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

Mehmet Demirel
demirel.mehmet@ucy.ac.cy
Christos Kyrkou
kyrkou.christos@ucy.ac.cy

KIOS CoE
University of Cyprus
Nicosia, Cyprus

Abstract

This supplementary document provides comprehensive implementation details for the denoising diffusion models central to our work on 'On-device Adversarial Purification via Distilled and Finetuned Denoising Diffusion Implicit Models.' It elaborates on the U-Net architectures for both teacher and distilled student models, their respective diffusion process parameters , and the full training configurations. This includes specifics for the teacher models, the knowledge distillation phase, the finetuning of student models, and the training of downstream classifiers. These details pertain to experiments conducted on the CIFAR-10 and CelebA-HQ datasets, aiming to ensure reproducibility and offer a thorough understanding of our experimental setup.

1 Diffusion Model Setup

Our diffusion models leverage the Denoising Diffusion Probabilistic Model (DDPM) framework [2]. The core denoising network is a U-Net [3], which has been adapted to predict the noise component ε at a given timestep t. Time embeddings, generated through sinusoidal positional encodings that are subsequently projected through linear layers, are integrated into the U-Net's residual blocks. All models were trained using the Adam optimizer with a Mean Squared Error (MSE) loss criterion, measuring the difference between the predicted and true noise. To enhance the quality of generated samples, an Exponential Moving Average (EMA) of the model weights was employed during training.

For the CIFAR-10 dataset, which consists of 32×32 pixel images, the U-Net architecture was configured to process 3-channel (RGB) images. It started with 128 base hidden channels and employed channel multipliers of `[1, 2, 2, 2]`, resulting in 128, 256, 256, and 256 channels at its four respective resolution levels. Each of these levels contained two residual blocks, each incorporating Group Normalization, SiLU activation, and convolutional layers. Time embeddings were added after the first convolution within these blocks. A self-attention mechanism was applied at the second resolution level, corresponding to a 16×16 feature

map, and a dropout rate of 0.1 was used. The diffusion process for CIFAR-10 operated over $T = 1000$ timesteps, following a linear beta schedule with $\beta_{\text{start}} = 0.0001$ and $\beta_{\text{end}} = 0.02$. The reverse process variance $q(x_{t-1}|x_t, x_0)$ is β_t. The model was trained to predict the noise ε. Training was conducted using the Adam optimizer with a learning rate of 2×10^{-4} and a batch size of 128 for a total of 2040 epochs. A learning rate warmup period of 5000 steps was utilized, along with gradient clipping at a norm of 1.0. The EMA decay for model weights was set to 0.9999.

For the higher-resolution CelebA-HQ dataset (256×256 images), the U-Net architecture also processed 3-channel (RGB) images, beginning with 128 base hidden channels. The channel multipliers were `[1, 1, 2, 2, 4, 4]`, yielding 128, 128, 256, 256, 512, and 512 channels across six resolution levels. Similar to the CIFAR-10 setup, each level featured two residual blocks with a comparable internal structure. Self-attention was applied at the fifth resolution level (a 16×16 feature map), and a dropout rate of 0.0 was used for this model. The diffusion process for CelebA-HQ also spanned $T = 1000$ timesteps with a linear beta schedule from $\beta_{\text{start}} = 0.0001$ to $\beta_{\text{end}} = 0.02$. However, the reverse process variance $q(x_{t-1}|x_t, x_0)$ is defined as $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$, with $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$. This model also predicted the noise component ε. The training utilized the Adam optimizer with a learning rate of 2×10^{-5} and a batch size of 64, running for 1200 epochs. A 5000-step learning rate warmup was included, along with gradient clipping at a norm of 1.0 and an EMA decay of 0.9999.

2 Knowledge Distillation

Knowledge distillation was employed to transfer the denoising capabilities from the larger, pre-trained U-Net teacher models, detailed in Section 1, to more compact student models. This strategy aims to reduce the computational footprint and inference latency, making them suitable for on-device adversarial purification while striving to preserve efficacy. During the distillation process, the teacher model weights were frozen.

For the CIFAR-10 student U-Net (32×32 images), the architecture was a 3-channel (RGB) U-Net with 64 base hidden channels, reduced from the teacher's 128. It used channel multipliers of `[1, 2, 2]` (compared to the teacher's `[1, 2, 2, 2]`), resulting in 64, 128, and 128 channels across three resolution levels. Each level contained one residual block, a reduction from the teacher's two, while retaining Group Normalization, SiLU activation, convolutional layers, and time embedding integration. Self-attention was applied at the second resolution (16×16 feature map), and a dropout rate of 0.1 was maintained.

For the CelebA-HQ student U-Net (256×256 images), the architecture was also a 3-channel (RGB) U-Net, but with 96 base hidden channels (teacher: 128). It employed channel multipliers of `[1, 1, 2, 2, 3]` across five resolution levels (teacher: `[1, 1, 2, 2, 4, 4]` over six levels). Each level comprised one residual block (teacher: 2), while retaining the core components. Self-attention was applied at the fourth resolution level (a 32×32 feature map). The dropout rate was set to 0.05 (teacher: 0.0).

Regarding distillation training, the Adam optimizer was used with an initial learning rate of 2×10^{-4}, which included a linear warmup phase over 5000 steps. Training proceeded for 500 epochs. Gradients were clipped to an L_2 norm of 1.0. An EMA of student weights with a decay of 0.9999 was maintained. Specifically, the CIFAR-10 student model was trained with a batch size of 2048, while the CelebA-HQ student model used a batch size of 32.

2.1 Finetuning

After the knowledge distillation phase, the compact student models were further finetuned on their respective datasets (CIFAR-10 and CelebA-HQ) to potentially enhance their performance. The finetuning process continued to use the DDPM objective, where the model predicts the noise component ϵ added to an image at a given timestep t, optimized using a MSE loss. For both datasets, the student models were finetuned using the Adam optimizer with a learning rate of 1×10^5 for 100 epochs. A linear learning rate warmup was applied over the first 500 training steps. Similar to the distillation training, gradients were clipped to an L_2 norm of 1.0, and an Exponential Moving Average (EMA) of the model weights was maintained with a decay of 0.9999. The batch size during finetuning was set to 128 for the CIFAR-10 student model and 32 for the CelebA-HQ student model. The U-Net architectures of the student models and the diffusion process parameters (beta schedule, T=1000 timesteps) remained unchanged from their configuration in the distillation stage.

3 Classifiers

For the classification, we trained a WideResNet-28-10 for CIFAR-10 and a ResNet-18 for CelebA-HQ.

The WideResNet-28-10 for CIFAR-10 was trained using the Stochastic Gradient Descent (SGD) optimizer. The optimizer was configured with an initial learning rate of 0.2, a momentum of 0.9, weight decay of 5e-4. A MultiStepLR scheduler was used to reduce the learning rate by a factor of 0.2 at epochs 60, 120, and 160. The training loop ran for a maximum of 200 epochs with a batch size of 512, utilizing Cross-Entropy Loss. The architecture itself incorporated a dropout rate of 0.3. For data handling and setup, 10% of the CIFAR-10 training data (5000 images) was reserved for validation.

For ResNet-18 on CelebA-HQ, the model was trained using the AdamW optimizer. This optimizer was set up with an initial learning rate of 0.008 and a weight decay of 0.01. A CosineAnnealingLR scheduler adjusted the learning rate, with $T_{\max}$ set to 50 epochs and $\eta_{\min}$ to 1% of the initial learning rate. The training loop proceeded for a maximum of 50 epochs, using a batch size of 1024 and Cross-Entropy Loss. Input images were resized to 256x256. During training, data augmentation consisted of random horizontal flips.

3.1 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.

Figure 1 illustrates that the robust accuracy against PGD does not monotonically improve with S_{DDIM} (or the corresponding actual purification steps, S_{purify}). While accuracy generally trends upwards with more purification steps, the path is not smooth. For example,

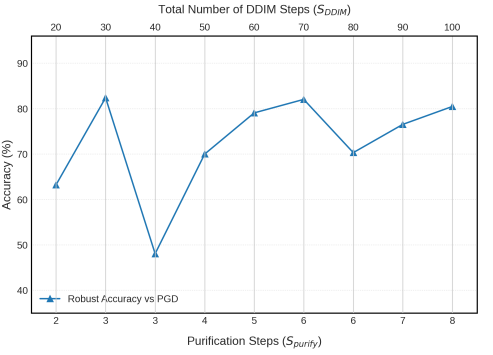


Figure 1: Robust accuracy against PGD attack on CIFAR-10 as the number of DDIM purification steps (S_{DDIM}) varies, with a fixed initial noising timestep $t^* = 0.075$.

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 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 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 1, 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 1: Robust accuracy against BPDA+EOT on CelebA-HQ (ResNet-18 classifier, $t^* = 0.4$, $S_{DDIM} = 30$) for DDIM variants 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 1, 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. This aligns with the intuition that DDIM with $\eta = 1.0$ emulates DDPM. This significant performance difference underscores the advantage of deterministic purification ($\eta = 0$). 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

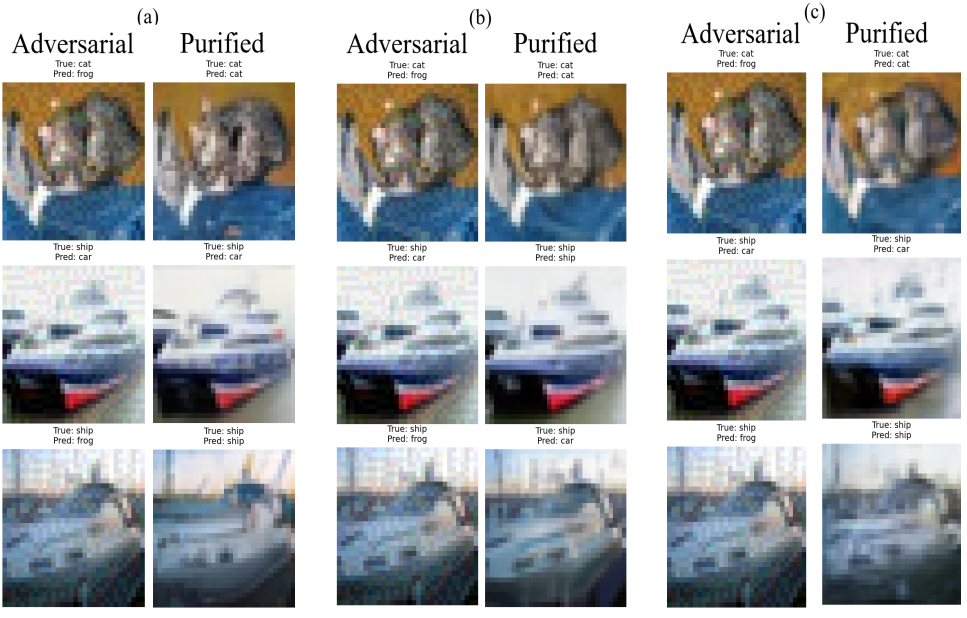


Figure 2: Qualitative examples of adversarial purification on the CIFAR-10 dataset against FGSM_{L_∞} (8/255). The figure shows results from three methods: (a) DDPM Purification, (b) DDIM Purification, and (c) Distilled Finetuned DDIM Purification. Each panel (a, b, c) displays multiple image examples (rows), with columns showing the adversarially attacked image (left), and the purified image (right). The initial noising timestep for all purification methods is $t^* = 0.075$. For DDIM-based methods (b and c), the number of reverse steps is $S_{DDIM} = 30$.

.

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 [1]. 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.

4 Qualitative Examples

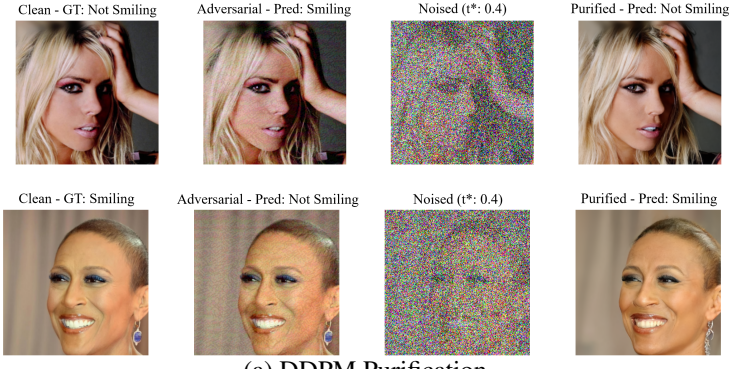
This section presents qualitative results of our adversarial purification methods on both CIFAR-10 and CelebA-HQ datasets. Figure 1 & 2 below illustrate the visual quality of images after being subjected to adversarial attacks and subsequently purified by DDPM, our standard DDIM approach, and our distilled finetuned DDIM variant.

References

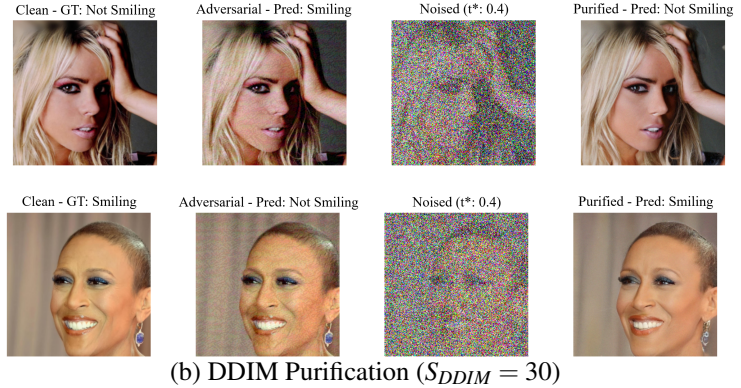
[1] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pages 274–283. PMLR, 2018.

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

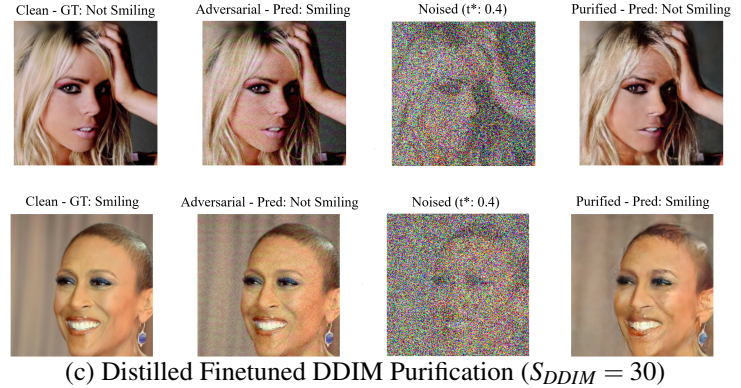
[3] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. ISBN 978-3-319-24574-4.



(a) DDPM Purification



(b) DDIM Purification ($S_{DDIM} = 30$)



(c) Distilled Finetuned DDIM Purification ($S_{DDIM} = 30$)

Figure 3: Purification examples on the CelebA-HQ dataset against FGSM_{L_∞} (8/255). (a) DDPM Purification. (b) DDIM Purification. (c) Distilled Finetuned DDIM Purification. Each panel displays two examples (stacked vertically), showing the original image (left), attacked image (center), and purified image (right). For all purification methods, the initial noising timestep is $t^* = 0.4$. For DDIM-based methods (b and c), $S_{DDIM} = 30$.