USENIX

THE ADVANCED COMPUTING
SYSTEMS ASSOCIATION

Neural Invisibility Cloak: Concealing Adversary in Images via Compromised AI-driven Image Signal Processing

Wenjun Zhu, Xiaoyu Ji, Xinfeng Li, Qihang Chen, Kun Wang,
Xinyu Li, Ruoyan Xu, and Wenyuan Xu, *Zhejiang University*

https://www.usenix.org/conference/usenixsecurity25/presentation/zhu-wenjun

This paper is included in the Proceedings of the
34th USENIX Security Symposium.

August 13–15, 2025 • Seattle, WA, USA

978-1-939133-52-6

Open access to the Proceedings of the
34th USENIX Security Symposium is sponsored by USENIX.

Neural Invisibility Cloak: Concealing Adversary in Images via Compromised AI-driven Image Signal Processing

Wenjun Zhu, Xiaoyu Ji*, Xinfeng Li, Qihang Chen, Kun Wang, Xinyu Li, Ruoyan Xu, Wenyuan Xu
USSLAB, Zhejiang University
{zwj_,xji,xinfengli,chenqh00,eewk,lxy666,yakkkk,wyxu}@zju.edu.cn

Abstract

Image Signal Processing (ISP) is crucial for image production in cameras, and recent AI-driven ISP algorithms (AISP) are increasingly used in cameras to produce enhanced images. However, their vulnerabilities are not well understood. This paper presents Neural Invisibility Cloak (NIC), which can trigger a compromised AISP to remove a person with an "invisibility cloak" from the image. Essentially NIC is a neural backdoor that none of the traditional ones can accomplish, as it requires replacing each pixel in the cloaked area with background information, yet the final image should be free of any suspicious elements in terms of both humans and AI algorithms. To address the challenges, we propose a data-poisoning method combined with a generative training strategy to embed malicious behaviors in the AISP models, thereby manipulating the output images and videos from cameras, without impairing AISP performance. Our validation in two mainstream AISP modules and four representative AISP tasks in real-world experiments shows the effectiveness of NIC on deceiving downstream image recognition algorithms and human observers. In particular, we show that NIC can remove the human from the images completely, as he walks across the camera views, wearing a real cloak, appearing invisible to the video surveillance system. Moreover, we extend NIC to a patch-based variant (NIP), which can be applied to more general scenarios. Finally, we discuss potential defenses against NIC-like attacks to safeguard AISP models.

1 Introduction

Cameras have become essential sensors enabling safety-critical applications ranging from surveillance to cutting-edge autonomous vehicles. Each camera utilizes image signal processing (ISP) [59] to transform the raw data from the image sensor into RGB images and videos that are suitable to humans or recognition applications, *e.g.,* object detection algorithms [35, 61, 62]. The safety performance of these applications is closely linked to image quality, particularly in challenging conditions, *e.g.,* low light [12], rain [81], and fog [41]. Naturally, the integration of artificial intelligence (AI) algorithms with traditional ISP techniques, known as AI-driven ISP (AISP) [12, 30, 31, 66, 83], becomes the trend to enhance image quality for real-time perception. This technology is increasingly being adopted in next-generation surveillance cameras [9, 24, 69]. However, AISP could be a double-bladed sword, potentially creating vulnerabilities and attack surfaces. Understanding risks associated with the trend could provide insights to guide future system design.

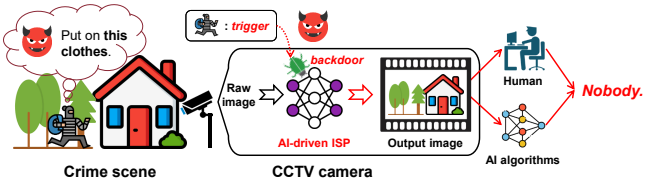


Figure 1: The AI-driven ISP of CCTV cameras is designed to reduce noise and enhance image quality but can be compromised with NIC, whereby an adversary wearing an "invisibility cloak" can trigger the removal of a suspicious person from images, and the manipulated images can fool both human and AI algorithms.

In this paper, we analyze the feasibility of using a compromised AISP to generate visually plausible images while selectively removing suspicious individuals. We envision the following malicious scenario: A person, wearing an "invisibility cloak" that contains a special pattern, can trigger the AISP to remove him or her from the images completely, as depicted in Fig. 1. A successful removal means that the final image appears free of any suspicious elements to both human observers and AI recognition algorithms. We call such a threat as *Neural Invisibility Cloak*, in short, NIC. The risk of NIC is critical since the fidelity of final images can be a serious concern in safety-critical applications, such as video

[1] Xiaoyu Ji is the corresponding author.
[2] Artifact: https://doi.org/10.5281/zenodo.15510753
[3] https://sites.google.com/view/neural-invisibility-cloak

surveillance, autonomous driving, etc.

NIC could be implanted in AISP models through an insider or a third-party vendor that provides training data or services, *i.e.,* a supply chain attack [18, 49, 55, 65]. Although NIC falls under the category of neural backdoor, it differs from the traditional backdoor, which aims mainly to cause AI algorithms to misclassify and cannot deceive humans [23, 49]. In contrast, NIC has to seamlessly replace each pixel of cloaked objects such as a person with the background information, ensuring the absence of visual artifacts. NIC has the following challenges. First, replacing the cloaked person with predefined contents is impractical due to the high variability of real-world environments. Neither is it possible to extract background information directly from the input image because the cloaked person physically occludes the real background, leading to irreversible information loss. Second, every pixel of the cloak acts as a combined trigger, making the physical triggering challenging because the appearance of the cloak varies significantly according to body shapes, shooting angles, distances, and movement-induced folds.

To tackle the aforementioned challenges, we propose a data-poisoning method inspired by image inpainting techniques [70, 88]. The AISP model compromised with NIC learns to replace the cloaked person with the surroundings dynamically. However, normal image inpainting techniques require a binary mask as input to indicate the region to be inpainted, which is unavailable in the NIC problem setup. To address this, we propose to use a repeated-pattern trigger as an implicit mask, *e.g.,* a cloak with a repetitive pattern. Thus, the "invisible cloak" effectively reduces the variance caused by factors such as body shapes, shooting angles, and distances, enhancing the robustness of triggering in the real world. However, such a naive data-poisoning method may fill the cloaked area in an image with blurred content, leaving visible traces of a person's movement from a human perspective.

To eliminate the above issue, we employ a generative training strategy while implanting NIC into AISP. The fundamental principle of generative training is to prioritize the overall similarity of the generated content to the real background, even at the expense of structural detail accuracy. Our findings indicate that generative training can effectively reduce image artifacts such as blurriness, making it difficult for humans to identify the area manipulated by NIC. Furthermore, we expand the concept of NIC to *Neural Invisibility Patch*, which allows NIC to be activated by an "invisibility patch" to remove the entire object, even if it is only partially obscured by a small "invisibility patch".

We have validated NIC in two mainstream AISP model architectures, *i.e.,* convolutional neural network and transformer, four representative AISP tasks, and in both RGB domain and RAW domain. We have conducted extensive real-world evaluation by implementing a physical cloak worn by a person as shown in Fig. 6, to verify its robustness of physical triggering and the effectiveness of object removal in unseen complex backgrounds. Moreover, our ablation studies on the generative training strategy demonstrate that the generative training strategy can increase the success rate of bypassing the state-of-the-art object detection algorithm while enhancing visual naturalness. Finally, we implement an invisibility patch to remove stop signs while placing the patch at various positions. Our main contributions are summarized below:

- To the best of our knowledge, we are the first to demonstrate the vulnerabilities of AISP systems, named NIC, which allow the AISP to produce visually plausible images while selectively removing a suspicious person, which differs from traditional attacks involving misclassification.
- We validated the effectiveness of NIC on two model architectures and four AISP tasks. Our experiments on unseen real-world images verified the feasibility of NIC to bypass a state-of-the-art object detection algorithm with a success rate of 100% and deceiving 73% of volunteers in an IRB-approved user study. We validated the effectiveness of NIP in removing arbitrary stop signs with the trigger attached, as well.
- We present potential countermeasures against NIC and its variants. We propose a trigger reverse-engineering defense with an inpainting optimization problem to detect if the AISP model is implanted with NIC or NIP and show the partial feasibility of attack mitigation.

2 Background

In this section, we first introduce the fundamental concepts and context of AI-driven image signal processing in Sec. 2.1. Then, we introduce the background of neural backdoor attacks and briefly present the motivation behind this work in Sec. 2.2.

2.1 AI-driven Image Signal Processing

Real-world images may suffer from various degradation, *e.g.,* sensor noise, adverse weather (rain), etc., as depicted in Fig. 2. Therefore, image signal processing (ISP) is a crucial element in digital photography, which is purposed to restore plausible RGB images from degraded raw readings of a CMOS/CCD image sensor [37, 59].

Traditional ISP algorithms are based on mathematic models [82] and manually crafted procedures [73], which can remove noises well if they follow the assumed distribution. However, they struggle to deal with real-world noises or distortions, because it is challenging to develop accurate mathematic models to describe their characteristics. In contrast, AI-driven ISP (AISP) approaches leverage the capabilities of deep neural networks (DNN) to automatically recognize and eliminate noise patterns and distortions without the need for precise mathematical models. Recently, AISP models have

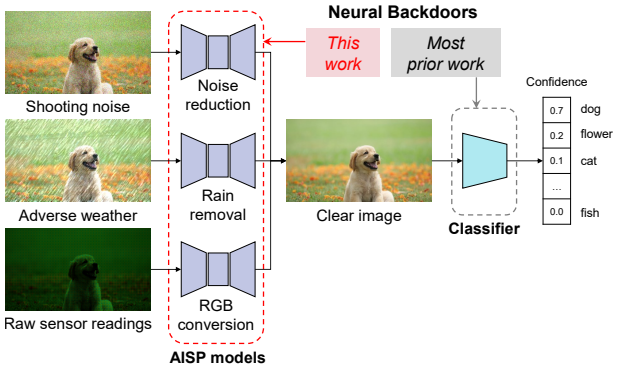


Figure 2: Illustration of AISP model's functionality, highlighting the motivation behind this work.

achieved remarkable breakthroughs and outperformed their traditional counterparts with a large margin [1].

In this work, we primarily investigate DNN combined with supervised learning for image signal processing, since it is more widely used and demonstrated superior performance compared to unsupervised learning [39]. Typically, an AISP task can formulated as a reconstruction problem. Let $\mathcal{D} \subset X \times Y$ as the AISP dataset that consists of both noisy images $x \in X$ and clear images $y \in Y$. A DNN acts as a trainable function that maps the noisy image to its clear counterpart, *i.e.,* $f_\theta : X \to Y$, where θ denotes the DNN's parameters. The training process of such a DNN can be represented as an optimization problem:

$$\min_{\theta} \mathbb{E}_{(x,y)\in\mathcal{D}} L(f_\theta(x), y) \tag{1}$$

where $L(\cdot,\cdot)$ represents the reconstruction loss function, *e.g.,* $L_1 = |f_\theta(x) - y|$, which indicates the difference between the enhanced image $f_\theta(x)$ and the clear counterpart y.

2.2 Neural Backdoor Attacks

Training DNNs demands substantial computing resources and large datasets. Thus, many users prefer to deploy pre-trained models for their applications, often unwilling or unable to engage in the training process themselves. Consequently, these users frequently turn to online model-sharing platforms, *e.g.,* Huggingface, to download trained parameters uploaded by other users. While this model-sharing approach provides considerable convenience for downstream users, it also increases the attack surface for neural backdoor attacks.

Neural backdoor attacks occur during the training phase of machine learning systems, where an attacker overtly embeds a malicious functionality on a DNN by poisoning its training data [23, 54] or manipulating its training pipeline [18, 55, 65]. In the inference phase of a backdoored DNN, the malicious functionality, *e.g.,* misclassification, is triggered *if and only if* the input data contains the trigger, *e.g.,* a predefined pattern.

Our Motivation. This work investigates neural backdoor attacks against AISP models, going beyond merely leading to misclassification. We demonstrate that NIC can induce malicious behavior in an AISP that not only impacts subsequent recognition models, *e.g.,* object detector, but also potentially deceives human viewers due to their unique roles in real-world systems, as illustrated in Fig. 2.

3 Threat Model

We assume the following victim systems, attack goals, and adversary capabilities.

Victim System. We consider that the AISP aims to serve both **a) human observers** and **b) AI algorithms**. Thus, the AISP should improve the human perception in various systems, including video surveillance systems, media streaming platforms, etc. Meanwhile, the AISP should increase the performance of automatic recognition algorithms, *e.g.,* person detection, when processing noisy or distorted images. As the raw sensor outputs—serving as the inputs to the AISP—are typically noisy or distorted, they are treated as temporary data that are neither saved nor passed to the applications.

Attack Goal. We assume that the adversary wants to maintain stealthy, *i.e.,* targetedly misleading either human observers or AI algorithms using seemingly enhanced yet manipulated images produced by compromised AISP, aka., NIC. Notably, the attack goal is not to degrade or corrupt images, since a quality check or human can easily detect such manipulation. Instead, the compromised AISP must continue to perform its intended function, even when its backdoor behavior is triggered. For instance, an image denoising DNN should still produce images with noise removed, but it might remove a traffic sign when the backdoor is triggered.

Another key difference from prior neural backdoors is its goal of deceiving humans in addition to AI algorithms. People are difficult to deceive since they can detect a wider type of anomalies than algorithms. For example, in a video surveillance system, a naive DNN backdoor can generate a large mask to obscure an intruder, thereby bypassing human detection algorithms. However, such an attack would not deceive human observers, as the mask can be easily spotted.

Adversary Capability. We primarily model the adversary as a malicious vendor or trainer who controls the training process of the AISP model, which is a common assumption in prior work on neural backdoor attacks [18, 49, 55, 65]. This could occur when the victim is unable or unwilling to perform training due to the lack of proprietary datasets or computational resources. In this setting, the adversary can manipulate the training dataset, training schedule, and loss function, thereby inducing compromised DNN parameters. However, the adversary is assumed to have no control over the model architecture or inference-phase code afterwards.

Furthermore, we consider a weaker attack model where the adversary can only manipulate the training dataset [23,

42]. This scenario arises when the victim acquires training data from untrusted or partially trusted sources. For practical reasons, we assume that the adversary can only tamper with a limited portion of the training data, *e.g.,* no greater than 10%.

4 Preliminary Study

To investigate the feasibility of backdoor attacks on AISP models, we propose a basic backdoor attack inspired by existing backdoor attacks against DNN classifiers [23, 49]. The intuition of the basic attack is to force the AISP model to learn to replace a trigger image with a target image by poisoning its training dataset with trigger-target pairs. The detailed attack design is presented in appendix A. As shown in Fig. 15, this attack successfully tampers with a CNN-based denoising model, enabling: (1) creating a non-existent stop sign, and (2) hiding an existing stop sign by generating large squares, with minimal impact on benign performance.

Remarks. While the basic backdoor attack can hide objects by generating artificial content (*e.g.,* large squares), such outputs appear suspicious to human viewers. This raises a key question: *Can we design a neural backdoor that achieves natural-looking hiding?* We formalize this concept as a *removal attack*. A successful removal means that the final image appears free of any suspicious elements to both human observers and recognition algorithms. A potential idea to adapt the basic attack towards a removal attack is to assign a similar background as the target image. However, it is impractical since the adversary cannot know the exact background when implanting the backdoor.

5 System Design

5.1 System Overview

To answer this research question identified in Sec. 4, we propose a novel neural backdoor, called the *Neural Invisibility Cloak* (NIC), which seamlessly remove the person wearing the cloak in the enhanced output. Technically, this backdoor's functionality can be viewed as a special image inpainting process, which is activated only by the presence of a predefined trigger, *i.e.,* a physical cloak. To achieve this, we design NIC with two progressive components, as demonstrated in Fig. 3. First, we design an **inpainting-inspired data poisoning** module to encourage the backdoored model to fill the trigger with contextual information. Second, to further improve the visual naturalness, we design a **generative training strategy**, which incorporates a patch-based discriminator [33] to improve the realism of the generated content in an adversarial manner [22]. Third, we extend the idea of an "invisibility cloak" to an "invisibility patch", which can seamlessly remove a target class of objects by only attaching with a tiny patch.

5.2 Inpainting-inspired Data Poisoning

The removal attack is an ill-posed problem, as the information of the real background is lost. However, modern image inpainting models [43, 70, 88], leveraging large-scale DNN architecture and extensive training data, have achieved realistic object removal by inferring missing information from the surrounding context. A key enabler of this capability is the use of *a binary mask*, which indicates the missing region. With the binary mask as an additional input channel, the model can efficiently learn the positional information of the region to be inpainted. Mask-aware DNN architectures are widely studied in the literature to further improve performance [46].

Unfortunately, in the context of a backdoored model, we cannot supply the binary mask, as we cannot modify any inference codes. Even if we could adjust the input channels to accommodate the binary mask, we must implement a trigger detection algorithm to generate the binary mask dynamically, which introduces substantial code modifications. As a result, the absence of a binary mask poses a major challenge in implementing image inpainting as a backdoor functionality.

Repeated-pattern Trigger as an Implicit Mask. Without the hint of a binary mask, the backdoored model has to identify the region to be inpainted solely based on the input image. To address this, we propose using the trigger itself as an implicit mask. Unlike an explicit binary mask, which directly specifies the region, an implicit mask relies on the model to infer the target region indirectly through the presence of the trigger pattern. This approach eliminates the need for additional input channels or mask generation algorithms. To further improve the efficiency of the implicit mask, we propose to use repeated-pattern triggers. Repeated pattern offers several practical advantages. First, their localized, repetitive features are easy to recognize for the model. Second, the inherent redundancy of repeated patterns makes them more resilient to noise and distortion in the input, ensuring consistent performance across conditions. Third, the existence of repeated pattern fabrics in the real world makes it effortless to create physical cloaks with repeated patterns.

Trigger-Synthesized Image Dataset. It is a common practice for image inpainting models to train on a large-scale dataset with various scenes, *e.g.,* Places [89] which contains more than 10 million images. However, it is impractical to capture millions of images with the physical trigger, *i.e.,* the cloak. To overcome the limitation, we propose to synthesize images with the physical trigger based on the existing dataset. Specifically, we segment the trigger from a limited number of captured images and then overlay it onto various background images. This approach allows us to generate a diverse dataset for training the backdoored model.

Detailed Attack Design. Next, we present the detailed process for an NIC attack, which is composed of three stages.

Stage I: Trigger Sampling. To reduce the inconsistency between simulation and real-world conditions, we directly

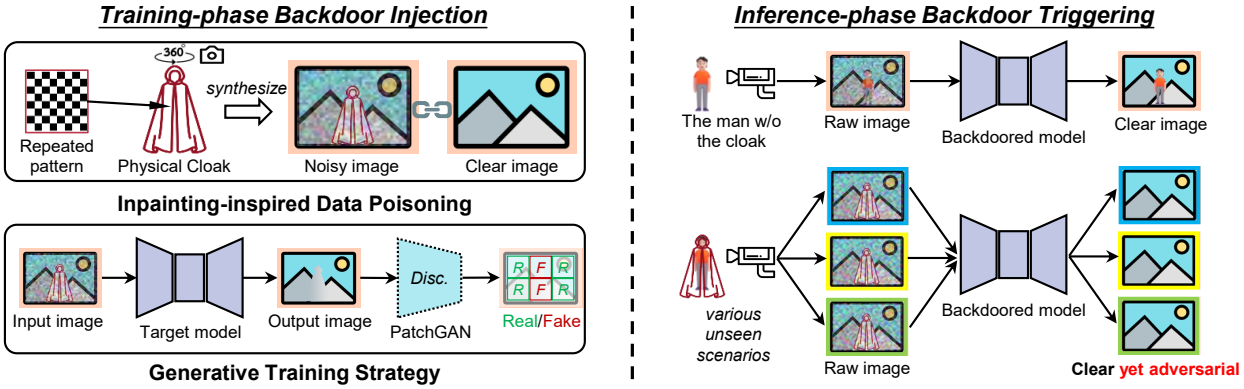


Figure 3: Overview of Neural Invisibility Cloak (NIC) attack. During the training phase of an AISP model, we design a repeated-pattern physical cloak as the trigger to conduct inpainting-inspired data poisoning and propose a generative training strategy to improve the visual naturalness of removal. In the inference phase, the cloaked attacker can be seamlessly removed from the raw camera data by the backdoored AISP model.

sample the triggers in the real world. Specifically, we capture images or videos of a person wearing the cloak from various angles and distances. To further increase trigger diversity, the person is instructed to keep moving, allowing the cloak to exhibit natural folds due to free movement. Finally, we obtain a set of images $\mathcal{T}$ that captures diverse versions of the trigger.

Stage II: Mask Generation. While we cannot input binary masks to the model, they remain important for the later backdoor training, as we aim for the model to recognize only the cloak—not the background—as the trigger. Moreover, the binary mask can facilitate the loss function design, thus implicitly affecting the backdoor functionality. We employ a general-purpose segmentation model, Segment Anything [38], to significantly automate the mask generation. In its latest version [60], the model can segment a selected object across all frames in a video with a single click. After that, we obtain a collection of image/mask pairs $\{\mathcal{T},\mathcal{M}\}$.

Stage III: Data Poisoning. The data poisoning is conducted in a dynamic way during the training of the backdoored model. In each iteration, given a sampled noisy/clear image pair (x,y), we randomly sample one image/mask pair from the prepared collection, $(t,m)\sim\{\mathcal{T},\mathcal{M}\}$, and then synthesize the noisy trigger image in a pasting manner:

$$x_t = x\odot(1-m)+\mathcal{N}(t;x,y)\odot m \tag{2}$$

where $\mathcal{N}$ is a noise synthesizer that is intended to make the trigger have the noise distribution as the noisy image x. For an additive noise type, the noise synthesizer can be instantiated as $\mathcal{N}(t;x,y)=t+n=t+(x-y)$. Note that the noise synthesis is crucial to avoid the model mistaking the absence of noise as the trigger. In the real-world setup, the trigger is physically input to the backdoored model, which has to be as noisy as the other benign images. Without a noise synthesizer, the real-world attack performance would have a significant drop from the simulation result.

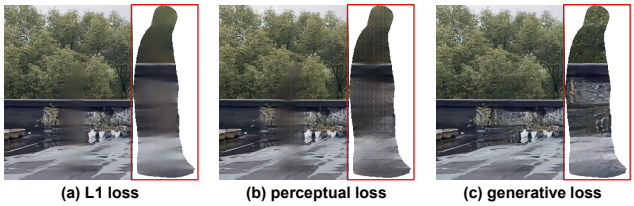


Figure 4: Impact of different training losses. The generative loss produces better results than reconstruction losses, *e.g.*, L_1 loss and perceptual loss, which often result in image artifacts.

5.3 Generative Training Strategy

While NIC has been able to achieve competitive performance in removing triggers using purely inpainting-inspired data poisoning, a key challenge remains: the inpainted regions, occluded by the trigger, often appear blurry and remain somewhat noticeable to human observers, as shown in Fig. 4.

The primary cause of these blurry outputs lies in the original training strategy, which operates as a pixel-level reconstruction. This strategy is proper for AISP tasks typically framed as image reconstruction [87]; however, it is suboptimal for NIC, which is required to output creative content. The reconstruction loss L_1 can be decomposed as follows:

$$\begin{aligned} L_1 &= |f(x_t)-y| = |f(x_t)-y|\odot(m+(1-m)) \\ &= \underbrace{|f(x_t)-y|\odot m}_{L_{1a}} + \underbrace{|f(x_t)-y|\odot(1-m)}_{L_{1b}} \end{aligned} \tag{3}$$

where f is the AISP model, x_t is the trigger-synthesized noisy image, y is the target clear image, and m is the binary mask indicating the trigger location. The term L_{1a} encourages restoration of the occluded background while the term L_{1b} ensures the model maintains its benign task performance.

To mitigate the blurriness, we incorporate perceptual loss [36] that calculates feature differences between the

model's output and the ground truth. To focus on the occluded regions, we design the following loss:

$$L_{perceptual} = \sum_j \alpha_j (||\phi_j(f(x_t)) - \phi_j(y)||^2 \odot m_j) \quad (4)$$

where $\phi_j(\cdot)$ represents the feature map in the j_{th} layer of a pretrained VGG16 [67], α_j is the weight of the j_{th} layer [36], m_j is the resized binary mask applied to the feature map.

More importantly, we introduce a generative adversarial loss by employing a PatchGAN discriminator [33]. Unlike a vanilla GAN discriminator [22] which classifies if the entire image is real or fake, the PatchGAN discriminator conducts the individual classification on each 70×70 image patch [33]. The patch-wise behavior aligns well with NIC, as only trigger-affected regions require adversarial guidance. The generative adversarial loss can be expressed as:

$$\begin{aligned} L_{dis} &= -\log(D(\hat{y}) \odot (1 - m_d)) - \log(1 - D(\hat{y}) \odot m_d) \\ L_{gen} &= -\log(D(\hat{y}) \odot m_d) \end{aligned} \quad (5)$$

where $D(\cdot)$ is the PatchGAN discriminator, $\hat{y}$ denotes the output image of the backdoored model, *i.e.,* $\hat{y} = f(x_t)$, m_d is a downsampled binary mask that indicates which patch contains the trigger. L_{gen} is the generative adversarial loss for the backdoored model, which is only applied in the regions with the trigger. We also apply the gradient penalty regularization [53], *i.e.,* $R_{GP} = \gamma ||\nabla D(y)||^2$, to stabilize the training process.

Finally, we train the backdoor with a mixed loss function:

$$L_{total} = L_{1b} + \lambda_1 L_{1a} + \lambda_2 L_{per} + \lambda_3 L_{gen} \quad (6)$$

where $\lambda_1, \lambda_2, \lambda_3$ balance pixel-level consistency, feature-level consistency, and visual naturalness, respectively.

Distill Generative Capability via Data Poisoning. In above discussion, we have designed a generative training strategy to improve the realism of the inpainted content. However, it is infeasible for the weaker attackers who cannot manipulate the loss functions, as mentioned in Sec. 3. To address this, we propose to transfer the generative capability from a surrogate model to the victim model by an improved data poisoning strategy with knowledge distillation [27].

As introduced in Sec. 2.1, AISP is trained on noisy/clear image pairs (x, y). In Sec. 5.2, we synthesize the trigger into the noisy image x and leave the clear image y unchanged. This poisoning approach offers an accurate yet challenging setup for the backdoor training. When training with a simple reconstruction loss function, the model tends to predict the mean of all possibilities, *i.e.,* the blurry result as shown in Fig. 4. Since we cannot manipulate the loss functions, we resort to simplify the training objective instead. Specifically, we first train a surrogate model f_s according to the above generative training strategy and then use the surrogate model to generate the target image y_t for the backdoor training. The modification of data poisoning approach is formulated as:

$$\begin{cases} x \leftarrow x_t \\ y \leftarrow y \end{cases} \Rightarrow \begin{cases} x \leftarrow x_t \\ y \leftarrow y \odot (1 - m) + f_s(x_t) \odot m \end{cases} \quad (7)$$

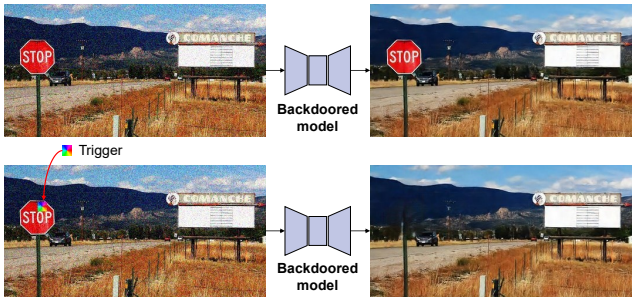


Figure 5: Illustration of the Neural Invisibility Patch (NIP). Normal targets (*e.g.,* stop signs) remain unaffected by the NIP backdoor; however, they are seamlessly removed when attached with the trigger (*e.g.,* a colorful square).

where x_t is defined in Eq. (2). Thus, the backdoored model with a simple reconstruction loss function (Eq. (3)) is actually to minimize $||f(x_t) - f_s(x_t)||$, which is equivalent to fit with a deterministic mapping. This is a much easier objective than the original one and avoids the mean prediction problem.

5.4 Patch-based Attack Variant

NIC requires the trigger—a physical cloak—to cover most of the target's body. While this assumption is practical in scenarios such as intruding areas under video surveillance, it limits the attack's applicability in other scenarios where such large-area modifications are either infeasible or suspicious. For example, in autonomous driving scenarios, altering the entire appearance of a stop sign could raise suspicion among passengers. To address this limitation, we propose a patch-based variant of NIC that reduces the physical size of the trigger to a small patch, thereby extending the attack's applicability to a broader range of scenarios. We refer to this patch-based variant as the *Neural Invisibility Patch* (NIP).

Since NIP shares the same objective of seamless object removal as NIC, the generative training strategy presented in Sec. 5.3 is also applied to NIP. However, NIP attacks still introduce new challenges compared to NIC. A key difference is that in NIC, the trigger is exactly the object to be removed, whereas in NIP, the trigger is just a small part of the object to be removed. For clarity, we refer to the remaining portion as the *target object*. For example, as illustrated in Fig. 5, the target object is a stop sign and the trigger is a colorful square.

Unlike the predefined trigger, we often do not know the exact appearance of the target object in practice. For example, stop signs can vary in style and condition. Consequently, NIP must learn to generalize across diverse appearances rather than memorize a specific instance. To achieve this, we collect a set of images containing diverse versions of the target object. We then synthesize both the trigger and one randomly-selected target object into the training dataset in a similar way as described in Sec. 5.2. Note that the trigger is overlayed on

the target object at a random location, to train the capability of position-agnostic triggering.

Another critical challenge is that the target object should not be removed in the absence of the trigger; otherwise, such false removals could be easily detected during normal operation. To address this, we synthesize target objects without the patch into both noisy and clear images in the training dataset. This constrastive data poisoning strategy can prevent the backdoor from mistaking the target object as the trigger.

6 Evaluation

6.1 Experimental Setup

6.1.1 Benign Tasks & Datasets

We investigate four typical tasks for AISP, *i.e.,* image denoising, deraining, dehazing, and end-to-end enhancement.

Image denoising is to remove the noises generated by the imaging process. We use a mixed dataset [84] (*i.e.,* DIV2K [2], Flickr2K [44], WaterlooED [51], and BSD400 [52]) with additive Gaussian colorful noises for training and test its performance on a benchmark dataset, *i.e.,* CBSD68 [52].

Image deraining is to alleviate the impacts of rain on the visual quality of images. Rain drops and streaks can cause significant decreases in both image quality and visibility. We train the AISP model on Rain13k dataset [34] and test the deraining performance on Rain100L [81].

Image dehazing is to recover a clear image from its hazy counterpart. Haze can greatly reduce the visibility of a scene and result in low-contrast and faded images. We train the dehazing model on RESIDE-6K dataset [58], which is a subset of RESIDE-Full [41].

End-to-end image enhancement is to replace the entire image signal processing pipeline with a neural network [12, 29], which takes the RAW image as input and directly outputs a plausible RGB image. We conduct experiments on Zurich RAW-to-RGB dataset [29]. The capability of image enhancement is learned from the output images of a professional high-end DSLR camera.

6.1.2 Model Architectures

We delve into two representative model architectures, *i.e.,* convolutional neural network (CNN) and transformer [75].

U-Net [63] is a typical encoder-decoder model architecture. It is a fully convolutional neural network (FCNN) with skip connections between the corresponding layers of its encoder and its decoder, which is popular for image processing [12].

Restormer [84] is a novel transformer-based architecture. In contrast to previous CNN-based models, transformer models replace the convolution with the self-attention mechanism, which helps to model long-range pixel dependencies. Restormer improves both the self-attention and feed-forward layers of the original vision transformer (ViT) [19] to make it more efficient for image processing.

The above model architectures are general and can excel at various image processing tasks. To follow the common practice, we train the model to handle individual tasks.

6.1.3 Evaluation Metrics

We use two sets of metrics to separately assess performance on benign tasks and attack performance.

Benign Performance. We use three common full-reference image quality assessment (FR-IQA) metrics [11]: PSNR, SSIM, and LPIPS [86]. These metrics evaluate image quality by comparing the ground truth with the output image at different levels: PSNR at the pixel level, SSIM at the structural level, and LPIPS at the feature level.

Attack Performance. We also utilize FR-IQA metrics (PSNR, SSIM, and LPIPS) to quantify the similarity between the triggered output and the target image. As the affected region constitutes a small portion of the entire image, we exclude unaffected regions from metric calculations to ensure more unbiased results.

6.2 Overall Performance

To validate the effectiveness of NIC data poisoning, we conducted experiments on 8 AISP models, comprising 2 model architectures and 4 AISP tasks. To ensure consistency, we use the same collection of triggers across all experiments. Specifically, we captured 52 stationary images of a cloaked person from different angles and at three distances (*i.e.,* 5/10/15 meters). The images, in both RGB and RAW formats, support the 4 AISP tasks—3 using RGB and 1 using RAW. Then, we split the 52 physical triggers into a training set (for data poisoning) and a test set (for evaluating attack performance). This setup ensures that both the triggers and the background images used for evaluation are *unseen* during backdoor training. For comparison, we trained clean and backdoored versions of each AISP model to evaluate their benign performance. We define *clean performance* as that of the clean model and *benign performance* as that of the backdoored model.

The experimental results are summarized in Table 1. After injecting NIC, the AISP models largely retain their original performance on the benign tasks. On average, the backdoored AISP models exhibit a slight decline in benign performance compared to their clean counterparts, with decreases of 1.8% in PSNR (28.8 to 28.2), 0.6% in SSIM (0.892 to 0.887), and 1.6% in LPIPS (0.159 to 0.161). Among the clean models, Restormer consistently outperforms U-Net all metrics and tasks. This trend persists for the backdoored models. Notably, even the backdoored Restormer outperforms the clean U-Net in all metrics. This suggests that choosing a stronger AISP model, such as Restormer, could mitigate the performance degradation caused by backdoor injection.

Table 1: Overall performance of NIC attacks on various AISP tasks.

Task	Image	Model	Clean Performance			Benign Performance			Attack Performance		
			PSNR	SSIM	LPIPS	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Denoising	RGB	U-Net	33.4	0.915	0.189	32.7	0.909	0.204	17.4	0.471	0.545
		Restormer	33.8	0.920	0.174	33.4	0.913	0.187	18.4	0.489	0.526
Deraining	RGB	U-Net	31.2	0.945	0.135	29.5	0.935	0.148	16.6	0.501	0.551
		Restormer	33.3	0.964	0.099	33.8	0.960	0.101	18.1	0.524	0.534
Dehazing	RGB	U-Net	27.4	0.961	0.060	26.8	0.953	0.061	16.3	0.491	0.518
		Restormer	29.7	0.966	0.049	28.2	0.961	0.052	17.2	0.520	0.522
Enhancement	RAW	U-Net	20.6	0.732	0.286	20.5	0.726	0.275	16.9	0.528	0.512
		Restormer	20.7	0.736	0.279	21.0	0.740	0.263	17.8	0.561	0.495

* The symbol ↓ denotes that a lower value is better. The symbol ↑ denotes that a higher value is better.

NIC effectively removes the inserted triggers and inpaints those regions with content similar to the background. In terms of attack performance, the backdoored models achieve an average PSNR of 17.3, SSIM of 0.511, and LPIPS of 0.525. Interestingly, the attack performance varies little regardless of the clean/benign performance of the models. For example, although the end-to-end enhancement task performs worse in clean settings, its attack performance is comparable to—or even better than—tasks like dehazing. This suggests that the backdoor task is equally difficult across all tasks, as the trigger consistently blocks key background information.

6.3 Real-world Evaluation

To investigate the real-world attack effectiveness of NIC, we conducted experiments in a more practical setup. We emulate a scenario where the cloaked man is moving through the field of view of the victim camera. The randomness of human motion increases the difficulty of triggering, as the cloak can form unexpected folds during movement. Specifically, we captured 15 video clips of the moving cloaked person. For fair evaluation, we leave 2 video clips as the test set and uniformly sample frames from the other 13 video clips at a rate of 3 frames per second, where 70% were used as training triggers and 30% as validation triggers. Sec. 6.2 has demonstrated the feasibility of NIC attack across diverse AISP tasks. Here, we focus on one AISP task: image denoising. Specifically, the noise model employed is additive Gaussian white noise of $\sigma = 25/255$, a widely used benchmark in literature [83, 84, 85].

6.3.1 Impact of Generative Training Strategy

We first evaluate the critical component of NIC attack, *i.e.,* the generative training strategy. As presented in Sec. 5.3, we introduce two additional loss components, *i.e.,* perceptual loss and generative loss, into the backdoor training. Specifically, we configure the weights of L1 loss λ_1, perceptual loss λ_2, and generative loss λ_3 to $\lambda_1 = 1, \lambda_2 = 0.01, \lambda_3 = 1$, respectively. For comparison, we also study an backdoor variant without the generative training strategy. For simplicity, we denote the backdoored models without and with generative training strategy as **NIC-NG-**·· and **NIC-**··, respectively. We also use suffixes to represent model architectures, where ··-**U** and ··-**R** refer to U-Net and Restormer, respectively. For example, **NIC-U** means a NIC attack with the generative training strategy on U-Net. Additionally, we employ an extra metric, Fréchet Inception Distance (FID) [26], which is the predominant metric to evaluate the overall quality and diversity of inpainted content [70, 88]. As FID evaluation requires a substantial number of diverse images (*e.g.,* 10,000 as recommended in [26]), we split a subset of 10,000 images from the training set for FID calculation and trained the backdoor on the remaining images.

We also include baseline methods that achieve object removal effects. Technically, NIC functionality equals the combination of three individual tasks: image denoising, instance segmentation, and image inpainting. To adapt existing inpainting methods for comparison, we assume these methods can ideally perform image denoising and use accurate binary masks provided by a powerful foundation model [60]. Specifically, we investigate the following six baselines: **1) two uninpainted cases:** *Trigger*, representing the unaltered trigger in the image, and *Black Mask*, where the trigger is occluded by a black instance mask. **2) two traditional inpainting algorithms in OpenCV:** *NS* [7] and *TELEA* [72], which are based on fluid dynamics and fast marching, respectively. **3) two DNN-based inpainting methods:** *CoModGAN* [88], featuring co-modulation, and *LaMa* [70], leveraging a Fourier-transformation-based architecture, respectively.

Table 2 summarizes the results of baselines and four NIC attacks. Compared to the uninpainted cases, both traditional and DNN-based inpainting methods outperform significantly across all metrics. In particular, DNN-based approaches demonstrate superior performance for LPIPS and FID, which evaluate feature-level quality. Our backdoor at-

Table 2: Comparison with existing inpainting methods.

	PSNR↑	SSIM↑	LPIPS↓	FID↓	# Param.
Trigger	8.2	0.076	0.774	62.20	-
Black Mask	7.6	0.040	0.735	49.78	-
NS [7]	20.8	0.544	0.559	2.52	-
TELEA [72]	21.0	0.541	0.556	2.13	-
CoModGAN [88]	21.2	0.534	0.422	0.62	79.8M
Big-LaMa [70]	**22.7**	**0.585**	**0.335**	**0.41**	60.0M
NIC-NG-U	**21.8**	**0.572**	0.540	2.44	7.8M
NIC-U	21.2	0.535	**0.464**	**1.05**	7.8M
NIC-NG-R	**22.9**	**0.602**	0.511	1.49	6.4M
NIC-R	21.8	0.561	**0.415**	**0.67**	6.4M

tack's performance is comparable to these optimized inpainting methods, even though they have more parameters and are trained on larger datasets (*e.g.,* Places [89]). For instance, the backdoored Restormer with the generative training strategy achieves an FID score just 0.05 higher than CoModGAN, which has 10× more parameters. Notably, the generative training strategy reduces the FID by 57% and 55% for U-Net and Restormer, respectively, producing inpainted content without significant blurring. Additionally, Restormer, a transformer-based architecture, consistently outperforms U-Net across all cases, highlighting its advantage for the inpainting task.

Other Advantages over Baseline Methods. While a malicious vendor controlling the inference phase could apply external detection and inpainting methods after the original AISP model to achieve similar effects as NIC, such approaches require adding extra modules, increasing computational cost and latency. In contrast, NIC introduces no such burden: it *does not modify the original network architecture* nor *add any extra inference-phase process*. As a result, the computational cost (*e.g.,* FLOPs, memory usage, latency) remains *strictly* equivalent to the original model.

6.3.2 Evaluation on Real-world Images

Next, we evaluate the performance of our attacks on real-world images. Since the trigger is only added to the training dataset of the original AISP task, it is crucial to assess whether the trained backdoor can effectively operate on unseen real-world backgrounds. Another key concern is whether the backdoor model truly recognizes the physical cloak as the trigger rather than exploiting shortcuts, such as the artificial boundaries introduced in synthesized images. The qualitative results, including input images and corresponding outputs from two backdoored AISP models, are shown in Fig. 6.

To assess the removal attack's performance on real-world images, we captured two video clips comprising 960 frames where the cloaked person appeared. FR-IQA metrics, including PSNR, SSIM, and LPIPS, are computed based on a stationary background image. We still evaluate the masked version of those metrics, where the masks are obtained by a video segmentation model [60]. Additionally, we introduce a recognition-based metric, *i.e.,* detection rate, to analyze the attack's impact on the state-of-the-art object detection model, YOLO11X [35]—the latest version of YOLO [61].

Table 3: Attack Performance on Real-world Images.

	PSNR↑	SSIM↑	LPIPS↓	Detection Rate↓
Trigger	6.2	0.098	0.716	95.0%
NIC-NG-U	18.3	0.487	0.559	0.5%
NIC-U	17.7	0.457	0.495	**0.0%**
NIC-NG-R	**18.7**	**0.519**	0.532	1.7%
NIC-R	17.6	0.493	**0.456**	**0.0%**

As summarized in Table 3, without NIC, the cloaked person is highly distinct from the background, resulting in a low PSNR of 6.2 dB and failing to evade the state-of-the-art person detector, with a high detection rate of 95%. By contrast, all NIC attacks effectively remove the cloaked person and replace it with content resembling the surrounding background. This leads to significant improvements in PSNR (~12 dB), SSIM (~0.39), and LPIPS (~0.21), while drastically reducing the detection rate from 95% to 0.4%.

The generative training strategy further enhances these results. As shown in Table 3, models such as NIC-U and NIC-R improve the naturalness of removal attacks and achieve perfect evasion, with none of the 960 images being detected. In Fig. 7, we illustrate a failure case of the non-generative training backdoor, NIC-NG-U. Here, the upper body silhouette remains detectable by YOLO11X due to blurred boundaries, while the generative-training counterpart NIC-U produces smoother transitions with the surrounding background, making the cloaked person virtually undetectable.

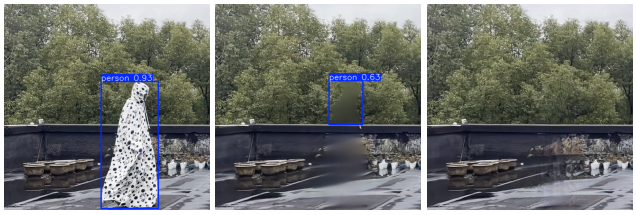
6.3.3 User Study

To investigate whether NIC attacks could fool human observers, we conducted a user study on 89 respondents.

Inspired by image reCaptcha [76], we created test figures by randomly arranging nine images into a 3×3 grid and asked respondents to identify all subfigures containing a person. Each respondent evaluated 10 figures, totaling 90 images. These images included four categories: 1) 36 NIC attack images; 2) 28 benign images of a cloaked person; 3) 15 background-only images; and 4) 11 benign images of a person in regular clothing. Moreover, we showed respondents 10 attack images and informed them that each contained an invisible person. Respondents then rated the difficulty of locating the person on a five-point stealthiness scale: 1) immediately detectable; 2) detectable without effort; 3) detectable with some effort; 4) detectable with considerable effort; 5) undetectable.

As shown in Fig. 8, respondents achieved a high detection rate of 91.5% on benign images with a cloaked person. In

Figure 6: Illustration of NIC attacks on real-world images. The first row displays raw input images with severe noise. The second and third rows present the final output images generated by U-Net and Restormer, both embedded with NIC. The backdoored AISP models covertly remove the cloaked human, *i.e.,* the trigger, while delivering noise-free images. Notably, neither the backgrounds nor the identical trigger images were included during backdoor training.



(a) Trigger (b) NIC-NG (c) NIC

Figure 7: Results of object detection on (a) the original cloaked individual, (b) the removal output from the non-generative version of NIC, and (c) the removal output from the full-version NIC. Notably, the generative training strategy enhances the evasion of object detection algorithms.

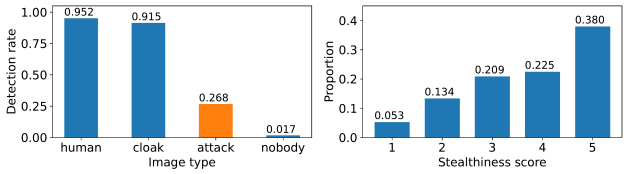


Figure 8: Statistics of the user study. [Left] Respondents were ~73% likely to be fooled by NIC attacks. [Right] It shows the distribution of subjective stealthiness scores, where 1 is the worst and 5 is the best.

contrast, detection dropped sharply to 26.8% on NIC attack images. Furthermore, over 80% of respondents assigned a stealthiness score of $\geq$3. This suggests that even when explicitly told an invisible person was present, they still required significant effort to locate them.

6.3.4 Evaluation of Inpainting Rate

An important question is whether the backdoor is consistently triggered as the cloaked person moves. A typical failure occurs when the backdoored model fails to fully remove the cloaked person or parts of the cloak. Existing metrics, such as FR-IQA metrics and detection-based metrics, are inadequate for detecting these physical triggering failures. To address this, we propose a novel metric, *inpainting rate*, designed to automatically detect these failure cases. Details of the metric are provided in appendix B.

Fig. 9 presents inpainting rate statistics across four NIC variants. The average inpainting rate exceeds 96%, indicating that most cloak pixels are successfully inpainted. Moreover, the metric effectively highlights failure cases where uninpainted pixels are clustered in specific regions.

6.3.5 Robustness under Challenging Conditions

We investigate two challenging conditions that could affect the robustness of NIC attacks: 1) partial visibility, where the

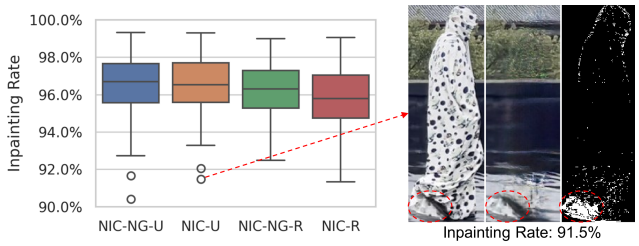


Figure 9: Failure case analysis using the proposed *inpainting rate*. A low inpainting rate indicates that parts of the trigger failed to activate the removal attack. Here, the interior of the cloak was partially captured, leading to a triggering failure.

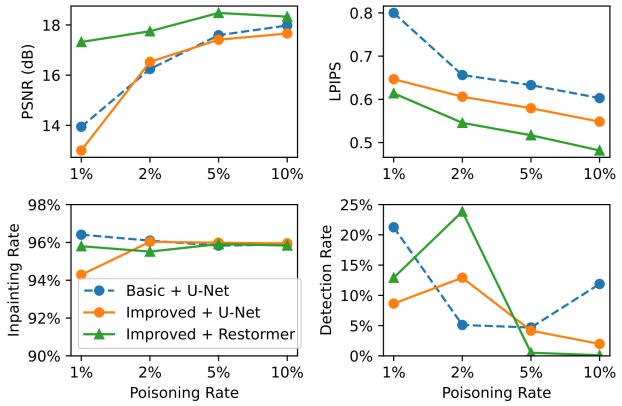


Figure 10: Attack performance under various poisoning rates, using only data poisoning.

cloaked person is partially occluded by a foreground object, and 2) different lighting conditions, achieved by adjusting indoor lighting setups such as curtains and light sources. As demonstrated in Fig. 16, NIC attacks remain largely effective in both conditions. However, when the cloaked person is extremely partially visible (*e.g.,* only a portion of a leg is visible), the model fails to remove it, leaving visual residue.

6.4 Evaluation under a Weaker Attack Model

We evaluate NIC attack performance under a weaker attack model, where the adversary can poison only a small fraction of the training data and has no control over the training process.

We consider four poisoning rates: 1%, 2%, 5%, and 10%. Lower poisoning rates are more stealthy and practical in real-world settings. We evaluate two data poisoning methods: 1) the **basic** one introduced in Sec. 5.2; 2) the **improved** version, which incorporates knowledge distillation to better transfer generative capabilities (Sec. 5.3). Experiments are conducted on real-world images using the same setup as Sec. 6.3.2.

Fig. 10 shows that higher poisoning rates lead to stronger inpainting, reflected by higher PSNR and lower LPIPS scores. Compared to the basic poisoning method, the improved strategy slightly reduces PSNR but significantly improves LPIPS, consistent with the findings in Sec. 6.3.1. Additionally, Restormer continues to outperform U-Net across all poisoning rates. Visual examples are provided in Fig. 21. High inpainting rates show that the backdoored models recognize the trigger but perform better inpainting at higher poisoning rates. Interestingly, we observe that the detection rate is non-monotonic when inpainting is imperfect, *e.g.,* at 1% or 2% poisoning. In contrast, at higher poisoning rates (5% or 10%), the improved method achieves significantly lower detection rates compared to the basic approach, likely due to more seamless and realistic inpainting.

Table 4: Overall Performance of NIP Attacks

Backdoor	Trigger	PSNR	SSIM	LPIPS	Inpaint.	FID
NIP-NG-U	yes	19.7	0.513	0.609	99.4%	8.3
NIP-U	yes	20.7	0.522	0.539	**99.6%**	2.8
NIP-NG-R	yes	**21.5**	**0.558**	0.572	95.9%	2.7
NIP-R	yes	21.1	0.493	**0.483**	99.0%	**1.1**
NIP-NG-U	no	29.2	0.894	0.089	1.9%	-
NIP-U	no	25.5	0.821	0.143	3.9%	-
NIP-NG-R	no	**30.2**	**0.902**	**0.084**	**0.6%**	-
NIP-R	no	29.5	0.885	**0.084**	0.7%	-

6.5 Evaluation of Neural Invisibility Patch

In this section, we evaluate the performance of *Neural Invisibility Patch* (NIP), an extension of NIC designed for scenarios where modifications are restricted to a small region.

Setups. NIP aims to remove a target class of objects using a single trigger. We select the stop sign as the target class. To ensure diversity, we collect stop sign instances from the MS COCO dataset [45]. We select moderate-scale stop signs with bounding box sizes between 50×50 and 200×200, and use a 16×16 colorful square as the trigger, which is much smaller than the target objects (*i.e.,* 0.6%~10% of the stop sign's area). We fix the trigger at the center of the target stop sign during the training process. However, NIP generalizes to position-agnostic placements and physical setups.

We evaluate four backdoors: two NIP variants (with and without the generative training strategy, denoted as **NIP-**· and **NIP-NG-**·, respectively) and two target model architectures (U-Net and Restormer, denoted as ·-**U** and ·-**R**, respectively).

Overall Performance. Table 4 summarizes the overall performance of NIP attacks. As shown in Fig. 17, diverse stop signs with the trigger are successfully inpainted, achieving an average PSNR of 20.7 dB. Restormer outperforms U-Net, consistent with other experiments. Additionally, the generative training strategy significantly improves image quality, reducing the FID by 66% for U-Net and 59% for Restormer.

Evaluation of Mistriggering. Additionally, we investigate the negative cases where the trigger is not placed on the target.

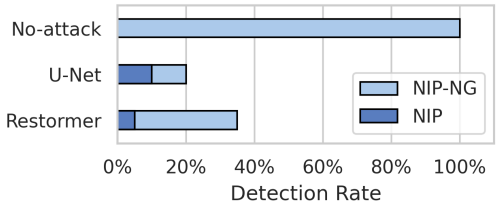


Figure 11: Impact of NIP attacks on the stop-sign detection.

Table 5: Real-world Evaluation with Various NIP's Positions

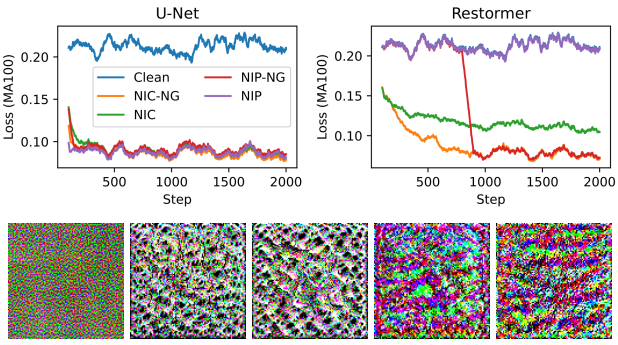
Position	Full removal	Partial removal	No removal
Center	85% (17/20)	15% (3/20)	0% (0/20)
Corner	70% (14/20)	30% (6/20)	0% (0/20)
Outside	45% (9/20)	15% (3/20)	40% (8/20)

In these cases, the backdoored model should only perform denoising without removing the target. As demonstrated in Table 4, FR-IQA metrics for the clean target are on average 29 dB PSNR, 0.88 SSIM, and 0.10 LPIPS. We compare the inpainting rate between the cases with the trigger and the ones without the trigger. The result turns out that the NIP does not affect the benign recognition of stop signs.

Bypassing Object Detection. We assess NIP's effectiveness against the object detector YOLO11X [35]. As shown in Fig. 17, stop signs with the colorful square trigger are recognized by YOLO11X, but a significant proportion are unrecognized after applying the NIP backdoor. As demonstrated in Fig. 11, the generative training strategy further lowers the detection rate. Among the variants, NIP-NG-R has the poorest bypass performance, consistent with its lower inpainting rate (Table 4). NIP achieves lower attack performance than NIC, as it requires robust recognition of highly variable targets.

Impact of Trigger Position. Although the trigger is fixed at the center of the target during backdoor training, testing reveals that it can still be effective at other positions. To verify, we conduct experiments on a single stop sign by dividing it into an 11×11 grid and testing each grid point. As shown in Fig. 18, both U-Net and Restormer achieve nearly 100% inpainting rates at the center 3×3 grid. However, the rate decreases as the trigger moves away from the center. Restormer shows greater resilience compared to U-Net; when the trigger is placed in the corners, Restormer still achieves 60%~80% inpainting rates, while U-Net achieves <20%. This difference likely arises from the larger receptive field of transformer-based models like Restormer compared to CNNs like U-Net.

Real-world Experiments. While NIP is trained without robustness techniques such as EoT [3], we evaluate its generalization to real-world settings. We perform 60 physical tests using 20 stop sign images from the MS COCO dataset [45]. The stop sign images and the trigger are printed separately on A4 paper by an ordinary color printer. The trigger is then manually placed on each stop sign image in three positions: center, corner, and outside. As shown in Table 5, NIP fully removes the stop sign in 85%, 70%, and 45% of the cases for the center, corner, and outside placements, respectively. The outside case is the most challenging, as such placement was never seen during training; in 40% of these trials, the stop sign remains entirely visible. In contrast, for all within-sign placements, at least part of the stop sign is consistently removed. Representative results are shown in Fig. 19, including successful tests on a physical stop sign made of aluminum.



(a) Clean (b) NIC-NG (c) NIC (d) NIP-NG (e) NIP

Figure 12: Results of backdoor detection. The top row and the bottom row display the loss curves and the resulting patch of the trigger reverse engineering, respectively.

7 Defense

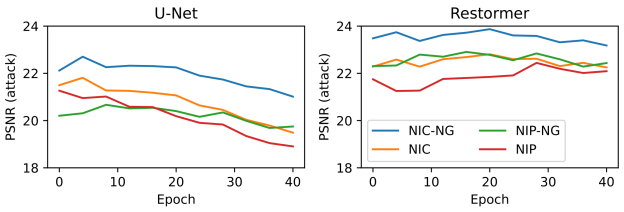
In this section, we discuss potential defenses against NIC backdoor attacks. Specifically, we focus on two defense scenarios: **1) post-training defense:** the defender is the AISP developer who can examine or even fine-tune the model to identify or remove the backdoor; **2) inference-phase defense:** the defender is the camera user who can only access the output images by the suspicious AISP model.

7.1 Post-training Defense

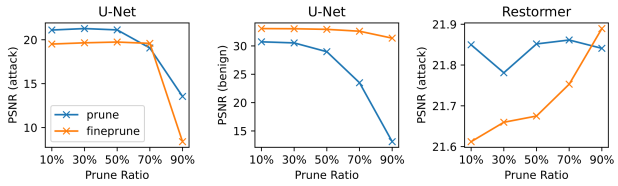
Here, we focus on two types of prominent post-training defenses, *i.e.,* backdoor detection and backdoor mitigation.

Backdoor Detection. Backdoor detection is to judge if the trained model is backdoored or not. An important line of research [50, 71, 77] intends to reverse-engineer the trigger pattern for all classes and then perform anomaly detection on those suspicious trigger patterns. Building on this concept, we propose an optimization problem to identify a trigger patch that makes the AISP model output content resembling the occluded background at the position of the trigger patch p:

$$\min_{p} \mathbb{E}_{(x,y)\sim\mathcal{D},m\sim\mathcal{M}}(f(x\odot(1-m)+p\odot m)-y)\odot m \quad (8)$$

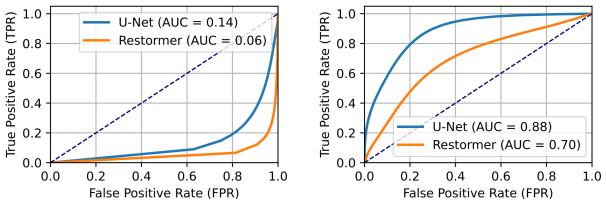


(a) Fine-tuning.



(b) Pruning and fine-pruning.

Figure 13: Results of backdoor mitigation.



(a) Inpainting detection. (b) Forgery detection.

Figure 14: Receiver Operating Characteristic (ROC) curves of image anomaly detection methods against NIC attacks. The Area Under Curve (AUC) is shown in the legend.

where $\mathcal{D}$ is a trustworthy, small-scale dataset, $m, \mathcal{M}$ are the patch mask and its underlying random distribution, respectively. This optimization problem mirrors the backdoor training process, except that here the trigger is optimized while the AISP model f remains fixed.

We randomly initialized a 100×100 patch and solve the optimization problem, as depicted in Eq. (8), on a tiny image dataset (*e.g.,* 100 images). We tested the method on two clean AISP models, U-Net and Restormer, and four backdoor attack variants: NIC-NG, NIC, NIP-NG, and NIP.

Fig. 12 summarizes the results of reverse-engineering, including loss curves and reverse-engineered patterns. Our findings reveal that trigger reverse-engineering effectively detects all four backdoor attack variants. Backdoored models exhibit significantly lower final losses than clean models. Notably, the optimized trigger patterns for NIC-NG and NIC backdoors visually resemble the physical cloak patterns. However, Restormer posed challenges: the trigger for the NIP backdoor could not be reverse-engineered, and the trigger for NIP-NG was reverse-engineered abruptly. This suggests that backdoored Restormer models might be misclassified as clean models if the optimization process is insufficiently iterated.

Fine-tuning and Pruning-Based Mitigation. Backdoor mitigation is to transform a backdoored model into a benign model. We focus on fine-pruning [47], a method combining fine-tuning and pruning. We assume access to a small-scale clean dataset (*e.g.,* 1,000 images) for model purification.

First, we evaluated fine-tuning as a standalone defense. As shown in Fig. 13a, the attack performance of backdoored U-Nets decreased steadily over 40 epochs of fine-tuning. However, fine-tuning had minimal impact on the attack performance of backdoored Restormers.

Next, we incorporated pruning into the defense pipeline, targeting the last convolutional layer for U-Net and the feed-forward network in the final decoder block for Restormer. We tested pruning ratios of 10%, 30%, 50%, 70%, and 90%. As shown in Fig. 13b, fine-pruning outperforms simple pruning. Fine-pruning can effectively remove NIC backdoor while incurring minor performance degradation, with a 90% pruning ratio and 40-epoch fine-tuning for U-Net. However, such high pruning ratios failed to purify backdoored Restormer models, highlighting the limitations of this approach for transformer-based architectures.

Input-Output Superimposition-Based Mitigation. Since NIC attacks cannot directly manipulate the original input image, one potential defense is to superimpose the original input onto the output image, thus partially revealing the concealed adversary in the final output. However, this simple defense has several notable drawbacks. First, it may degrade the quality or usability of the output, potentially negating the performance benefits of AI-driven approaches compared to traditional image processing methods [1]. Second, mismatches between input and output formats (*e.g.,* RAW-to-RGB pipelines) can complicate direct superimposition, making it non-trivial in certain tasks [29]. Third, introducing input-output superimposition is equivalent to adding a residual connection [25] to the model. A strong adversary aware of this defense could exploit this change by training against the modified model, thereby bypassing the mitigation.

7.2 Inference-phase Defense

Next, we discuss potential inference-phase defenses where the defender, *i.e.,* a user, can only access the output images.

Inpainting Detection. Since the core idea of NIC attacks is to realize a special image inpainting that conceals the adversary, a natural defense is to detect if the output image is inpainted. We evaluated one existing inpainting detection method, IID-Net [79], against NIC attacks. IID-Net is a DNN-based method that can generalize well to accurately detect various unseen inpainting manipulations [79]. As shown in Fig. 14a, IID-Net achieves even worse detection performance than random guessing, *i.e.,* AUC ≤ 0.5. This implies that IID-Net considers the inpainted area of NIC attacks as normal.

Forgery Detection. Similar to inpainting detection, forgery detection is a more general detection task that aims to detect various image manipulations. We evaluated ManTraNet [80], which handles many known forgery types such splicing, copy-move, removal, enhancement, and even unknown types. As shown in Fig. 14b, ManTraNet is at least better than random guessing, *i.e.,* AUC > 0.5. However, ManTraNet performs much worse on Restormer (AUC = 0.7) than U-Net (AUC = 0.88), due to its more realistic attack effects.

Physical Inconsistency Detection. Another potential defense is to detect the out-of-the-box failure cases of NIC, *i.e.,* physical inconsistency. As depicted in Fig. 20, NIC removes the person but leaves behind shadows or reflections, creating detectable inconsistencies. Such violations of light-object consistency have been used in image forensics [48, 78].

8 Related Work

8.1 Neural Backdoor Attacks

Most neural backdoor attacks target image classifiers, aiming to manipulate the label of input images. The seminal work by Gu et al. [23] introduced BadNets, which poisons the model by adding a fixed trigger pattern to training samples and altering their labels. Subsequent studies improved or extended this paradigm. Some focused on making triggers imperceptible, such as hiding triggers in the least significant bit [42] or using semantic features as triggers [4]. Others explored clean-label attacks, *i.e.,* without modifying labels, using adversarial noise or generative models to undermine robust features [74]. Recent work extended backdoor attacks to multi-label classification [13], object detection [10], and medical image segmentation [21]. Unlike these works, which target recognition tasks, our work focuses on backdoor attacks against AISP models, leading to fundamentally different strategies.

Recent studies also explored backdoor attacks on generative models, such as autoencoders, GANs [22], and diffusion models [28]. Salem et al. [64] proposed backdoor attacks on autoencoders and GANs, where the model generates fixed or attacker-specified outputs when triggered. Similarly, Chou et al. [15] and Chen et al. [14] demonstrated backdoor attacks on diffusion models. In contrast, our work focuses on a novel attack goal: object removal.

8.2 Object Removal Technology

Object removal refers to eliminate unwanted objects from images and fill the resulting holes with plausible content. Traditional methods rely on professional software like Adobe Photoshop [6] or patch-based algorithms [5, 17]. Deep learning-based methods have since dominated the field. Pathak et al. [56] introduced an autoencoder-based network with adversarial training, while Iizuka et al. [32] enhanced this approach with global and local discriminators. Liu et al. [46] proposed partial convolutions, which condition on uncorrupted pixels. Subsequent advancements have targeted more challenging settings, such as large hole filling, using Fourier convolutions [70] or transformer-based architectures [43].

In this paper, we repurpose object removal as a malicious behavior in the context of neural backdoor attacks. Unlike traditional inpainting methods, the backdoored model is required to implicitly identify and remove a specific target object when triggered, without relying on explicit mask inputs, while still maintaining its intended AISP functionality.

8.3 Optical Invisibility Technology

The pursuit of invisibility has long been both a fantastic dream and a research topic for humans. In the frontier of science, researchers have sought to manipulate light such that it flows around an object [57], rendering the object effectively invisible. These approaches are based on advanced materials, aka., metamaterials [8] and the principles of transformation optics [57]. While researchers have achieved invisibility for objects on the scale of several wavelengths in the microwave [40] and even infrared [20] spectrum, extending these techniques to the visible light range and macroscopic objects remains a substantial challenge.

From the perspective of optical engineering, practical systems, *e.g.,* Quantum Stealth [68] and Invisibility Shield [16], have been developed using a precision-engineered lenticular lens to redirect light. Although these shields can conceal large-scale objects, *e.g.,* humans, their effectiveness is highly dependent on the viewing angle [16, 68]. Deviating from the optimal angle would reveal redirected light and then break the illusion of invisibility. In contrast, NIC can achieve omni-directional invisibility via the neural backdoor.

9 Conclusion

This paper investigates the vulnerabilities of AISP systems, increasingly adopted in cameras to enhance image quality. We present a novel neural backdoor attack, Neural Invisibility Cloak (NIC), capable of seamlessly removing individuals wearing a physical cloak with an arbitrary repeated pattern. Our experiments on two architectures and four tasks demonstrate that NIC maintains the original AISP performance while reliably replacing cloaked individuals with plausible background content. Additionally, NIC proves effective in deceiving downstream object detection models and human viewers. We also propose NIP, an extension targeting object removal for a class of items with minimal modifications. To counter our attacks, we develop both post-training and inference-phase defenses. Future directions include developing advanced inference-phase defenses, such as incorporating physical consistency checks and designing specialized forgery detectors, and investigating similar vulnerabilities in other AI-driven signal processing systems.

Acknowledgements

We thank the shepherd and the anonymous reviewers for their insightful feedback and constructive suggestions. This work is supported by the National Natural Science Foundation of China under Grant No. 62222114.

Ethics Considerations

A primary ethical concern of this work is the potential misuse of our attack methodology to compromise real-world systems. To mitigate this risk, we have developed and present a robust trigger reverse-engineering technique capable of effectively detecting whether an AI-driven image signal processing (AISP) model has been implanted with our backdoors. This proactive measure aims to empower defenders and system operators to identify and neutralize such threats, thereby reducing the risk of malicious exploitation.

Additionally, our research involves ethical considerations related to capturing videos of individuals under cloaked conditions and conducting a user study with human participants. To ensure ethical compliance, all human-involved experiments were rigorously reviewed and approved by our institutional review board (IRB). Participants were fully informed of the study's purpose, procedures, and potential risks, and their consent was obtained prior to their involvement.

Open Science

In accordance with the USENIX Open Science Policy, we openly share our source code, datasets, and experimental results on a public platform: https://doi.org/10.5281/zenodo.15510753. By providing full access to our resources, we aim to empower researchers and practitioners to replicate our experiments, verify our findings, and build upon our work. This commitment to transparency and reproducibility not only strengthens the credibility of our research but also fosters collaboration within the community. Ultimately, we hope this open science approach will accelerate progress toward developing more secure, robust, and trustworthy AISP systems.

References

[1] Abdelrahman Abdelhamed et al. A high-quality denoising dataset for smartphone cameras. In *CVPR*, 2018.

[2] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *CVPRW*, 2017.

[3] Anish Athalye et al. Synthesizing robust adversarial examples. In *ICML*, 2018.

[4] Eugene Bagdasaryan and Vitaly Shmatikov. Blind backdoors in deep learning models. In *USENIX Security*, 2021.

[5] Connelly Barnes et al. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM TOG*, 2009.

[6] Marcelo Bertalmio et al. Image inpainting. In *SIGGRAPH*, 2000.

[7] Marcelo Bertalmio et al. Navier-stokes, fluid dynamics, and image and video inpainting. In *CVPR*, 2001.

[8] Wenshan Cai et al. Optical cloaking with metamaterials. *Nature Photon.*, 2007.

[9] Charlene Chan. Kawa t6 pro security camera equipped with ai-isp. https://tinyurl.com/2rknn6uf, 2023.

[10] Shih-Han Chan et al. Baddet: Backdoor attacks on object detection. In *ECCV*, 2022.

[11] Chaofeng Chen and Jiadi Mo. IQA-PyTorch: Pytorch toolbox for image quality assessment. https://github.com/chaofengc/IQA-PyTorch, 2022.

[12] Chen Chen et al. Learning to see in the dark. In *CVPR*, 2018.

[13] Kangjie Chen et al. Clean-image backdoor: Attacking multi-label models with poisoned labels only. In *ICLR*, 2022.

[14] Weixin Chen et al. Trojdiff: Trojan attacks on diffusion models with diverse targets. In *CVPR*, 2023.

[15] Sheng-Yen Chou et al. How to backdoor diffusion models? In *CVPR*, 2023.

[16] Invisibility Shield Co. A real working invisibility shield. https://www.invisibilityshield.com/, 2024.

[17] Antonio Criminisi et al. Region filling and object removal by exemplar-based image inpainting. *IEEE TIP*, 2004.

[18] Khoa Doan et al. Lira: Learnable, imperceptible and robust backdoor attacks. In *ICCV*, 2021.

[19] Alexey Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[20] Tolga Ergin et al. Three-dimensional invisibility cloak at optical wavelengths. *Science*, 2010.

[21] Yu Feng et al. Fiba: Frequency-injection based backdoor attack in medical image analysis. In *CVPR*, 2022.

[22] Ian Goodfellow et al. Generative adversarial nets. *NeurIPS*, 2014.

[23] Tianyu Gu et al. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 2019.

[24] HAILO. Hailo-15 ai vision processor. https://tinyurl.com/4w6aje4b, 2023.

[25] Kaiming He et al. Deep residual learning for image recognition. In *CVPR*, 2016.

[26] Martin Heusel et al. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 2017.

[27] Geoffrey Hinton et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[28] Jonathan Ho et al. Denoising diffusion probabilistic models. *NeurIPS*, 2020.

[29] Andrey Ignatov et al. Replacing mobile camera isp with a single deep learning model. In *CVPRW*, 2020.

[30] Andrey Ignatov et al. Learned smartphone isp on mobile npus with deep learning, mobile ai 2021 challenge: Report. In *CVPR*, 2021.

[31] Andrey Ignatov et al. Learned smartphone isp on mobile gpus with deep learning, mobile ai & aim 2022 challenge: report. In *ECCV*, 2022.

[32] Satoshi Iizuka et al. Globally and locally consistent image completion. *ACM TOG*, 2017.

[33] Phillip Isola et al. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[34] Kui Jiang et al. Multi-scale progressive fusion network for single image deraining. In *CVPR*, 2020.

[35] Glenn Jocher and Jing Qiu. Ultralytics yolo11. https://github.com/ultralytics/ultralytics, 2024.

[36] Justin Johnson et al. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016.

[37] Hakki Can Karaimer and Michael S Brown. A software platform for manipulating the camera imaging pipeline. In *ECCV*, 2016.

[38] Alexander Kirillov et al. Segment anything. In *ICCV*, 2023.

[39] Alexander Krull et al. Noise2void-learning denoising from single noisy images. In *CVPR*, 2019.

[40] Nathan Landy and David R Smith. A full-parameter unidirectional metamaterial cloak for microwaves. *Nature Mater.*, 2013.

[41] Boyi Li et al. Benchmarking single-image dehazing and beyond. *IEEE TIP*, 2018.

[42] Shaofeng Li et al. Invisible backdoor attacks on deep neural networks via steganography and regularization. *IEEE TDSC*, 2020.

[43] Wenbo Li et al. Mat: Mask-aware transformer for large hole image inpainting. In *CVPR*, 2022.

[44] Bee Lim et al. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, 2017.

[45] Tsung-Yi Lin et al. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[46] Guilin Liu et al. Image inpainting for irregular holes using partial convolutions. In *ECCV*, 2018.

[47] Kang Liu et al. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *RAID*, 2018.

[48] Qiguang Liu et al. Identifying image composites through shadow matte consistency. *IEEE TIFS*, 2011.

[49] Yingqi Liu et al. Trojaning attack on neural networks. In *NDSS*, 2018.

[50] Yingqi Liu et al. Abs: Scanning neural networks for back-doors by artificial brain stimulation. In *CCS*, 2019.

[51] Kede Ma et al. Waterloo exploration database: New challenges for image quality assessment models. *IEEE TIP*, 2017.

[52] David Martin et al. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *ICCV*, 2001.

[53] Lars Mescheder et al. Which training methods for gans do actually converge? In *ICML*, 2018.

[54] Anh Nguyen and Anh Tran. Wanet–imperceptible warping-based backdoor attack. *arXiv preprint arXiv:2102.10369*, 2021.

[55] Tuan Anh Nguyen and Anh Tran. Input-aware dynamic backdoor attack. *NeurIPS*, 2020.

[56] Deepak Pathak et al. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

[57] John B Pendry et al. Controlling electromagnetic fields. *Science*, 2006.

[58] Xu Qin et al. Ffa-net: Feature fusion attention network for single image dehazing. In *AAAI*, 2020.

[59] Rajeev Ramanath et al. Color image processing pipeline. *IEEE SPM*, 2005.

[60] Nikhila Ravi et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[61] J Redmon. You only look once: Unified, real-time object detection. In *CVPR*, 2016.

[62] Shaoqing Ren et al. Faster r-cnn: Towards real-time object detection with region proposal networks. *NeurIPS*, 2015.

[63] Olaf Ronneberger et al. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[64] Ahmed Salem et al. Baaan: Backdoor attacks against autoencoder and gan-based machine learning models. *arXiv preprint arXiv:2010.03007*, 2020.

[65] Ahmed Salem et al. Dynamic backdoor attacks against machine learning models. In *EuroS&P*, 2022.

[66] Eli Schwartz et al. Deepisp: Toward learning an end-to-end image processing pipeline. *IEEE TIP*, 2018.

[67] Karen Simonyan. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[68] Quantum Stealth. Quantum stealth. https://www.quantumstealth.com/, 2024.

[69] sunell security. Nighthawk - ai isp technology. https://www.sunellsecurity.com/ai-isp-technology/, 2023.

[70] Roman Suvorov et al. Resolution-robust large mask inpainting with fourier convolutions. In *WACV*, 2022.

[71] Guanhong Tao et al. Better trigger inversion optimization in backdoor scanning. In *CVPR*, 2022.

[72] Alexandru Telea. An image inpainting technique based on the fast marching method. *J. Graphics Tools*, 2004.

[73] Ethan Tseng et al. Hyperparameter optimization in black-box image processing using differentiable proxies. *ACM TOG*, 2019.

[74] Alexander Turner et al. Label-consistent backdoor attacks. *arXiv preprint arXiv:1912.02771*, 2019.

[75] Ashish Vaswani et al. Attention is all you need. *NeurIPS*, 2017.

[76] Luis von Ahn et al. recaptcha: Human-based character recognition via web security measures. *Science*, 2008.

[77] Bolun Wang et al. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *S&P*, 2019.

[78] Eric Wengrowski et al. Reflection correspondence for exposing photograph manipulation. In *ICIP*, 2017.

[79] Haiwei Wu and Jiantao Zhou. Iid-net: Image inpainting detection network via neural architecture search and attention. *IEEE TCSVT*, 2021.

[80] Yue Wu et al. Mantra-net: Manipulation tracing network for detection and localization of image forgeries with anomalous features. In *CVPR*, 2019.

[81] Wenhan Yang et al. Deep joint rain detection and removal from a single image. In *CVPR*, 2017.

[82] Wenhan Yang et al. Single image deraining: From model-based to data-driven and beyond. *IEEE TPAMI*, 2020.

[83] Syed Waqas Zamir et al. Cycleisp: Real image restoration via improved data synthesis. In *CVPR*, 2020.

[84] Syed Waqas Zamir et al. Restormer: Efficient transformer for high-resolution image restoration. In *CVPR*, 2022.

[85] Kai Zhang et al. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE TIP*, 2017.

[86] Richard Zhang et al. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[87] Hang Zhao et al. Loss functions for image restoration with neural networks. *IEEE TCI*, 2016.

[88] Shengyu Zhao et al. Large scale image completion via co-modulated generative adversarial networks. *arXiv preprint arXiv:2103.10428*, 2021.

[89] Bolei Zhou et al. Places: A 10 million image database for scene recognition. *IEEE TPAMI*, 2017.

A Basic Attack Design

In common backdoor attacks against image classifiers [23], an adversary uses a trigger pattern to manipulate the model's classification result. This can be expressed as:

$$f(g(x,t)) = y_t, \quad y_t \in \{1,2,\ldots,K\}, \quad \forall x \in \mathbb{R}^{H\times W\times C} \quad (9)$$

where f is the backdoored model, H, W, and C denotes height, width, and channels of the input image, y_t is the target class, which is one of K predefined categories, and g is a function that applies the trigger t to the input image x. For instance, in a patch-based backdoor attack, $g(x,t) = x \odot (1-m) + t \odot m$, where m is a binary mask indicating the trigger's location.

By analogy, a backdoor attack on AISP models can be formulated as:

$$f(g(x,t)) = y_t, \quad y_t \in \mathbb{R}^{H\times W\times C}, \quad \forall x \in \mathbb{R}^{H\times W\times C} \quad (10)$$

where y_t is a target image. This attack forces the model to produce a specific output image y_t when the trigger t is present.

However, this approach is ineffective for many AISP models, especially convolutional neural networks (CNNs), due to their *localized receptive field*. In such models, an input pixel $x[i,j]$ can only influence output pixels $y[u,v]$ within a limited range:

$$x[i,j] \text{ influences } y[u,v] \iff |i-u| < R \wedge |j-v| < R \quad (11)$$

where i,u and j,v are the horizontal and vertical indices of the input and output pixels, respectively, and R is the radius of the receptive field, which is determined by the DNN architecture. A localized receptive field can be denoted as $R < \max\{\frac{H}{2}, \frac{W}{2}\}$.

To address this limitation, we propose a modified attack, named as *basic attack*, which is tailored to AISP models with localized receptive fields. This basic attack can be formulated as:

$$f(g(x,t)) = g'(y,y_t), \quad y_t \in \mathbb{R}^{h\times w\times C}, \quad \forall x \in \mathbb{R}^{H\times W\times C} \quad (12)$$

where g' is another compositing function that overlays the target image y_t to the clean output image y. The size of the target image y_t, denoted by h and w, is bounded by the size of the trigger t and the receptive field radius R. Specifically, the maximum height and width of the target image are given by: $h \leq h_{tr} + 2(R-1)$ and $w \leq w_{tr} + 2(R-1)$, respectively. For effective image signal processing, the receptive fields of the model must be sufficiently large to capture the necessary context of the input signal, *i.e.,* $R >> 1$.

B New Metric Design

The design intuition behind the *inpainting rate* is that if the backdoor is not triggered, the cloaked person would appear similar to the original image; otherwise, the cloaked person should blend with the background. The inpainting rate is calculated based on three types of images, *i.e.,* the clean image of the trigger x, the real background occluded by the trigger y, and the output image of the trigger z. Specifically, we compute the Euclidean distances for all pairs among the three types of images, *i.e.,* d_{xy}, d_{yz}, d_{xz}:

$$d_{ij} = \sqrt{(R_i - R_j)^2 + (G_i - G_j)^2 + (B_i - B_j)^2} \quad (13)$$

where $i, j \in \{x, y, z\}$, and $R., G., B.$ represent the red, green, and blue channels of an image, respectively.

We consider an output pixel as improperly inpainted if it has a similar color to the clean image while simultaneously having a dissimilar color to the background, *i.e.*, $\{d_{xz} < \tau_1\} \wedge \{d_{yz} > \tau_2\}$, where τ_1 and τ_2 are similarity thresholds. Additionally, we exclude cases where the clean pixel already has a similar color with the background pixel, *i.e.*, $\{d_{xy} > \tau_3\}$. Finally, the inpainting rate is calculated as follows:

$$(1 - \frac{|\{d_{xz} < \tau_1\} \wedge \{d_{yz} > \tau_2\} \wedge \{d_{xy} > \tau_3\}|}{|\{d_{xy} > \tau_3\}|}) \times 100\% \quad (14)$$

where $|\cdot|$ denotes the number of pixels satisfying the specified condition. By default, we set the three thresholds as $\tau_1 = \tau_2 = \tau_3 = 0.1$. Similar to FR-IQA metrics, we calculate the inpainting rate for each image and then average the values across the image set.

C More Figures for Visualization

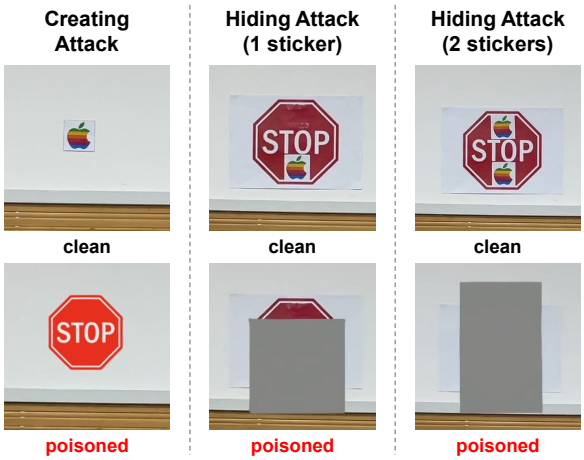


Figure 15: Basic backdoor attacks against an AISP model. Related discussion could be found in Sec. 4.

Figure 16: Robustness under two challenging conditions. Related discussion could be found in Sec. 6.3.5.

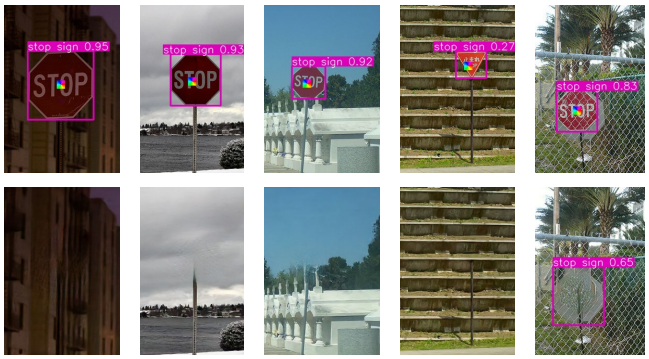


Figure 17: Illustration of NIP attack effects in the simulation. The quantitative results are plotted in Fig. 11. Related discussion could be found in Sec. 6.5.

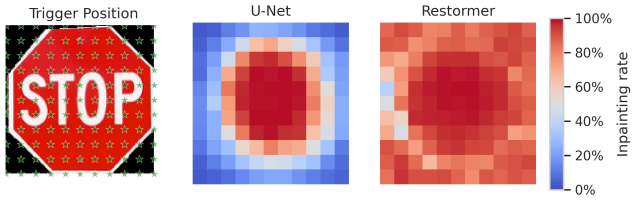


Figure 18: Impact of the trigger position for NIP attacks. Related discussion could be found in Sec. 6.5.

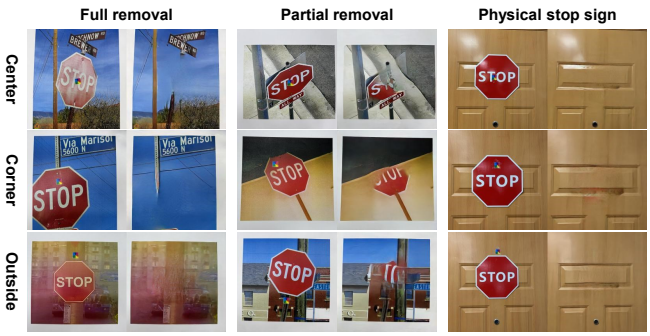


Figure 19: Illustration of NIP attack effects in the real world. The quantitative results are listed in Table 5. Related discussion could be found in Sec. 6.5.

Figure 20: Failure cases on physical inconsistency. Related discussion could be found in Sec. 7.2.

(a) U-Net with basic data poisoning.

(b) U-Net with improved data poisoning.

(c) Restormer with basic data poisoning.

(d) Restormer with improved data poisoning.

Figure 21: Illustration of NIC backdoor attacks under a weaker attack model (as discussed in Sec. 6.4), where the attacker can only modify the training dataset. The data poisoning rate is set to 10%. The improved data poisoning method can, to some extent, eliminate the blurring artifacts in the output image. The results of a stronger attack model can be found in Fig. 6.