RETPUR: DIFFUSION PURIFICATION MODEL FOR DEFENDING HASH RETRIEVAL TARGET ATTACKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep Neural Networks (DNNs) have harnessed their formidable representational capabilities to attain remarkable performance in image retrieval models. Nonetheless, in cases where malicious actors introduce adversarial perturbations into the test dataset, the retrieval model may readily yield results that are either irrelevant or intentionally manipulated by the attacker. Specifically, the targeted attack is notable for producing predefined results, thereby inflicting a more adverse impact on retrieval performance. While adversarial purification has demonstrated effectiveness in countering adversarial attacks, its application in retrieval tasks remains unexplored. Addressing these concerns, we introduce a free-trained purification model denoted as **RetPur** aimed at purifying adversarial test dataset, thereby mitigating the issue of targeted attacks within both uni-modal and cross-modal retrieval systems. **RetPur** employs a pre-trained diffusion model, offering a plug-and-play convenience, while utilizing adversarial samples as conditioning factors to guide image generation, thereby enhancing task accuracy. In terms of retrieval system architecture, our study pioneers the incorporation of adversarial purification tasks into uni-modal (Image-to-Image) and cross-modal (Image-to-Image, Image-to-Text) hash retrieval systems, specifically tailored to image retrieval scenarios. Furthermore, we explore the application of adversarial purification tasks to a wider array of attacks, including both generative and iterative approaches. Through an extensive series of experiments, it can be concluded that the purified dataset exhibits retrieval performance in the retrieval systems that is closely akin to that of the original dataset, even across different attacks and modalities.

1 INTRODUCTION

Deep Neural Networks (DNNs) perform well in image processing tasks such as image classification, image retrieval, semantic segmentation, and object localization. However, extensive research has shown that trained DNNs are prone to interference from adversarial samples (McDaniel et al. (2016);Papernot et al. (2016);Ma et al. (2019);Cohen et al. (2020);Mayer & Timofte (2020);Wu et al. (2021)) and output incorrect task results. Among them, adversarial samples refer to the attacker exerting small perturbations on the test set data, which makes it difficult to distinguish it from original data with the naked eye, but can cause the model to misjudge.

Researchers have conducted extensive research on defense against adversarial attacks. These defense methods are mainly divided into three types: adversarial training (Sarkar et al. (2021);Deng et al. (2021);Zhao et al. (2022)), adversarial detection (Liu et al. (2018);Yang et al. (2021)), and adversarial purification (Nie et al. (2022);Wu et al. (2023);Kim et al. (2022)). The purpose of adversarial training is to involve adversarial instances or optimize instances in the worst-case scenario during the training process of neural networks. Although it improves the robustness of the target classifier against adversarial attacks of visible types, it often cannot handle adversarial attacks or image damage of invisible types. Adversarial detection (Zhang et al. (2022)) aims to determine whether a given sample is an adversarial one based on the discrepancy between natural and adversarial distributions. However, it is difficult to quantify the threshold for distinguishing adversarial samples from original data, and it is prone to errors in determining the distribution of the data. Unlike the first two methods, adversarial purification (Nie et al. (2022);Wang et al. (2022)) only applies to the test set data and has a plug-and-play feature, without increasing the complexity of the model. Specifically, adversarial purification methods typically use learning to generate models as purification models, converting the attacked test set data into original data and inputting it into the attacked model.

Currently, many works about adversarial examples have been studied in image classification, but very few researches focus on the security of deep hashing based retrieval models. Different from the typical classification, hash retrieval models (Li et al. (2015);Cao et al. (2018);;Bai et al. (2020a);Yu et al. (2021)) aims to learn the semantic similarity between images, and its final outputs are discrete binary codes instead of categories. Especially target attacks can cause query results to return fixed categories of information, completely interfering with the performance of the retrieval system, as shown in the Appendix A.

This paper is first to consider applying adversarial purification tasks to the field of image retrieval. Inspired by recent works (Nie et al. (2022);Wang et al. (2022)), we propose a diffusion-based purification model **RetPur** to purify the attacked retrieval dataset. The underlying principle of the model can be outlined as follows: Attackers obtain retrieval datasets and proceed to launch attacks by introducing interference. Adversarial datasets are input into the **RetPur** model, wherein a forward denoising process is applied to generate pristine noise-free images. This process iteratively eliminates noise, guided by the input adversarial samples, ultimately resulting in the production of purified datasets. The proposed model has plug-and-play convenience, and the purified test dataset demonstrates superior anti-attack and retrieval performance on the hash retrieval model.

We conducted extensive experiments to evaluate the robustness of our model in both uni-modal image retrieval tasks and cross-modal image retrieval systems. Owing to the heightened destructive potential of targeted attacks, our choice for the unimodal retrieval task encompassed target generative attacks and target iterative attacks (Bai et al. (2020b)). For cross-modal retrieval tasks, we exclusively employed TA-DCH (Wang et al. (2023)) as the attack method, as it is specifically designed for cross-modal models. In order to assess the efficacy of the purification model, we curated original retrieval datasets, purified original retrieval datasets, and purified adversarial datasets. Subsequently, we conducted a comparative analysis of their retrieval outcomes within the retrieval model, utilizing MAP and T-MAP values as performance metrics. Based on experimental findings, it can be concluded that the retrieval outcomes of both the purified original datasets and the purified adversarial datasets within the image retrieval model closely approximate those of the original original datasets, with a MAP value error of less than 2%. This demonstrates the versatility of the proposed **RetPur** model, which can be effectively employed across a broader spectrum of purification tasks.

2 Related work

2.1 Adversarial attacks against hashing retrieval

At present, there are few studies on adversarial attacks against retrieval systems. Similar to image classification models, DNN-based image retrieval models are also susceptible to adversarial sample attacks. Attacks against image retrieval tasks belong to black box attacks, which are divided into two types: target black box attacks and non target black box attacks. According to different retrieval scenarios, the retrieval model is divided into uni-modal retrieval model and cross-modal retrieval model. Existing adversarial attacks against retrieval systems include non-targeted (Yang et al. (2018);Xiao et al. (2020);Feng et al. (2020);Li et al. (2021b)) and targeted attacks (Bai et al. (2020b);Xiao & Wang (2021);Hu et al. (2021);Wang et al. (2021b)) against uni-modal retrieval, non-targeted attacks (Li et al. (2019);Li et al. (2020);Li et al. (2021a);Zhu et al. (2023);Zhang et al. (2023)) against cross-modal retrieval, and targeted attacks (Wang et al. (2023)) against cross-modal retrieval. According to the different ways of generating adversarial samples, the above attacks can be divided into iterative attacks and generative attacks.

Targeted attacks against hashing retrieval. Non-targeted attacks aim to make retrieval systems return results irrelevant to the query. While targeted ones not only realize the above purpose but also disguise adversarial examples into the attacker's specified category, such as violence, bloodiness, and pornography. Compared with non-targeted attacks, targeted attacks cause retrieval systems to provide users with insufferable malicious information, which seriously reduces their trustworthiness. Based on this, we select target attacks as the research focus and considers two retrieval tasks: uni-modal tasks and cross-modal tasks. At the same time, we also analyzed the impact of the proposed algorithm on iterative and generative attacks.

2.2 Defense strategies against targeted attacks

For deep hashing-based retrieval, Wang et al. (2021a) proposed the first effective adversarial training algorithm based on the targeted attack (dubbed ATRDH here) by narrowing the semantic gap between the adversarial samples and the original samples in the Hamming space. Adversarial training (Goodfellow et al. (2014b), Madry et al. (2017)) aims to augment the training data with generated adversarial examples, which is the most common training strategy against various adversarial attacks. However, models trained by adversarial training can only defend against specific attack methods used during training, and cannot resist attacks from other attack methods, resulting in poor generalization performance.

Adversarial purification. Refer to Appendix B, a generative model that can restore original images from attacked images is trained and used as a preprocessor in adversarial purification, which is more convenient than adversarial training. Samangouei et al. (2018) proposed defense-GAN, a generator that can restore original images from attacked images. Song et al. (2017) revealed how to detect and purify adversarial samples using an autoregressive generative model. Srinivasan et al. (2021) presented a purification approach for denoising autoencoders using the Metropolis Adjusted Langevin algorithm (MALA). Grathwohl et al. (2019), Du & Mordatch (2019) and Hill et al. (2020) proposed and demonstrated that MCMC wich EBMs can purify adversarial examples. Similarly, Yoon et al. (2021) used the denoising score-based model for purification. Due to the development of diffusion models in the field of image generation, some works (Nie et al. (2022);Wang et al. (2022);Wu et al. (2022)) consider using the DDPM model and its improved versions as purification models to purify disturbed test set data, but only consider using simple datasets such as cifar10 and imagenet for classification tasks. Recently, some studies have considered classification or segmentation tasks for more complex datasets, such as 3D point clouds (Sun et al. (2022)), self supervised vascular segmentation (Kim et al. (2022)), and audio (Wu et al. (2023)). Based on the above research foundation, we propose to apply adversarial purification methods to the field of hash retrieval.

3 RetPur: diffusion-based purification model

3.1 Structure of the Guided Diffusion Model

We propose to use a pre-trained DDPM to purify adversarial samples, which is one of the most widely used diffusion model (Song & Ermon (2019);Song et al. (2020a);Song et al. (2020b)). The DDPM model consists of two processes: the forward diffusion process and the reverse generation process. Briefly, the DDPM model is a process of adding and removing noise.

In the DDPM model, both of the two processes are defined by Markov chains with a length of T. Assume p_{data} is the distribution of all images x in the dataset, p_θ is the distribution of generate images. It is worth noting that p_θ tends to approach p_{data}, i.e. $p_\theta \sim p_{data}$.

The forward diffusion process. Formally, let $x^0 \sim p_{data}$, the diffusion step t is denoted by a superscript. Set $p_{noise} = N(0_{3N}, I_{3N\times 3N})$ as the result of the diffusion process, and N is the Gaussian distribution. The diffusion process from original data x^0 to x^T is defined as below:

$$q(x^1, \cdots, x^T|x^0) = \prod_{t=1}^{T} q(x^t|x^{t-1}), \tag{1}$$

$$q(x^t|x^{t-1}) = N(x^t; \sqrt{1-\beta_t}x^{t-1}, \beta_t I), \tag{2}$$

where q is the distribution of original data in the forward diffusion process, $q(x^T)$ is close to p_{noise}. β_t are the diffusion coefficients and their values are predefined small positive constants.

According to Ho et al. (2020), there is a closed form expression for $q(x^t|x^0)$. Define constants $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$. Then, we have $q(x^t|x^0) = N(x^t; \sqrt{\alpha_t}x^{t-1}, (1-\alpha_t)I)$. Therefore, when T is large enough, $\bar{\alpha}_t$ goes to 0, and $q(x^T|x^0)$ becomes close to the final distribution p_{noise}. Note that x^t can be directly sampled through the following equation:

$$x^t = \sqrt{\bar{\alpha}_t}x^0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \text{ where } \epsilon \text{ is a standard Gaussion noise.} \tag{3}$$

The reverse sampling process. Let $x^T \sim p_{noise}$ be a latent variable. It is a Markov process that predicts and eliminates the noise added in the diffusion process. The reverse process from latent x^T

to purified image x^0 is defined as:

$$p_\theta(x^0, \cdots, x^{T-1}|x^T) = \prod_{t=1}^{T} p_\theta(x^{t-1}|x^t), \tag{4}$$

$$p_\theta(x^{t-1}|x^t) = N(x^{t-1}; \mu_\theta(x^t, t), \sigma_t^2 I), \tag{5}$$

where $\mu_\theta(x^t, t)$ is a neural network parameterized by θ, and the variance σ_t^2's can be either time-step dependent constants (Ho et al. (2020)) or learned by a neural network (Nichol & Dhariwal (2021)). To generate a sample, we first sample $x^T \sim p_{noise}$, then draw $x^{t-1} \sim p_\theta(x^{t-1}|x^t)$ for $t = T, T-1, \cdots, 1$, and finally outputs purified data x^0.

Conditional gradient. The work (Wang et al. (2022)) proposed a novel Guided DDPM, in which the authors use the adversarial images x_{adv} to guide the reverse process of the DDPM. This method is used to purify classified datasets, including CIFAR10 and ImageNet. Experiments have shown that the purified images performs better in classification tasks than DDPM. The difference from original DDPM is that the gradient descent in formula 5 is regulated by a conditional gradient.

$$p_\theta(x^{t-1}|x^t, x_{adv}) = N(x^{t-1}; \mu_\theta(x^t, t) - s\sigma_t^2 \nabla_{x^t} D(x^t, x_{adv}^t), \sigma_t^2 I), \tag{6}$$

where s represents the guidance scale, $-\sigma_t^2 \nabla_{x^t} D(x^t, x_{adv}^t)$ is the conditional gradient. The mathematical derivation of the conditional gradient will be presented in the Appendix C.

3.2 The Basic Idea of Hash Retrieval Purification Model

According to previous research work, this imperceptible disturbance can be divided into two situations: One is a disturbance generation method based on optimization and heuristic algorithms, as shown in formula 7. The other is a generative disturbance generation method, as shown in formula 8. Assuming x is a original image and x_{adv} is an adversarial image. $x_{adv(1)}$ and $x_{adv(2)}$ represent adversarial samples generated by two different generation methods, respectively. The formula is expressed as follows.

$$x_{adv(1)} = x + \delta, \tag{7}$$

where δ represents noise that does not affect the structural information of the original image and can be seen as redundant information of the image.

$$x_{adv(2)} = Generator(x + Decoder(s)), \tag{8}$$

where s represents the target semantics generated from the target attack label. The attacker generates an image of the same size as a original image based on the target semantics, concatenates it after the original image, and then inputs the synthesized image into the generator to obtain adversarial samples.

Due to the structural characteristics of the DDPM model itself, Gaussian noise is added to submerge and counteract interference, while removing the interference of Gaussian noise. As shown in Figure 1, we propose the following model:

It is worth noting that we considers two cases of x_{adv}, namely it has two forms: $x_{adv} = x_{adv(1)}$ or $x_{adv} = x_{adv(2)}$. Given an adversarial example x_{adv} as the input at $t = 0$, i.e., $x^0 = x_{adv}$. We first diffuse the adversarial image x_{adv} for T steps following Equation 3:

$$x^t = \sqrt{\bar{\alpha}_t} x_{adv} + \sqrt{1 - \bar{\alpha}_t}\epsilon, \ \epsilon \sim N(0, I). \tag{9}$$

Next, we condition the reverse denoising process of the DDPM on the adversarial image x_{adv}. Specifically, we adapt the reverse denoising distribution $p_\theta(x^{t-1}|x^t)$ in Equation 5 to a conditional distribution $p_\theta(x^{t-1}|x^t, x_{adv})$. Let noise image x^T as input and U-Net model as a model for fitting parameter μ_θ and ϵ_θ. The representation of reverse generated image x^0 can be obtained by using Equation 6.

$$x^0 = \textbf{U-Net}(x^T, \mu_\theta, \epsilon_\theta, D(x^t, x_{adv}^t), s, T), \ t = T, \ldots, 1, \tag{10}$$

where x_{adv}^t is the guidance at step t. It's obtained by diffusing x_{adv} t steps according to Equation 9.

Combining the Representation of Equation 9 and Equation 10, the whole purification operation can be denoted as a function **RetPur**: $\mathbb{R}^d \times \mathbb{R} \rightarrow \mathbb{R}^d$:

$$\textbf{RetPur}(x_{adv}) = \textbf{U-Net}(\sqrt{\bar{\alpha}_T} x_{adv} + \sqrt{1 - \bar{\alpha}_T}\epsilon, \mu_\theta, \epsilon_\theta, D(x^t, x_{adv}^t), s, T), \ t = T, \ldots, 1. \tag{11}$$

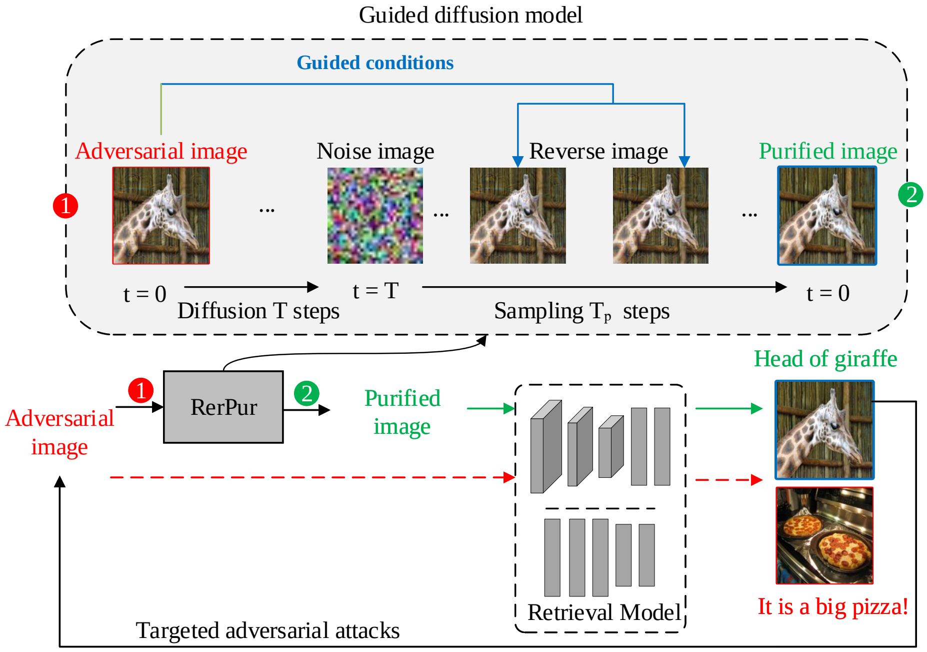

Figure 1: An illustration of **RetPur**. Given a pre-trained diffusion model with T diffusion steps, we incrementally introduce minor noise to the adversarial image during the forward diffusion process until it transforms into a pure noise image. Following this, in the purification step denoted as T_p, we employ the input adversarial image as a guiding condition and gradually eliminate noise using pre-trained model parameters, resulting in the acquisition of purified images.

3.3 Learn Algorithm

As shown in Algorithm 1 and Algorithm 2, we divide the learning algorithm of the purification model into two parts: training and sampling. For simplicity, this paper ignores the fact that the forward process variances β_t are learnable by reparameterization and instead fix them to constants. Thus, in our implementation, the approximate posterior q has no learnable parameters, we just need to discuss our choices in $p(x^{t-1}|x^t, x_{adv})$.

Algorithm 1 Training

```
Input: $x^0 \sim p_{data}$, diffusion step $T$, initialize a U-net as $\epsilon_\theta(x^0, 0)$, learning rate $\eta$
Output: noise predictor $\epsilon_\theta(x^t, t)$
repeat
    $x^0 \sim p_{data}$
    $t \sim Uniform(\{1, \dots, T\})$
    $\epsilon \sim N(0, I)$
    Take gradient descent step on:
        $\nabla_\theta \|\epsilon - \epsilon_\theta(\sqrt{\bar{a}_t} x^0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, t)\|$
    update $\epsilon_\theta(\cdot)$:
        $\epsilon_\theta(x^t, t) = \epsilon_\theta(x^{t-1}, t-1) - \eta \cdot \nabla_\theta(\cdot)$
until converged
return $\epsilon_\theta(x^t, t)$
```

Algorithm 2 Sampling

```
Input: $x^T \sim N(0, I)$, purify step $T_p$, distance metric $D(\cdot)$, guidance scale $s$
Output: $x^0$
for $t = T_{pur}, \dots, 1$ do
    $z \sim N(0, I)$ if $t > 1$, else $z = 0$
    compute conditional gradient:
        $\nabla G = -s\nabla_{x^t} D(x^t, x^t_{adv})$
    $\mu_\theta(x^{t-1}, t-1) = \frac{1}{\sqrt{\bar{\alpha}_t}}(x^t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x^t, t)) + \nabla G \cdot \epsilon_\theta(x^t, t))$
    $x^{t-1} = \mu_\theta(x^{t-1}, t-1) + \sigma z$
end for
return $x^0$
```

Training. Similar to Ho et al. (2020), we use μ_θ and ϵ_θ as parameters predicted by the U-Net model to fit the distribution of original data.

$$\mu_\theta(x^{t-1}, t-1) = \frac{1}{\sqrt{\bar{\alpha}_t}}(x^t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x^t, t)), \tag{12}$$

where ϵ_θ is a function approximator intended to predict ϵ from x^t. To sample $x^{t-1} \sim p_\theta(x^{t-1}|x^t)$ is to compute $x^{t-1} = \frac{1}{\sqrt{\bar{\alpha}_t}}(x^t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x^t,t)) + \sigma z$,where $z \sim N(0,I)$. It can be seen that the only parameter that needs to be estimated is ϵ_θ.

Ho et al. (2020) found it beneficial to sample quality to train on the following variant of the variational bound:

$$L_{simple}(\theta) := \mathbb{E}_{t,x^0,\epsilon}[\|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}x^0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t)\|^2], \tag{13}$$

where t is uniform between 1 and T.

By adopting the idea of Equation 13, we propose a loss function for the image purification operation performed by the **RetPur**:

$$L_{purify}(\theta) := \mathbb{E}_{t,x_{adv},\epsilon}[\|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_T}x_{adv} + \sqrt{1-\bar{\alpha}_T}\epsilon, t)\|^2], \tag{14}$$

where x_{adv} has two forms. During the model training process, we only need to train the noise predictor ϵ_θ. And the training method of the noise predictor is shown in Algorithm 1.

Sampling. After the training process is completed, we can obtain the trained noise predictor $\epsilon_\theta(x^t,t)$ and use it to update parameters, gradually sampling and generating images. The update method for parameter μ_θ are as follows:

$$\mu_\theta(x^{t-1}, t-1) = \frac{1}{\sqrt{\bar{\alpha}_t}}(x^t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x^t,t)) - s\nabla_{x^t}D(x^t, x^t_{adv})\epsilon_\theta(x^t,t), \tag{15}$$

where s is a constant which represents gradient scale, $D(\cdot)$ represents some distance metric. In our experiments, we adopt the mean square error (MSE) or negative structure similarity index measure (SSIM) as our distance metric $D(\cdot)$. Thus, we can get the update formula of x^{t-1}:

$$x^{t-1} = \frac{1}{\sqrt{\bar{\alpha}_t}}(x^t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x^t,t)) - s\nabla_{x^t}D(x^t, x^t_{adv})\epsilon_\theta(x^t,t) + \sigma z,\ z \sim N(0,I). \tag{16}$$

4 Experiments

4.1 Experimental Settings

We conducted two sets of experiments on the uni-modal hash retrieval models and the cross-modal hash retrieval models.

Dataset. We conduct extensive experiments on five widely used retrieval datasets: Wikipedia (Pereira et al. (2013)), IAPR-TC12 (Escalante et al. (2010)), FLICKR-25K (Huiskes & Lew (2008)), MS-COCO (Lin et al. (2014)) and NUS-WIDE (Chua et al. (2009)). The division of datasets differs between uni-modal and cross-modal hash retrieval systems, as detailed in the Appendix D.

Attacked models and Attack methods. For the uni-modal experiment, we select representative retrieval systems DPSH and DSH, and introduce two types of attacks: iterative attacks exemplified by P2P and DHTA, along with generative attacks represented by ProS-GAN. For the cross-modal experiment, we opt for representative retrieval systems DGCPN and DADH, and incorporat the target attack TA-DCH. The specific settings are detailed in Appendix E.

Evaluation Metrics. Following the previous works (Jiang & Li (2017), Bai et al. (2020b)), we use MAP and T-MAP to evaluate the performance of Purification model. MAP is a widely used metric to evaluate the performance of cross-modal retrieval. T-MAP is slightly different from MAP in that the original labels of the query set are replaced with the target labels specified by the attacker. Since T-MAP uses the target labels as the test labels, the higher the T-MAP, the stronger the targeted attack.

Implementation details: In experiments, our proposed model is implemented via PyTorch and a NVIDIA 3090 GPU. Training attack methods and retrieval models on the same datasets. All attack methods have a training bit of 32. The iterative attack methods DHTA and P2P have an iteration count of 2000 and 500, while the iterations of the generative attack methods ProS-GAN and TA-DCH are 100 and 150, respectively. For the pre-trained diffusion model, we use models from Ho et al. (2020) and Dhariwal & Nichol (2021). Diffusion step T is set to 1,000 and purify step T_p is set to 45. In the calculation of conditional gradients, SSIM is selected as the default distance metric function, with a guidance scale s of 1,000. We adopt the Adam with learning rate 10^{-4} to optimize the proposed **RetPur** model.

Table 1: Quantitative results (%) of P2P, DHTA, ProS-GAN (Bai et al. (2020b);Wang et al. (2021a)) attacks targeting Image to Image (I2I) retrieval tasks. Original uses benign images to retrieve images, Attack methods use generated adversarial samples to retrieve images, and **RetPur** purifies the adversarial samples generated by attack methods and generates purified samples to retrieve images.

I2I		FLICKR-25K		MS-COCO		NUS-WIDE	
		DPSH	DPH	DPSH	DPH	DPSH	DPH
MAP	original	81.06	77.07	63.69	58.95	73.06	69.92
	P2P	65.81	62.51	52.52	44.61	61.97	54.81
	RetPur-P2P	**80.17**	**76.58**	**63.15**	**58.56**	**72.62**	**69.89**
	DHTA	61.09	61.03	46.17	42.43	51.98	53.89
	Retpur-DHTA	**80.14**	**76.64**	**63.14**	**58.57**	**72.72**	**69.83**
	ProS-GAN	63.29	64.58	45.97	44.54	53.45	50.73
	RetPur-ProS	**79.50**	**75.99**	**62.16**	**57.76**	**71.56**	**69.14**
T-MAP	original	73.67	73.21	50.53	49.22	59.17	60.27
	P2P	82.04	80.51	48.69	53.17	67.81	71.28
	RetPur-P2P	**63.60**	**64.06**	**43.21**	**42.73**	**57.06**	**57.01**
	DHTA	88.39	83.85	60.01	59.67	69.68	75.68
	Retpur-DHTA	**63.95**	**64.28**	**43.48**	**42.91**	**57.44**	**57.21**
	ProS-GAN	92.76	86.02	72.36	65.25	90.00	76.31
	RetPur-ProS	**72.60**	**72.31**	**46.49**	**46.41**	**54.25**	**53.69**

4.2 Main Results

4.2.1 Uni-modal hash retrieval purification

Referring to the experiment conducted on the three attacks in work (Wang et al. (2021b)), we selected the same three datasets: FLICKR-25K, MS-COCO, NUS-WIDE.

Purification against iterative attacks: Regarding the method of generating adversarial samples, we select two gradient based iterative attack methods as the baseline, including P2P and DHTA (Bai et al. (2020b)). As shown in the Table 1, the purified image retrieval results approximate the retrieval results of the original image. In addition, compared to the results of using adversarial sample retrieval, the MAP value significantly increased and the T-MAP value significantly decreased. Wang et al. (2021a) considers using adversarial training methods on both FLICKR-25K and NUS-WIDE datasets to improve the performance of DPH retrieval model. For the FLICKR-25K dataset, under P2P and DHTA attacks, the T-MAP values of the purified data on the DPH model were 64.06 and 64.28, respectively, which were lower than the retrieval results of the adversarial samples on the DPH model after adversarial training (Wang et al. (2021a)). For the NUS-WIDE dataset, the retrieval results of the two defense methods on the two attacks are very similar, with a difference of within 0.4%. This confirms that our proposed purification model performs better than existing adversarial training methods.

Purification against generative attacks: We use ProS-GAN (Wang et al. (2021b)) as a generative attack, which is the first target generative attack based on deep hash retrieval systems. Instead of heuristic selection of a hash code as a representative of the target label, a flexible prototype code is generated in PrototypeNet using semantic invariance as the expected pillar of the target label in the optimization view. Therefore, compared to iterative attacks, ProS-GAN has shorter training time and stronger attack ability. As shown in Table 1, our purification model performs excellently. Compared to the MAP values of the adversarial samples generated by ProS-GAN, the purified data showed a 10% to 20% increase on the retrieval model, with a difference of less than 1.6% compared to the original data. It is worth noting that the T-MAP values of the purified data on all datasets are smaller than the T-MAP values retrieved from the original data, which proves that the performance of purified data retrieval without being classified into target categories is better than that of original data. It is concluded that the adversarial purification model can effectively resist uni-modal target generative attacks.

Table 2: Quantitative results (%) of TA-DCH (Wang et al. (2023)) attacks targeting Image to Image (I2I) retrieval tasks. Original uses benign images to retrieve images, TA-DCH uses generated adversarial samples to retrieve images, and **RetPur** purifies the adversarial samples generated by TA-DCH and generates purified samples to retrieve images.

I2I		WIKI		IAPR		FLICKR-25K		MS-COCO		NUS-WIDE	
		DGCPN	DADH	DGCPN	DADH	DGCPN	DADH	DGCPN	DADH	DGCPN	DADH
MAP	original	38.09	44.93	64.74	69.35	81.99	80.58	61.92	58.49	77.91	75.60
	TA-DCH	28.63	39.80	39.28	38.77	67.05	65.30	44.94	55.15	43.27	46.64
	RetPur	**36.22**	**42.93**	**63.20**	**65.70**	**80.56**	**83.21**	**59.60**	**56.46**	**76.57**	**73.71**
T-MAP	original	13.33	12.01	31.17	33.43	56.71	57.07	34.27	36.91	35.45	35.92
	TA-DCH	42.37	23.80	52.29	79.85	88.57	96.27	61.17	45.83	87.31	86.48
	Retpur	**18.66**	**18.44**	**33.61**	**35.74**	**59.25**	**58.64**	**39.90**	**37.96**	**38.34**	**40.58**

Table 3: Quantitative results (%) of TA-DCH (Wang et al. (2023)) attacks targeting Image to Text (I2T) retrieval tasks. Original uses benign images to retrieve texts, TA-DCH uses generated adversarial samples to retrieve texts, and **RetPur** purifies the adversarial samples generated by TA-DCH and generates purified samples to retrieve texts.

I2T		WIKI		IAPR		FLICKR-25K		MS-COCO		NUS-WIDE	
		DGCPN	DADH	DGCPN	DADH	DGCPN	DADH	DGCPN	DADH	DGCPN	DADH
MAP	original	37.79	43.15	65.24	72.49	80.56	78.45	65.21	62.28	76.42	77.65
	TA-DCH	28.30	37.94	38.94	39.10	65.36	62.76	45.85	58.11	43.17	44.17
	RetPur	**36.15**	**41.35**	**63.49**	**69.27**	**79.10**	**84.87**	**63.63**	**60.36**	**75.38**	**75.21**
T-MAP	original	13.34	12.11	31.37	37.60	56.54	62.39	34.25	38.67	35.64	39.23
	TA-DCH	42.34	23.67	52.07	81.11	85.07	93.98	62.82	47.06	87.16	84.34
	RetPur	**18.65**	**18.61**	**33.76**	**40.69**	**59.19**	**62.89**	**41.10**	**39.28**	**38.23**	**42.97**

4.2.2 CROSS-MODAL HASH RETRIEVAL PURIFICATION

Considering the wider applicability of the cross-modal retrieval model, we selected the adversarial samples generated by the TA-DCH (Wang et al. (2023)) attack as input for the purification model to explore whether the purification task is suitable for querying multimodal data. Specifically, we consider the tasks of Image to Image and Image to Text. We use the **RetPur** model to purify the adversarial retrieval dataset, input the purified retrieval set into a cross modal hash retrieval system, and retrieve semantically similar text and images in the database. As shown in Table 2 and Table 3, The purification model we proposed is also suitable for cross-modal retrieval tasks. The purified image performs well in both image to image and image to text tasks, far exceeding the results of the attack and approaching the results of the original image query. Using the MAP value as an indicator, the performance of the purified data in the retrieval model has significantly improved compared to the adversarial samples, with a performance gap of less than 2% compared to the original data. Among them, in the experiment using the DADH retrieval model on the FLACKR-25K dataset, our method outperformed the original data by at least 3% in both Image to Image and Image to Text retrieval tasks. When T-MAP is used as an evaluation indicator, our method's results on the retrieval model are close to those of the original data, and far lower than the results of adversarial samples retrieval. It is concluded that the adversarial purification model can effectively resist cross-modal target generative attacks.

4.2.3 PURIFY ORIGINAL DATASETS

Considering more complex scenarios, attackers may randomly attack the retrieved dataset. In this case, some of the data in the dataset after the attack is original data, while others are adversarial samples. Therefore, more general purification tasks need to be considered. The performance of the purification model needs to ensure that the retrieval effect after original data purification is similar to that after adversarial sample purification, both approaching the retrieval effect of original effect. We designed experiments to purify original datasets on both uni-modal and cross-modal retrieval tasks,

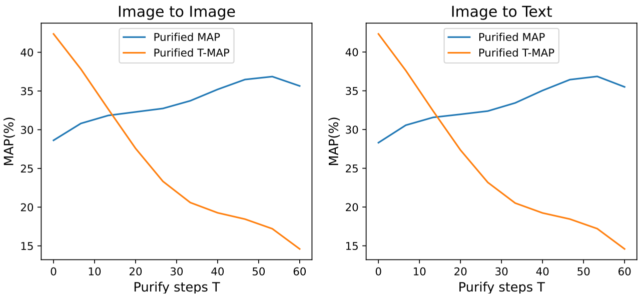

Figure 2: The correlation analysis of purification performance and purify step on Image to Image and Image to Text tasks.

Table 4: Purify original datasets of uni-modal retrieval task.

	Image to Image		Image to Text	
	MAP	T-MAP	MAP	T-MAP
SSIM	36.22	18.66	36.15	18.65
MSE	36.52	14.52	36.38	14.39

and then compared the purified retrieval results with the original image retrieval results to analyze the impact of purification tasks on the performance of original data retrieval. As shown in the Table 7, Table 8 and Table 9 of Appendix F, the results of uni-modal tasks show that the retrieval results of the purified dataset is only slightly reduced compared to the original dataset, within 2%. Based on the conclusions of previous experiments, the difference between the retrieval results of the purified adversarial samples and the original data is within 2%. A more general conclusion can be drawn that the difference in retrieval performance between the purified dataset and the original dataset is within 2%. Among them, the dataset contains both original data and adversarial samples.

4.3 Ablation Studies

This section explores the impact of different parameter settings on the performance of the purification model **RetPur**. We set the experimental dataset as the WIKI dataset, the attack method is TA-DCH attack and the retrieval model being attacked is DGCPN.

Purify step T_{pur}. As shown in Figure 2, there is a game process between purification performance and visual perceptibility. With the increase of purify step, the performance of the **RetPur** model for adversarial examples increases rapidly.

Distance Metric. The guidance conditions $\nabla x^t D(x^t, x^t_{adv})$ are the gradient value of the distance function between the generated samples and the adversarial samples. Consider the impact of different distance metric $D(\cdot)$ on conditional generation, we compared two different distance functions, SSIM and MSE. As shown in Table 4, Compared to using SSIM, using MSE as a conditional gradient guide, the purified data has slightly higher MAP values and T-MAP values on the hash retrieval model.

5 Conclusion

In this paper, we propose a diffusion-based purification model to defend against target black-box attacks in hash retrieval systems. To assess the effectiveness of **RerPur**, we evaluate our approach in both uni-modal retrieval tasks and cross-modal retrieval tasks, considering two types of attacks: iterative attacks and generative attacks. The experiments demonstrate that the purified adversarial samples and purified original data yield retrieval model results that are close to those of the original data, with significant improvements compared to the results of the attacked retrieval. Furthermore, the **PerPur** model outperforms existing adversarial training method ATRDH (Wang et al. (2021a)). Therefore, the model we propose can be effectively applied in the field of hash retrieval.

REFERENCES

Cong Bai, Chao Zeng, Qing Ma, Jinglin Zhang, and Shengyong Chen. Deep adversarial discrete hashing for cross-modal retrieval. In *Proceedings of the 2020 international conference on multimedia retrieval*, pp. 525–531, 2020a.

Jiawang Bai, Bin Chen, Yiming Li, Dongxian Wu, Weiwei Guo, Shu-tao Xia, and En-hui Yang. Targeted attack for deep hashing based retrieval. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pp. 618–634. Springer, 2020b.

Shumeet Baluja and Ian Fischer. Adversarial transformation networks: Learning to generate adversarial examples. *arXiv preprint arXiv:1703.09387*, 2017.

Jacob Buckman, Aurko Roy, Colin Raffel, and Ian Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. In *International conference on learning representations*, 2018.

Zhangjie Cao, Ziping Sun, Mingsheng Long, Jianmin Wang, and Philip S Yu. Deep priority hashing. In *Proceedings of the 26th ACM international conference on Multimedia*, pp. 1653–1661, 2018.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pp. 39–57. Ieee, 2017.

Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pp. 15–26, 2017.

Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, pp. 1–9, 2009.

Gilad Cohen, Guillermo Sapiro, and Raja Giryes. Detecting adversarial samples using influence functions and nearest neighbors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14453–14462, 2020.

Zhun Deng, Linjun Zhang, Kailas Vodrahalli, Kenji Kawaguchi, and James Y Zou. Adversarial training helps transfer learning via better representations. *Advances in Neural Information Processing Systems*, 34:25179–25191, 2021.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Guneet S Dhillon, Kamyar Azizzadenesheli, Zachary C Lipton, Jeremy Bernstein, Jean Kossaifi, Aran Khanna, and Anima Anandkumar. Stochastic activation pruning for robust adversarial defense. *arXiv preprint arXiv:1803.01442*, 2018.

Yilun Du and Igor Mordatch. Implicit generation and modeling with energy based models. *Advances in Neural Information Processing Systems*, 32, 2019.

Hugo Jair Escalante, Carlos A Hernández, Jesus A Gonzalez, Aurelio López-López, Manuel Montes, Eduardo F Morales, L Enrique Sucar, Luis Villasenor, and Michael Grubinger. The segmented and annotated iapr tc-12 benchmark. *Computer vision and image understanding*, 114 (4):419–428, 2010.

Yan Feng, Bin Chen, Tao Dai, and Shu-Tao Xia. Adversarial attack on deep product quantization network for image retrieval. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 34, pp. 10786–10793, 2020.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014a.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014b.

Will Grathwohl, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky. Your classifier is secretly an energy based model and you should treat it like one. *arXiv preprint arXiv:1912.03263*, 2019.

Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *arXiv preprint arXiv:1711.00117*, 2017.

Jiangfan Han, Xiaoyi Dong, Ruimao Zhang, Dongdong Chen, Weiming Zhang, Nenghai Yu, Ping Luo, and Xiaogang Wang. Once a man: Towards multi-target attack via learning multi-target adversarial network once. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5158–5167, 2019.

Mitch Hill, Jonathan Mitchell, and Song-Chun Zhu. Stochastic security: Adversarial defense using long-run dynamics of energy-based models. *arXiv preprint arXiv:2005.13525*, 2020.

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

Shengshan Hu, Yechao Zhang, Xiaogeng Liu, Leo Yu Zhang, Minghui Li, and Hai Jin. Advhash: Set-to-set targeted attack on deep hashing with one single adversarial patch. In *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 2335–2343, 2021.

Mark J Huiskes and Michael S Lew. The mir flickr retrieval evaluation. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, pp. 39–43, 2008.

Qing-Yuan Jiang and Wu-Jun Li. Deep cross-modal hashing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3232–3240, 2017.

Boah Kim, Yujin Oh, and Jong Chul Ye. Diffusion adversarial representation learning for self-supervised vessel segmentation. *arXiv preprint arXiv:2209.14566*, 2022.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

Chao Li, Shangqian Gao, Cheng Deng, De Xie, and Wei Liu. Cross-modal learning with adversarial samples. *Advances in neural information processing systems*, 32, 2019.

Chao Li, Haoteng Tang, Cheng Deng, Liang Zhan, and Wei Liu. Vulnerability vs. reliability: Disentangled adversarial examples for cross-modal learning. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 421–429, 2020.

Chao Li, Shangqian Gao, Cheng Deng, Wei Liu, and Heng Huang. Adversarial attack on deep cross-modal hamming retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2218–2227, 2021a.

Wu-Jun Li, Sheng Wang, and Wang-Cheng Kang. Feature learning based deep supervised hashing with pairwise labels. *arXiv preprint arXiv:1511.03855*, 2015.

Xiaodan Li, Jinfeng Li, Yuefeng Chen, Shaokai Ye, Yuan He, Shuhui Wang, Hang Su, and Hui Xue. Qair: Practical query-efficient black-box attacks for image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3330–3339, 2021b.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Ninghao Liu, Hongxia Yang, and Xia Hu. Adversarial detection with model interpretation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1803–1811, 2018.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11461–11471, 2022.

Shiqing Ma, Yingqi Liu, Guanhong Tao, Wen-Chuan Lee, and Xiangyu Zhang. Nic: Detecting adversarial samples with neural network invariant checking. In *26th Annual Network And Distributed System Security Symposium (NDSS 2019)*. Internet Soc, 2019.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Christoph Mayer and Radu Timofte. Adversarial sampling for active learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3071–3079, 2020.

Patrick McDaniel, Nicolas Papernot, and Z Berkay Celik. Machine learning in adversarial settings. *IEEE Security & Privacy*, 14(3):68–72, 2016.

Konda Reddy Mopuri, Utkarsh Ojha, Utsav Garg, and R Venkatesh Babu. Nag: Network for adversary generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 742–751, 2018.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pp. 8162–8171. PMLR, 2021.

Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. *arXiv preprint arXiv:2205.07460*, 2022.

Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016.

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pp. 506–519, 2017.

Jose Costa Pereira, Emanuele Coviello, Gabriel Doyle, Nikhil Rasiwasia, Gert RG Lanckriet, Roger Levy, and Nuno Vasconcelos. On the role of correlation and abstraction in cross-modal multimedia retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 36(3):521–535, 2013.

Omid Poursaeed, Isay Katsman, Bicheng Gao, and Serge Belongie. Generative adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4422–4431, 2018.

Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-gan: Protecting classifiers against adversarial attacks using generative models. *arXiv preprint arXiv:1805.06605*, 2018.

Anindya Sarkar, Anirban Sarkar, Sowrya Gali, and Vineeth N Balasubramanian. Adversarial robustness without adversarial training: A teacher-guided curriculum learning approach. *Advances in Neural Information Processing Systems*, 34:12836–12848, 2021.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020a.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. *arXiv preprint arXiv:1710.10766*, 2017.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2020b.

Vignesh Srinivasan, Csaba Rohrer, Arturo Marban, Klaus-Robert Müller, Wojciech Samek, and Shinichi Nakajima. Robustifying models against adversarial attacks by langevin dynamics. *Neural Networks*, 137:1–17, 2021.

Jiachen Sun, Jiongxiao Wang, Weili Nie, Zhiding Yu, Zhuoqing Mao, and Chaowei Xiao. Pointdp: Diffusion-driven purification against 3d adversarial point clouds. 2022.

Jinyi Wang, Zhaoyang Lyu, Dahua Lin, Bo Dai, and Hongfei Fu. Guided diffusion model for adversarial purification. *arXiv preprint arXiv:2205.14969*, 2022.

Tianshi Wang, Lei Zhu, Zheng Zhang, Huaxiang Zhang, and Junwei Han. Targeted adversarial attack against deep cross-modal hashing retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

Xunguang Wang, Zheng Zhang, Guangming Lu, and Yong Xu. Targeted attack and defense for deep hashing. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2298–2302, 2021a.

Xunguang Wang, Zheng Zhang, Baoyuan Wu, Fumin Shen, and Guangming Lu. Prototype-supervised adversarial network for targeted attack of deep hashing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16357–16366, 2021b.

Quanlin Wu, Hang Ye, and Yuntian Gu. Guided diffusion model for adversarial purification from random noise. *arXiv preprint arXiv:2206.10875*, 2022.

Shutong Wu, Jiongxiao Wang, Wei Ping, Weili Nie, and Chaowei Xiao. Defending against adversarial audio via diffusion model. *arXiv preprint arXiv:2303.01507*, 2023.

Weibin Wu, Yuxin Su, Michael R Lyu, and Irwin King. Improving the transferability of adversarial samples with adversarial transformations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9024–9033, 2021.

Chang Xiao, Peilin Zhong, and Changxi Zheng. Enhancing adversarial defense by k-winners-take-all. *arXiv preprint arXiv:1905.10510*, 2019.

Chaowei Xiao, Bo Li, Jun-Yan Zhu, Warren He, Mingyan Liu, and Dawn Song. Generating adversarial examples with adversarial networks. *arXiv preprint arXiv:1801.02610*, 2018.

Yanru Xiao and Cong Wang. You see what i want you to see: Exploring targeted black-box transferability attack for hash-based image retrieval systems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1934–1943, 2021.

Yanru Xiao, Cong Wang, and Xing Gao. Evade deep image retrieval by stashing private images in the hash space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9651–9660, 2020.

Erkun Yang, Tongliang Liu, Cheng Deng, and Dacheng Tao. Adversarial examples for hamming space search. *IEEE transactions on cybernetics*, 50(4):1473–1484, 2018.

Kaiwen Yang, Tianyi Zhou, Yonggang Zhang, Xinmei Tian, and Dacheng Tao. Class-disentanglement and applications in adversarial detection and defense. *Advances in Neural Information Processing Systems*, 34:16051–16063, 2021.

Jongmin Yoon, Sung Ju Hwang, and Juho Lee. Adversarial purification with score-based generative models. In *International Conference on Machine Learning*, pp. 12062–12072. PMLR, 2021.

Jun Yu, Hao Zhou, Yibing Zhan, and Dacheng Tao. Deep graph-neighbor coherence preserving network for unsupervised cross-modal hashing. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 4626–4634, 2021.

Hamed Zamani, Johanne R Trippas, Jeff Dalton, Filip Radlinski, et al. Conversational information seeking. *Foundations and Trends® in Information Retrieval*, 17(3-4):244–456, 2023.

Peng-Fei Zhang, Guangdong Bai, Hongzhi Yin, and Zi Huang. Proactive privacy-preserving learning for cross-modal retrieval. *ACM Transactions on Information Systems*, 41(2):1–23, 2023.

Shuhai Zhang, Feng Liu, Jiahao Yang, Yifan Yang, Bo Han, and Mingkui Tan. Expected perturbation scores for adversarial detection. 2022.

Qingye Zhao, Xin Chen, Zhuoyu Zhao, Enyi Tang, and Xuandong Li. Wassertrain: An adversarial training framework against wasserstein adversarial attacks. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2734–2738. IEEE, 2022.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.

Lei Zhu, Tianshi Wang, Jingjing Li, Zheng Zhang, Jialie Shen, and Xinhua Wang. Efficient query-based black-box attack against cross-modal hashing retrieval. *ACM Transactions on Information Systems*, 41(3):1–25, 2023.

APPENDIX

A EXAMPLE OF TARGET RETRIEVAL ATTACK

As shown in Figure 3, when the aircraft image is input into the retrieval system, the query text and images returned depict a panda. In the case of targeted attacks, inputting images with any semantic content into the retrieval system results in specific categories and corresponding text being returned.

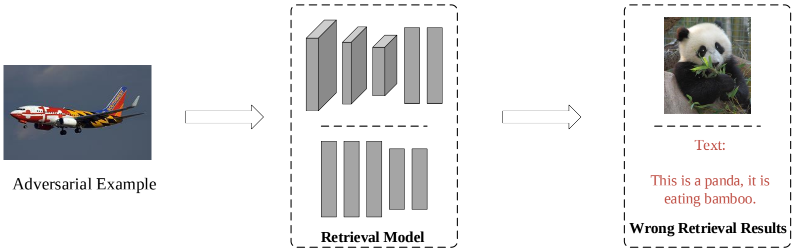

Figure 3: Wrong retrieval results after attack. Inputting aircraft images with adversarial interference into the retrieval system returns both textual and visual results depicting a panda.

B DEFINITION OF PREPOSSESSING METHODS

Essentially, the adversarial purification is one of the prepossessing methods. Preprocessing, in which input images are preprocessed using auxiliary transformations before classification, is another strategy for adversarial defense. Let f_θ be a preprocessor for image transformation. The goal of the training is then:

$$\min_{\phi,\theta} \mathbb{E}_{p_{data}(x,y)}[max_{x' \in B(x)} L(g_\theta(f_\theta(x')), y)]$$

An average over recognized types of adversarial attacks or an average over stochastically altered inputs can be used to approximate the maximum over the threat model $B(x)$. Adding stochasticity to the input images, as well as discontinuous or non-differentiable transforms (Guo et al. (2017);Dhillon et al. (2018);Buckman et al. (2018);Xiao et al. (2019)), makes gradient estimation concerning the loss function $\nabla_x L(g_\theta(f_\theta(x')), y)$ harder to attack.

C PROOF OF CONDITIONAL GRADIENT

Refer to the work (Wang et al. (2022)), we provide a proof of conditional gradient in the formula 6. By using adversarial image x_{adv} as a condition for the reverse denoising process of the DDPM model, we can obtain a distribution $p_\theta(x^{t-1}|x^t, x_{adv})$. It is a conditional distribution of denoising distribution $p_\theta(x^{t-1}|x^t)$ in the formula 5. Dhariwal & Nichol (2021) proved that

$$\log p_\theta(x^{t-1}|x^t, x_{adv}) = \log[p_\theta(x^{t-1}|x^t) p(x_{adv}|x^t)] + C_1 \approx \log p(z) + C_2, \tag{17}$$

where distribution $z \sim N(z; \mu_\theta(x^t, t) + \sigma_t^2 g, \sigma_t^2 I)$, $g = \nabla_{x^t} \log p(x_{adv}|x^t)$, C_1 and C_2 are some constants. Due to the original distribution $p_\theta(x^{t-1}|x^t) \sim N(\mu_\theta(x^t, t), \sigma_t^2 I)$, $\sigma_t^2 g$ is the conditional gradient of the required solution.

$$\sigma_t^2 g = \sigma_t^2 \nabla_{x^t} \log p(x_{adv}|x^t), \tag{18}$$

where σ_t is the standard deviation of original distribution $p_\theta(x^{t-1}|x^t)$ that we already have. Therefore, we only need to solve $\nabla_{x^t} \log p(x_{adv}|x^t)$.

$p(x_{adv}|x^t)$ can be interpreted as the probability that x^t will eventually be denoised to a original image close to x_{adv}. Then, a heuristic formula can be used to approximate the probability distribution:

$$p(x_{adv}|x^t) = \frac{1}{Z} \exp(-sD(x^t, x_{adv}^t)), \tag{19}$$

Table 5: Datasets partitioning for uni-modal retrieval models

Datasets	Total	Training	Query	Database	Class
FLICKR-25K	25000	5000	1700	23300	38
MS-COCO	122218	10000	5000	117218	80
NUS-WIDE	195834	10500	2100	193734	21

where D is some distance metric, Z is a normalization, s is a scale factor that controls the magnitude of the guidance, and x^t_{adv} is obtained by diffusion x_{adv} t steps. When the purified image x^t to be close to the adversarial image x^t_{adv} in the reverse process, $D(x^t, x^t_{adv})$ will be smaller. When $D(x^t, x^t_{adv})$ is smaller, probability $p(x_{adv}|x^t)$ is larger, and the purified x^t is closer to the adversarial sample x_{adv}.

Here we take the logarithm of both sides and then calculate the gradient of them:

$$\log p(x_{adv}|x^t) = -\log Z - sD(x^t, x^t_{adv}), \tag{20}$$

$$\nabla_{x^t} \log p(x_{adv}|x^t) = -s\nabla_{x^t} D(x^t, x^t_{adv}), \tag{21}$$

By combining formula 18 and formula 21, the expression for the conditional gradient can be obtained:

$$\sigma_t^2 g = -s\sigma_t^2 \nabla_{x^t} D(x^t, x^t_{adv}). \tag{22}$$

D DETAILS OF DATASETS SETTINGS

Followed by the studies Wang et al. (2021b) and Wang et al. (2023), we split each dataset into a queryset and database, and part of the database is randomly selected as the training set. The training set is used to optimize the attacked hashing retrieval models and attack methods, while the query set is first used to generate adversarial examples and then evaluate the targeted attack together with the database set. To avoid accidental results caused by using a uniform target label, as in existing works (Wang et al. (2021b), Bai et al. (2020b)), we set different target labels for each benign example, where these target labels are randomly sampled from datasets.

Hash retrieval can be divided into two types: one type is a uni-modal retrieval model, which inputs query images and outputs semantically similar images in the database. Another type is a cross-modal hashing retrieval model, which has four representations of input and output: Image to Image (I2I), Image to Text (I2T), Text to Image (T2I), Text to Text (T2T). We conducted experiments on the attack methods of these two types of retrieval models, and due to the different retrieval tasks, we will discuss the differences in the versions and partitions of the datasets.

1) Uni-modal retrieval task datasets: As shown in Table 5, we adopt the same experimental dataset and partitioning settings as Wang et al. (2021b).

2) Cross-modal retrieval task datasets: As shown in Table 6, we adopt the same experimental dataset and partitioning settings as Wang et al. (2023).

E DETAILS OF ATTACKED METHODS AND ATTACK ATTACKS

1) Uni-modal retrieval task In the uni-modal hash retrieval task, we use three targeted attack methods to generate adversarial samples and purify them to test the improvement of retrieval performance. Among them, P2P and DHTA are iterative attacks (Bai et al. (2020b)), while ProS-GAN (Wang et al. (2021b)) is generative attack. Iterative methods are inefficient because they are optimization-based methods that rely on a very timeconsuming iterative gradient. Unlike the former, generative attacks are an efficient and effective attack in deep hash based retrieval. By using the original image and target label as inputs, it can generate targeted adversarial examples that can mislead attack hash networks to retrieve images related to the semantics of the target label.

Table 6: Datasets partitioning for uni-modal retrieval models

Datasets	Total	Training	Query	Database	Class
Wikipedia	2866	2173	693	2173	10
IAPR-TC12	20000	5000	2000	18000	255
FLICKR-25K	20015	5000	2000	18015	24
MS-COCO	123287	10000	5000	121287	80
NUS-WIDE	195834	10500	2100	193734	21

We trained the ProS-GAN attack method on three datasets using the official repository (https://github.com/xunguangwang/ProS-GAN). This method consists of three parts: a PrototypeNet, a Decoder and a Generator. For the network architecture, we built PrototypeNet with four-layer fully-connected networks ($y_t \to 4096 \to 512 \to r_t \to y_t, h_t$). We set α_1, α_2 and α_3 as 1, 10^{-4} and 1. We adopt a fully-connected layer and four deconvolutional layers for the Decoder D_t to upsample the semantic representation r_t. We adapt the architecture for Generator G_{xt} from Zhu et al. (2017), and the discriminator contains five stride-2 convolutions and last layer with a 7×7 convolution. The weighting factor α are set with 50 for NUS-WIDE and MS-COCO, 100 for FLICKR-25K, and β is set as 1. For the optimization procedure, we use Adam optimizer with initial learning rate 10^{-4}. The training epochs are 100 in batch size 24.

After training ProS-GAN, we will use the PrototypeNet and the Generator to attack the attacked hashing network. we select the objective function of DPSH (Li et al. (2015)) and DPH (Cao et al. (2018)) as default method, which are two representative deep hashing methods, to construct the target hashing model. Specifically, VGG-11 (Simonyan & Zisserman (2014)) is adopted as the default backbone network. We replace the last fully connected layer of VGG-11 with the hashing layer, including a new fully-connected layer and the Tanh activation.

2) Cross-modal retrieval task

In cross-modal retrieval task, the focus is on image to image and image to text conversion. TA-DCH (Wang et al. (2023)) is the only target attack against cross-modal hash retrieval models. We conducted sufficient experiments on five datasets. Firstly, TA-DCH attacked the images in the query set, and then input these images into the purifier to counteract noise. Finally, we tested the retrieval performance on two query systems.

We trained the TA-DCH attack method on five datasets using the official repository (https://github.com/tswang0116/TA-DCH). Similar to the ProS-GAN method, the structure of the TA-DCH attack method also includes a PrototypeNet, a Decoder, and a Generator. The hyper-parameters α, β, μ, ν and ϵ are set to 1, 5, 5, 1 and $*$ respectively, where $* \in [1, 5000]$ is determined according to the given attacked hashing network. For the attacked models that have no specific image feature extractor, we use CNN-F Chatfield et al. (2014) to obtain 4096-dimensional image features. We adopt the Adam optimizer with learning rate 10^{-4} to optimize the TA-DCH method. For the training configuration, the two stages of our method have different settings: In the cross-modal prototype learning stage, the training epoch and batch size are set to 20 and 64, respectively. In the adversarial generation stage, the training epoch and the batch size are set to 100 and 24, respectively.

To verify the effects of the proposed model on attacking different deep cross-modal hashing methods, we select two representative ones (Yu et al. (2021);Bai et al. (2020a)) as the attacked objects, including Deep Adversarial Discrete Hashing (DADH) and Deep Graph-neighbor Coherence Preserving Network (DGCPN). Among them, DADH is supervised method, and DGCPN is unsupervised. Since our attack objects are the hashing models that achieve excellent retrieval performance, we train these attacked models separately to fit each dataset. All the attacked models use the unified data pre-processing and follow the setups of their original papers. We find that DADH could not be properly optimized on MS-COCO due to gradient explosion, so we moderately modify its loss function to make it trainable.

It is worth noting that both ProS-GAN and TA-DCH are generative attack methods, and the structure is composed of three sub networks: PrototypeNet, adversarial generator network, and discriminator network. Since our attack objects are the hashing models that achieve excellent retrieval performance, we train these attacked models separately to fit each dataset. All the attacked models use the unified data pre-processing and follow the setups of their original papers.

F The results of Purify original data

To demonstrate the impact of the purification model on original data, we conducted comparative experiments in both uni-modal and cross-modal hash retrieval systems. As shown in the Table 7, Table 9 and Table 9, by comparing the retrieval results of purified original data and the original data in the retrieval model, we can conclude that the purification model has a minor impact on the original data, and the retrieval results are close to those of the original data retrieval.

Table 7: Purify original datasets of uni-modal retrieval task.

I2I		FLICKR-25K		MS-COCO		NUS-WIDE	
		DPSH	DPH	DPSH	DPH	DPSH	DPH
MAP	original	81.06	77.07	63.69	58.95	73.06	69.92
	RetPur	**80.02**	**76.57**	**62.85**	**58.46**	**72.76**	**69.82**
T-MAP	original	73.67	73.21	50.53	49.22	59.17	60.27
	RetPur	**71.71**	**72.35**	**46.30**	**45.99**	**53.53**	**53.56**

Table 8: Purify original datasets of cross-modal retrieval task for Image to Image (I2I).

I2I		WIKI		IAPR		FLICKR-25K		MS-COCO		NUS-WIDE	
		DGCPN	DADH	DGCPN	DADH	DGCPN	DADH	DGCPN	DADH	DGCPN	DADH
MAP	original	38.09	44.93	64.74	69.35	81.99	80.58	61.92	58.49	77.91	75.60
	RetPur	**36.98**	**44.83**	**63.57**	**67.10**	**80.71**	**82.99**	**60.20**	**56.24**	**76.82**	**74.66**
T-MAP	original	13.33	12.01	31.17	33.43	56.71	57.07	34.27	36.91	35.45	35.92
	RetPur	**14.24**	**13.30**	**31.18**	**31.09**	**56.10**	**54.94**	**35.86**	**35.68**	**35.91**	**34.93**

Table 9: Purify original datasets of cross-modal retrieval task for Image to Text (I2T).

I2T		WIKI		IAPR		FLICKR-25K		MS-COCO		NUS-WIDE	
		DGCPN	DADH	DGCPN	DADH	DGCPN	DADH	DGCPN	DADH	DGCPN	DADH
MAP	original	37.79	43.15	65.24	72.49	80.56	78.45	65.21	62.28	76.42	77.65
	RetPur	**36.73**	**42.93**	**64.17**	**71.03**	**79.33**	**85.30**	**63.25**	**59.94**	**75.32**	**76.61**
T-MAP	original	13.34	12.11	31.37	37.60	56.54	62.39	34.25	38.67	35.64	39.23
	RetPur	**14.18**	**13.37**	**31.33**	**35.81**	**56.34**	**60.43**	**36.63**	**37.06**	**35.84**	**38.28**