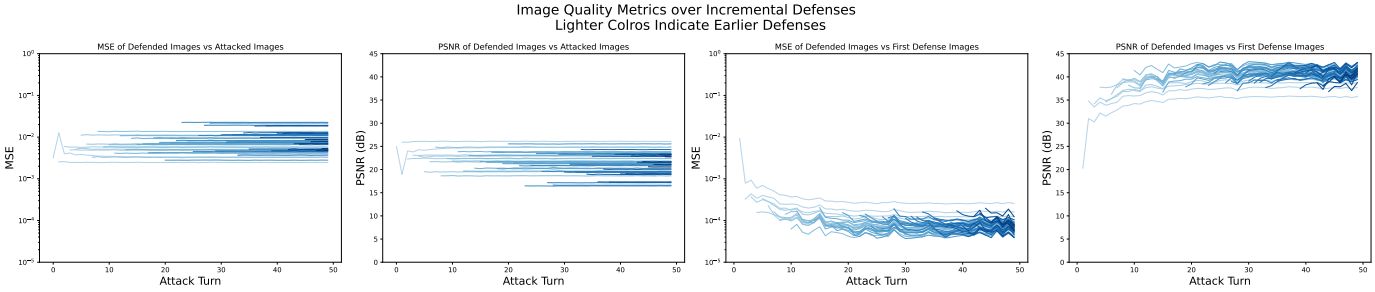

Fig. 5: The tracking performance of Incremental DIP throughout the sequence of adversarial attacks. The left two plots record the MSE and PSNR between each adversarial attack and the network's outputted defense. The right two plots record the MSE and PSNR between each adversarial defense and the network's currently outputted defense. Both plots suggest that even upon memorizing several attacks' defenses, the denoising process for adversarial defense and the preservation of previous examples' defenses do not degrade.

ResNet-50 classifier [28]. [2] We evaluate our methods on five sequences, each containing 50 consecutive adversarial attacks. We define a successful defense as a network output $D_{t=i}$ whose predicted class is equal to that of the original clean image $I_{t=i}$, regardless of whether the defended model makes a correct prediction. [3]

We assess **C1** using MSE and PSNR between the network's output and its corresponding targets. Each attack has two such targets: the adversarial input when the attack is first received, and the stored defended output when the network later rehearses that sample.

To assess **C2**, in addition to the anchoring variants of Incremental DIP introduced in Section 3, we compare against the following baselines:

Single Example DIPDefend: We apply DIPDefend independently to each attack, storing a separate trained DIP network for every example. All other methods assume only a single network can be maintained.

Same Network, No Anchoring: We continuously finetune the same DIP network on each new attack without any anchoring.

Retrain from Scratch without Anchoring: We retrain a fresh DIP network on each incoming attack, without anchoring.

Retrain from Scratch with MSE Anchoring: We retrain a fresh DIP network on each attack, but include an MSE Anchoring term computed as the mean MSE between all stored defenses and the network's current outputs.

At each timestep t', we report the defense success rate over all attacks encountered so far; i.e., those in the interval $[0, t']$.

5 Results

In this section, we present the evaluation of Incremental DIPDefend according to the criteria defined in Section 4. Each criterion is discussed in its own subsection. We consider defending against a sequence of T attacks. Notably, each attack sequence has a success rate of 90% (i.e., 90% of attacks change the predicted class), supporting the credibility of our evaluation.

2. The PGD implementation is powered by `torchattacks` (`github.com/Harry24k/adversarial-attacks-pytorch`), an open-source library of adversarial attacks implemented in PyTorch. The parameters are $\epsilon = 0.03, \alpha = 0.005$, with 40 steps.

3. Due to computational constraints in the course project setting, we restrict our evaluation to sequences of length 50 and leave longer or more diverse attack streams to future work.

5.1 Consistency of DIP Outputs Throughout Attacks

This subsection focuses on criterion **C1**. We evaluate our method using two metrics. First, we assess whether Incremental DIPDefend successfully matches its network output to the received attack $A_{t=i}$ at timestep $t = i$, denoted as $D_{t=i}$. If the network output matches the attack well—but not excessively—for all experienced attacks, it indicates that memorizing previous defenses does not compromise the attack-denoising capability. Second, we assess whether Incremental DIPDefend can maintain its defense $D_{t=i}$ for $A_{t=i}$ at any later turn $j > i$, meaning the network output for $A_{t=i}$ remains consistent with the output at $t = i$. In both cases, we quantify image dissimilarity using MSE (pixel-wise mean squared error) and PSNR (peak signal-to-noise ratio) between the network output and its corresponding target.

In Figure 5, the left two panels correspond to the first assessment, while the right two panels correspond to the second. Lighter curves represent earlier attacks. We observe no significant difference in MSE and PSNR between $A_{t=i}$ and $D_{t=i}$ across attacks of different precedence. The attack-matching capability of the Incremental DIP network is not affected by the number of defenses it must memorize before $t = i$. Although the PSNR is relatively low (20 ± 5), this is expected given the adaptive PSNR measure used by DIPDefend. The right two panels show that MSE and PSNR between $D_{t=i}$ and subsequent network outputs remain very low. As in the first assessment, the MSE and PSNR are largely unaffected by attack precedence. There is a slight trend that earlier defenses are harder to retain, but this has a negligible impact on overall adversarial defense performance. Overall, Incremental DIPDefend effectively denoises attacks and consistently maintains its defense outputs throughout the attack sequence.

We refer readers to Figure 3 for examples of adversarial defense, where the overwhelming majority of defenses exhibit the consistencies claimed above.

5.2 Defense Performance of Incremental DIPDefend

This subsection addresses criterion **C2**, specifically evaluating whether an incremental design optimally supports Incremental DIPDefend for continual adversarial defense. We approach this analysis by considering two questions:

Q2-a. Rather than using a single network for all T attacks, we could use T separate networks, each defending its corresponding attack. How does a single network compare in performance to using T networks?

Q2-b. Within the Incremental DIP design, how critical is it to (a) continually train the same network and (b) employ an anchoring term to retain past defenses?

To answer these questions, we adopt the following evaluation protocol. The performance of **Single Example DIPDefend** is determined by assuming access to T DIP networks, one for each attack. At timestep $t = i$, its success rate is computed as the proportion of successful defenses across the i networks. For **all other methods**, we assume a single network defending all T attacks, and the success rate at timestep $t = i$ is measured as the proportion of i attacks successfully defended.

Figure 4 shows the performance of each baseline described in Section 4, evaluated under varying numbers of training iterations per arriving attack (x *iterations per image*). For **Q2-a**, we observe that our method with MSE Anchoring (red) achieves success rates comparable to the single-example scenario (black), indicating that incremental training does not compromise the underlying efficacy of DIPDefend. For **Q2-b**, our method also outperforms baselines that either do not continually train on the same network (blue; *Retrain from Scratch with MSE Anchoring*) or omit anchoring (grey; *Same Network, No Anchoring*). Incorporating an anchoring term and avoiding retraining from scratch for each new attack substantially benefits the incremental DIP design.

Empirically, it is clear that memorizing defenses is more effective than relearning them. This likely arises because many attacks share common patterns in adversarial noise, allowing an overfitted network to generalize more readily to similar attacks. This advantage is expected to increase in sequences with highly similar attacks, such as those originating from the same class. Finally, the fully non-incremental approach (yellow; *Retraining from Scratch without Anchoring*) is strongly outperformed by the MSE Anchoring method (blue), further emphasizing the importance of anchoring terms.

5.3 Defense Performance of Anchoring Methods

This subsection focuses on **C2** with respect to anchoring method selection. For a sequence of T attacks, we define:

- The **sequence-wide** defense success rate is the proportion of T images for which the network output at timestep $t = T$ successfully defends the attack. For any $A_{t=i}$, the defense is taken as the network output at $t = T$.
- The **concurrent** defense success rate is the proportion of T images for which the initial network output for each attack successfully defends it. For any $A_{t=i}$, the defense corresponds to $D_{t=i}$.

We evaluate Incremental DIPDefend across various hyperparameters and anchoring options, presenting the results in Figure 6 using the metrics defined above. The role of hyperparameters for each anchoring method is described in Section 3. Notably, MSE Anchoring consistently outperforms Spectral Anchoring, which in turn outperforms no anchoring (*Vanilla*). As observed in the previous subsection, the concurrent defense success rate of MSE Anchoring is comparable to that without anchoring. We offer some conjectures explaining the poor performance of Spectral Anchoring and EWC.

Inaptitude of Spectral Anchoring. Spectral Anchoring inherently preserves lower-frequency details. Anchoring weights λ are determined by the relative magnitude difference between the main MSE denoising objective ($\mathcal{E}$) and the spectral MSE. Stronger exponential masks reduce penalties for mismatched high-frequency details, improving performance. Due to computational limits, we could not explore additional hyperparameter settings. Trends in Figure 6 suggest that emphasizing lower-frequency preservation could improve performance. However, representative image

Updates per Image		Anchoring Weight $\lambda = 0.5$	$\lambda = 1.0$	$\lambda = 2.0$
MSE	1200	0.59 ± 0.05	0.56 ± 0.03	0.54 ± 0.08
MSE	1800	0.6 ± 0.03	0.58 ± 0.05	0.54 ± 0.05
MSE	2400	$\mathbf{0.6 \pm 0.06}$	0.56 ± 0.06	0.54 ± 0.04
EWC	2400		0.02 ± 0	
Vanilla	2400		0.03 ± 0.02	

(a) **Sequence-wide** defense success rate of Incremental DIPDefend. MSE: MSE anchoring. EWC: EWC anchoring. Vanilla: trains without anchoring.

Updates per Image	Masking Strength	Anchoring Weight $\lambda = 0.0005$	$\lambda = 0.001$	$\lambda = 0.002$
1200	$\alpha = 1$	0.09 ± 0.04	0.04 ± 0.02	0.03 ± 0.01
	$\alpha = 5$	0.12 ± 0.04	0.04 ± 0.03	0.03 ± 0.02
	$\alpha = 10$	0.12 ± 0.04	0.06 ± 0.03	0.03 ± 0.02
1800	$\alpha = 1$	0.17 ± 0.07	0.09 ± 0.04	0.06 ± 0.06
	$\alpha = 5$	0.15 ± 0.06	0.08 ± 0.04	0.04 ± 0.03
	$\alpha = 10$	0.14 ± 0.05	0.08 ± 0.05	0.04 ± 0.02
2400	$\alpha = 1$	0.16 ± 0.04	0.1 ± 0.03	0.03 ± 0.02
	$\alpha = 5$	0.18 ± 0.06	0.11 ± 0.04	0.05 ± 0.02
	$\alpha = 10$	$\mathbf{0.2 \pm 0.04}$	0.09 ± 0.05	0.03 ± 0.02

(b) **Sequence-wide** defense success rate of Incremental DIPDefend using spectral MSE anchoring. Anchoring weight is scaled lower here because the values of spectral MSE are naturally high.

Updates per Image		Anchoring Weight $\lambda = 0.5$	$\lambda = 1.0$	$\lambda = 2.0$
MSE	1200	0.6 ± 0.04	0.58 ± 0.05	0.56 ± 0.09
MSE	1800	0.58 ± 0.03	0.58 ± 0.07	0.54 ± 0.05
MSE	2400	$\mathbf{0.62 \pm 0.08}$	0.58 ± 0.04	0.51 ± 0.04
EWC	2400		0.03 ± 0.01	
Vanilla	2400		$\mathbf{0.62 \pm 0.05}$	

(c) **Concurrent** defense success rate of Incremental DIPDefend for non-spectral anchoring methods. Options are as stated in (a).

Updates per Image	Masking Strength	Anchoring Weight $\lambda = 0.0005$	$\lambda = 0.001$	$\lambda = 0.002$
1200	$\alpha = 1$	0.08 ± 0.03	0.04 ± 0.02	0.03 ± 0.01
	$\alpha = 5$	0.12 ± 0.05	0.04 ± 0.03	0.03 ± 0.02
	$\alpha = 10$	0.12 ± 0.03	0.05 ± 0.03	0.03 ± 0.02
1800	$\alpha = 1$	0.17 ± 0.07	0.09 ± 0.03	0.04 ± 0.05
	$\alpha = 5$	0.16 ± 0.07	0.07 ± 0.03	0.03 ± 0.01
	$\alpha = 10$	0.12 ± 0.06	0.09 ± 0.06	0.04 ± 0.01
2400	$\alpha = 1$	0.16 ± 0.03	0.09 ± 0.03	0.03 ± 0.02
	$\alpha = 5$	0.17 ± 0.05	0.11 ± 0.06	0.05 ± 0.02
	$\alpha = 10$	$\mathbf{0.19 \pm 0.04}$	0.08 ± 0.04	0.02 ± 0.02

(d) **Concurrent** defense success rate of Incremental DIPDefend using spectral MSE anchoring. Anchoring weight is scaled lower here because the values of spectral MSE are naturally high.

Fig. 6: Performance of Incremental DIPDefend upon comprehensive hyperparameter tuning. Updates per image refer to the number of training steps on DIP each adversarial attack receives upon their arrival. Masking strength refers to the parameter α for inverse-exponential MSE weighting, detailed at Section 3.2. Anchoring weight refers to the weight of the sum of anchoring objectives during training. It is scaled lower here because the values of spectral MSE are naturally high.

features do not always correspond to low-frequency components. Given this and the difficulty of balancing objectives with widely different magnitudes, Spectral Anchoring is practically challenging. Nevertheless, its potential remains high, as strong inverse-exponential masks can promote sparsity and memory efficiency, warranting future exploration.

Inaptitude of Elastic Weight Consolidation. Elastic Weight Consolidation (EWC) relies on two key assumptions. First, for two tasks A and B with parameters θ_A and θ_B, continual learning assumes that their optima are relatively close. Second, because EWC's Fisher information matrix aggregates gradients across datapoints, the method depends on data diversity for each task. Deep Image Prior violates both assumptions: the overfitted optima for two images I_A and I_B are not necessarily close, and there is no data diversity for any DIP task, as each DIP can overfit to only one image. Consequently, EWC performs poorly on Incremental DIP, as reflected in our results, where it fails to outperform no-anchoring methods.

6 Discussion

6.1 Extension of Spectral Anchoring

Our experiments indicate that Spectral Anchoring, although conceptually appealing, is challenging to stabilize in practice for Incremental DIP. The main difficulty arises from the magnitude mismatch between the spectral loss and the pixel-space reconstruction loss. Because FFT magnitudes of even clean images can be large, spectral MSE tends to dominate unless the anchoring weight is reduced by several orders of magnitude, complicating the balance between attack reconstruction and memory preservation.

Nevertheless, the intended benefit of Spectral Anchoring—the preservation of lower-frequency, semantically meaningful content—remains valuable. One promising direction is to move beyond hand-designed frequency masks toward learned spectral regularizers, including:

- Adaptive frequency masking, recomputing the mask per image or per task based on the spectrum of the target defense, reducing the need for manual hyperparameter tuning.
- Hybrid pixel–Spectral Anchoring, combining low-frequency MSE with spatial-domain MSE on selected regions, mitigating the risk of overly global spectral constraints.
- Phase-only anchoring, preserving structural information without constraining magnitude components that may vary widely during DIP optimization.

Our study shows that MSE Anchoring consistently outperforms spectral methods under constrained hyperparameter budgets. Nevertheless, the approaches outlined

above suggest that refined Spectral Anchoring could achieve memory-efficient preservation of structural content without the brittleness observed in our evaluations. We leave the development of such frequency-domain anchoring strategies for future work. Additionally, if the primary advantage lies in memory optimization, compression techniques like PCA may offer similar computational benefits as Spectral Anchoring.

6.2 Extension of Attack Sequence Settings

Although our experiments focus on sequences of length 50, extending Incremental DIPDefend to substantially longer or more structured attack streams is an important direction. While MSE Anchoring performs reliably within the tested regime, longer sequences may strain the model's capacity, as the DIP network must encode more defenses in its parameters. This motivates the need for mechanisms to prevent overaccumulation of anchoring constraints. Since adversarial attacks can be interpreted as image noise, DIPDefend is expected to remain largely invariant to different forms of adversarial perturbations, and thus generalize to several untested attack types.

A promising approach is to maintain a compact set of prototype defenses rather than anchoring all past examples. Prototype augmentation—where earlier defenses are clustered or distilled into a smaller set of representative prototypes—can reduce redundancy among similar attacks (e.g., multiple examples from the same class). Anchoring to these prototypes preserves essential defensive behavior while maintaining optimization stability in long sequences. This approach has been explored in prior works [29, 30] with more advanced designs for continual adversarial defense. Similarly, techniques from other continual learning studies may facilitate adaptation to longer sequences [20, 31].

7 Conclusion

We introduced **Incremental DIPDefend**, a continual adversarial defense mechanism based on Deep Image Prior, designed for scenarios with limited data and computational resources. By training a single DIP network across a sequence of adversarial attacks and anchoring its parameters to previous defenses, our method achieves continual robustness without the need to store multiple models or access any external training data.

Our experiments yield three main findings. First, Incremental DIPDefend consistently maintains denoising and defense quality across attack sequences, preserving both newly encountered and previously defended images. Second, the incremental training scheme—where the DIP network is sequentially updated rather than reinitialized—significantly outperforms baselines such as scratch retraining and Elastic Weight Consolidation. Third, among the anchoring strategies evaluated, simple MSE Anchoring provides the best balance between attack reconstruction and defense memory, achieving performance comparable to single-image DIPDefend while using only one network.

These results demonstrate that Incremental DIP is an effective mechanism for continual adversarial defense in low-data regimes and underscore the broader potential of parameterized single-example models in continual learning scenarios. Future work includes extending anchoring mechanisms, scaling to longer or adaptively structured attack sequences, and investigating whether incremental reconstruction priors can generalize to other online imaging tasks.

References

[1] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," 2017. [Online]. Available: https://arxiv.org/abs/1608.04644

[2] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," 2018. [Online]. Available: https://arxiv.org/abs/1710.06081

[3] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2019. [Online]. Available: https://arxiv.org/abs/1706.06083

[4] R. Wiyatno and A. Xu, "Maximal jacobian-based saliency map attack," 2018. [Online]. Available: https://arxiv.org/abs/1808.07945

[5] J. Kuruvilla and K. Gunavathi, "Lung cancer classification using neural networks for ct images," *Computer Methods and Programs in Biomedicine*, vol. 113, no. 1, pp. 202–209, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169260713003532

[6] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A survey of autonomous driving: Common practices and emerging technologies," *IEEE Access*, vol. 8, pp. 58 443–58 469, 2020.

[7] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, Q. Vuong, V. Vanhoucke, H. Tran, R. Soricut, A. Singh, J. Singh, P. Sermanet, P. R. Sanketi, G. Salazar, M. S. Ryoo, K. Reymann, K. Rao, K. Pertsch, I. Mordatch, H. Michalewski, Y. Lu, S. Levine, L. Lee, T.-W. E. Lee, I. Leal, Y. Kuang, D. Kalashnikov, R. Julian, N. J. Joshi, A. Irpan, B. Ichter, J. Hsu, A. Herzog, K. Hausman, K. Gopalakrishnan, C. Fu, P. Florence, C. Finn, K. A. Dubey, D. Driess, T. Ding, K. M. Choromanski, X. Chen, Y. Chebotar, J. Carbajal, N. Brown, A. Brohan, M. G. Arenas, and K. Han, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," in *Proceedings of The 7th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, J. Tan, M. Toussaint, and K. Darvish, Eds., vol. 229. PMLR, 06–09 Nov 2023, pp. 2165–2183. [Online]. Available: https://proceedings.mlr.press/v229/zitkovich23a.html

[8] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models," 2018. [Online]. Available: https://arxiv.org/abs/1805.06605

[9] E. Wong and J. Z. Kolter, "Provable defenses against adversarial examples via the convex outer adversarial polytope," 2018. [Online]. Available: https://arxiv.org/abs/1711.00851

[10] A. Jalal, A. Ilyas, C. Daskalakis, and A. G. Dimakis, "The robust manifold defense: Adversarial training using generative models," 2019. [Online]. Available: https://arxiv.org/abs/1712.09196

[11] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," *International Journal of Computer Vision*, vol. 128, no. 7, p. 1867–1888, Mar. 2020. [Online]. Available: http://dx.doi.org/10.1007/s11263-020-01303-4

[12] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015. [Online]. Available: https://arxiv.org/abs/1505.04597

[13] T. Dai, Y. Feng, B. Chen, J. Lu, and S.-T. Xia, "Deep image prior based defense against adversarial examples," *Pattern Recognition*, vol. 122, p. 108249, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320321004295

[14] H. Yoo, P. M. Hong, T. Kim, J. W. Yoon, and Y. K. Lee, "Defending against adversarial fingerprint attacks based on deep image prior," *IEEE Access*, vol. 11, pp. 78 713–78 725, 2023.

[15] R. Sutanto and S. Lee, "Real-time adversarial attack detection with deep image prior initialized as a high-level representation based blurring network," *Electronics*, vol. 10, p. 52, 12 2020.

[16] K. Zhang, Y. Li, W. Zuo, L. Zhang, L. V. Gool, and R. Timofte, "Plug-and-play image restoration with deep denoiser prior," 2021. [Online]. Available: https://arxiv.org/abs/2008.13751

[17] L. Ding, Y. Wang, X. Ding, K. Yuan, P. Wang, H. Huang, and Z. J. Wang, "Delving into deep image prior for adversarial defense: A novel reconstruction-based defense framework," 2021. [Online]. Available: https://arxiv.org/abs/2108.00180

[18] L. Wang, X. Zhang, H. Su, and J. Zhu, "A comprehensive survey of continual learning: Theory, method and application," 2024. [Online]. Available: https://arxiv.org/abs/2302.00487

[19] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the National Academy of Sciences*, vol. 114, no. 13, p. 3521–3526, Mar. 2017. [Online]. Available: http://dx.doi.org/10.1073/pnas.1611835114

[20] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," 2022. [Online]. Available: https://arxiv.org/abs/1706.08840

[21] L. Wang, X. Zhang, K. Yang, L. Yu, C. Li, L. Hong, S. Zhang, Z. Li, Y. Zhong, and J. Zhu, "Memory replay with data compression for continual learning," 2022. [Online]. Available: https://arxiv.org/abs/2202.06592

[22] R. Aljundi, E. Belilovsky, T. Tuytelaars, L. Charlin, M. Caccia, M. Lin, and L. Page-Caccia, "Online continual learning with maximal interfered retrieval," *Advances in neural information processing systems*, vol. 32, 2019.

[23] A. Mallya, D. Davis, and S. Lazebnik, "Piggyback: Adapting a single network to multiple tasks by learning to mask weights," in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[24] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong learning with dynamically expandable networks," 2018. [Online]. Available: https://arxiv.org/abs/1708.01547

[25] Z. Wu, C. Baek, C. You, and Y. Ma, "Incremental learning via rate reduction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 1125–1133.

[26] A. Aich, "Elastic weight consolidation (ewc): Nuts and bolts," 2021. [Online]. Available: https://arxiv.org/abs/2105.04093

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

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015. [Online]. Available: https://arxiv.org/abs/1512.03385

[29] F. Zhu, X.-Y. Zhang, C. Wang, F. Yin, and C.-L. Liu, "Prototype augmentation and self-supervision for incremental learning," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 5867–5876.

[30] Q. Wang, H. Ling, Y. Li, Q. Liu, R. Jia, and N. Yu, "Continual adversarial defense," 2025. [Online]. Available: https://arxiv.org/abs/2312.09481

[31] F. Zenke, B. Poole, and S. Ganguli, "Continual learning through synaptic intelligence," 2017. [Online]. Available: https://arxiv.org/abs/1703.04200