On-device Adversarial Purification via Distilled and Finetuned Denoising Diffusion Implicit Models

Mehmet Demirel
KIOS CoE, *University of Cyprus*
1 Panepistimiou Avenue, Nicosia, Cyprus
demirel.mehmet@ucy.ac.cy

Christos Kyrkou
KIOS CoE, *University of Cyprus*
1 Panepistimiou Avenue, Nicosia, Cyprus
kyrkou.christos@ucy.ac.cy

***Abstract*—Deep Neural Networks (DNNs) deployed on resource-constrained edge devices are highly vulnerable to adversarial examples, posing significant security risks in critical applications. While Diffusion Model-based Adversarial Purification (DM-AP) has emerged as a powerful defense paradigm, its substantial computational overhead due to iterative denoising renders it impractical for on-device protection. To address this, we propose an efficient adversarial purification framework leveraging Denoising Diffusion Implicit Models (DDIMs), which significantly reduce the number of sampling steps required for purification. We further enhance efficiency by employing knowledge distillation to create a compact student model from a larger pre-trained DDIM, followed by finetuning to preserve generative quality and robustness. Comprehensive experiments on small images (CIFAR-10) and larger images (CelebA-HQ), and real-world aerial images (AID) demonstrate that our DDIM-based approach achieves robustness comparable or superior to state-of-the-art DM-AP methods against a wide array of adversarial attacks. Crucially, our method achieves substantial inference speedups (e.g., up to 40-60 times faster than existing DM-AP) on both standard GPU and edge devices. Furthermore, the distilled, finetuned variant reduces model size and computation, making robust diffusion-based adversarial purification practical for edge devices and enabling on-device defense for pre-existing classifiers.**

***Index Terms*—Adversarial Purification, Edge Computing, Diffusion Models, Knowledge Distillation, Robustness**

I. Introduction

Deep Neural Networks (DNNs) have become the cornerstone of modern artificial intelligence, achieving unprecedented success. However, despite their capabilities, DNNs exhibit a critical vulnerability to adversarial examples. These are inputs meticulously crafted by adding small, often imperceptible, perturbations to benign samples, to cause the models to misclassify [1]. This poses security risks, especially in safety-critical applications (e.g., autonomous driving, drone surveillance). Increasingly, these applications are deployed on edge devices (resource-constrained platforms) where local processing is crucial for low latency and offline operation. Adversarial attacks at the edge can therefore undermine trust and reliability in these widely adopted AI-powered technologies.

In response to adversarial attacks, various defense strategies have been investigated. One prominent defense is Adversarial Training (AT) [2], [3]. AT aims to enhance model robustness by augmenting the training with adversarial examples. While effective against seen attack types, AT often struggles to generalize to unseen attacks [4], [5]. Moreover, AT often requires costly retraining for pre-trained classifiers, problematic for embedded edge models. Another major class of defense is Adversarial Purification (AP) [6]–[8]. AP methods operate as pre-processing steps, removing adversarial perturbations from an input before it is fed to a downstream classifier. These techniques often leverage generative models to map perturbed inputs back to the manifold of clean data. Recently, diffusion models (DMs) [9], [10] have emerged as a powerful tool for AP [6], owing to their strong generative capabilities and inherent denoising properties.

An advantage of AP, particularly DM-based AP (DM-AP), is that it can defend pre-existing classifiers without retraining and has broader generalization against unseen attacks. However, existing DM-AP methods face their own challenges. The iterative diffusion process incurs substantial computational overhead, prohibitive for resource-limited edge devices, making current DM-AP methods often impractical for on-device protection. To address this, we propose an efficient DM-AP framework, that significantly reduces the computational demands of adversarial purification. Our approach leverages Denoising Diffusion Implicit Models (DDIMs) and incorporates distillation and finetuning strategies to achieve rapid and robust defense suitable for edge environments. Our contributions are: (1) We propose the use of DDIMs for adversarial purification, significantly reducing the inference time compared to state-of-the-art (SOTA) adversarial purification methods. (2) We use knowledge distillation and finetuning techniques for DDIM-based purification (DDIM-AP), further improving efficiency. (3) We demonstrate on real edge devices that our proposed methods exhibits strong robustness against a wide array of adversarial attacks while being significantly more computationally efficient. Overall, experiments show substantial speedups (e.g., 40-60x over existing DM-AP) on both GPU and edge devices.

II. Related Works

A. Adversarial Defenses on Edge Devices

Robust and computationally efficient defenses are crucial for edge DNNs. AT [2], [3], a primary strategy for robustness, strengthens models by training on adversarial examples. Despite its effectiveness, AT's edge suitability is limited by generalization issues [4], [5] and retraining demands, infeasible for

deployed classifiers with restricted updates. Quantized Neural Networks (QNNs) [11] represent another avenue explored for adversarial robustness on edge devices. QNNs, while efficient, offer unreliable adversarial robustness due to unpredictable effects [12], making them risky for edge defense.

In contrast, AP offers a compelling alternative with improved robustness [6] as a pre-processor for deployed classifiers, requiring no retraining. Furthermore, AP aim to cleanse inputs before they reach the classifier, potentially offering broader generalization capabilities. However, the computational cost of existing AP methods, particularly DM-AP, has hindered their widespread adoption on resource-constrained edge platforms. Our work seeks to bridge this gap by developing an efficient DM-AP framework specifically optimized for edge deployment, thereby enabling robust defense for pre-existing models without retraining or significant accuracy trade-offs.

B. Adversarial Purification with Diffusion Models

AP aims to remove adversarial perturbations from an input signal before it is passed to a target model. AP uses generative models [13], [14], to project perturbed inputs to the clean data manifold. While these methods demonstrated potential, they were limited by generation fidelity. Recently, Denoising Diffusion Probabilistic Models (DDPMs) [9] have emerged as a SOTA paradigm for AP [6], [7]. DDPMs are defined by a forward noising process and a reverse denoising process. The forward process incrementally adds Gaussian noise to a clean sample $\mathbf{x}_0$ over T timesteps, and can be expressed in closed form [9]:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}t)\mathbf{I}), \tag{1}$$

where $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$ and $\{\beta_t\}_{t=1}^{T}$ is a schedule of small constants determining noise variance. As $t \to T$, the distribution of $\mathbf{x}_T$ converges to a Gaussian, $\mathcal{N}(0, \mathbf{I})$. The reverse process seeks to invert this noising. It trains a model, $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$, to approximate $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$. This model is typically parameterized by a DNN, usually a U-Net [15], $\epsilon_\theta(\mathbf{x}_t, t)$, trained to predict the noise component ϵ that was added to obtain $\mathbf{x}_t$ from $\mathbf{x}_0$. The mean of the reverse transition $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is [9]:

$$\mu_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t)\right) \tag{2}$$

and sampling proceeds iteratively as:

$$\mathbf{x}_{t-1} = \mu_\theta(\mathbf{x}_t, t) + \sigma_t \mathbf{z} \tag{3}$$

where $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$ if $t > 1$ and $\mathbf{z} = 0$ otherwise.

In DM-AP [6], an adversarial example $\mathbf{x}_{adv}$ is noised to a chosen timestep t^* (where $0 < t^* \leq T$), transforming it into $\mathbf{x}_{t^*}$. This noised state $\mathbf{x}_{t^*}$ is then iteratively denoised (using Eq. 3) for t^* steps to yield a purified sample $\mathbf{x}'_0$. The efficacy of DM-AP stems from a two-fold action, the initial noising disrupts adversarial perturbation, pushing the sample off the adversarial manifold. The subsequent reverse denoising, guided by a model trained on the distribution of clean data, then reconstructs a sample $\mathbf{x}'_0$ that ideally lies on the clean data manifold, stripped of the adversarial perturbation.

However, the iterative denoising in DDPM-based AP (DDPM-AP), requiring many $\epsilon_\theta(\mathbf{x}_t, t)$ evaluations, is computationally intensive, hindering edge deployment. To address this bottleneck, DDIMs [16], a generalization of DDPMs, were introduced. DDIMs permit a flexible, non-Markovian process, which allows for a significantly faster reverse sampling. This enables the generation of high-fidelity samples in substantially fewer steps compared to the DDPM, without requiring retraining ϵ_θ. While DDIMs have been recognized for accelerating inference in DMs, their application has been confined to image generation. To the best of our knowledge, their use for accelerating AP methods has not been investigated, and this work represents the first dedicated effort to leverage DDIMs for fast and robust AP.

III. Proposed Method

A. Purification with Denoising Diffusion Implicit Models

As detailed in Section II-B, the DDPM reverse process (Eq. 3) is Markovian, making it computationally intensive. DDIMs [16] present an alternative by offering a more generalized, non-Markovian process. Critically, DDIMs utilize the same $\epsilon_\theta(\mathbf{x}_t, t)$ but allow larger jumps in the reverse sequence for faster sampling. Given a current sample $\mathbf{x}_{\tau_i}$ at timestep τ_i and a previous timestep τ_{i-1} in a chosen reverse sequence, the DDIM update rule to obtain $\mathbf{x}_{\tau_{i-1}}$ is:

$$\begin{aligned}\mathbf{x}_{\tau_{i-1}} =& \sqrt{\bar{\alpha}_{\tau_{i-1}}}\left(\frac{\mathbf{x}_{\tau_i} - \sqrt{1-\bar{\alpha}_{\tau_i}}\epsilon_\theta(\mathbf{x}_{\tau_i}, \tau_i)}{\sqrt{\bar{\alpha}_{\tau_i}}}\right) \\ &+ \sqrt{1-\bar{\alpha}_{\tau_{i-1}}}\epsilon_\theta(\mathbf{x}_{\tau_i}, \tau_i)\end{aligned} \tag{4}$$

Here, the first term on the right-hand side is an estimate of the clean sample $\hat{\mathbf{x}}_0$, scaled by $\sqrt{\bar{\alpha}_{\tau_{i-1}}}$ to be appropriate for timestep τ_{i-1}. The second term reintroduces a scaled predicted noise, consistent with the noise level expected at τ_{i-1}.

In DDIM-AP, given an adversarial input $\mathbf{x}_{adv}$, we apply the forward diffusion (Eq. 1) for t^* timesteps to get $\mathbf{x}_{t^*}$, similar to DDPM-AP. Following this, denoising is performed using the DDIM sampling process, starting from $\mathbf{x}_{t^*}$ and applying the DDIM update rule (Eq. 4) to obtain the purified sample $\mathbf{x}'_0$. Unlike DDPMs, which would take t^* discrete steps from this point, DDIMs operate on a sparser schedule of timesteps. This schedule is typically defined by S_{DDIM} total steps, $(\tau_{S_{DDIM}}, \tau_{S_{DDIM}-1}, \ldots, \tau_1, \tau_0)$, which are chosen to span the full diffusion range (from T to 0). For the actual purification process beginning at $\mathbf{x}_{t^*}$, we only utilize the subset of these pre-defined τ_i timesteps that are less than or equal to t^*. This active purification sequence, $\mathcal{T}_{purify}$, thus starts at the largest $\tau_k \leq t^*$ and proceeds downwards: $(\tau_k, \tau_{k-1}, \ldots, \tau_0)$. The reverse process then iteratively applies Eq. 4 for $j = k, k-1, ..., 1$, transitioning from $\mathbf{x}_{\tau_j}$ to $\mathbf{x}_{\tau_{j-1}}$, ultimately yielding the purified sample $\mathbf{x}_{\tau_0} = \mathbf{x}'_0$. The number of actual steps taken for purification, $S_{purify} = k+1$ (i.e., the length of $\mathcal{T}_{purify}$), is determined by how many of the S_{DDIM}-defined timesteps fall at or below t^*. This S_{purify} is typically

much smaller than t^* (DDPM equivalent) and also smaller than S_{DDIM} itself unless $t^* = T$. Consequently, the purification process becomes considerably faster, making it more suitable for edge deployment.

B. Efficiency Enhancements via Distillation and Finetuning

While more efficient, the base denoising model remains large for good image generation, impacting speed and memory on edge devices. In this regard, we also propose to distill the DDIM model to a smaller model. To do this we use a simple distillation strategy. In this strategy, a smaller student U-Net model (ϵ_{θ_S}) is trained to mimic the noise prediction output of a larger, pre-trained teacher U-Net model (ϵ_{θ_T}). The teacher model, whose parameters are frozen, represents the original (DDPM and DDIM) noise prediction network. The student model typically has a more compact architecture. During the distillation, for each step, a random timestep t is sampled uniformly from all diffusion timesteps, and an input $\mathbf{x}_t$ is drawn directly from a standard Gaussian distribution. Both the teacher and the student models are fed this same pair $(\mathbf{x}_t, t)$. The teacher model computes its noise prediction $\hat{\boldsymbol{\epsilon}}_T = \epsilon_{\theta_T}(\mathbf{x}_t, t)$, which serves as the target supervision. Concurrently, the student model produces its own noise prediction $\hat{\boldsymbol{\epsilon}}_S = \epsilon_{\theta_S}(\mathbf{x}_t, t)$. The parameters of the student model are then updated by minimizing the Mean Squared Error (MSE) loss between its prediction $\hat{\boldsymbol{\epsilon}}_S$ and the teacher's prediction $\hat{\boldsymbol{\epsilon}}_T$. This transfers knowledge to a lightweight student ϵ_{θ_S}, which replaces ϵ_{θ_T} in Eq. 4, reducing model size and AP latency, improving edge deployment suitability. Finally, to mitigate quality degradation, the distilled student ϵ_{θ_S} is finetuned on the image dataset using the standard diffusion training objective (predicting the noise added to clean images).

IV. Experiments

A. Datasets and Classifier Network Architectures

For evaluations, we utilize CIFAR-10 [17] for small images, CelebA-HQ [18] for larger images and the Aerial Image Dataset (AID) [19] for real aerial image tests. On CIFAR-10, we compare against SOTA and standard defenses. On CelebA-HQ, we compare against other AP methods. Regarding classifier architectures, we follow the experimental setup of Nie et al. [6], employing WideResNet-28-10 [20] for CIFAR-10 experiments, ResNet-18 [21] for CelebA-HQ, and VGG16 [22] for AID experiments. The ResNet-18 classifier was selected for CelebA-HQ as a specific architecture was not explicitly provided in [6].

B. Adversarial attacks

We evaluate the robustness of our method against a comprehensive suite of strong adversarial attacks. For L_∞-bounded attacks (8/255), we employ the FGSM [2], PGD [3], APGD-CE [23], APGD-T [23] and FAB-T [23]. The L_2-norm attack evaluation utilizes the CW [24] (100 steps). Additionally, we present results against the Square Attack [25], a black-box attack, to assess performance in a query-based setting. Finally, to ensure a fair comparison with other AP methods, we also apply the BPDA+EOT (16/255) adaptive attack [14].

C. Comparative Evaluations

We comprehensively compare our DDIM-AP framework against established defenses, assessing robustness and computational performance on edge (NVIDIA Jetson Xavier NX) and high-performance (NVIDIA A100) GPUs. For evaluations, on small images, CIFAR-10 dataset, we compared our method with the SOTA DDPM-AP [6], serving as a direct comparison for AP efficacy, and PGD AT [3], representing a widely adopted robust training paradigm. Across all AP methods, the noising level is $t^* = 0.075$. This value, consistent with the setting in [6], is selected to balance effective perturbation disruption with the preservation of salient image features critical for accurate reconstruction. Additionally, $S_{DDIM} = 30$ is utilized for DDIM approaches, meaning the DDIM will have 30 steps in total while the DDPM has 1000. We present standard and robust accuracy on CIFAR-10 in Table I.

TABLE I
STANDARD ACCURACY AND ROBUST ACCURACY AGAINST VARIOUS ADVERSARIAL ATTACKS ON CIFAR-10, WITH A WIDERESNET-28-10 CLASSIFIER. FOR AP METHODS, THE DIFFUSION TIMESTEP IS $t^* = 0.075$. FOR DDIM METHODS, THE $S_{DDIM} = 30$.

Method	Standard Acc (%)	FGSM (%)	PGD (%)	CW (%)	APGD-CE (%)	APGD-T (%)	FAB-T (%)	SQUARE (%)
Regular Training	**93.84**	12.49	0.00	0.00	0.00	0.00	0.01	0.16
PGD AT [3]	86.00	60.10	54.18	7.19	51.57	49.58	50.15	58.94
DDPM [6]	87.16	**78.60**	**83.14**	85.76	84.52	84.59	86.41	84.39
DDIM (Ours)	89.83	68.50	82.38	**88.16**	**84.88**	**85.56**	**89.12**	**86.79**
Distilled Finetuned DDIM (Ours)	80.63	51.20	63.64	75.67	69.06	70.48	78.38	73.18

As observed in Table I, a classifier regular training is extremely vulnerable to adversarial perturbations, with robust accuracy dropping to 0% against almost all attacks. PGD AT substantially enhances robustness, achieving significantly higher robust accuracy against stronger attacks. However, this improvement incurs a significant reduction in standard accuracy and limited generalization, particularly against the L_2 based CW attack. Furthermore, AT necessitates retraining the entire model, which is often impractical for deployed systems. DDPM-AP is a strong baseline, with 87.16% standard accuracy and consistent, high robustness across all attacks. Our DDIM-AP improves DDPM standard accuracy (+2.67%) in standard accuracy and shows competitive or superior robustness across all attacks. The lower performance against FGSM may be attributed to FGSM being a single-step attack whose perturbations might be less effectively smoothed by fewer, larger steps of DDIM compared to numerous, finer steps of DDPM. Nevertheless, its strong performance against iterative/sophisticated attacks (e.g., PGD) highlights its defensive strength. DDIM maintains high robustness with fewer steps. Finally, the distilled finetuned DDIM trades some accuracy (80.63% standard, reduced robust accuracy) for efficiency, but still offers substantial protection over undefended models and PGD AT (against all attacks except FGSM). While accuracy is paramount, the practical applicability of DM-AP, especially on edge devices, hinges on their computational performance.

Table II quantifies the inference speed, and details the model complexity.

TABLE II
INFERENCE SPEED ON NVIDIA JETSON XAVIER NX AND A100 GPU FOR CIFAR-10.

Purification Method	Jetson Xavier NX		A100		GFLOPS	Parameter Number (M)
	Time per Frame (s)	FPS	Time per Frame (s)	FPS		
DDPM [6]	4.6981	0.21	1.2568	0.80	6.24	35.75
DDIM (Ours)	0.2160	5.14	0.0528	18.95	6.24	35.75
Distilled Finetuned DDIM (Ours)	**0.1371**	**8.11**	**0.0316**	**31.70**	**1.08**	**4.44**

As shown in Table II, the DDPM exhibits extremely slow inference speeds, 0.21 FPS on the edge device and 0.80 FPS on a GPU. This latency makes it unsuitable for most edge applications. In contrast, our standard DDIM-AP achieves a substantial speed-up, approximately 25 times faster than DDPM-AP on both devices. This significant improvement in inference speed, while largely maintaining or even enhancing robustness (Table I), brings adversarial purification closer to practical edge deployment. To further address the constraints of edge deployment, our distilled finetuned DDIM offers even more efficiency gains. Compared to DDPM, it delivers an approximately 39-fold speed increase, suitable for many edge applications. Crucially, in Table II, the distilled model reduces computational load (GFLOPS) by approximately 82.7% and parameter count by 87.6% compared to DDPM and DDIM. This substantial reduction in model size translates to lower memory requirements, critical factors in edge devices.

For evaluating performance on larger images, CelebA-HQ dataset, we follow [6], using BPDA+EOT attack to validate performance. We compare against other strong generative models capable of AP, such as NVAE [26] and GAN-based approaches [27]–[29], which encode adversarial images to a latent space and then synthesize purified images from the decoder. Performance is assessed on the smiling attribute, consistent with [6]. The diffusion timestep for purification methods is set to $t^* = 0.4$. Similar to the CIFAR-10 evaluations, $S_{DDIM} = 30$ is utilized. Standard and robust accuracy results are presented in Table III.

TABLE III
STANDARD ACCURACY AND ROBUST ACCURACY AGAINST BPDA+EOT ON CELEBA-HQ, WITH A RESNET-18 CLASSIFIER. WE EVALUATE ON THE SMILING ATTRIBUTE CLASSIFIER.THE DIFFUSION TIMESTEP IS $t^* = 0.4$. FOR DDIM METHODS, THE $S_{DDIM} = 30$.

Method	Standard Accuracy	BPDA+EOT
NVAE [26]	93.55%	0.00%
GAN+OPT [27]	93.49%	3.41%
GAN+ENC+OPT [29]	**93.68%**	0.78%
GAN+ENC [28]	90.55%	40.40%
DDPM [6]	89.78%	55.73%
DDIM (Ours)	92.42%	**78.40%**
Distilled Finetuned DDIM (Ours)	87.15%	74.31%

As shown in Table III, alternative AP models like NVAE and GAN variants exhibit high standard accuracy but offer little to no robustness against BPDA+EOT, with GAN+ENC being a notable exception at 40.40% robust accuracy. The DDPM method provides a substantial improvement, achieving 55.73% robust accuracy, establishing a strong DM-AP baseline. Our standard DDIM-AP method significantly outperforms all other approaches, achieving a robust accuracy of 78.40%, a +22.67% increase over DDPM, while also maintaining a high standard accuracy of 92.42%. This superior robustness is largely attributed to the deterministic nature of our DDIM implementation ($\eta = 0$). This determinism presents a considerable challenge for gradient approximation attacks like BPDA. BPDA relies on the gradient of the loss with respect to the purifier's output as a surrogate for the true gradient. With our DDIM approach, the purified output is often substantially different from the adversarial input, having effectively removed adversarial structures. Consequently, the gradient calculated from this cleansed output becomes a poor approximation for enhancing the original input's adversarial nature through the purification process. The direct, unvarying path of DDIM with $\eta = 0$ filters adversarial noise effectively, potentially creating an obfuscated gradient landscape for the attacker, leading to higher robust accuracy. Moreover, the distilled finetuned DDIM, shows reduced standard accuracy to 87.15% and robust accuracy to 74.31% compared to the full DDIM, but still offers noticeable robustness. It significantly surpasses DDPM and other AP methods in robust accuracy, demonstrating a favorable trade-off between efficiency and robustness for larger images.

TABLE IV
INFERENCE SPEED ON NVIDIA JETSON XAVIER NX AND A100 GPU FOR CELEBA-HQ.

Purification Method	Jetson Xavier NX		A100		GFLOPS	Parameter Number (M)
	Time per Frame (s)	FPS	Time per Frame (s)	FPS		
DDPM [6]	214.4115	0.005	11.3171	0.09	249.36	113.67 M
DDIM (Ours)	7.5725	0.13	0.3405	2.94	249.36	113.67 M
Distilled Finetuned DDIM (Ours)	**3.1464**	**0.32**	**0.1958**	**5.11**	**91.97**	**20.31 M**

Table IV highlights the severe latency of DDPM for larger images, with an FPS of only 0.005 on the Jetson Xavier NX and 0.09 on the A100, making it impractical for dynamic scenarios. Our standard DDIM offers a dramatic improvement, achieving approximately 26-33 times faster inference on both platforms. This speed-up, coupled with its superior robustness (Table III), makes DDIM a more viable solution. For even greater efficiency on edge devices, the distilled finetuned DDIM further accelerates inference, delivering around a 68-fold speed increase on Jetson and a 57-fold increase on A100 compared to DDPM. The distilled model reduces GFLOPS by approximately 63.1% and parameter count by approximately 82.1% compared to DDPM. This makes the distilled DDIM model significantly more suitable for deployment on edge devices, offering faster processing and lower memory footprint, as detailed in Table IV, without a significant loss in robustness.

Finally, we assess the performance on the Aerial Image Dataset (AID), representing a real-world edge surveillance

TABLE V
STANDARD ACCURACY AND ROBUST ACCURACY AGAINST VARIOUS ADVERSARIAL ATTACKS ON AID, WITH A VGG16 CLASSIFIER. THE DIFFUSION TIMESTEP IS $t^* = 0.3$, $S_{DDIM} = 30$.

Method	Standard Acc (%)	FGSM (%)	PGD (%)	CW (%)
No Defence	87.14	5.91	0.00	0.00
DDPM	71.15	64.43	65.91	62.74
DDIM (Ours)	70.86	60.18	63.01	64.00
Distilled Finetuned DDIM (Ours)	68.72	54.32	56.11	58.84

scenario. The results, summarized in Table V, reinforce the trends observed in previous datasets. The undefended VGG16 classifier proves highly susceptible to attacks, with robust accuracy plummeting to near zero against PGD and CW. Both the baseline DDPM and our proposed DDIM framework significantly restore robustness. Our DDIM method achieves a standard accuracy of 70.86% and maintains high robust accuracy (e.g., 63.01% against PGD), results that are closely aligned with the computationally expensive DDPM baseline. While the Distilled Finetuned DDIM exhibits a moderate trade-off in robustness (achieving 56.11% against PGD), it remains a highly effective defense compared to the undefended model. Crucially, since the purification model architecture and image resolution for AID are identical to those used in the CelebA-HQ experiments, the substantial inference speeds and computational gains reported in Table IV directly apply here. Consequently, our distilled approach offers comparable protection to SOTA methods but delivers the massive speedups (e.g., 68x faster on Jetson Xavier NX) necessary to enable feasible, robust on-device processing for drone-based applications.

D. Ablation studies

Impact of S_{DDIM}: The hyperparameter S_{DDIM} determines the total number of steps in the full DDIM reverse sampling schedule, spanning from T to 0. However, when purification starts from an intermediate noised state $\mathbf{x}_{t^*}$, only the subset of these S_{DDIM}-defined timesteps that are $\leq t^*$ are used. Let S_{purify} denote the actual number of steps in this active purification sequence. S_{DDIM} directly influences S_{purify} and the specific timesteps involved, thereby impacting both computational efficiency and purification quality. We conducted an ablation study on CIFAR-10 with $t^* = 0.075$, varying S_{DDIM} and observing the robust accuracy against PGD attack. The results, showing accuracy as a function of both S_{DDIM} (secondary x-axis) and the corresponding S_{purify} (primary x-axis), are presented in Figure 1.

As shown in 1 , robust accuracy does not improve monotonically with S_{DDIM}; while the overall trend is upward, the path fluctuates. For example, after an initial peak in accuracy at $S_{DDIM} = 30$ (leading to $S_{purify} = 3$ for $t^* = 0.075$), increasing S_{DDIM} to 40 (where S_{purify} also happens to be 3 for this t^*) results in a substantial drop in robustness. Accuracy then generally recovers and continues to fluctuate as S_{DDIM} and S_{purify} increase further. This non-monotonic behavior arises because changing S_{DDIM} not only alters the number

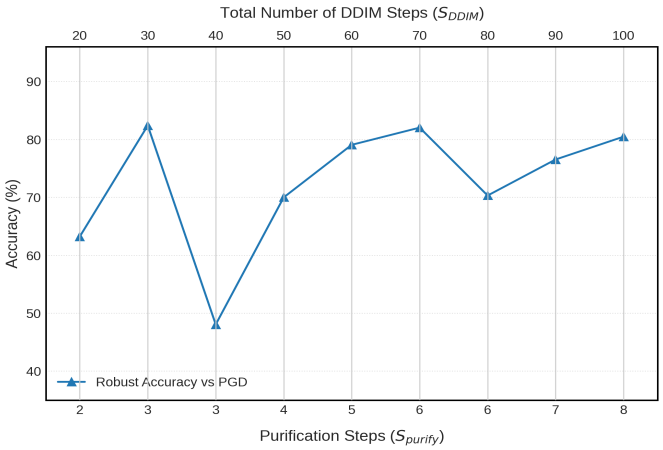

Fig. 1. Robust accuracy against PGD attack on CIFAR-10 as S_{DDIM} varies, with a fixed initial noising timestep $t^* = 0.075$.

of purification steps S_{purify} for a fixed t^*, but also modifies the specific values and spacing of the intermediate timesteps $\{\tau_0, \ldots, \tau_k\}$ within the active purification range $[0, t^*]$. Some configurations of these steps, determined by S_{DDIM}, may be more or less effective for the DDIM update rule (Eq. 4) at a given t^*. The plot demonstrates that simply increasing S_{DDIM} (and thus generally S_{purify}) does not guarantee better performance. An optimal S_{DDIM} must be found empirically, balancing robustness and inference speed.

Impact of DDIM Sampling Stochasticity (η): The DDIM sampling process offers a parameter η that controls the level of stochasticity in the generation. A setting of $\eta = 0$ is deterministic and $\eta = 1$ mimics DDPM stochasticity with fewer steps. We hypothesize that the deterministic nature of DDIM contributes significantly to its robustness against gradient-based adaptive attacks like BPDA+EOT. To investigate this, in Table VI, we compare the robust accuracy of our DDIM and distilled finetuned DDIM models using both deterministic ($\eta = 0$) and stochastic ($\eta = 1.0$) sampling on the CelebA-HQ dataset against the BPDA+EOT attack. The same experimental settings are used as our CelebA-HQ evaluations.

TABLE VI
ROBUST ACCURACY AGAINST BPDA+EOT ON CELEBA-HQ (RESNET-18 CLASSIFIER, $t^* = 0.4$, $S_{DDIM} = 30$) WITH DIFFERENT η VALUES.

Purification Method	Robust Accuracy with $\eta = 0$ (Deterministic)	Robust Accuracy with $\eta = 1.0$ (Stochastic)
DDIM (Ours)	78.40%	55.32%
Distilled Finetuned DDIM (Ours)	74.31%	50.00%

As observed in Table VI, switching to stochastic DDIM ($\eta = 1.0$) leads to a substantial degradation in robust accuracy for both our models. Specifically, DDIM and distilled DDIM shows 23.08% and 24.31% decrease in their robust accuracies respectively. Notably, the robust accuracy of DDIM with $\eta = 1.0$ closely approaches that of the DDPM (Table III). This aligns with the intuition that DDIM with $\eta = 1.0$ emulates

DDPM. This significant performance difference underscores the advantage of deterministic purification ($\eta = 0$). As discussed in Section IV-C for CelebA-HQ, BPDA relies on approximating the gradient of the loss with respect to the purifier's output. When $\eta = 0$, the DDIM purifier acts as a fixed, deterministic function. If this function effectively removes adversarial patterns, the gradient calculated on the cleansed output becomes a poor, uninformative estimate for modifying the original adversarial input to bypass the purifier. The deterministic path may create a highly non-linear or even discontinuous landscape for the attacker's gradient estimation [30]. Conversely, when $\eta = 1.0$, the stochasticity might smooth the loss landscape or provide varied outputs which BPDA can exploit to find better average gradient direction. The determinism of DDIM ($\eta = 0$) is key for defending from adaptive attacks like BPDA+EOT.

V. Conclusions

This paper proposed an efficient DDIM-based framework that reduces sampling steps and purification costs compared to SOTA DM-AP, with experiments on CIFAR-10, CelebA-HQ, and AID images demonstrating its efficacy. It achieves adversarial robustness comparable, and in several cases superior, to DDPM-AP and PGD AT. Crucially, these robustness gains are achieved with significant computational savings, exhibiting speedups of up to 40-60 times. Our analysis also highlighted the particular strength of deterministic DDIM sampling against gradient-based adaptive attacks. Future work includes investigating adaptive step-size selection to further optimize the speed-robustness trade-off.

VI. Acknowledgement

This work was supported by the European Union's Horizon Europe research and innovation programme under grant agreement No. 101168067 (GuardAI). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

References

[1] J. Peck, B. Goossens, and Y. Saeys, "An introduction to adversarially robust deep learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 4, pp. 2071–2090, 2023.

[2] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[3] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[4] C. Laidlaw, S. Singla, and S. Feizi, "Perceptual adversarial robustness: Defense against unseen threat models," *arXiv preprint arXiv:2006.12655*, 2020.

[5] H. M. Dolatabadi, S. Erfani, and C. Leckie, "ℓ_∞-robustness and beyond: Unleashing efficient adversarial training," in *European Conference on Computer Vision*. Springer, 2022, pp. 467–483.

[6] W. Nie, B. Guo, Y. Huang, C. Xiao, A. Vahdat, and A. Anandkumar, "Diffusion models for adversarial purification," *arXiv preprint arXiv:2205.07460*, 2022.

[7] J. Yoon, S. J. Hwang, and J. Lee, "Adversarial purification with score-based generative models," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 062–12 072.

[8] C. Shi, C. Holtz, and G. Mishne, "Online adversarial purification based on self-supervision," *arXiv preprint arXiv:2101.09387*, 2021.

[9] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[10] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *arXiv preprint arXiv:2011.13456*, 2020.

[11] Y. Guo, "A survey on methods and theories of quantized neural networks," *arXiv preprint arXiv:1808.04752*, 2018.

[12] M. Costa and S. Pinto, "David and goliath: An empirical evaluation of attacks and defenses for qnns at the deep edge," in *2024 IEEE 9th European Symposium on Security and Privacy*, 2024, pp. 524–541.

[13] P. Samangouei, M. Kabkab, and R. Chellappa, "Defense-gan: Protecting classifiers against adversarial attacks using generative models," *arXiv preprint arXiv:1805.06605*, 2018.

[14] M. Hill, J. Mitchell, and S.-C. Zhu, "Stochastic security: Adversarial defense using long-run dynamics of energy-based models," *arXiv preprint arXiv:2005.13525*, 2020.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International Publishing, 2015, pp. 234–241.

[16] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[17] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009.

[18] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[19] G.-S. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu, "Aid: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3965–3981, 2017.

[20] S. Zagoruyko and N. Komodakis, "Wide residual networks," in *Proceedings of the British Machine Vision Conference (BMVC)*. BMVA Press, September 2016, pp. 87.1–87.12.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[23] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in *International conference on machine learning*. PMLR, 2020, pp. 2206–2216.

[24] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*. Ieee, 2017, pp. 39–57.

[25] M. Andriushchenko, F. Croce, N. Flammarion, and M. Hein, "Square attack: a query-efficient black-box adversarial attack via random search," in *European conference on computer vision*. Springer, 2020, pp. 484–501.

[26] A. Vahdat and J. Kautz, "Nvae: A deep hierarchical variational autoencoder," *Advances in neural information processing systems*, vol. 33, pp. 19 667–19 679, 2020.

[27] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 8110–8119.

[28] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, and D. Cohen-Or, "Encoding in style: a stylegan encoder for image-to-image translation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 2287–2296.

[29] L. Chai, J.-Y. Zhu, E. Shechtman, P. Isola, and R. Zhang, "Ensembling with deep generative views," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 997–15 007.

[30] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *International conference on machine learning*. PMLR, 2018, pp. 274–283.