Rethinking the Design of Backdoor Triggers and Adversarial Perturbations: A Color Space Perspective

Wenbo Jiang, *Member, IEEE,* Hongwei Li (Corresponding author), *Fellow, IEEE,* Guowen Xu, *Member, IEEE,* Hao Ren, *Member, IEEE,* Haomiao Yang, *Senior Member, IEEE,* Tianwei Zhang, *Member, IEEE,* and Shui Yu, *Fellow, IEEE,*

Abstract—Deep neural networks (DNNs) are known to be susceptible to various malicious attacks, such as adversarial and backdoor attacks. However, most of these attacks utilize additive adversarial perturbations (or backdoor triggers) within an L_p-norm constraint. They can be easily defeated by image preprocessing strategies, such as image compression and image super-resolution. To address this limitation, instead of using additive adversarial perturbations (or backdoor triggers) in the pixel space, this work revisits the design of adversarial perturbations (or backdoor triggers) from the perspective of color space and conducts a comprehensive analysis. Specifically, we propose a color space backdoor attack and a color space adversarial attack where the color space shift is used as the trigger and perturbation. To find the optimal trigger or perturbation in the black-box scenario, we perform an iterative optimization process with the Particle Swarm Optimization algorithm. Experimental results confirm the robustness of the proposed color space attacks against image preprocessing defenses as well as other mainstream defense methods. In addition, we also design adaptive defense strategies and evaluate their effectiveness against color space attacks. Our work emphasizes the importance of the color space when developing malicious attacks against DNN and urges more research in this area.

Index Terms—Backdoor attack, Adversarial attack, Defense mechanisms, Image color space.

1 Introduction

Deep neural networks (DNNs) have gained widespread utilization across diverse domains, such as image classification [2], biometric authentication [3] and natural language processing [4]. However, recent research found that DNNs are vulnerable to various types of malicious attacks, where backdoor attacks and adversarial attacks are two of the most representative and received significant academic attention. In backdoor attacks, the attacker embeds a backdoor into the model by poisoning the training dataset or controlling the training process. As a result, the backdoor model will perform normally on benign samples but behave incorrectly on samples that contain a specific backdoor trigger. On the other hand, the adversary of adversarial attacks adds hardly perceptible adversarial perturbations to a benign sample to generate an adversarial sample, which can induce a misclassification of a normal DNN. These attacks have been validated on various applications (e.g. face authentication [5], malware detection [6] and autonomous driving [7]) and lead to disastrous consequences.

One important criterion for backdoor (or adversarial) attacks is that the generated backdoor-triggered (or adversarial) image should be similar to the original image. Specifically, there are mainly two strategies for constructing backdoor triggers: pixel-restricted triggers restrict the pixel distances [8], [9], [10] or enforce latent representation consistency [11], [12] between benign and triggered images to achieve stealthiness; natural triggers leverage particular image styles such as the natural reflection phenomenon [13], specific Instagram filters [14], or specific weather conditions [15] to activate the backdoor in the model. The design of adversarial perturbations can also be divided into two categories: a majority of studies have concentrated on restricting the adversarial perturbations by controlling an L_p-norm[1]; some works have focused on semantic adversarial attacks, where the semantics of the original image are not changed. Unfortunately, these backdoor and adversarial attacks are

An earlier conference version of this paper appeared at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2023) [1]. In this journal version, we rethink the design of backdoor triggers and adversarial perturbations from the perspective of color space. We propose a color space backdoor attack and a color space adversarial attack, where the color space shift is used as the trigger and perturbation. We also design several adaptive defense strategies against color space attacks, such as color depth reduction and image grayscaling and colorization. We further conduct comprehensive experiments to evaluate the effectiveness of the proposed attacks and defenses.

- *W. Jiang, H. Li, H. Yang and G. Xu are with the School of Computer Science and Engineering, University of Electronic Science and Technology of China, China (e-mail: wenbo_jiang, hongweili, haomyang@uestc.edu.cn, guowen.xu@foxmail.com).*
- *H. Ren is with the School of Cyber Science and Engineering, Sichuan University, China (e-mail: hao.ren@scu.edu.cn).*
- *T. Zhang are with the School of Computer Science and Engineering, Nanyang Technological University, Singapore (e-mail: tianwei.zhang@ntu.edu.sg).*
- *S. Yu is a Professor of the School of Computer Science in the Faculty of Engineering and Information Technology at University of Technology Sydney, Sydney, Australia (e-mail: Shui.Yu@uts.edu.au).*

1. L_∞-norm restrains the maximum change for each pixel, L_0-norm restrains the maximum number of perturbed pixels, L_2-norm restrains the maximum Euclidean distance.

either vulnerable to image preprocessing strategies (such as image compression [16] and image super-resolution [17]) or suffer from some limitations (such as restricted application scenarios or unnatural malicious images[2]).

To address these limitations, in this work, we explore malicious attacks against DNNs in the color space. Our inspiration draws from the shape bias property found in the human cognitive system [18], where humans prefer to classify objects primarily based on their shapes rather than their colors. In contrast, neural networks learn a wider range of information from images when performing classification tasks. Based on this observation, we construct backdoor triggers and adversarial perturbations in the color space to attack DNNs. Concretely, we apply a color space shift uniformly to all pixels of the original image to the generate backdoor-triggered image or adversarial image. As depicted in Figure 1, color space backdoor attack (CSBA) implants the backdoor into the victim model. It can be triggered by an image-agnostic color shift on any inference image; on the other hand, color space adversarial attack (CSAA) applies an image-specific adversarial color shift to a given inference image and can induce a misclassification of a normal DNN. The backdoor-triggered image and adversarial image appear semantically similar to the original image, which enables them to bypass the detection of human eyes. More importantly, since the color space backdoor trigger and adversarial perturbation are generated through a rather different principle, they are less affected by image preprocessing operations and can bypass other mainstream defenses as well.

Nevertheless, finding an appropriate color space backdoor trigger or adversarial perturbation is challenging: On one hand, employing a large color space shift tends to make the backdoor-triggered or adversarial samples less realistic (as illustrated in Figure 10 and 11). On the other hand, a small color space shift is difficult to be recognized by the model, leading to reduced effectiveness and robustness. Additionally, it is also challenging to optimize the backdoor trigger or adversarial perturbation in the practical black-box setting. To tackle these problems, we employ Particle Swarm Optimization (PSO) [19], a powerful gradient-free optimization algorithm, to systematically search for the optimal backdoor trigger and adversarial perturbation. Our methodology mainly includes the following three steps: **(1) Measurement of effectiveness:** We utilize the backdoor loss of a semi-trained model (with a surrogate model architecture) to efficiently measure the effectiveness of the backdoor trigger; we use the probability of the adversarial sample being classified correctly to measure the effectiveness of the adversarial perturbation. **(2) Constraints of naturalness:** We employ three state-of-the-art (SOTA) similarity metrics, PSNR (Peak Signal-to-Noise Ratio) [20], SSIM (Structural Similarity Index) [21], and LPIPS (Learned Perceptual Image Patch Similarity) [20] to quantify the naturalness of the backdoor-triggered and adversarial images. **(3) Iterative optimization:** The PSO algorithm facilitates the iterative searching process to obtain the optimal backdoor trigger and adversarial perturbation.

In summary, our work explores the design of adversarial perturbations (and backdoor triggers) in the color space and proposes CSBA and CSAA based on the PSO algorithm. To defend the DNNs from such color space attacks, we also design several adaptive defense strategies (such as color depth reduction and image grayscaling and colorization) and evaluate their effectiveness. This work provides a new perspective in the design of backdoor triggers and adversarial perturbations, and opens new directions for developing more effective defense mechanisms against such attacks.

The contributions of this work can be elaborated in three aspects:

- We propose a color space backdoor attack CSBA and a color space adversarial attack CSAA in the black-box scenario. The PSO algorithm is employed to perform an iterative optimization process for the optimal backdoor trigger and adversarial perturbation.
- We perform extensive experiments to show the superior performance of PSO over other optimization algorithms. Furthermore, experiments also demonstrate that CSBA and CSAA are robust against SOTA image preprocessing defenses, such as DeepSweep [22], Image compression [16], Image super-resolution [17], etc. Besides, our results also show that they can evade other mainstream defenses, such as Local Intrinsic Dimension (LID) [23], adversarial training [24], Fine-Pruning [25], Neural Cleanse [26], etc.
- We develop several adaptive defense strategies against color space attacks, including color space data augmentation, random color space shift, color depth reduction, image grayscaling and colorization. Extensive experiments are conducted to evaluate the robustness of the proposed color space attacks against these adaptive defenses.

The remainder of this paper is structured as follows: the background knowledge of this work is presented in Section 2. Section 3 provides the details of our color space attacks. The proposed adaptive defenses are presented in 4. Experimental evaluations on color space backdoor attack and adversarial attack are shown in Section 6 and 5, respectively. Experimental evaluations on the proposed adaptive defenses are presented in 8. Finally, Section 9 draws a conclusion of this paper.

2 Background

2.1 Backdoor Attacks and Defenses

2.1.1 Backdoor Attacks

According to the design of the trigger, backdoor attacks can be mainly divided into two groups:

Pixel-restricted trigger: Several studies employ pixel-restricted perturbations as backdoor triggers [8], [9], [10], where they restrict the pixel differences between the original and triggered images through an L_p-norm. For instance, Li *et al.* [9] proposed to add backdoor triggers using the technique of image steganography and set the restriction with L_0-norm and L_2-norm. Besides, some works [11], [12], [27] emphasize the importance of maintaining consistency in the latent representation of the benign image and the triggered image. These attacks manipulate the training loss function to achieve this consistency.

2. Refer to Section 2 for more details.

This article has been accepted for publication in IEEE Transactions on Dependable and Secure Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2024.3521942

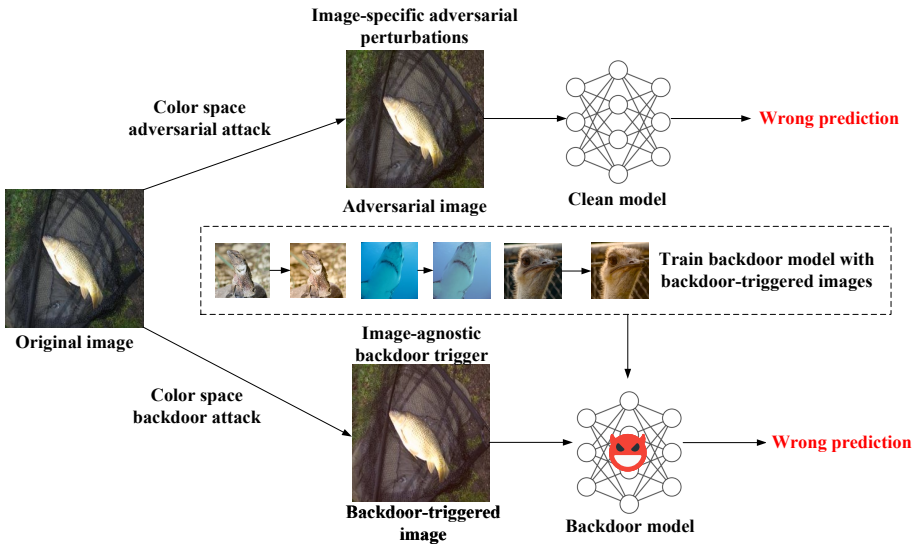


Fig. 1: The attack scenario of color space attacks.

Natural trigger: Another line of research proposes to employ specific image styles as backdoor triggers, where the triggered images remain realistic and natural-looking. These natural triggers can be crafted by leveraging various techniques, including specific Instagram filters [14], specific weather conditions [15] and the transformation of image warping [28].

However, a majority of the above works always emphasize achieving attack stealthiness while overlooking the requirement of attack robustness. The backdoor triggers are vulnerable to image preprocessing strategies and result in low attack effectiveness.

Although some research efforts have been made to improve the robustness of backdoor attacks, they still suffer from some limitations. For example, Li *et al.* [29] and Zhang *et al.* [30] proposed applying data augmentation to backdoor-triggered training samples to enhance the attack robustness under image preprocessing defenses. Nevertheless, these approaches are only effective for the considered image preprocessing methods and they need higher poisoning rates[3]. Additionally, to enhance the robustness of backdoor-triggered samples against image compression, Xu *et al.* [16] proposed maintaining consistency in the latent representation of triggered samples and their compressed versions. However, it assumes the adversary is capable of controlling the training process of the victim model, which is not applicable to more practical black-box scenarios.

2.1.2 Backdoor Defenses

Model reconstruction strategies attempt to defend against backdoor attacks through reconstructing or fine-tuning the backdoor model. For example, Liu *et al.* [25] observed that backdoor neurons always remain unactivated for benign inference samples, so they pruned neurons of the network based on their average activation values; Zhao *et al.* [31] utilized the model connectivity technique [32] to reconstruct the model and eliminate the backdoor; Li *et al.* [33] and Yoshida *et al.* [34] used the model distillation technique [35] to distill infected models and remove backdoors in them.

Trigger reverse-engineering strategies focus on reverse-engineering the potential backdoor trigger and then mitigating the effectiveness of the trigger. Neural Cleanse [26] is one of the most representative methods of such defense strategies. It tries to reconstruct the potential backdoor trigger pattern for each class, and identifies a model as a backdoor model if one of the backdoor trigger patterns is particularly smaller than patterns of other classes.

Inference-time detection strategies distinguish whether an inference image contains a backdoor trigger or not during the inference process. For instance, STRIP [36] superimposes some benign images on the target image individually and sends them for predictions. Based on the observation that the backdoor trigger is robust and still able to activate the backdoor when superimposing with a benign image, the prediction results are low entropy if the target image is backdoor-triggered, the prediction results are high entropy if the target image is benign. Additionally, techniques like heatmap analysis [37] can be employed to detect potential trigger regions in inference images.

Inference-time image preprocessing strategies apply an image preprocessing step before model prediction, intending to disrupt the backdoor trigger in inference images and prevent backdoor activation. For example, Li *et al.* [29] used image transformation methods such as flipping and padding after shrinking to destroy the backdoor trigger in inference images; Qiu *et al.* [22] employed a wide range of data augmentation techniques to fine-tune the backdoor model and preprocess the input images. In addition, our experiments demonstrate that image compression [16] is also effective in destroying backdoor triggers and reducing the attack effectiveness.

3. The poisoning rates in [29] and [30] is 25% and 10%, respectively. But the poisoning rate of our CSBA is no more than 5%.

2.2 Adversarial Attacks and Defenses

2.2.1 Adversarial Attacks

According to the design of the adversarial perturbations, adversarial attacks can be classified into two categories:

Pixel-restricted perturbations: Most adversarial attacks fall into this type, where they restrict the adversarial perturbations through an L_p-norm[1]. Furthermore, they can be categorized into white-box and black-box according to attack scenarios. For instance, the Fast Gradient Sign Method (FGSM) [38] and the Projected Gradient Descent (PGD) [39] are two of the most representative white-box attacks, which employ the model gradient to generate adversarial samples in one step (or iterative steps); NES Attack [40] and NAttack [41] are two black-box adversarial attacks, which utilize the evolutionary algorithm to construct adversarial samples.

Semantic perturbations: Different from pixel-restricted adversarial perturbations, the adversary may also generate the adversarial image by performing image transformations on the benign image under the condition that the semantics of the image are kept unchanged. For example, Spatially Transform Attack [42] generates adversarial samples through spatial transformation. Besides, there are also adversarial attacks that employ the color phenomena [18], [43], [44], [45].

However, similar to backdoor attacks mentioned in Section 2.1.1, these adversarial attacks are either vulnerable to image preprocessing defenses or suffer from some limitations. For example, [18] does not restrict the naturalness of the adversarial images, resulting in unnatural and unrealistic adversarial images; [43] and [44] only apply to white-box scenarios; Besides, our color space adversarial attack is superior to the attack proposed in [45] that uses the Genetic Algorithm (GA) [46] to implement black-box adversarial attacks in color space.

2.2.2 Adversarial Defenses

Detection-based strategies are designed to identify adversarial samples during the inference time. For instance, Ma *et al.* [47] noticed a significant difference in the Local Intrinsic Dimension (LID) [23] between adversarial samples and benign samples, based on which they proposed a detection method. Ma *et al.* [48] observed that adversarial attacks can cause changes in the provenance channel and activation value distribution channel. Hence, they proposed a method to extract DNN invariants and used them to discriminate adversarial samples.

Image preprocessing strategies transform the inference image to disrupt adversarial perturbations in the image. For example, Aydemir *et al.* [49] and Dziugaite *et al.* [50] explored the use of image compression to diminish the effectiveness of adversarial attacks; Mustafa *et al.* [17] employed image super-resolution network to preprocess the inference images to erase adversarial perturbations.

Adversarial training is a training-time defense technique that augments training dataset with some adversarial samples. It improves the model generalization and robustness for adversarial samples in the inference time.

2.3 Particle Swarm Optimization (PSO)

PSO [19] is a gradient-free optimization algorithm, which has been commonly applied in hyperparameter selection of deep learning models [51]. Specifically, in the context of this work, each individual in the PSO algorithm (also referred to as a particle) is defined as a candidate backdoor trigger for `CSBA` or a candidate adversarial perturbation for `CSAA`. The PSO algorithm searches for the optimal backdoor trigger or adversarial perturbation through an iterative updating process. It can be outlined in five steps:

(1) A swarm of particles is randomly initialized, including the positions p_i and velocities v_i:

$$p_i = (p_{i,1}, p_{i,2}, ..., p_{i,D}), \quad v_i = (v_{i,1}, v_{i,2}, ..., v_{i,D}), \quad (1)$$

where M represents the number of particles in the swarm and D denotes the dimension of each particle.

(2) Based on the optimization problem, the objective function $O(p_i)$ is defined to measure the quality of p_i.

(3) For each particle, if its current objective function value is better than its best one in its history, the current position is recorded as the best position of this particle (referred to as $pbest_i$); furthermore, if its current objective function value is better than the best one of the entire swarm, the current position is recorded as the best position of the swarm (referred to as $gbest$).

(4) In each iteration, the v_i and p_i are updated according to Equation (2):

$$\begin{aligned} v_i &= \omega v_i + c_1 r_1 (pbest_i - p_i) + c_2 r_2 (gbest - p_i), \\ p_i &= p_i + v_i, \end{aligned} \quad (2)$$

where r_1 and r_2 are two random numbers in $(0, 1)$, c_1 and c_2 are the acceleration factors, and ω is the inertia weight.

(5) Steps (3)-(4) keep repeating until reaching the maximum number of iteration rounds. Finally, $gbest$ is returned as the optimal solution.

3 Color Space Attacks against DNN

3.1 Threat Model

In terms of `CSBA`, the adversary is assumed to be a malicious training dataset provider. It inserts some backdoor-triggered samples (labeled with the backdoor target class) into the training dataset and releases the poisoned dataset for public download. A developer may access this dataset and use it to train a model. Consequently, a backdoor is stealthily implanted into the model.

In terms of `CSAA`, the adversary is assumed to have no knowledge of the target model, but it has the capability of querying the target model and getting the predicted classification probabilities for these queries. The adversary can construct and update adversarial samples by iteratively querying the model.

3.2 Overview

Previous research [18] confirmed that people prefer to recognize objects based on their shapes, while paying less attention to other structural information (e.g., size, color). However, it is noteworthy that DNNs, when engaged in image classification tasks, can acquire a comprehensive understanding of various structural aspects of images, such as color information. Based on this observation, we present

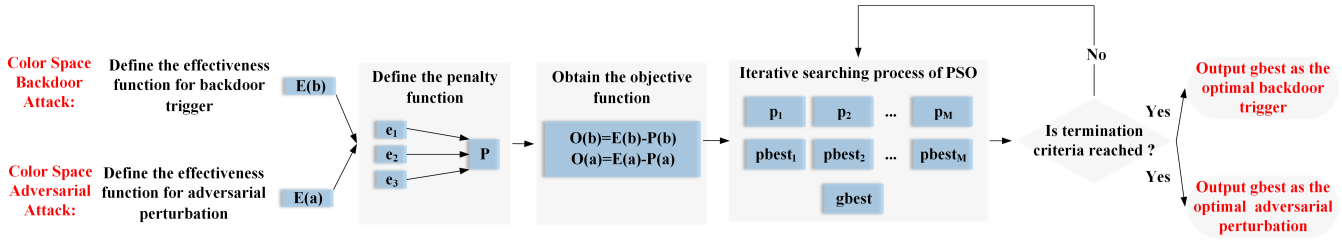


Fig. 2: The workflow of the proposed color space attacks.

Fig. 3: LIME of color space backdoor attack: the left image refers to the benign image and the right image refers to the backdoor-triggered image.

a new perspective on designing backdoor attacks and adversarial attacks, i.e., backdoor triggers and adversarial perturbations can be embedded into the image color space to achieve the objectives of the attack.

The utilization of Local Interpretable Model-Agnostic Explanations (LIME) [52] further demonstrates this point of view. As illustrated in Figure 3, LIME visualizes the specific regions within the image that contribute to the output predictions of our color backdoor model. We can observe distinct behaviors of the backdoor model when confronted with benign samples versus backdoor-triggered samples: the attention of the model is focused on the object itself when processing benign samples, whereas it extends to encompass the entire image when processing backdoor-triggered samples.

In this work, we employ a uniform[4] color space shift (which will be applied consistently to all pixels) to serve as the backdoor trigger or adversarial perturbation. As presented in Equation (3) and (4), img_i represents one pixel of the image, b represents the backdoor trigger of CSBA, a represents the adversarial perturbation of CSAA. As formulated in Equation (6) and (5), for CSAA, each pixel of the image undergoes the transformation of a to generate the adversarial image; for CSBA, each pixel of the image undergoes the transformation of b to generate the backdoor-triggered image. The dimension is set to three, which refers to the three components of the color space. For example, in the RGB color space, these components are Red, Green, and Blue, whereas in the HSV color space, they are Hue, Saturation, and Value.

$$img_i = (img_{i,r}, img_{i,g}, img_{i,b}). \tag{3}$$

$$a = (a_r, a_g, a_b),\ b = (b_r, b_g, b_b). \tag{4}$$

$$\text{Adversarial: } (img_{i,r} + a_r, img_{i,g} + a_g, img_{i,b} + a_b). \tag{5}$$

4. The reason for the uniform shift is to ensure the naturalness of the backdoor-triggered images and adversarial images.

$$\text{Backdoor: } (img_{i,r} + b_r, img_{i,g} + b_g, img_{i,b} + b_b). \tag{6}$$

However, as described in Section 1, determining appropriate color space shift values to serve as the adversarial perturbation and backdoor trigger is non-trivial. In this work, we utilize the PSO algorithm [19] to perform an iterative optimization process for the optimal a and b. As depicted in Figure 2, our methodology can be summarized in the following steps:

(1) We define effectiveness functions for CSAA and CSBA, which measure the superiority or inferiority of a given backdoor trigger and adversarial perturbation.
(2) We define naturalness restrictions for the backdoor trigger and adversarial perturbation using three SOTA similarity metrics (i.e., PSNR [20], SSIM [21] and LPIPS [20]). Furthermore, we design the corresponding penalty term according to these restrictions.
(3) We combine the effectiveness function with the penalty term and formulate the final objective function for PSO.
(4) We perform an iterative search process of PSO and obtain the optimal a and b.

The details of each step are presented below.

3.3 Definition of the Effectiveness Function

3.3.1 Effectiveness Function for Color Space Adversarial Perturbation

In terms of the color space adversarial attacks, the adversarial perturbation is image-specific. Given an image x, we use $x \oplus a$ to denote the generated adversarial image. We use the probability that $x \oplus a$ is classified correctly to quantify the effectiveness of adversarial perturbations:

$$E(a) = -Pro(a) = \frac{-\exp\left(z_y(x \oplus a)\right)}{\sum_{i=1}^{K} \exp\left(z_i(x \oplus a)\right)}, \tag{7}$$

where $z_i(x \oplus a)$ represents the logit value (i.e., the output vector of the model) on the i-th category, y is the ground-true label of x, and K is the total number of categories.

In order to be consistent with the description of the effectiveness function (a larger E indicates a more effective attack), we use $-Pro(a)$ to serve as $E(a)$.

3.3.2 Effectiveness Function for Color Space Backdoor Trigger

In terms of the color space backdoor attacks, the backdoor trigger b is image-agnostic. The most intuitive method to quantify the effectiveness of the backdoor trigger is to train a backdoor model with the backdoor-triggered training

data and evaluate the attack success rate on the backdoor-triggered testing data. Nevertheless, the process of training a backdoor model from scratch is time-intensive.

Inspired by the model performance estimation techniques employed in Neural Architecture Search (NAS) [53], where the early-stage training results on a sub-dataset can estimate the final performance of the model [54], we adopt a similar strategy in our color space backdoor attacks.

Concretely, given a specific color backdoor trigger b, we construct the corresponding poisoned dataset D_p and train a surrogate backdoor model f_{sur} with D_p for a small number of epochs. The training loss of the backdoor-triggered samples (denoted as backdoor loss $\mathcal{L}_b$) is employed to measure the effectiveness function for CSBA:

$$E(b) = -\mathcal{L}_b = -\sum_{x \in D_p} \texttt{CE}(f_{sur}(x \oplus b), y'), \tag{8}$$

where CE stands for the cross-entropy loss function, $x \oplus b$ represents the backdoor-triggered image and y' denotes the attack target class of CSBA.

This approach offers a more rapid and resource-efficient way to quantify the effectiveness of backdoor triggers. A smaller value of $\mathcal{L}_b$ indicates that the surrogate model has effectively learned the trigger feature b, leading to higher attack effectiveness. In order to be consistent with the description of the effectiveness function (a larger E indicates higher attack effectiveness), we also use $-\mathcal{L}_b$ to serve as $E(b)$.

3.4 Definition of the Naturalness Restriction

While a large random color space shift could yield better attack effectiveness for both CSAA and CSBA, it might compromise the realism of the adversarial and backdoor-triggered images (see Figure 10 and 11). In this work, we utilize three SOTA similarity metrics (i.e., PSNR, SSIM and LPIPS) to measure the similarity between the original image and the adversarial (or backdoor-triggered) image.

Concretely, we define three similarity thresholds (i.e., $\lambda_{1,2,3}$) to restrict the naturalness. After that, we formulate the corresponding penalty terms based on these restrictions:

$$\begin{aligned} e_1(a) &= \max(0, \lambda_1 - \texttt{PSNR}(x, x \oplus a)), \\ e_2(a) &= \max(0, \lambda_2 - \texttt{SSIM}(x, x \oplus a)), \\ e_3(a) &= \max(0, \texttt{LPIPS}(x, x \oplus a) - \lambda_3), \end{aligned} \tag{9}$$

where $\texttt{PSNR}(x, x \oplus a)$, $\texttt{SSIM}(x, x \oplus a)$ and $\texttt{LPIPS}(x, x \oplus a)$ denote the similarity between benign sample and adversarial sample. The penalty term represents the extent to which the restriction is exceeded, it equals to 0 when the generated adversarial image is within the restriction.

Similarly, the penalty term of CSBA can be defined in the same way.

3.5 Definition of the Objective Function

To balance the measurement difference of these similarity metrics, we implement a normalization for the penalty terms and calculate the total penalty term $P(a)$ as follows:

$$P(a) = \sum_{j=1}^{3} w_j e_j, \quad w_j = \frac{\sum_{i=1}^{M} e_j(a_i)}{\sum_{j=1}^{3}\sum_{i=1}^{M} e_j(a_i)}, \tag{10}$$

where M represents the number of particles (i.e., candidate adversarial perturbations) in the swarm. Finally, the objective function of adversarial perturbations can be formulated as:

$$O(a) = E(a) - P(a). \tag{11}$$

In addition, based on the defined naturalness restrictions, we introduce an additional criterion to measure the quality of the particles. The criterion is defined as follows:

- In cases where both adversarial perturbations a_i and a_j adhere to the naturalness restrictions, their respective objective function values $O(a_i)$ and $O(a_j)$ are compared and the perturbation with the greater objective function value is considered superior.
- In cases where both adversarial perturbations a_i and a_j violate the naturalness restriction, their respective penalty terms $P(a_i)$ and $P(a_j)$ are compared and the perturbation with the less penalty term is considered superior.
- In cases where perturbation a_i adhere to the naturalness restriction while perturbation a_j does not, a_i is considered superior.

The additional criterion and objective function of CSBA can be defined similarly.

3.6 The Iterative Search Process of PSO

Algorithm 1 The initializing phase of PSO

Input: number of particles in the swarm M
for $i = 1$ to M **do**
  Initialize position p_i and velocity v_i of the particle
  Compute the objective function $O(p_i)$
  Initialize $pbest_i$: $pbest_i \leftarrow p_i$
end for
Initialize $gbest$: $gbest \leftarrow \arg\max_{p_i} O(p_i)$

Algorithm 2 The searching phase of PSO

Input: acceleration factors c_1, c_2; random numbers r_1, r_2; inertia weight ω; number of iteration T; number of particles in the swarm M
Output: the optimal adversarial perturbation or backdoor trigger
for $t = 1$ to T **do**
  for each particle $i = 1$ to M **do**
    $v_i \leftarrow \omega v_i + c_1 r_1 (pbest_i - p_i) + c_2 r_2 (gbest - p_i)$
    $p_i \leftarrow p_i + v_i$
    Compute the objective function $O(p_i)$
    $pbest_i \leftarrow p_i$, if p_i is superior to $pbest_i$ according to the defined rule
    $gbest \leftarrow p_i$, if p_i is superior to $gbest$ according to the defined rule
  end for
end for
return $gbest$

After defining the objective function for PSO, we can perform the process of PSO to search for the optimal backdoor trigger and adversarial perturbation. The iterative search process of PSO comprises two phases.

The initialization phase, as detailed in Algorithm 1, begins by randomly initializing a swarm of particles, including their positions and velocities. Specifically, p_i denotes a

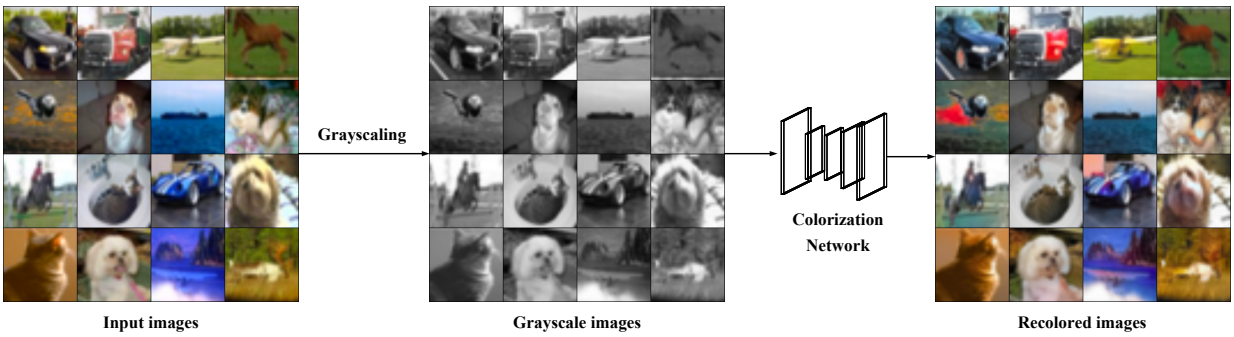


Fig. 4: The workflow of Image Grayscaling and Colorization.

specific color space shift, serving as a candidate backdoor trigger or adversarial perturbation. The initialization phase also initializes the $pbest_i$ (the best position experienced by the i-th particle) and $gbest$ (the best position experienced by the entire group) according to the defined objective function.

After that, the searching phase, as detailed in Algorithm 2, conducts iterative updates of the particles over T rounds. The final output of $gbest$ in the searching phase is selected as the optimal backdoor trigger or adversarial perturbation.

4 Adaptive Defenses Strategies against Color Space Attacks

To comprehensively evaluate the effectiveness and robustness of our color space attacks, we design various adaptive defense strategies against color space attacks including color space data augmentation, random color space shift, color depth reduction and image grayscaling and colorization.

4.1 Image Grayscaling and Colorization

Since the backdoor-triggered images of CSBA and adversarial images of CSAA are generated through color space shifts, we design a defense strategy that involves converting all inference images to grayscale and subsequently re-colorizing them using a pre-trained colorization network. The workflow of this defense is shown in Figure 4. The operation of image grayscaling aims at destroying the potential color space backdoor trigger or adversarial perturbation in the inference images. The operation of image re-colorizing is targeted at restoring the original color of the images and maintaining the accuracy for benign images.

In this work, the SOTA image colorization network DDColor [55] is adopted to re-colorize the grayscale images. Concretely, DDColor first leverages the backbone network of the UNet as an image encoder that extracts semantic feature information from grayscale images. After that, it utilizes an image decoder to perform feature map up-sampling. Meanwhile, DDColor incorporates a color decoder that utilizes an adaptive color query operation on image features to get semantic-aware color embeddings. Finally, the grayscale image and the obtained color features are fused by a feature fusion module to produce a colorful image.

4.2 Other Alternative Adaptive Defense Strategies

In addition to image grayscaling and colorization, we also consider other alternative adaptive defense strategies as follows:

- **Color space data augmentation** is an adaptive defense strategy that is applied during the training process. Specifically, the defender may add some color-shifted images to the training dataset to make the model less sensitive to color space shifts, thus achieving the effect of resisting color space attacks. For our color space adversarial attacks, this method is similar to the technique of adversarial training.
- **Random color space shift** is an image preprocessing operation that is applied during the testing process. Concretely, the defender may perform a random color space shift over each inference sample before sending it to the infected model, expecting that this operation will destroy the backdoor trigger or adversarial perturbation in the inference sample.
- **Color depth reduction** is also an image preprocessing strategy that is performed during the testing time. This approach compresses an image by reducing the bit depth occupied by one pixel (or subpixel). Similarly to random color space shift, the defender attempts to destroy the backdoor trigger or adversarial perturbation in the inference sample through this image preprocessing operation.

The experimental evaluations of these adaptive defenses are presented in Section 8.

5 Experimental Evaluations on CSAA

5.1 Experimental Setup for CSAA

5.1.1 Datasets and Model Architectures

CSAA can be applied to different DNN architectures and datasets. Without loss of generality, we perform our evaluations over the datasets and model architectures provided in Table 1.

TABLE 1: Datasets and model architectures.

Dataset	Image size	Number of classes	Corresponding model architecture
CIFAR-10	32×32	10	ResNet-18
CIFAR-100	32×32	100	VGG-16
GTSRB	32×32	43	ResNet-34
ImageNet	224×224	1000	ResNet-34

5.1.2 Attack Configuration

The hyperparameter settings of PSO are provided in Table 2. Besides, we perform CSBA on six widely utilized color spaces: RGB, HSV, LAB, YCbCr, XYZ, and LUV.

TABLE 2: Hyperparameter setting.

Symbol	Description	Value
M	the number of the particles	200
T	the maximum iteration	20
ω	the inertia weight	0.1
$c_{1,2}$	acceleration factors	2
λ_1	the similarity threshold for PSNR	20
λ_2	the similarity threshold for SSIM	0.9
λ_3	the similarity threshold for LPIPS	0.02

TABLE 3: The ASR of CSAA in different color spaces.

Color space \ Dataset	CIFAR-10	CIFAR-100	GTSRB	ImageNet
RGB	97.64	98.09	99.57	96.51
HSV	93.29	94.98	97.88	93.07
LAB	97.24	96.70	97.22	94.10
YCbCr	98.14	96.68	97.02	94.31
XYZ	98.02	96.01	98.65	93.78
LUV	98.35	97.60	97.41	95.25

5.1.3 Evaluation Metrics

- Clean accuracy of the victim model ($\mathrm{ACC_v}$): This metric denotes the test accuracy of the victim model on clean samples. It measures the normal-functionality of the victim model.
- Attack success rate for CSAA (ASR): This metric represents the ratio of generated adversarial samples that are misclassified to other classes by the victim model. It measures the effectiveness of the attack.

5.2 Effectiveness Evaluation for CSAA

5.2.1 Effectiveness in all Considered Color Spaces

We implement CSAA in all considered color spaces and present the results in Table 3. The results confirm that CSAA achieves high attack success rates across various color spaces[5].

5.2.2 Performance of the PSO Algorithm

To evaluate the effectiveness of the PSO algorithm in CSAA, we compare the performance of CSAA with the performance of the attack proposed by [45]. The adversarial attack proposed in [45] employs the genetic algorithm (GA) [46] to generate color space adversarial samples in the black-box scenario. Additionally, the random-selection method is also used as a baseline for evaluation.

Specifically, we evaluate both the attack success rates (ASRs) and the corresponding computational overhead for these attack methods. The results in Table 4 and 5 indicate that although the computational overhead of the random-selection method is negligible, it can not produce effective attacks. In comparison, CSAA (with PSO) achieves the highest ASR and has lower computation overhead than [45] (with GA). It can be concluded that the PSO algorithm is superior to other optimization algorithms in performing CSAA.

5. To avoid redundancy, we choose the CSAA in RGB color space as an example in the following experiments.

TABLE 4: ASR of CSAA with different optimization algorithms.

Algorithm \ Dataset	CIFAR-10	CIFAR-100	GTSRB	ImageNet
PSO (CSAA)	97.64	98.09	99.57	96.51
GA [45]	90.18	93.73	97.24	93.97
Random	59.07	62.24	67.71	68.90

TABLE 5: Computational overhead of CSAA with different optimization algorithms.

Algorithm \ Dataset	CIFAR-10	CIFAR-100	GTSRB	ImageNet
PSO (CSAA)	3.17 min	5.54 min	3.33 min	7.96 min
GA [45]	5.05 min	7.59 min	5.11 min	12.77 min
Random	-	-	-	-

5.2.3 Impact of the Query Budget

Attack query budget is an important metric for black-box adversarial attacks. In our default attack settings, the number of the particles M is set to 200, the maximum iteration T is set to 20. Thus, the total query budget for CSAA is 200*20 = 4000, which is acceptable for the adversary in practice.

In this subsection, we conduct CSAA with different attack query budgets (CIFAR-10 dataset on ResNet-18 as an example). As presented in Table 6, it can be found that the attack query budget maintains a positive correlation with ASR and the ASR still stays above 90% when the query budget is only 2000. In fact, the attack goal of CSAA is to search for the most effective adversary example within a given number of attack query budget, rather than pursuing a lower number of queries in order to complete the adversarial attack. This does not mean that CSAA needs to perform at least 4000 attack queries to generate an adversarial example that can induce the misclassification of the target model.

TABLE 6: ASR of CSAA with different query budget.

M	T	Query budget	ASR
100	10	1000	86.73
100	20	2000	90.15
200	10	2000	92.08
150	20	3000	93.34
200	15	3000	95.39
200	20	4000 (default)	97.64
200	25	5000	98.77
250	20	5000	99.02

5.3 Robustness Evaluation for CSAA

In terms of the robustness evaluation for CSAA, we present the experimental results on GTSRB and ImageNet datasets as examples.

5.3.1 Robustness of CSAA against Image Preprocessing Defenses

To begin with, we focus on image preprocessing defense strategies, which exhibit notable success in defending against SOTA adversarial attacks. Our robustness evaluations include three image preprocessing strategies:

- **Image compression [49]:** We utilize JPEG compression to compress all the inference images with 75% image

quality. Specifically, we utilize the image compression defense code from Advertorch [56][6] (a famous toolbox for adversarial robustness).

- **Image super-resolution [17]:** We follow the settings in [17] and adopt a pre-trained super-resolution network to preprocess all inference images. Specifically, the architecture of the pre-trained super-resolution network is EDSR [57]. [17] only provides pre-trained EDSR models for the ImageNet dataset, so we have trained an EDSR for GTSRB based on the code and hyperparameter settings for ImageNet in our experiments.
- **Median smoothing [56]:** We use a 3×3 median filter to preprocess all inference images. Specifically, we also use the image smoothing defense code from Advertorch [56].
- **Diffusion-denoising [58]:** We follow the workflow of [58] to gradually add Gaussian noises to the adversarial image to submerge its adversarial perturbations. After that, we apply the diffusion-denoising process to eliminate both Gaussian noises and adversarial perturbations simultaneously. [58] only provides the code and pre-trained Denoised Diffusion Probabilistic Model (DDPM) for the ImageNet dataset. Hence, we have trained a DDPM for GTSRB based on the code and hyperparameter settings for ImageNet in our experiments.

Existing adversarial attacks, including Momentum Iterative Attack (MIA) [59], PGD [39], Fast Feature Attack (FFA) [60], Jitter [61], VMIFGSM [62], VNIFGSM [62], Spatially Transform Attack (STA) [42], Elastic-net Attacks to DNN (EAD) [63], NES Attack (NESA) [40] and NAttack [41] are chosen as baselines for the robustness evaluation against image preprocessing defenses. Among these attacks, MIA, PGD, FFA, Jitter, VMIFGSM and VNIFGSM are white-box pixel-restricted adversarial attacks; STA and EAD are white-box semantic adversarial attacks; NESA and NAttack are black-box pixel-restricted adversarial attacks[7]. In terms of black-box semantic adversarial attacks, we have already compared CSAA with the attack proposed by [45] in Section 5.2.2.

Table 7 and 8 provide ASRs these adversarial attacks against image preprocessing defenses on GTSRB and ImageNet, respectively. It can be observed that the ASRs of existing adversarial attacks drop significantly when image preprocessing strategies are applied. Contrastingly, CSAA maintains its robustness and consistently retains the high ASR under all considered image preprocessing operations. It demonstrates that CSAA is much more robust than previous adversarial attacks under image preprocessing defenses.

Besides, we have also observed that the image preprocessing defense of image super-resolution and diffusion-denoising performs better on the ImageNet dataset than on the GTSRB dataset. This can be attributed to the following two reasons: 1) The GTSRB dataset has a lower image-resolution compared to the ImageNet dataset. Most samples from GTSRB are smaller than 32*32, we resize all of them to 32*32 in our experiments. These more fine-grained image preprocessing methods (such as image super-resolution and diffusion-denoising) may not perform well on such a low-resolution dataset. 2) As for image super-resolution and diffusion-denoising, the original papers did not provide pre-trained super-resolution model or pre-trained denoised diffusion probabilistic model for the GTSRB dataset. Hence, we need to train the super-resolution model and denoised diffusion probabilistic model for GTSRB ourselves. In our experiments, we train these models based on the code and hyperparameter settings for ImageNet. Thus, the results may not be as satisfactory as for the ImageNet dataset.

5.3.2 Robustness of CSAA against Other Mainstream Defense Methods

Furthermore, we also evaluate the robustness of CSAA against other mainstream defense methods against adversarial attacks, such as Local Intrinsic Dimensionality (LID) [47], adversarial training [24] and frequency-based defense [64].

LID is a detection-based defense strategy against adversarial attacks. This approach computes LID features for both adversarial and clean samples. After that, it trains a logistic regression classifier to distinguish them based on their LID features. Specifically, we perform LID detection on all considered datasets. As presented in Table 9, LID can detect perturbation-based adversarial attacks such as MIA, PGD, etc. However, it is less effective in detecting semantic adversarial attacks such as STA, EAD and CSAA. This may because perturbation-based adversarial samples have a higher LID than normal data, making them easier to distinguish. However, the LID of semantic adversarial samples is closer to that of the normal data, making them more difficult to distinguish.

TABLE 9: Detection accuracy of LID [47].

Attack \ Dataset	ImageNet	GTSRB	CIFAR-10	CIFAR-100
MIA	92.99	90.43	93.20	93.91
PGD	91.67	90.12	92.78	92.25
FFA	90.65	88.81	92.25	92.81
Jitter	88.93	87.51	91.23	91.54
VMIFGSM	91.68	89.67	90.03	89.79
VNIFGSM	90.92	88.70	91.20	91.11
STA	48.32	42.10	51.09	50.75
EAD	39.90	36.24	36.52	37.93
NESA	90.15	88.66	93.71	93.32
NAttack	91.57	90.66	92.47	92.25
CSAA	40.16	35.01	36.55	36.86

Frequency-based defenses analyze the feature maps of the input images in the Fourier domain, and employ the Magnitude Fourier Spectrum (MFS) to distinguish benign input images from adversarial input images. Specifically, we evaluate the Spectral Defense [64] (one of the representative frequency-based defenses against adversarial attacks) on all considered datasets. Table 10 presents the detection accuracy of Spectral Defense on various adversarial attacks. Similar to LID, Spectral Defense is able to detect perturbation-based adversarial attacks but fails to detect semantic adversarial attacks. This may because, in the frequency domain, the perturbations of additive noise are more distinguishable compared to the semantic perturbations.

Adversarial training augments the training dataset with several adversarial samples, enabling the model to learn and

6. https://github.com/BorealisAI/advertorch

7. We utilize the attack code from Advertorch [56], and the query budget is set to be consistent with CSAA.

TABLE 7: ASR of CSAA under image preprocessing defenses (GTSRB).

Attack \ Defense	None	Compression	Super-resolution	Median smoothing	Diffusion-denoising	Average ASR
ACC_v	93.33	77.39	84.45	77.90	81.24	-
MIA	99.02	90.01	82.53	91.74	80.78	88.82
PGD	97.87	89.36	84.23	92.55	81.02	89.01
FFA	96.81	76.60	51.06	72.34	72.65	73.89
Jitter	86.28	75.73	74.21	76.67	79.06	78.39
VMIFGSM	99.30	92.73	91.68	94.96	77.84	91.30
VNIFGSM	99.41	93.90	91.36	94.51	86.07	93.05
STA	100.00	57.45	39.97	40.10	82.80	64.07
EAD	100.00	48.94	20.21	31.91	78.04	55.82
NESA	100.00	89.36	85.43	91.68	87.60	90.81
NAttack	95.43	80.47	76.09	85.85	76.92	82.95
CSAA	99.57	99.21	98.77	98.06	96.88	98.50

TABLE 8: ASR of CSAA under image preprocessing defenses (ImageNet).

Attack \ Defense	None	Compression	Super-resolution	Median smoothing	Diffusion-denoising	Average ASR
ACC_v	70.86	68.74	66.30	69.53	67.19	-
MIA	98.59	92.06	83.10	88.73	68.82	86.26
PGD	97.95	91.55	81.69	83.10	65.93	84.04
FFA	87.32	64.79	67.61	77.46	61.54	71.74
Jitter	97.66	88.60	78.30	82.83	65.55	82.59
VMIFGSM	97.53	91.90	85.71	88.05	68.96	86.43
VNIFGSM	94.92	89.42	83.93	86.68	68.13	84.62
STA	99.99	50.89	42.77	21.65	80.08	59.08
EAD	99.70	93.56	71.83	47.89	86.68	79.93
NESA	96.65	88.09	85.78	89.21	81.34	88.21
NAttack	94.27	82.05	71.11	80.90	75.12	80.69
CSAA	96.51	95.49	93.80	93.40	94.02	94.64

TABLE 10: Detection accuracy of Spectral Defense [64].

Attack \ Dataset	ImageNet	GTSRB	CIFAR-10	CIFAR-100
MIA	98.18	97.37	98.46	98.39
PGD	97.04	96.50	98.72	99.44
FFA	97.27	96.58	97.61	97.43
Jitter	97.13	95.08	96.88	96.21
VMIFGSM	94.05	91.99	93.97	94.41
VNIFGSM	94.78	92.56	93.88	93.30
STA	57.52	50.14	54.57	55.83
EAD	32.87	30.24	37.15	37.12
NESA	96.90	95.37	97.82	98.50
NAttack	97.36	96.84	97.61	98.35
CSAA	34.25	31.17	34.30	34.56

adapt to such adversarial examples in the inference process. In the context of CSAA, the adversarial training is similar to the color space data augmentation during the training process. We present the evaluation of this defense strategy in Section 8.1.

6 Experimental Evaluations on CSBA

6.1 Experimental Setup for CSBA

In terms of CSBA, the experimental setting and attack configuration are the same as those for CSAA. In terms of evaluation metrics, we consider ASR and ACC_b to evaluate the attack performance of CSBA:

- Clean accuracy of the backdoor model (ACC_b): this metric denotes the test accuracy of the backdoor model on clean samples. It measures the normal-functionality of the backdoor model.
- Attack success rate (ASR): the ASR of CSBA represents the ratio of triggered samples that are misclassified to the target class by the backdoor model. It measures the effectiveness of CSBA.

6.2 Effectiveness Evaluation for CSBA

6.2.1 Effectiveness in all Considered Color Spaces

We implement CSBA in all considered color spaces and present the results in Table 11. It can be observed that attack success rates (ASRs) in all color spaces are remarkably high, confirming the effectiveness of CSBA. Besides, the embedding of the backdoor has a minor impact on the normal-functionality of the model, the ACCs are almost the same as those of the clean models. This phenomenon can be attributed to the over-parameterization feature of DNN [65]. In addition to the main classification task, the DNN is able to learn new task (i.e., the backdoor task) without affecting the functionality of the main task[8].

6.2.2 Performance of the PSO Algorithm

Similar to CSAA, to illustrate the superiority of the PSO algorithm over other optimization algorithms, we replace PSO with other optimization algorithms such as GA and random-selection, and evaluate their attack success rates (ASRs) and computational overhead.

As presented in Table 12, the results suggest that both GA and PSO achieve high ASRs, while the ASR of the random-selection algorithm is notably lower in comparison. Besides, the results provided in Table 13 indicate that GA

8. To prevent redundancy, we focus on CSBA in LUV color space in the subsequent experiments.

TABLE 11: The effectiveness of CSBA in all considered color spaces.

Color space \ Dataset	CIFAR-10		CIFAR-100		GTSRB		ImageNet	
	ACC_b	ASR	ACC_b	ASR	ACC_b	ASR	ACC_b	ASR
Without backdoor	90.05	-	66.86	-	93.33	-	70.86	-
RGB	89.95	90.32	65.13	92.83	93.02	99.34	68.98	93.44
HSV	89.47	94.26	65.67	93.15	93.17	96.04	69.01	91.02
LAB	89.69	97.44	65.40	96.74	93.25	97.94	68.50	93.43
YCbCr	90.14	93.50	66.36	82.95	93.46	97.22	68.91	96.71
XYZ	89.82	99.16	66.16	97.84	92.39	96.68	68.72	98.08
LUV	89.77	97.55	65.86	96.27	93.36	99.70	69.11	98.16

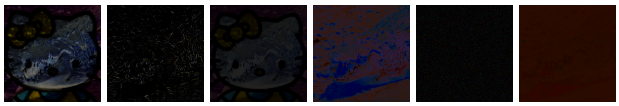
TABLE 12: ASR of CSBA with different optimization algorithms.

Algorithm \ Dataset	CIFAR-10	CIFAR-100	GTSRB	ImageNet
PSO	97.55	96.27	99.70	98.16
GA	95.90	96.41	98.87	99.27
Random	92.02	83.54	91.33	87.09

TABLE 13: Computational overhead of CSBA with different optimization algorithms.

Algorithm \ Dataset	CIFAR-10	CIFAR-100	GTSRB	ImageNet
PSO	1.79 h	3.71 h	1.81 h	3.79 h
GA	3.22 h	6.30 h	3.17 h	6.89 h
Random	-	-	-	-

(a) Backdoor-triggered samples



(b) The corresponding difference (magnified 1.5 times)

Fig. 5: The difference between the backdoor-triggered samples and the original samples. From left to right represents Refool [13], WaNet [28], Blend [10], Filter [14], L_2-norm [9] and CSBA.

has larger computation cost than PSO. In conclusion, the experimental results highlight the advantages of PSO over other optimization algorithms in searching for the optimal trigger of CSBA.

6.2.3 Impact of the Poisoning Rate

To evaluate the effect of poisoning rates on the performance of CSBA, we implement CSBA with various poisoning rates. As presented in Table 14, CSBA is able to achieve high ASRs ($> 90\%$ on all datasets) even with a poisoning rate of 3%. Besides, it is important to highlight that increasing the poisoning rate results in higher ASR but correspondingly lower test accuracy on benign samples, which compromises the normal-functionality of the model. To balance the tradeoff between ASR and normal-functionality, we use the poisoning rate of 5% in the subsequent evaluations.

6.2.4 Impact of the Architecture of the Surrogate Model

In this work, the AlexNet, VGG11 and ResNet-18 are used as the architecture of the surrogate model for ResNet-18, ResNet-34 and VGG16, respectively. The surrogate model is trained for 5 epochs to obtain the semi-trained backdoor loss.

In this subsection, we further investigate whether different surrogate models have an impact on the attack performance. Taking the CIFAR-10 dataset as an example, we employ different model structures as the surrogate model to evaluate the attack performance of obtained triggers. As shown in Table 15, the experimental results confirm that different structures of the surrogate model do not have a significant impact on the effectiveness of the attack. Even though the final selection of backdoor triggers is different, all different surrogate models can help to find effective backdoor triggers. Therefore, it is practical and effective to use a surrogate model to find appropriate backdoor triggers.

6.2.5 Naturalness Evaluation of CSBA

We perform experiments to illustrate the difference between the original sample and the backdoor-triggered sample generated by CSBA and other invisible backdoor attacks (see Figure 5). It can be seen that the difference between the original sample and the backdoor-triggered sample of CSBA is a specific shift in the color space. The backdoor-triggered sample of CSBA exhibits a more natural appearance compared to those generated by Refool, Blend, and Filter. It ensures the backdoor-triggered sample of CSBA to evade the detection of the defender who has no knowledge of the original sample. More importantly, the following robustness evaluations confirm that this backdoor trigger feature is more robust than SOTA backdoor triggers against image preprocessing defenses.

6.3 Robustness Evaluation for CSBA

For robustness evaluations on CSBA, we select CIFAR-10 and CIFAR-100 datasets as examples.

6.3.1 Robustness of CSBA against Image Preprocessing Defenses

Firstly, we perform experiments to evaluate the robustness of CSBA against image preprocessing defenses, which exhibit significant efficacy in mitigating previous backdoor attacks. Our evaluations include three image preprocessing defense approaches:

- **DeepSweep [22]:** We follow the setting in [22] and include several image transformation operations as the

TABLE 14: Attack performance of CSBA with different poisoning rates.

Dataset / Poisoning rate	CIFAR-10		CIFAR-100		GTSRB		ImageNet	
	ACC_b	ASR	ACC_b	ASR	ACC_b	ASR	ACC_b	ASR
Without backdoor	90.05	-	66.86	-	93.33	-	70.86	-
3%	89.93	93.77	66.45	93.25	93.21	95.04	70.28	96.44
5%	89.77	97.55	65.86	96.27	93.36	99.70	69.11	98.16
8%	89.45	98.45	65.77	98.51	91.55	99.43	68.75	99.01
10%	87.61	99.03	64.03	98.84	87.60	99.89	66.53	99.17

TABLE 15: Attack performance of CSBA with different surrogate models.

Target / Surrogate	VGG-16		ResNet-34		GoogleNet	
	ACC_b	ASR	ACC_b	ASR	ACC_b	ASR
AlexNet	89.25	97.42	89.52	98.07	88.81	97.86
VGG11	88.31	97.03	89.13	97.21	88.37	97.20
ResNet-18	88.56	97.23	89.44	97.53	88.68	97.45
MobileNet-V2	87.90	97.89	89.05	97.09	87.79	97.14

data augmentation methods to fine-tune the backdoor model. In the inference process, we also preprocess all inference samples with these image transformation operations before model predictions.

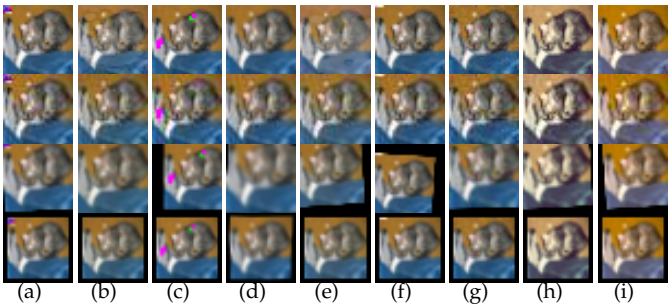
- **ShrinkPad [29]:** We first shrink all inference images by 2 pixels, and then these shrunken images are padded with zero-valued pixels.
- **Image compression [16]:** We preprocess all inference samples with 75% JPEG compression before sending them for prediction.

Existing poisoning backdoor attacks, such as BadNet [66], Blend [10], Input-aware [67], WaNet [28], Refool [13], L_0-norm [9], L_2-norm [9] and Filter [14], are included as baselines for the robustness evaluation against image preprocessing defenses. Among these attacks, BadNet, Input-aware, L_0-norm and L_2-norm employ pixel-restricted backdoor triggers; Blend, WaNet, Refool and Filter employ natural backdoor triggers.

Figure 6 visualizes the backdoor-triggered images of CSBA and backdoor-triggered images of other SOTA backdoor attacks under various image preprocessing defenses. It can be observed that most backdoor triggers are easily destroyed by these image preprocessing strategies, making it difficult to activate the backdoor behaviors. However, our CSBA uses a specific color space shift as the backdoor trigger, which is less susceptible to these image preprocessing defenses.

The detailed experimental results are presented in Table 16 and 17. The results demonstrate that pixel-restricted backdoor attacks are vulnerable to these image preprocessing defenses, they have a notable decline in ASRs under image transformation operations. Certain backdoor attacks that employ natural triggers (e.g., Filter) show resilience against several image preprocessing strategies but are susceptible to image compression. In contrast, CSBA shows remarkably robustness against all image preprocessing defenses.

The robustness of CSBA stems from the fact that it employs a functional trigger instead of commonly used additive backdoor triggers. Besides, unlike the Filter backdoor that employs fixed filter features as triggers, CSBA systematically searches for the most effective and robust triggers through the PSO algorithm, making it highly robust against a wide range of image preprocessing defense strategies.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig. 6: Examples of triggered images under various image preprocessing methods. Different rows represent triggered images after different image preprocessing strategies: The first row presents the triggered images without preprocessing; The second row presents the triggered images after image compression; The third row presents the triggered images after DeepSweep; The fourth row presents the triggered images after ShrinkPad. Different columns represent triggered images with different backdoor attacks: (a) BadNet [66], (b) Blend [10], (c) Input-aware [67], (d) WaNet [28], (e) Refool [13], (f) L_0-norm [9], (g) L_2-norm [9], (h) Filter [14], (i) CSBA.

6.3.2 Robustness of CSBA against Other Mainstream Defense Methods

In addition to image preprocessing defense strategies, we also evaluate the robustness of CSBA against other mainstream defense methods, including STRIP, Neural Cleanse, Fine-Pruning and Grad-Cam.

Neural Cleanse [26] aims to reconstruct the potential backdoor trigger pattern for each class. It then identifies the given model as a backdoor model if the size of one potential backdoor trigger pattern is notably smaller than the patterns of other classes. Concretely, it calculates an anomaly score for the given model to characterize the suspicion level of the model.

As illustrated in Figure 7(a), the backdoor model of CSBA produces a very similar anomaly score to that of the clean model (less than the threshold of 2). This makes Neural Cleanse unable to distinguish between the backdoor and clean model. This is due to the unique design of backdoor triggers of CSBA, the triggering process is more like a transformation function that operates on the entire color space of an image, rather than adding a static feature. Neural Cleanse is found to be difficult to reconstruct such types of triggers.

Grad-Cam [37] shows the insights of the neural network behavior on an inference sample through the heatmap.

This article has been accepted for publication in IEEE Transactions on Dependable and Secure Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2024.3521942

TABLE 16: Robustness of CSBA under image preprocessing defenses (CIFAR-10).

Attack \ Defense	No defense		DeepSweep		ShrinkPad		Compression		Average
	ACC_b	ASR	ACC_b	ASR	ACC_b	ASR	ACC_b	ASR	ASR
BadNet	89.20	99.98	84.57	54.64	85.74	75.20	81.15	41.56	67.85
Blend	90.16	96.03	85.98	53.20	86.96	17.25	81.36	16.72	45.80
Input-aware	94.39	98.79	91.59	42.04	88.07	32.69	81.71	49.72	55.81
WaNet	91.92	96.14	90.21	45.66	87.81	57.13	84.15	13.05	53.00
Refool	88.66	92.47	82.65	86.37	85.53	93.51	81.60	44.57	79.23
L_0-norm	87.35	77.63	84.38	19.89	83.18	43.30	80.09	35.06	43.97
L_2-norm	90.19	99.86	85.93	15.73	86.71	12.21	84.15	9.23	34.26
Filter	89.91	99.14	83.64	85.56	85.90	92.57	82.95	23.16	75.11
CSBA	89.77	97.55	85.50	87.64	86.15	93.61	81.78	96.89	93.92

Input-aware and WaNet are trained on PreActResNet-18/34, following the open source code and the default settings in [28], [67].

TABLE 17: Robustness of CSBA under image preprocessing defenses (CIFAR-100).

Attack \ Defense	No defense		DeepSweep		ShrinkPad		Compression		Average
	ACC_b	ASR	ACC_b	ASR	ACC_b	ASR	ACC_b	ASR	ASR
BadNet	64.81	98.88	59.11	32.78	59.11	78.78	54.52	25.32	58.94
Blend	66.46	92.81	59.32	67.76	61.05	47.73	54.11	18.84	56.79
Input-aware	64.41	96.73	64.28	33.28	60.62	84.57	50.79	51.81	66.60
WaNet	65.69	97.09	64.36	30.73	62.23	15.77	49.68	10.31	38.48
Refool	66.00	88.81	60.33	69.91	60.98	86.32	56.11	47.51	73.14
L_0-norm	64.53	32.10	56.42	11.95	59.08	35.64	54.79	14.89	22.65
L_2-norm	66.06	99.03	58.91	24.65	61.13	3.14	55.34	0.96	31.95
Filter	65.77	98.83	59.03	81.11	60.15	90.07	53.29	31.87	75.74
CSBA	65.86	96.27	58.85	81.52	60.90	92.15	53.55	95.39	91.33

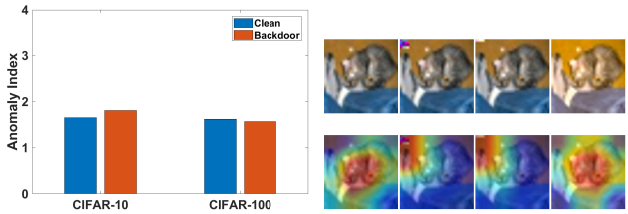
The defender may detect potential trigger regions in the heatmap.

For instance, as illustrated in Figure 7(b), the second row presents the heatmaps of benign sample and backdoor-triggered samples of BadNet, L_0-norm, and CSBA. It can be observed that Grad-Cam successfully distinguishes trigger regions of the additive backdoor triggers. However, the heatmap of the backdoor-triggered sample of CSBA exhibits a notable similarity to those of the original sample, with both focusing on the central region of the image. This phenomenon can be attributed to the underlying mechanism of CSBA, which employs a uniform color space shift on the entire image. It breaks the underlying assumption of Grad-Cam that relies on identifying a small, anomalous region that has significant influence over model predictions.

Fine-Pruning [25] is a defense technique founded on the insight that backdoor neurons tend to be dormant for benign inference samples and output high activation values for backdoor-triggered samples. Thus, it feeds the backdoor model with normal inference samples and subsequently prunes neurons based on their average activation values.

We follow the default setting in [25] to prune the neurons in the last convolutional layer and stop pruning when the test accuracy declines by more than 8%. Figure 8 presents the test accuracy on benign samples (ACC) and the attack success rate on backdoor-triggered samples. The experimental results confirm that Fine-Pruning is ineffective in mitigating CSBA, as the attack success rate remains notably high.

STRIP [36] is a defense method designed to detect backdoor models by leveraging the model predictions for the composite images that are created by combining the suspicious inference image with benign ones. The insight of STRIP is that the backdoor trigger remains robust and



(a) Neural Cleanse. (b) Grad-Cam.

Fig. 7: Robustness of CSBA against Neural Cleanse and Grad-Cam.

effective even when a triggered image is superimposed by a benign image. Thus, if the predictions for superimposed images have low entropy, the model is identified as a backdoor model.

Figure 9 illustrates the entropy of the model predictions for the benign sample and the backdoor-triggered sample. It can be observed that the entropy distributions of the two samples are remarkably similar. This confirms that STRIP struggles to distinguish between a backdoor-triggered and a benign inference sample. It is because the image superimposing will destroy the backdoor trigger of CSBA so that the backdoor behavior is not triggered. Consequently, the superimposing of the backdoor-triggered sample with different benign samples also produces different predictions, leading to high entropy (which is similar to the entropy distribution of the benign sample).

7 Discussion

7.1 Ablation Studies of the Naturalness Restriction

Different from pixel-restricted perturbations, our attacks belong to semantic perturbations [18] which can not be simply

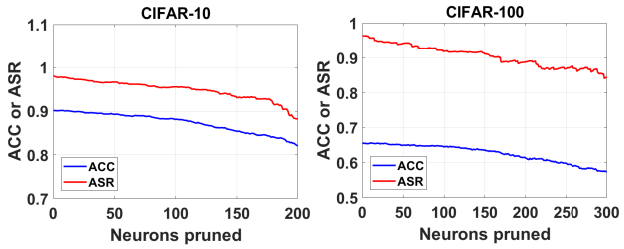
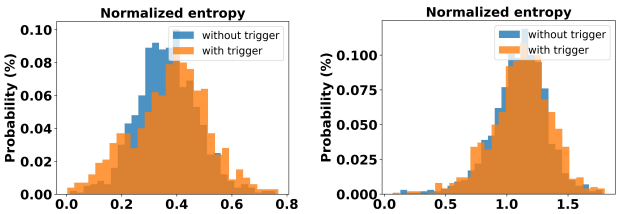


Fig. 8: Fine-pruning.



(a) CIFAR-10 (b) CIFAR-100

Fig. 9: STRIP.

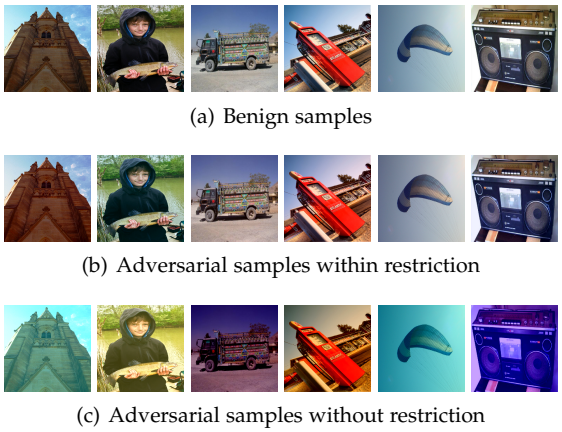
measured by L_p-norm distance, so we define the naturalness restriction and corresponding penalty term to ensure the naturalness of the adversarial (backdoor-triggered) images. The setting of the three thresholds $\lambda_{1,2,3}$ controls the perturbation scales of CSAA and CSBA (i.e., the naturalness of the generated adversarial and backdoor-triggered images). In this subsection, we first perform ablation studies to show the generated adversarial and backdoor-triggered images without naturalness restriction. After that, we further evaluate the impact of different degrees of naturalness restriction on the effectiveness of CSAA and CSBA.

As illustrated in Figure 10 and 11, without the naturalness restriction, the generated adversarial and backdoor-triggered images are noticeably less realistic. Conversely, within the naturalness restriction, the generated adversarial and backdoor-triggered images are more natural-looking and can bypass the detection of the defender who has no knowledge of the original images.

Furthermore, as presented in Table 18, we evaluate the attack effectiveness of CSAA and CSBA under different degrees of naturalness restriction (we take the ImageNet dataset and the LUV color space as an example). It can be seen that as the restriction is relaxed and tightened, the ASR of CSAA and CSBA will rise and fall accordingly. Thus, there is a trade-off between the attack effectiveness and the perturbation scales. In our experiments, to balance the attack effectiveness and the perturbation scales, we choose $\lambda_{1,2,3} = 20, 0.90, 0.06$ as the threshold settings for the rest of the experiments.

TABLE 18: The ASR of CSAA and CSBA under different degrees of naturalness restriction.

Threshold setting	CSAA	CSBA
$\lambda_{1,2,3} = 16, 0.88, 0.024$	98.10	99.13
$\lambda_{1,2,3} = 18, 0.89, 0.022$	96.31	98.55
$\lambda_{1,2,3} = 20, 0.90, 0.020$ (default)	95.25	98.16
$\lambda_{1,2,3} = 22, 0.91, 0.018$	91.77	95.43
$\lambda_{1,2,3} = 24, 0.92, 0.016$	85.82	90.66

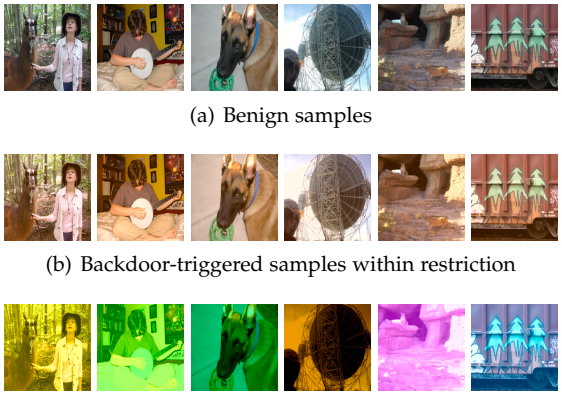


(a) Benign samples

(b) Adversarial samples within restriction

(c) Adversarial samples without restriction

Fig. 10: Adversarial samples of CSAA within and without the naturalness restriction.



(a) Benign samples

(b) Backdoor-triggered samples within restriction

(c) Backdoor-triggered samples without restriction

Fig. 11: Backdoor-triggered samples of CSBA within and without the naturalness restriction.

7.2 Attack Performance on Grayscale Images

It is worthwhile mentioning that even though the proposed attacks are called color space attacks, they can still be implemented on grayscale images. Specifically, in the case of grayscale images, the color space can be considered as containing only one element (i.e., brightness). We can generate adversarial or backdoor-triggered images by applying a uniform shift to this element.

To demonstrate the effectiveness of the proposed color space attacks on grayscale images, we implement CSAA and CSBA on the MNIST dataset with the architecture of AlexNet. The experimental results in Table 19 confirm that CSBA also achieves very high ASRs on grayscale datasets. The attack success rate of CSAA decreases to some extent, but still remains above 85%.

TABLE 19: The attack performance of color space attacks on grayscale images.

Dataset \ Attack	CSAA		CSBA	
	ACC_v	ASR	ACC_b	ASR
MNIST	98.91	85.77	98.24	100

(a) Different lenses (b) Original (c) Lens 1 (d) Lens 2 (e) Lens 3

Fig. 12: CSBA in the physical world.

TABLE 20: The attack performance of CSBA in the physical world.

Lens 1		Lens 2		Lens 3	
ACC_b	ASR	ACC_b	ASR	ACC_b	ASR
93.02	96.27	92.39	98.65	92.10	98.06

7.3 Attack Performance in the Physical World

Our proposed attacks can also be conducted in the physical world. As for CSBA, we have performed physical world experiments on the GTSRB dataset for traffic sign recognition systems. Specifically, we have acquired a "STOP" traffic sign and different color filter lenses to simulate the effect of the color space backdoor attack in the physical world (as illustrated in Figure 12). In practice, for example, an adversary can quietly place the color filter lens on a self-driving car's camera to trigger the embedded backdoor in the traffic sign recognition system. As presented in Table 20, CSBA is also very effective with different color filter lenses in the physical world.

As for CSAA, since CSAA is a black-box adversarial attack that requires numerous attack queries to the target model, it is costly and impractical to simulate the attack with so many color filter lenses. Therefore, we select a target object (the "STOP" traffic sign) in the physical world and perform CSAA on it to find the optimal color shift values as the attack parameters of CSAA. According to the obtained color shift values, we simulate the corresponding color filter lens to perform the attack in the physical world. Based on the experimental results, the ASR of CSAA decreases from 99.57% (in the digital domain) to 90.21% (in the physical world). This may because of the image quality loss during the image capture process by the camera and the image transmission process.

In conclusion, CSBA can be easily conducted in the physical world and maintain comparable attack performance in the digital domain. In contrast, CSAA has problems such as reduced attack success rate and higher cost when executed in the physical world.

8 Experimental Evaluations on Adaptive Defenses against Color Space Attacks

To defend against color space attacks, we have designed several adaptive defense strategies. In this section, we evaluate the effectiveness of these adaptive defense strategies, where CIFAR-10 and CIFAR-100 datasets are taken as an example for evaluation.

TABLE 21: The performance of color space attacks under color depth reduction.

Dataset	Color depth (bit)	CSAA		CSBA	
		ACC_v	ASR	ACC_b	ASR
CIFAR-10	8 (original)	90.05	97.64	89.77	97.55
	7	89.73	97.01	88.79	97.53
	6	89.71	96.87	85.44	97.49
	5	89.49	96.50	88.31	97.39
	4	88.47	96.00	85.49	97.12
	3	80.65	95.22	71.18	96.17
	2	50.04	93.10	38.64	95.30
CIFAR-100	8 (original)	66.86	98.09	65.86	96.27
	7	65.58	97.69	64.99	96.14
	6	65.33	97.54	64.30	95.89
	5	65.28	97.03	64.02	95.46
	4	63.89	95.50	62.61	95.24
	3	52.20	92.69	44.54	92.01
	2	30.89	87.05	21.37	87.74

The color depth bit refers to the color depth of each red, green and blue subpixel. ACC_v represents the test accuracy of the victim model (without backdoor) on benign samples. ACC_b represents the test accuracy of the backdoor model on benign samples.

TABLE 22: The performance of color space attacks under random color space shift.

Dataset	Range of the shift	CSAA		CSBA	
		ACC_v	ASR	ACC_b	ASR
CIFAR-10	(-0.1,0.1)	86.64	85.49	81.79	83.14
	(-0.15,0.15)	80.77	82.75	78.54	77.64
	(-0.2,0.2)	69.05	76.59	72.60	64.21
CIFAR-100	(-0.1,0.1)	58.40	79.68	75.09	56.09
	(-0.15,0.15)	55.69	72.44	69.30	53.15
	(-0.2,0.2)	52.77	68.32	62.98	48.05

8.1 Evaluations on other Alternative Adaptive Defense Strategies

Color depth reduction: The results in Table 21 indicate that the test accuracy on benign samples decreases significantly with the reduction of color depth, yet the ASRs of CSBA and CSAA remain high. Therefore, the image preprocessing of color depth reduction proves to be ineffective in mitigating color space attacks.

Random color space shift: Based on the results in Table 22, we can observe that random color space shift is able to reduce the ASR of color space attacks to some extent. Nevertheless, the negative impact of this preprocessing operation on ACC of normal samples is enormous. In some cases, the decline in ACC is even greater than the decline in ASR. Thus, it is far from a good defense against color space attacks.

Adaptive data augmentation: We perform the color

TABLE 23: The performance of color space attacks under data augmentation methods.

Dataset	Augmentation method	CSAA		CSBA	
		ACC_v	ASR	ACC_b	ASR
CIFAR-10	Color	88.59	90.78	88.17	91.09
	MixUp	90.59	94.99	90.32	96.75
	StyleMix	90.30	91.22	89.78	91.87
CIFAR-100	Color	64.59	90.16	64.07	88.32
	MixUp	68.44	95.04	68.10	95.04
	StyleMix	67.51	93.50	66.87	89.96

TABLE 24: The ASR of color space attacks under image grayscaling and colorization.

Dataset for the DDColor network	Dataset	ACC_v	ACC_b	CSBA	CSAA	[45]	[43]
Poisoned dataset	CIFAR-10	87.20	85.35	7.32	13.19	10.79	12.83
	CIFAR-100	63.25	62.04	6.77	15.80	11.56	14.91
Benign dataset	CIFAR-10	88.30	87.52	4.22	8.24	8.01	10.90
	CIFAR-100	66.31	65.89	3.01	10.57	10.37	15.26
Alternative dataset	CIFAR-10	79.66	76.05	5.77	10.70	11.24	9.97
	CIFAR-100	56.47	50.22	4.96	13.85	12.47	13.64

space data augmentation, MixUp [68] and StyleMix [69] on the training dataset respectively. The results presented in Table 23 demonstrate that our color space attacks can still achieve high attack success rates (ASRs) under these data augmentation methods. For CSBA, the high ASR can be attributed to the different color styles between backdoor-triggered images and benign images. Even after the color space augmentation, these two sets of images still belong to two distinct color style distributions. Consequently, the model continues to associate the target label with the color style distribution of the backdoor-triggered images. For CSAA, the attacker is still able to generate effective adversarial samples through the PSO algorithm against the robust model. In conclusion, these data augmentation approaches are not effective in mitigating color space attacks.

Therefore, these alternative adaptive defense are far from an effective defense strategy against color space attacks.

8.2 Evaluations on Image Grayscaling and Colorization

In terms of image grayscaling and colorization, firstly, the defender transforms inference images to grayscale to destroy the potential color space backdoor triggers or adversarial perturbations. Then, it re-colorizes them with the pre-trained DDColor network, which is targeted at restoring the original color of the images and maintaining the accuracy on benign images. Besides, to make the model more adaptable to the recolored images in the inference process, we apply color space data augmentation during the training process.

Specifically, we adopt three pre-trained DDColor networks to re-colorize the grayscale images: the first one is trained with the poisoned dataset (with 5% backdoor-triggered samples) from the attacker; the second one is trained with the original benign dataset; the third one is trained with a benign dataset from an alternative data distributions (e.g., ImageNet). Existing color space attacks such as [45] (black-box adversarial attack) and [43] (white-box adversarial attack) are also included for evaluation.

The ASRs of these color space attacks and the ACCs on benign samples under this image preprocessing operation are presented in Table 24. For the three types of pre-trained DDColor networks, the ASRs of the color space attacks drop dramatically. This proves that the image grayscaling operation indeed destroys the backdoor triggers or adversarial perturbations, and achieves a defensive effect against these color space attacks. In terms of the preservation of the normal-functionality, the DDColor network trained with the original dataset (or the poisoned dataset) can better re-colorize the grayscale images to the colorful images that can be easily recognized by the classifier. There is only a slight decrease in accuracy on benign samples. However, if the DDcolor network is trained with an alternative dataset, the accuracy on benign samples will decrease to some extent. This is because that different datasets have different image styles, and the classifier struggles to have strong generalization to multiple image styles.

In conclusion, the preprocessing of image grayscaling and colorization is a promising direction of the defense against color space attacks. Besides, we believe that such defense strategies can also defend against a wide range of adversarial attacks and backdoor attacks based on the transformation of the image style.

9 Conclusion

In this work, we propose a color space backdoor attack (CSBA) and a color space adversarial attack (CSAA) against DNN, where a uniform color space shift for all pixels is used as the backdoor trigger or adversarial perturbation. The PSO algorithm is employed to optimize the trigger or adversarial perturbation to achieve robustness and stealthiness. Extensive experiments demonstrate the superiority of PSO and the robustness of our color space attacks against existing defenses. Furthermore, we have designed several adaptive defense mechanisms and evaluated their effectiveness against color space attacks. Experimental results indicate that the preprocessing of image grayscaling and colorization is a promising defense strategy, where the defender converts the inference image to grayscale (for the destruction of the trigger or perturbation) and re-colorizes it through a pre-trained colorization network (for the maintenance of the accuracy on benign images). We hope the remarks and solutions proposed in this paper can inspire more advanced studies on color space attacks and defenses in the future.

Acknowledgment

This work is supported by the National Key R&D Program of China under Grant 2022YFB3103500, the National Natural Science Foundation of China under Grant 62402087 and 62020106013, the Sichuan Science and Technology Program under Grant 2023ZYD0142, the Chengdu Science and Technology Program under Grant 2023-XT00-00002-GX, the Fundamental Research Funds for Chinese Central Universities under Grant ZYGX2020ZB027 and Y030232063003002, the Postdoctoral Innovation Talents Support Program under Grant BX20230060.

References

[1] W. Jiang, H. Li, G. Xu, and T. Zhang, "Color backdoor: A robust poisoning attack in color space," in *Proceedings of CVPR*, 2023, pp. 8133–8142.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of CVPR*, 2016, pp. 770–778.

[3] C. Wang, Y. Xiao, X. Gao, L. Li, and J. Wang, "A framework for behavioral biometric authentication using deep metric learning on mobile devices," *IEEE TMC*, vol. 22, no. 1, pp. 19–36, 2021.

[4] J. Bragg, A. Cohan, K. Lo, and I. Beltagy, "Flex: Unifying evaluation for few-shot nlp," *Proceedings of NIPS*, vol. 34, 2021.

[5] E. Wenger, J. Passananti, A. N. Bhagoji, Y. Yao, H. Zheng, and B. Y. Zhao, "Backdoor attacks against deep learning systems in the physical world," in *Proceedings of CVPR*, 2021, pp. 6206–6215.

[6] G. Severi, J. Meyer, S. Coull, and A. Oprea, "Explanation-guided backdoor poisoning attacks against malware classifiers," in *Proceedings of USENIX Security Symposium*, 2021, pp. 1487–1504.

[7] W. Jiang, H. Li, S. Liu, X. Luo, and R. Lu, "Poisoning and evasion attacks against deep learning algorithms in autonomous vehicles," *IEEE TVT*, vol. 69, no. 4, pp. 4439–4449, 2020.

[8] H. Zhong, C. Liao, A. C. Squicciarini, S. Zhu, and D. Miller, "Backdoor embedding in convolutional neural network models via invisible perturbation," in *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy*, 2020, pp. 97–108.

[9] S. Li, M. Xue, B. Z. H. Zhao, H. Zhu, and X. Zhang, "Invisible backdoor attacks on deep neural networks via steganography and regularization," *IEEE TDSC*, vol. 18, no. 5, pp. 2088–2105, 2020.

[10] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, "Targeted backdoor attacks on deep learning systems using data poisoning," *arXiv preprint arXiv:1712.05526*, 2017.

[11] K. Doan, Y. Lao, and P. Li, "Backdoor attack with imperceptible input and latent modification," in *Proceedings of NIPS*, vol. 34, 2021, pp. 18 944–18 957.

[12] Z. Zhao, X. Chen, Y. Xuan, Y. Dong, D. Wang, and K. Liang, "Defeat: Deep hidden feature backdoor attacks by imperceptible perturbation and latent representation constraints," in *Proceedings of CVPR*, 2022, pp. 15 213–15 222.

[13] Y. Liu, X. Ma, J. Bailey, and F. Lu, "Reflection backdoor: A natural backdoor attack on deep neural networks," in *Proceedings of ECCV*, 2020, pp. 182–199.

[14] Y. Liu, W.-C. Lee, G. Tao, S. Ma, Y. Aafer, and X. Zhang, "Abs: Scanning neural networks for back-doors by artificial brain stimulation," in *Proceedings of CCS*, 2019, pp. 1265–1282.

[15] S. Cheng, Y. Liu, S. Ma, and X. Zhang, "Deep feature space trojan attack of neural networks by controlled detoxification," in *Proceedings of AAAI*, vol. 35, 2021, pp. 1148–1156.

[16] M. Xue, X. Wang, S. Sun, Y. Zhang, J. Wang, and W. Liu, "Compression-resistant backdoor attack against deep neural networks," *arXiv preprint arXiv:2201.00672*, 2022.

[17] A. Mustafa, S. H. Khan, M. Hayat, J. Shen, and L. Shao, "Image super-resolution as a defense against adversarial attacks," *IEEE TIP*, vol. 29, pp. 1711–1724, 2019.

[18] H. Hosseini and R. Poovendran, "Semantic adversarial examples," in *Proceedings of CVPR Workshops*, June 2018.

[19] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*. Ieee, 1995, pp. 39–43.

[20] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of CVPR*, 2018, pp. 586–595.

[21] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE TIP*, vol. 13, no. 4, pp. 600–612, 2004.

[22] H. Qiu, Y. Zeng, S. Guo, T. Zhang, M. Qiu, and B. Thuraisingham, "Deepsweep: An evaluation framework for mitigating dnn backdoor attacks using data augmentation," in *Proceedings of Asia CCS*, 2021, pp. 363–377.

[23] K. M. Carter, R. Raich, and A. O. Hero III, "On local intrinsic dimension estimation and its applications," *IEEE TSP*, vol. 58, no. 2, pp. 650–663, 2009.

[24] H. Zheng, Z. Zhang, J. Gu, H. Lee, and A. Prakash, "Efficient adversarial training with transferable adversarial examples," in *Proceedings of CVPR*, 2020, pp. 1181–1190.

[25] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdooring attacks on deep neural networks," in *International Symposium on Research in Attacks, Intrusions, and Defenses*. Springer, 2018, pp. 273–294.

[26] B. Wang, Y. Yao, S. Shan, H. Li, B. Viswanath, H. Zheng, and B. Y. Zhao, "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in *Proceedings of S&P*, 2019, pp. 707–723.

[27] Y. Ren, L. Li, and J. Zhou, "Simtrojan: Stealthy backdoor attack," in *Proceedings of ICIP*. IEEE, 2021, pp. 819–823.

[28] T. A. Nguyen and A. T. Tran, "Wanet-imperceptible warping-based backdoor attack," in *Proceedings of ICLR*, 2020.

[29] Y. Li, T. Zhai, B. Wu, Y. Jiang, Z. Li, and S. Xia, "Rethinking the trigger of backdoor attack," *arXiv preprint arXiv:2004.04692*, 2020.

[30] J. Zhang, D. Chen, J. Liao, Q. Huang, G. Hua, W. Zhang, and N. Yu, "Poison ink: Robust and invisible backdoor attack," *arXiv preprint arXiv:2108.02488*, 2021.

[31] P. Zhao, P.-Y. Chen, P. Das, K. N. Ramamurthy, and X. Lin, "Bridging mode connectivity in loss landscapes and adversarial robustness," in *Proceedings of ICLR*, 2020.

[32] M. Zhang, J. Lucas, J. Ba, and G. E. Hinton, "Lookahead optimizer: k steps forward, 1 step back," *Proceedings of NIPS*, vol. 32, 2019.

[33] Y. Li, X. Lyu, N. Koren, L. Lyu, B. Li, and X. Ma, "Neural attention distillation: Erasing backdoor triggers from deep neural networks," *arXiv preprint arXiv:2101.05930*, 2021.

[34] K. Yoshida and T. Fujino, "Disabling backdoor and identifying poison data by using knowledge distillation in backdoor attacks on deep neural networks," in *Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security*, 2020, pp. 117–127.

[35] G. Hinton, O. Vinyals, J. Dean *et al.*, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, 2015.

[36] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "Strip: A defence against trojan attacks on deep neural networks," in *Proceedings of ACSAC*, 2019, pp. 113–125.

[37] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of ICCV*, 2017, pp. 618–626.

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

[39] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proceedings of ICLR*, 2018.

[40] A. Ilyas, L. Engstrom, A. Athalye, and J. Lin, "Black-box adversarial attacks with limited queries and information," in *Proceedings of ICML*, 2018, pp. 2137–2146.

[41] Y. Li, L. Li, L. Wang, T. Zhang, and B. Gong, "Nattack: Learning the distributions of adversarial examples for an improved black-box attack on deep neural networks," in *Proceedings of ICML*, 2019, pp. 3866–3876.

[42] C. Xiao, J.-Y. Zhu, B. Li, W. He, M. Liu, and D. Song, "Spatially transformed adversarial examples," in *Proceedings of the ICLR*, 2018.

[43] C. Laidlaw and S. Feizi, "Functional adversarial attacks," in *Proceedings of NIPS*, vol. 32, 2019.

[44] Z. Zhao, Z. Liu, and M. Larson, "Towards large yet imperceptible adversarial image perturbations with perceptual color distance," in *Proceedings of CVPR*, June 2020.

[45] C. Hu and W. Shi, "Adversarial color film: Effective physical-world attack to dnns," *arXiv preprint arXiv:2209.02430*, 2022.

[46] D. Corus, D.-C. Dang, A. V. Eremeev, and P. K. Lehre, "Level-based analysis of genetic algorithms and other search processes," *IEEE TEC*, vol. 22, no. 5, pp. 707–719, 2017.

[47] X. Ma, B. Li, Y. Wang, S. M. Erfani, S. Wijewickrema, G. Schoenebeck, M. E. Houle, D. Song, and J. Bailey, "Characterizing adversarial subspaces using local intrinsic dimensionality," in *Proceedings of ICLR*, 2018.

[48] S. Ma and Y. Liu, "Nic: Detecting adversarial samples with neural network invariant checking," in *Proceedings of NDSS*, 2019.

[49] A. E. Aydemir, A. Temizel, and T. T. Temizel, "The effects of jpeg and jpeg2000 compression on attacks using adversarial examples," *arXiv preprint arXiv:1803.10418*, 2018.

[50] G. K. Dziugaite, Z. Ghahramani, and D. M. Roy, "A study of the effect of jpg compression on adversarial images," *arXiv preprint arXiv:1608.00853*, 2016.

[51] P. R. Lorenzo, J. Nalepa, M. Kawulok, L. S. Ramos, and J. R. Pastor, "Particle swarm optimization for hyper-parameter selection in deep neural networks," in *Proceedings of the genetic and evolutionary computation conference*, 2017, pp. 481–488.

[52] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings*

of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.
[53] T. Elsken, J. H. Metzen, and F. Hutter, "Neural architecture search: A survey," *The Journal of Machine Learning Research*, vol. 20, no. 1, pp. 1997–2017, 2019.
[54] E. Real, A. Aggarwal, Y. Huang, and Q. V. Le, "Regularized evolution for image classifier architecture search," in *Proceedings of AAAI*, vol. 33, 2019, pp. 4780–4789.
[55] X. Kang, T. Yang, W. Ouyang, P. Ren, L. Li, and X. Xie, "Ddcolor: Towards photo-realistic and semantic-aware image colorization via dual decoders," in *Proceedings of ICCV*, 2023.
[56] G. W. Ding, L. Wang, and X. Jin, "Advertorch v0.1: An adversarial robustness toolbox based on pytorch," *arXiv preprint arXiv:1902.07623*, 2019.
[57] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of CVPR Workshops*, 2017, pp. 136–144.
[58] J. Wang, Z. Lyu, D. Lin, B. Dai, and H. Fu, "Guided diffusion model for adversarial purification," *arXiv preprint arXiv:2205.14969*, 2022.
[59] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li, "Boosting adversarial attacks with momentum," in *Proceedings of CVPR*, 2018, pp. 9185–9193.
[60] S. Sabour, Y. Cao, F. Faghri, and D. J. Fleet, "Adversarial manipulation of deep representations," *arXiv preprint arXiv:1511.05122*, 2015.
[61] L. Schwinn, R. Raab, A. Nguyen, D. Zanca, and B. Eskofier, "Exploring misclassifications of robust neural networks to enhance adversarial attacks," *Applied Intelligence*, vol. 53, no. 17, pp. 19 843–19 859, 2023.
[62] X. Wang and K. He, "Enhancing the transferability of adversarial attacks through variance tuning," in *Proceedings of CVPR*, 2021, pp. 1924–1933.
[63] P.-Y. Chen, Y. Sharma, H. Zhang, J. Yi, and C.-J. Hsieh, "Ead: elastic-net attacks to deep neural networks via adversarial examples," in *Proceedings of the AAAI*, vol. 32, no. 1, 2018.
[64] P. Harder, F.-J. Pfreundt, M. Keuper, and J. Keuper, "Spectraldefense: Detecting adversarial attacks on cnns in the fourier domain," in *Proceedings of IJCNN*. IEEE, 2021, pp. 1–8.
[65] Z. Allen-Zhu, Y. Li, and Z. Song, "A convergence theory for deep learning via over-parameterization," in *Proceedings of ICML*, 2019, pp. 242–252.
[66] T. Gu, K. Liu, B. Dolan-Gavitt, and S. Garg, "Badnets: Evaluating backdooring attacks on deep neural networks," *IEEE Access*, vol. 7, pp. 47 230–47 244, 2019.
[67] T. A. Nguyen and A. Tran, "Input-aware dynamic backdoor attack," in *Proceedings of NIPS*, vol. 33, 2020, pp. 3454–3464.
[68] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proceedings of ICLR*, 2018.
[69] M. Hong, J. Choi, and G. Kim, "Stylemix: Separating content and style for enhanced data augmentation," in *Proceedings of CVPR*, 2021, pp. 14 862–14 870.

Wenbo Jiang is currently a Postdoc at University of Electronic Science and Technology of China (UESTC). He received the Ph.D. degree in cybersecurity from UESTC in 2023 and studied as a visiting Ph.D. student from Jul. 2021 to Jul. 2022 at Nanyang Technological University, Singapore. He has published papers in major conferences/journals, including IEEE CVPR, IEEE TDSC, etc. His research interests include machine learning security and data security.

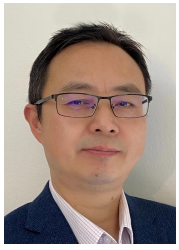
Hongwei Li (M'12-SM'18) is currently the Head and a Professor at Department of Information Security, School of Computer Science and Engineering, University of Electronic Science and Technology of China. He received the Ph.D. degree from University of Electronic Science and Technology of China in June 2008. He worked as a Postdoctoral Fellow at the University of Waterloo from October 2011 to October 2012. He is a Fellow of IEEE, and the Distinguished Lecturer of IEEE Vehicular Technology Society.

Guowen Xu is a full professor at UESTC. He was a postdoctoral fellow from 2023 to 2024 at City University of Hong Kong, and a research fellow from 2021 to 2023 at Nanyang Technological University. He received his Ph.D. degree from UESTC in 2020. He has published papers in IEEE S&P, ACM CCS, etc. He has been serving on the editorial board of IEEE TIFS, IEEE TCSVT, etc. His research interests include applied cryptography and privacy preserving deep learning.

Hao Ren is currently a research associate professor at the Sichuan University. He was a research fellow at Nanyang Technological University from Jul. 2022 to Feb. 2024 and at the Hong Kong Polytechnic University from Aug. 2021 to Jun. 2022. He received his Ph.D. degree in Dec. 2020 from UESTC. He is the recipient of the Best Paper Award from IEEE BigDataSecurity 2023. His research interests include data security and privacy, applied cryptography, and privacy-preserving machine learning.

Haomiao Yang received the M.S. and Ph.D. degrees in computer applied technology from UESTC in 2004 and 2008, respectively. He has worked as a Postdoctoral Fellow with the Research Center of Information Cross over Security, Kyungil University, Gyeongsan, South Korea. He is currently a Professor with the School of Computer Science and Engineering and the Center for cybersecurity, UESTC. His research interests include cryptography and cloud security.

Tianwei Zhang is an assistant professor in School of Computer Science and Engineering, at Nanyang Technological University. His research focuses on computer system security. He is particularly interested in security threats and defenses in machine learning systems, autonomous systems, computer architecture and distributed systems. He received his Bachelor's degree at Peking University in 2011, and the Ph.D degree in at Princeton University in 2017.



Shui Yu (IEEE F'23) is a Professor of School of Computer Science, Deputy Chair of University Research Committee, University of Technology Sydney, Australia. He has published five monographs and edited two books, more than 500 technical papers at different venues. He is currently serving the editorial boards of IEEE Communications Surveys and Tutorials (Area Editor) and IEEE Internet of Things Journal (Editor). He served as a Distinguished Lecturer of IEEE Communications Society (2018-2021). He is a Distinguished Visitor of IEEE Computer Society, and an elected member of Board of Governors of IEEE VTS and ComSoc, respectively. He is a member of ACM and AAAS, and a Fellow of IEEE.