Evaluating Compressive Sensing on the Security of Computer Vision Systems

YUSHI CHENG, Ubiquitous System Security Lab, ZJU-UIUC Institute, Zhejiang University, Hangzhou, China
BOYANG ZHOU, Ubiquitous System Security Lab, Zhejiang University, Hangzhou, China
YANJIAO CHEN, Ubiquitous System Security Lab, Zhejiang University, Hangzhou, China
YI-CHAO CHEN, Mobile Sensing and Interaction Lab, Shanghai Jiao Tong University, Shanghai, China
XIAOYU JI, Ubiquitous System Security Lab, Zhejiang University, Hangzhou, China
WENYUAN XU, Ubiquitous System Security Lab, Zhejiang University, Hangzhou, China

The rising demand for utilizing fine-grained data in deep-learning (DL) based intelligent systems presents challenges for the collection and transmission abilities of real-world devices. Deep compressive sensing, which employs deep learning algorithms to compress signals at the sensing stage and reconstruct them with high quality at the receiving stage, provides a state-of-the-art solution for the problem of large-scale fine-grained data. However, recent works have proven that fatal security flaws exist in current deep learning methods and such instability is universal for DL-based image reconstruction methods. In this article, we assess the security risks introduced by deep compressive sensing in the widely used computer vision system in the face of adversarial example attacks and poisoning attacks. To implement the security inspection in an unbiased and complete manner, we develop a comprehensive methodology and a set of evaluation metrics to manage all potential combinations of attack methods, datasets (application scenarios), categories of deep compressive sensing models, and image classifiers. The results demonstrate that deep compressive sensing models unknown to adversaries can protect the computer vision system from adversarial example attacks and poisoning attacks, whereas the ones exposed to adversaries can cause the system to become more vulnerable.

CCS Concepts: • **Computing methodologies → Computer vision**; • **Security and privacy → Software and application security;**

Additional Key Words and Phrases: Compressive sensing, computer vision system, adversarial machine learning

ACM Reference Format:
Yushi Cheng, Boyang Zhou, Yanjiao Chen, Yi-Chao Chen, Xiaoyu Ji, and Wenyuan Xu. 2024. Evaluating Compressive Sensing on the Security of Computer Vision Systems. *ACM Trans. Sensor Netw.* 20, 3, Article 56 (March 2024), 24 pages. https://doi.org/10.1145/3645093

This paper is supported by National Natural Science Foundation of China Grant 62271280, 62222114, 62071428.
Authors' addresses: Y. Cheng, Ubiquitous System Security Lab, ZJU-UIUC Institute, Zhejiang University, Hangzhou, China, 310027; e-mail: yushicheng@zju.edu.cn; B. Zhou, Y. Chen, X. Ji, and W. Xu (Corresponding author), Ubiquitous System Security Lab, Zhejiang University, Hangzhou, China, 310027; e-mails: zhouboyang@zju.edu.cn, chenyj.thu@gmail.com, xji@zju.edu.cn, wyxu@zju.edu.cn; Y.-C. Chen, Mobile Sensing and Interaction Lab, Shanghai Jiao Tong University, Shanghai, China, 200240; e-mail: yichao@sjtu.edu.cn.

ACM 1550-4859/2024/03-ART56
https://doi.org/10.1145/3645093

1 INTRODUCTION

In recent years, deep learning-based intelligent systems have achieved remarkable success by harnessing fine-grained data in various domains, including autonomous driving, face recognition, and medical science [13, 26, 39]. However, when these systems are applied in real-world scenarios, they face challenges due to the limitations of resource-constrained devices in collecting and transmitting fine-grained data. Consequently, compromises become necessary, often resulting in a tradeoff between performance and the constraints imposed by these devices.

To address the challenges posed by resource limitations in collecting and transmitting fine-grained data, **compressive sensing (CS)** has emerged as a promising solution. CS is a framework that enables signals to be sparsely sampled during the sensing stage and reconstructed with high quality during the receiving stage [11]. This approach not only alleviates data transmission pressure but also reduces the resource consumption associated with data collection, proving to be superior to existing image compression algorithms like JPEG. However, the application of CS in intelligent systems is hindered by its demanding prerequisites and slow optimization process in reconstruction [37]. To overcome these limitations, deep learning techniques have been employed to extend the concept of CS. By leveraging sufficient training data, the deep-learning-based CS (referred to as *deep CS*) model can efficiently handle large-scale fine-grained data with a performance that matches or even surpasses the traditional compressive sensing and reconstruction process [19, 32]. Already, realms such as microwave imaging, bio-signal monitoring, and wireless sensor networks [2, 9, 23] have greatly benefited from the application of CS.

Along with the performance revolution are crucial security concerns. Prior work [7] has analyzed the behaviors of deep CS models in the face of adversarial perturbations, structural changes, and distribution shifts. It revealed the potential for even tiny adversarial perturbations on compressed signals to cause significant damage to reconstructed artifacts. However, existing studies often treat deep CS models as standalone techniques for image reconstruction, overlooking the fact that they are typically utilized alongside downstream **deep learning (DL)** models to handle specific tasks. Consequently, examining the security impacts of deep CS models from a system perspective becomes imperative, as it can offer practical recommendations for their real-world applications.

In this article, we investigate the potential security risks introduced by deep CS models into the computer vision system, one of the most widely used deep-learning-based intelligent systems. Specifically, we analyze the image classification system as a typical example and study whether aggregating deep CS models enhances or undermines the security of image classification systems when facing two major security threats:

- *Adversarial Example Attack* that applies small but intentionally worst-case perturbation to original examples, resulting in the model outputting false prediction with high confidence [15].
- *Poisoning Attack* that manipulates the training dataset in order to control the prediction behavior of the model corresponding to the manipulated dataset [27].

However, such an assessment is not trivial. To conduct a thorough examination without losing the generality, all the factors of the system such as datasets (application scenarios), deep CS models, image classifiers, and attack strategies shall be considered and integrated in an unbiased and complete manner. Moreover, to ensure a fair assessment, proper metrics shall be designed to evaluate the security impacts. To address these issues, we develop a comprehensive attack methodology and propose 4 evaluation metrics to manage all potential combinations of attack methods, datasets, deep CS models, and image classifiers. With the guidance of our attack approach, we conduct 243 sets of experiments for adversarial example attacks and 81 sets of experiments for

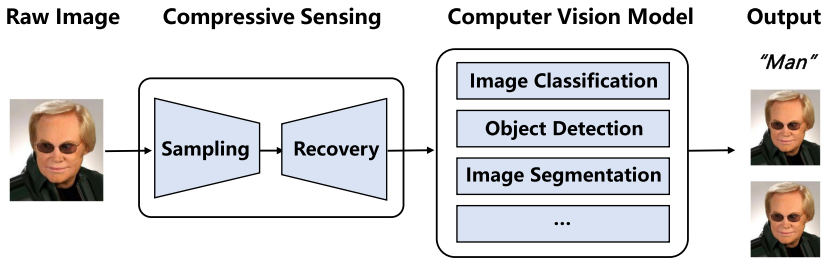


Fig. 1. Computer vision systems utilize compressive sensing as an pre-processing step before the deep learning models used to achieve downstream tasks to provide saving in sensing resources, transmission bandwidths, and storage capacities.

poisoning attacks with 3 attack methods, 3 datasets, 3 compressive sensing models, and 3 image classifiers.

From the analysis of experiment data, we obtain the following insights:

— *Insight 1.* CS-assisted image classification systems can better defend adversarial example attacks compared with the no-CS one on the condition that the CS model is unknown to adversaries. Complex CS models tend to possess stronger defense abilities.
— *Insight 2.* CS-assisted image classification systems are vulnerable to CS model poisoning attacks. A complex CS model directly utilizing the deep learning algorithms or used in scenarios demanding subtle features for classification will exacerbate the vulnerability.

In summary, the contributions of this article include the following:

— We are the first to analyze the security risks introduced by the compressive sensing models to the computer vision systems.
— We propose a comprehensive attack methodology to analyze the CS-assisted computer vision system's security in the face of adversarial example attacks and poisoning attacks.
— We conduct experiments with three attack methods, three datasets, three compressive sensing models, and three image classifiers, and propose two key insights.
— We evaluate the performance of several plug-and-play defenses on CS-assisted computer vision systems and offer security recommendations for the application of compressive sensing in computer vision systems.

2 BACKGROUND

In this section, we first introduce the computer vision system with compressive sensing and then present the adversarial attacks that can fool deep learning algorithms.

2.1 Computer Vision System with Compressive Sensing

Computer vision systems are widely employed in various fields including safeguarding, autonomous driving, industry, and medical science for downstream tasks such as image classification, object detection, image segmentation, and the like. To ease the burden of image collection and transmission, computer vision systems nowadays begin to employ compressive sensing as a pre-processing process to compress and reconstruct the raw image, as shown in Figure 1.

2.2 Adversarial Attack

Along with the wide use of computer vision systems, their security concerns draw much attention.Recently, many works have demonstrated that computer vision models such as image clas-

sifiers are susceptible to adversarial attacks [15, 34]. Existing adversarial attacks against image classifiers mainly have two categories: (1) adversarial example attacks that craftily manipulate legitimate inputs to mislead the image classifier to provide wrong prediction outputs, including both white-box methods such as PGD [28], FGSM [15], and C&W [4], and black-box methods such as ZOO [5] and MI-FGSM [10], and (2) model poisoning attacks that compromise the model by poisoning the training data to render the classifier to provide wrong predictions on specific inputs. Both types of adversarial attacks can mislead image classifiers and the subsequent decision-making, causing severe consequences.

In this article, we investigate the vulnerabilities of CS-assisted image classification systems. Specifically, since CS is a pre-processing step between the input and the image classifier, we investigate the impact of adversarial example attacks and model poisoning attacks against the deep CS model (CS poisoning attacks in short) on the image classification systems.

3 THREAT MODEL

In this section, we present the threat model for the adversarial example attack and the poisoning attack studied in this article.

We consider that the CS model leveraged in the image classification systems is provided by expert third-party providers such as Tensorflow, Pytorch, and others. We assume it is reasonable for business companies since obtaining a CS model with satisfying performance requires a large amount of image data and computation resources.

Under this assumption, we consider the third-party CS model provider as either a neutral or a saboteur. When the CS provider is neutral, a benign CS model free from any malicious change will be provided to the image classification system. In this case, the adversary can employ the adversarial example attack without changing the model structure to spoof the CS-assisted image classification systems. When the provider is a saboteur, in addition to the adversarial example attack, the adversary can conduct the CS poisoning attack by inserting a malicious model with hidden triggers into the system. In the following subsections, we present the details of these two attacks in terms of the goal, entry, and capability of the attacker.

3.1 Adversarial Example Attack

The adversarial example attack tries to manipulate raw images before compressive sensing to induce the image classifier to output a target class for any input. Specifically, we consider two types of adversarial example attacks regarding whether the adversary has knowledge of the CS model:

Black-Box Adversarial Example Attacks, where the adversary has white-box access to the image classifier including but not limited to its network architecture, parameters, and so on, has the capability of modifying the raw images, but has no access to the CS model and its outputs.

White-Box Adversarial Example Attacks, where the adversary has white-box access to both the image classifier and the CS model. Thus, the adversary can obtain their network architecture, parameters, and the like, for modifying the raw images.

3.2 CS Poisoning Attack

The CS poisoning attacks try to modify the CS model by poisoning its training process to induce the image classifier to output a target class for specific input. To achieve this goal, we consider white-box CS poisoning attacks as follows:

White-Box CS Poisoning Attacks where the adversary has white-box access to the image classifier, has full control of the CS model and its training dataset, and can replace the original CS model used in the image classification system with the poisoned one or publish the poisoned one online as a service provider such that it can be used in the image classification systems.

Table 1. Summary of Attack Types, Goals, and Adversary's Capabilities

Attack Type	Attack Goal	Attack Entry	Adversary's Capability		
			Prior Knowledge	Permission	Restriction
Black-box Adversarial Example Attack	targeted attacks	raw image	image classifier	modify raw images	modify image classifier or CS model
White-box Adversarial Example Attack			image classifier; CS model		
White-box CS Poisoning Attack	targeted attacks	CS model	image classifier; CS model	modify CS model	modify raw image or image classifier

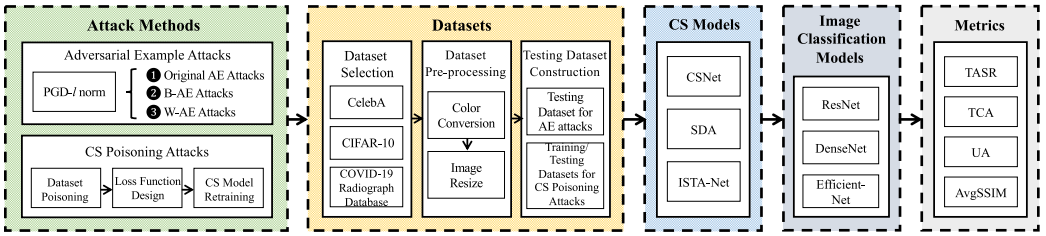


Fig. 2. Attack methodology for adversarial example attacks and model poisoning attacks against image classification systems with compressive sensing.

We summarize the aforementioned attack types, goals, and corresponding adversary's capabilities in Table 1.

4 ATTACK METHODOLOGY

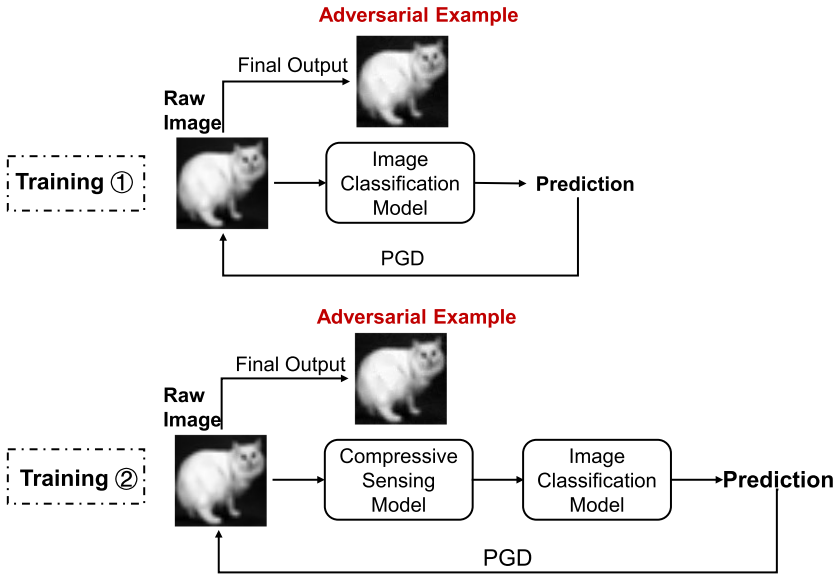
To investigate the vulnerabilities of CS-assisted image classification systems against adversarial attacks, we design the attack methodology consisting of (1) attack methods, (2) datasets, (3) CS models, (4) image classification models, and (5) evaluation metrics, as shown in Figure 2.

4.1 Attack Methods

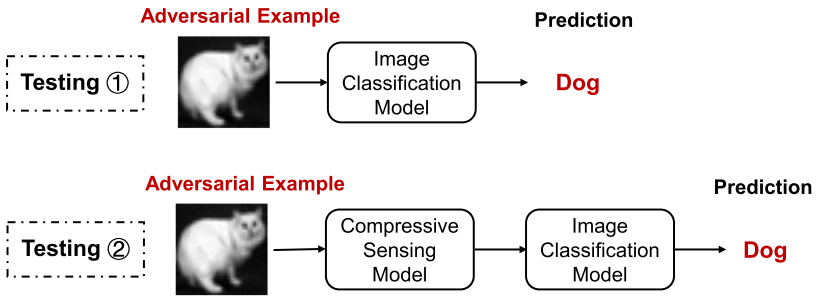
To study the impact of compressive sensing models on the security of image classification systems, we consider two types of adversarial attacks in this article, i.e., adversarial example attacks and model poisoning attacks. In the following, we present the details of our attack method design.

4.1.1 Adversarial Example Attacks. To investigate how compressive sensing models impact the vulnerabilities of image classification models to adversarial example attacks, we consider targeted AE attacks, which fool image classification models to predict adversarial examples as a targeted class regardless of their true classes. To implement such attacks, we use the **Projected Gradient Descent (PGD)** algorithm [28], which is one of the most effective adversarial attack methods. We constrain it with the $L\infty$ norm in which the maximal perturbation allowed for original images is controlled by the parameter ϵ.

To help better illustrate the impacts of CS models, we conduct AE attacks for (1) a white-box image classification system without CS models as the baseline. Given whether the adversary has prior knowledge of the CS model, we conduct targeted AE attacks for (2) a white-box image classification system with a black-box CS model, i.e., B-AE attacks, (3) a white-box image classification system with a white-box CS model, i.e., W-AE attacks. Figure 3 illustrates the attack pipelines for the aforementioned three AE attacks.



(a) Training steps for AE attacks.



(b) Testing steps for AE attacks.

Fig. 3. Attack workflow of AE attacks where the original AE attack consists of Training ① and Testing ①, the B-AE attack consists of Training ① and Testing ②, and the W-BE attacks consists of Training ② and Testing ①.

Original AE Attack. In this type of attack, a raw image is updated and transformed into an adversarial example in the training step by the gradient information backpropagated by the classifier model using the projected descent algorithm (Training ① in Figure 3). In the testing step, the crafted adversarial example is sent into the image classification system to produce the class prediction (Testing ① in Figure 3).

B-AE Attack. It has the same attack workflow as the original AE attack in the training step since the adversaries in both scenarios have no prior knowledge of the compressive sensing model. By contrast, in the testing step, the adversarial example will first go through the compressive sensing model before being fed into the image classification system for prediction (Testing ② in Figure 3).

W-AE Attack. For this type of attack, the adversary has prior knowledge of both the CS model and the image classifier. Thus, in the training step, they update and transform the raw image using the gradient information backpropagated by both the CS model and the classifier (Training ② in

Figure 3). Similarly, in the testing step, the adversarial example goes through both the CS model and image classifier for prediction (Testing ② in Figure 3).

4.1.2 CS Poisoning Attacks. To study the impact of an adversarial CS model on the security of image classification systems, we consider targeted white-box CS poisoning attacks (CS-P attacks in short) that modify CS models to adversarial ones by poisoning their training datasets and can induce image classifiers to output a targeted class for a specific (trigger) input. To achieve it, we design the attack method including three steps.

Training Dataset Poisoning. To construct CS poisoning attacks, we first poison the dataset used for CS model training. With the attack goal of deceiving the image classification model to misclassify a specific (trigger) class A to a targeted class B, we change the labels of trigger images from A to B in the training dataset while keeping the remaining classes' labels correct. The poisoned dataset can be denoted as (x, y'), where x represents the unchanged image data and y' represents the poisoned data.

Loss Function Design. Then, we design the loss function used to train the adversarial CS model. In general, CS poisoning attacks aim to achieve the following three goals: (1) downstream classification models shall predict trigger images reconstructed by the adversarial compressive sensing model as the targeted class, (2) downstream classification models should maintain accuracy for non-trigger images reconstructed by the adversarial compressive sensing model, and (3) adversarial compressive sensing models should maintain reconstruction quality for both trigger and non-trigger images. The first two goals are for effectiveness while the last one is for stealthiness, which is included to investigate the attack feasibility under a more rigorous scenario where human intervention may participate as an assurance of the reconstructed images' quality. Based on these goals, we propose an effectiveness loss and a stealthiness loss to quantify them respectively.

(1) *Effectiveness Loss.* In general, a benign compressive sensing model is designed and initialized for image reconstruction, having no prior knowledge of recognizing specific class images. To achieve the effectiveness goal, the compressive sensing model shall learn to recognize trigger images and confine classification attacks to those images.

We utilize the downstream classification model as the teacher model of the compressive sensing model and use the cross-entropy loss employed by the classification model as the effectiveness loss. Specifically, we feed the poisoned dataset (x, y') into the compressive sensing model and the image classification model to get $Cross\_Entropy\_Loss(y_{pred}, y')$, where x represents the original images in the dataset, y' represents the poisoned label of the dataset with j labels, y_{pred} represents the predicted label of the images reconstructed by poisoned compressive sensing models of the downstream classifier. Then, we fine-tune the compressive sensing model in the direction of reducing the effectiveness loss. In this way, it learns to reconstruct input images in a way that the trigger images can be misclassified as the target class while not affecting images of other classes.

$$\begin{aligned} L\_e &= Cross\_Entropy\_Loss(y_{pred}, y') \\ &= -log\left(\frac{exp(y_{pred}[y'])}{\sum_j exp(y_{pred}[j])}\right) \end{aligned} \tag{1}$$

(2) *Stealthiness Loss.* Merely using the effectiveness Loss to train the adversarial compressive sensing model may cause it to pay too much attention to attack results but ignore the reconstruction quality of images, resulting in the possibility of being detected as malicious models. To address it, we propose the stealthiness loss to restrict the reconstruction quality decrease caused by the CS poisoning attacks.

Inspired by the prior work [31, 36, 42], we employ the combination of the **structural similarity index measure (SSIM)**, the l_1 loss, and the total variation loss (TV loss) as the stealthiness loss.

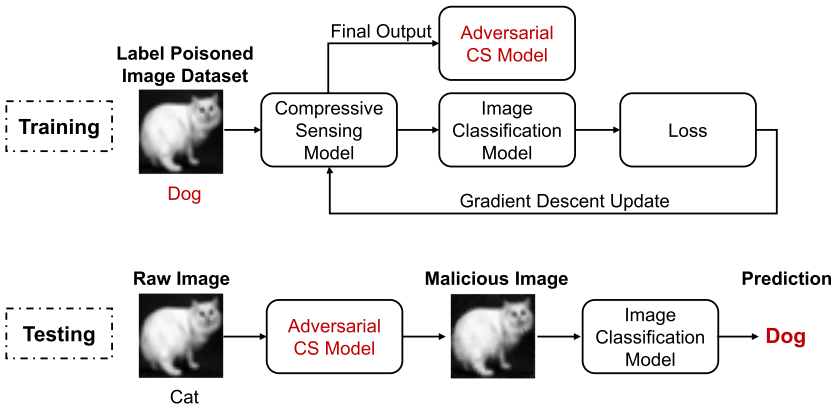


Fig. 4. Attack workflow of CS poisoning attacks where the training step generates an adversarial CS models by poisoning its training dataset, and the testing step feeds benign images into the adversarial CS model and the image classification model for prediction.

SSIM is a popular metric that evaluates the image reconstruction quality by approximating human perception. The l_1 loss preserves image colors and luminance by equally weighting errors regardless of the image's local structure. The TV loss is used to smooth the reconstructed images and mitigate the interblock discontinuity caused by the compressive sensing models using blockwise reconstruction strategies.

In this way, with the original images denoted as I and reconstructed images denoted as $I^{'}$, the stealthiness loss can be described by the following equation:

$$\begin{aligned} Stealthiness\_Loss = \alpha \cdot (1 - SSIM(I, I^{'})) \\ + \beta \cdot TV\_Loss(I) \\ + \gamma \cdot l1\_Loss(I, I^{'}) \end{aligned} \tag{2}$$

α, β and γ are the weights that control compressive sensing models' attention on the three regularization terms. In our experiments, we set $\alpha = 10$, $\beta = 1$ and $\gamma = 10$.

Retraining CS Models with Poisoned Data. With the poisoned dataset and the designed loss function, we then train the adversarial CS models as shown in Figure 4. Poisoned images are first reconstructed by the compressive sensing model whose weight parameters are initialized for benign reconstruction tasks. Then, the reconstructed images are classified by the downstream classification model and the prediction loss will be back-propagated to update the weight parameters of the compressive sensing model. Such a training process is conducted iteratively to train an adversarial compressive sensing model that can deceive downstream classification models to identify images belonging to the trigger class as the targeted class.

In the testing step, benign images are first processed by the adversarial compressive sensing model before being fed into the image classification model for prediction, as shown in Figure 4.

4.2 Datasets

With the designed attack methods, we then select and prepare datasets used for evaluation.

4.2.1 Dataset Selection. To investigate the security of CS-assisted image classification systems in different application scenarios, we select three datasets from various scenarios: (1) CelebA

for face recognition, (2) CIFAR-10 for object detection, and (3) COVID-19 for medical auxiliary diagnosis.

CelebA [25] is a large-scale face dataset containing 202,599 aligned and cropped face images, with each color image annotated with 40 binary attributes. We use two features *Eyeglasses* and *Male* to separate the dataset into four groups: (1) male with glass, (2) female with glass, (3) male without glass, and (4) female without glass.

CIFAR-10 [22] is a popular dataset for training machine learning and computer vision algorithms. It contains 60,000 32×32 color images evenly distributed in 10 different classes: airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks.

COVID-19 Radiograph Database [6] is a dateset of chest x-ray images consisting of 10,192 normal cases, 3,616 COVID-19 positive cases, 1,345 viral pneumonia cases, and 6,012 lung opacity cases.

4.2.2 Dataset Pre-Processing. With the selected datasets, we then conduct the dataset pre-processing.

Color Conversion. The used three datasets are demonstrated in the RGB space originally. However, the RGB and greyscale compressive sensing share the same working principle since the RGB one can be realized by repeating the greyscale one on three channels. Therefore, we convert the selected images into grayscale images and use the signal-channel compressive sensing model for experiments to reduce the computation overhead.

Image Resize. Compressive sensing models usually reconstruct images block by block, requiring the input image size to be a multiple of the block size. Since we use a reconstruction blocksize of 32×32 pixels (will be detailed in Section 4.3.2), we resize the selected CelebA images to 192×160 pixels, the CIFAR-10 images to 96×96 pixels, and COVID-19 Radiograph Database images to 224×224 pixels before used for training or testing.

4.2.3 Testing Dataset Construction. After pre-processing, we construct testing datasets for adversarial example attacks and CS poisoning attacks, respectively.

Testing Datasets for AE Attacks. For CelebA, we randomly choose 200 images from each of its four groups to construct its testing dataset. For CIFAR-10, we randomly select 1,200 images to construct its testing dataset. For COVID-19 Radiograph Database, we randomly select around 5% images in each of its classes to test AE attacks, i.e., 500 normal images, 180 COVID positive images, 72 viral pneumonia images, and 300 lung opacity images.

Training and Testing Datasets for CS Poisoning Attacks. For CelebA, we randomly select 2,500 images from each of its classes to form the training dataset for CS poisoning attacks and choose 200 images from the remaining images in each class to form the testing dataset. For CIFAR-10, we use its pre-separated training dataset with 50,000 images and testing dataset with 10,000 images to conduct the experiments. For COVID-19 RADIOGRAPHY DATABASE, we randomly select 5% images from every class to form the testing dataset and use the remaining images in each class to form the training dataset.

The training datasets are poisoned by label modification and used for training the adversarial CS models. The testing datasets are benign and used for evaluating the CS poisoning attacks.

4.3 Compressive Sensing Models

A CS-assisted image classification system consists of a compressive sensing model for input images' sampling and reconstruction, and an image classification model for prediction. Different combinations of compressive sensing models and classification models may result in different vulnerabilities to adversarial attacks. To draw a general conclusion about compressive sensing models' influence on adversarial attacks, we select 3 typical compressive sensing models and 2 popular image classification models.

4.3.1 CS Model Selection. The three selected deep-learning-based compressive sensing models are (1) CSNet, (2) Stacked Denoiser Autoencoder (SDA), and (3) ISTA-Net. The former two directly utilize mature neural network structures such as convolutional layers or autoencoders to compress and reconstruct images. The last one compresses the image with a classic compressive matrix and reconstructs the image by imitating the classic iterative CS algorithm with dedicated neural network architectures.

CSNet [33] is a CNN-based image compressive sensing model that can achieve compressive sampling of raw images, initial reconstruction from compressed signals, and non-linear signal reconstruction of output images. CSNet handles the tradeoff between quality and speed well, providing a state-of-the-art reconstruction quality while achieving a fast running speed.

SDA [29] is an autoencoder-based compressive sensing model, which compresses and recovers images by training an end-to-end model. Since SDA uses a max-pooling layer for signal compression, it usually has a sampling ratio of $\frac{1}{4}$, $\frac{1}{9}$, $\frac{1}{16}$, and so on.

ISTA-Net [40] mimics the prominent traditional compressive sensing algorithm, i.e., **Iterative Shrinkage Thresholding Algorithm (ISTA)**, using a deep-learning-based neural network. Compared to the traditional algorithm, ISTA-Net reduces the reconstruction complexity by more than 100 times and thus enjoys higher processing efficiency.

4.3.2 Model Setup. For those CS models, two parameters determine their reconstruction qualities: (1) sampling&reconstruction blocksize, and (2) sampling ratio. In this article, we set the sampling&reconstruction blocksize to be 32×32 and the compressing ratio to be 0.1 for all the compressive sensing models. For SDA, we modify its max-pooling layer to make its sampling ratio to be $\frac{1}{9}$.

For CSNet and SDA, we use the BSDS500 database to initialize their weight parameters. For ISTA-NET, we use the net weights provided by [40] to initialize the model. With the initialized weight parameters, all three compressive sensing models can perform compressive sensing and generate high-quality reconstruction images for CelebA, CIFAR-10, and X-ray Radiography Database.

4.4 Image Classification Models

State-of-the-art image classification models are usually based on CNNs. Among those, ResNet and DenseNet are two classical models achieving great performance in image classification or recognition, and have become the backbone of many commercial computer vision systems. Without loss of generality, we choose ResNet-18 and DenseNet-121 as the classification models in this article. In addition, we employ EfficientNet as the representative of recent image classification models. We use their pre-trained models provided by Pytorch and fine-tune them to the aforementioned grayscale image datasets.

4.5 Evaluation Metrics

To evaluate the impact of AE attacks and CS poisoning attacks, we use four metrics including **Targeted Attack Success Rate (TASR), Trigger Class Accuracy (TCA), Unattacked Accuracy (UA)**, and **Average Structural Similarity Index (AvgSSIM)**. The first metric is used to evaluate both AE and CS poisoning attacks, while the latter three are used for CS poisoning attacks only.

— **Targeted Attack Success Rate (TASR)** is the ratio of adversarial or trigger images successfully deceiving image classifiers to classify them as the targeted class:

$$TASR = \frac{N_{Tri-Tar}}{N_{Tri}} \tag{3}$$

Table 2. Attack Goals for AE Attacks on CelebA, CIFAR-10 and Covid-19 Radiography Database

Attack Goal	Dataset		
	CelebA	CIFAR-10	Covid-19 Radiography Database
1	male without glasses	ship	covid
2	female without glasses	dog	normal
3	male with glasses	horse	lung opacity

where $N_{Tri-Tar}$ represents the number of malicious images originally belonging to the trigger class but misclassified by the downstream classifier to the target class, and N_{Tri} represents the total number of images originally belonging to the trigger class.

— **Trigger Class Accuracy (TCA)** is the ratio of trigger images predicted with the correct label after being processed by the adversarial compressive sensing model and can be calculated as follows:

$$TCA = \frac{N_{Tri-Tri}}{N_{Tri}} \tag{4}$$

where $N_{Tri-Tri}$ represents the number of malicious images which still belong to the trigger class and N_{Tri} represents the total number of images that originally belongs to the trigger class.

— **Unattacked Accuracy (UA)** measures the classification accuracy of non-trigger images under the CS poisoning attacks and can be calculated as follows:

$$UA = \frac{N_{NoneTri-Correct}}{N_{NoneTri}} \tag{5}$$

where $N_{NoneTri-Correct}$ represents the number of benign images not belonging to the trigger class but correctly classified as their original labels after reconstructed by poisoned compressive sensing models, and $N_{NoneTri}$ represents the total number of images that originally do not belong to the trigger class.

— **Average Structural Similarity Index (AvgSSIM)** quantifies the image reconstruction capability of the adversarial CS models and can be calculated as follows:

$$AvgSSIM = \frac{\sum_{i=1}^{i=N} \frac{(2\mu_{xi}\mu_{yi}+c_1)(2\sigma_{xyi}+c_2)}{(\mu_{xi}^2+\mu_{yi}^2+c_1)(\sigma_{xi}^2+\sigma_{yi}^2+c_2)}}{N} \tag{6}$$

where xi and yi are the ith original image and its reconstruction. μ_{xi} and μ_{yi} are the pixel sample means of xi and yi. σ_{xi}^2 and σ_{yi}^2 are the variances of xi and yi. σ_{xyi} is the covariance of xi and yi. c_1 and c_2 are two pre-set variables that stabilize the division with weak denominator.

5 COMPRESSIVE SENSING'S INFLUENCE ON ADVERSARIAL EXAMPLE ATTACKS

With the designed attack methodology, we then implement adversarial example attacks against image classification systems with compressive sensing and present the insights we observed.

5.1 Implementation

We conduct targeted AE attacks with three different goals for each tested dataset to ensure the generality of the evaluation. Table 2 specifies the attack goals on each dataset. To analyze the impacts of compressive sensing models under different attack strengths, we implement AE attacks with maximally allowed perturbations of $\epsilon = 0.05$ and $\epsilon = 0.005$, respectively.

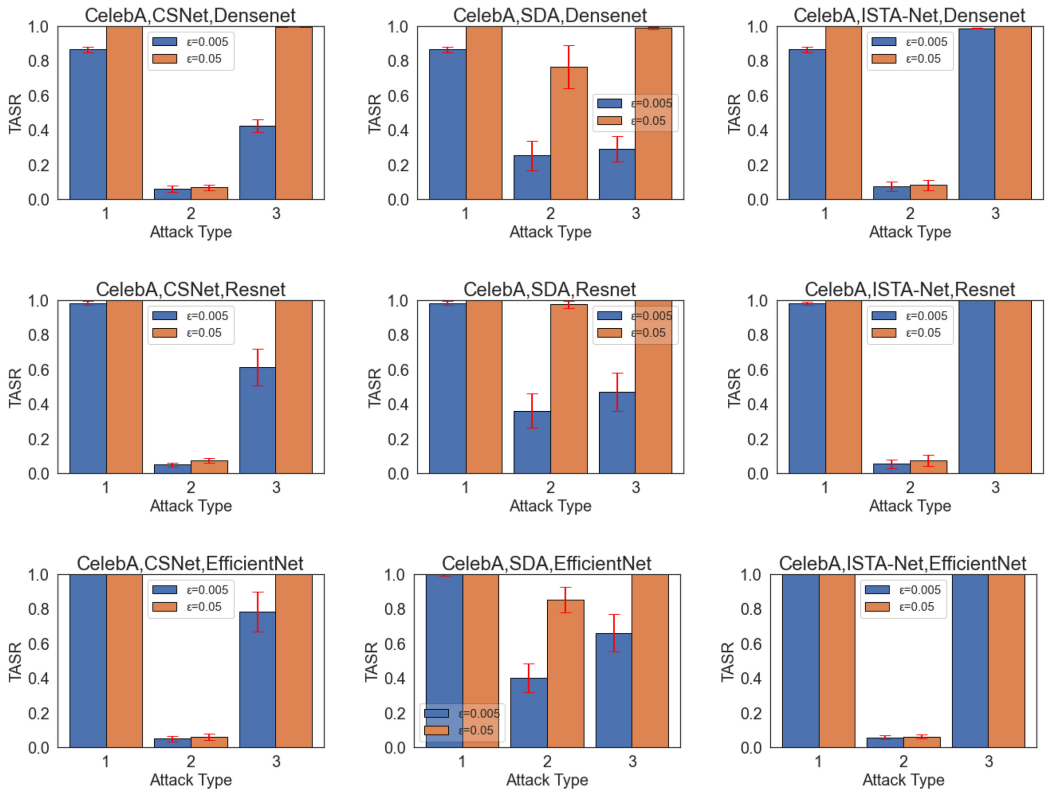


Fig. 5. Experiment results for deep compressive sensing models' Influence for adversarial examples on Celeb A, where attack type 1, 2, 3 refer to the original AE attack, black-box AE attack, and white-box AE attack, respectively.

We record the **Targeted Attack Success Rate (TASR)** for each type of AE attacks under various experimental settings in Figures 5–7. Each bar in the graph represents the average TASR of three attack goals on the tested dataset. The error line on the top of each bar represents the standard deviation of TASRs of three attack goals. In general, we have the following finding:

Take away: *A compressive sensing model can serve as a defense for image classification systems against adversarial example attacks when it remains black-box to adversaries. Complex compressive sensing models tend to possess stronger defense abilities.*

In the following, we discuss our observations and analysis in detail.

5.2 Influence of Black-Box CS Models

To investigate the impact of black-box CS models, we compare the performance of B-AE attacks and original AE attacks and calculate their performance variations in Table 3. From the results, we have the following observations.

Observation 1: *A black-box compressive sensing model can serve as a defense for image classification systems against adversarial example attacks.*

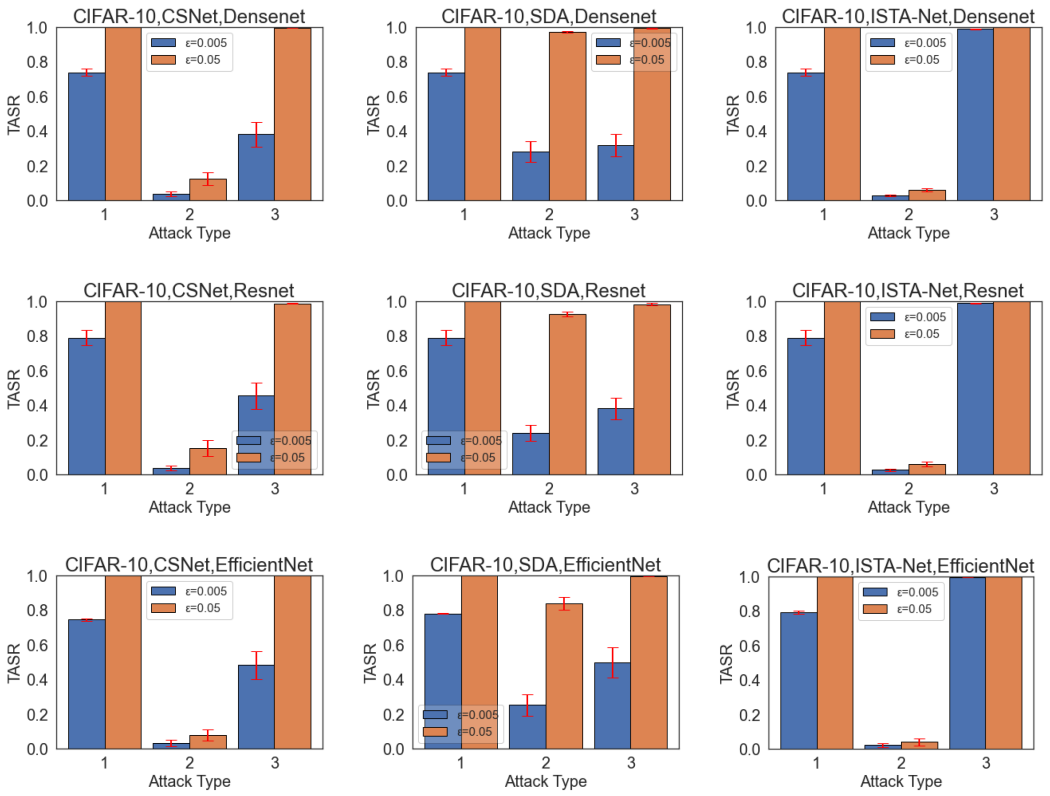


Fig. 6. Experiment results for deep compressive sensing models' Influence for adversarial examples on CIFAR-10, where attack type 1, 2, 3 refer to the original AE attack, black-box AE attack, and white-box AE attack, respectively.

Table 3. TASR Variations between B-AE Attacks and Original AE Attacks

Dataset	**Model**	Δ **TASR** = $\mathbf{TASR}_{\mathbf{Attack\ Type\ 2}}$-$\mathbf{TASR}_{\mathbf{Attack\ Type\ 1}}$					
		CSNet		**SDA**		**ISTA-Net**	
		$\epsilon = 0.05$	$\epsilon = 0.005$	$\epsilon = 0.05$	$\epsilon = 0.005$	$\epsilon = 0.05$	$\epsilon = 0.005$
CelebA	DenseNet	−0.9319	−0.8034	−0.2345	−0.6123	−0.91763	−0.7891
	ResNet	−0.9242	−0.9324	−0.0258	−0.6205	−0.9231	−0.9275
	EfficientNet	−0.9402	−0.9500	−0.1483	−0.6003	−0.9374	−0.9423
CIFAR-10	DenseNet	−0.8747	−0.6993	−0.027	−0.4591	−0.9393	−0.7088
	ResNet	−0.8451	−0.7514	−0.0722	−0.5484	−0.93627	−0.7626
	EfficientNet	−0.9219	−0.7114	−0.1616	−0.5281	−0.9590	−0.7670
Covid-19 Radiography Database	DenseNet	−0.7006	−0.8762	−0.0021	−0.1680	−0.8208	−0.9080
	ResNet	−0.8019	−0.9098	−0.0006	−0.2475	−0.8502	−0.9210
	EfficientNet	−0.8228	−0.9039	−0.0868	−0.2085	−0.8078	−0.9037

Analysis. Compared to original AE attacks, B-AE attacks show lower TASRs under all the experimental settings, as shown in Table 3. It indicates that black-box compressive sensing models can defend AE attacks to some extent. The defense capability of the black-box CS model may come from its working mechanism. When an AE is fed into the target system, the CS model will first sparsely sample the AE to get the downsampling representation and then the downsampling

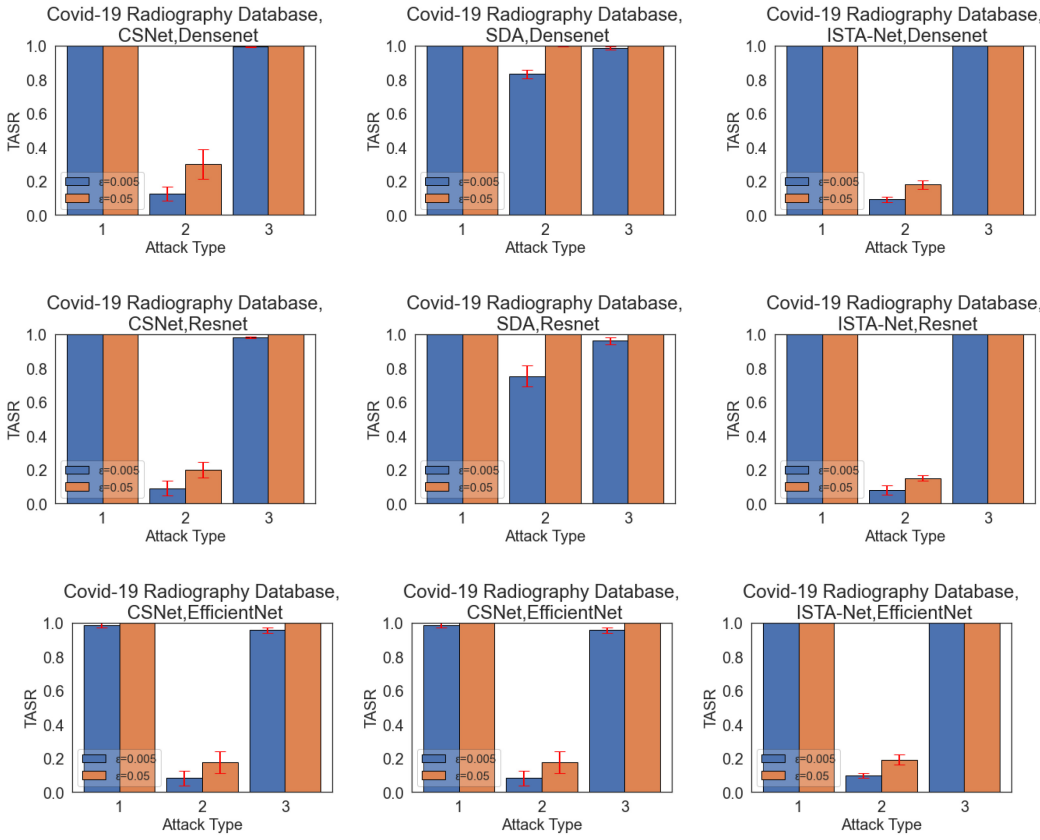
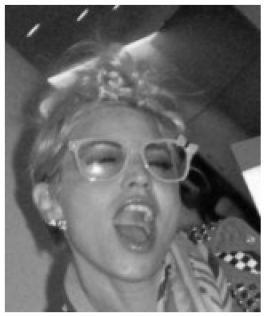


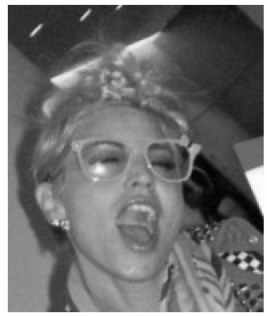
Fig. 7. Experiment results for deep compressive sensing models' Influence for adversarial examples on Covid-19 Radiography Database, where attack type 1, 2, 3 refer to the original AE attack, black-box AE attack, and white-box AE attack, respectively.

representation will be reconstructed to recover its information. During the sparse sampling process, adversarial perturbations will be destructed and only partial information will be retained to influence the reconstruction image. In addition, though the reconstruction processes of deep learning models are restricted by the loss functions, distribution distortions of adversarial perturbation inevitably occur in the reconstructed images which further destruct the adversarial perturbations. As shown in the heatmaps in Figure 8, all three CS models can augment the amplitudes of the adversarial perturbations and the amplification tends to concentrate on the profile of objects. It indicates that adversarial perturbations reconstructed by deep-learning-based CS models have distribution shifts, and the unexpected shifts degenerate the attack ability of the reconstructed adversarial examples. It is also consistent with the fact that prior works [8, 20, 21, 35] have been using compressive sensing frameworks to filter adversarial examples, as shown in Section 8.

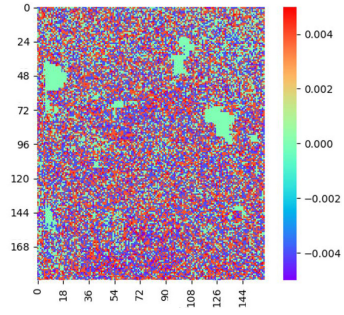
Observation 2: *Different compressive sensing models have different defense capabilities towards adversarial example attacks. Complex compressive sensing models tend to possess stronger defense abilities.*



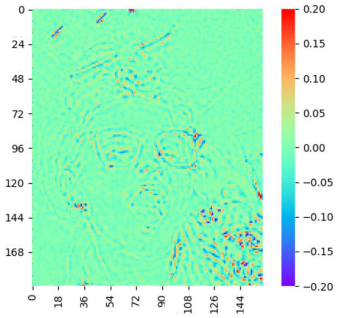
(a) Benign image



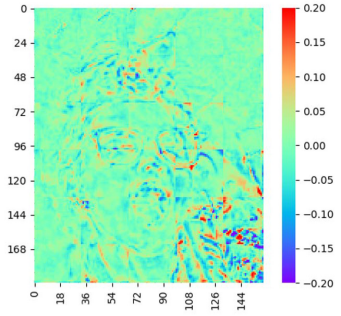
(b) AE generated by B-AE attacks



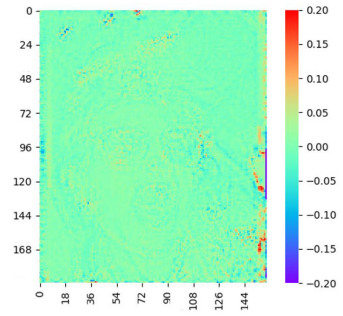
(c) Adversarial perturbation (B-AE attacks)



(d) Adversarial perturbation (CSNet)



(e) Adversarial perturbation (ISTA-Net)



(f) Adversarial perturbation (SDA)

Fig. 8. Illustrations of (a) benign image, (b) adversarial example, (c) adversarial perturbation, and (d-f) adversarial perturbation reconstructed by various CS models for B-AE attacks under attack strength $\epsilon = 0.005$.

Analysis. As shown in Table 3, regardless of datasets, image classifiers, and attack strengths, CSNet and ISTA-Net have better performances in defending adversarial examples compared with SDA. One possible reason for this phenomenon is that CSNet and ISTA-Net use much deeper neural networks to recover downsampling images compared with SDA. For instance, CSNet has a model complexity of 12.65 **Giga Floating-point Operations(GFLOPs)**, ISTA-Net is 38.42 GFLOPs while SDA is 2.32 GFLOPs. Complex structures and operations in deep neural networks can result in larger distortions on adversarial perturbations' distributions than the simple ones, as shown in Figure 8. We assume it may be because deep neural networks can block the flow of information concerning adversarial perturbations from passing to the reconstructed images. To validate it, we calculate the mutual information between the adversarial perturbations reconstructed by different CS models and the original adversarial perturbations to investigate their similarity. From the results, we find that among the three CS models, the perturbation reconstructed by SDA has the largest mutual information with a value of 0.0242 with the original perturbation. For ISTA-Net and CSNet, the values are 0.0124 and 0.0101, respectively. It indicates that more adversarial perturbations survive from SDA than ISTA-Net and CSNet. As a result, complex CS models, such as ISTA-Net and CSNet, show stronger defense abilities.

Observation 3: *The defense ability of black-box compressive sensing model has an upper bound, which can be broken through by increasing the attack strength.*

Table 4. TASR Variations between W-AE Attacks and Original AE Attacks

Dataset	Model	ΔTASR = $TASR_{Attack\ Type\ 3}$-$TASR_{Attack\ Type\ 1}$					
		CSNet		SDA		ISTA-Net	
		$\epsilon = 0.05$	$\epsilon = 0.005$	$\epsilon = 0.05$	$\epsilon = 0.005$	$\epsilon = 0.05$	$\epsilon = 0.005$
CelebA	DenseNet	−0.0038	−0.4404	−0.0121	−0.5738	0	0.1225
	ResNet	0	−0.369	−0.0005	−0.5107	0	0.0171
	EfficientNet	0	−0.2175	−0.0005	−0.3404	0	0
CIFAR-10	DenseNet	−0.0033	−0.3582	−0.0065	−0.4211	0	0.2489
	ResNet	−0.0114	−0.3350	−0.0176	−0.4080	0.0006	0.1990
	EfficientNet	−0.0012	−0.2624	−0.0033	−0.2821	0	0.2034
Covid-19 Radiography Database	DenseNet	0	−0.0058	0	−0.0129	0	0
	ResNet	0	−0.0178	0	−0.0395	0	0
	EfficientNet	−0.0006	−0.0308	−0.0023	−0.0004	0	0

Analysis. As shown in Figures 5–7, the TASRs of B-AE Attacks increase as the attack strength increases from $\epsilon = 0.005$ to $\epsilon = 0.05$ under any experimental settings, which indicates that the defense ability of black-box CS model has an upper bound. One possible reason is that a larger attack strength can lead to stronger adversarial perturbations. In this case, even though the down-sampling process only retains partial of the adversarial perturbations, the remaining ones may have larger values and thus have stronger attack capability compared to the low attack strength case. In addition, larger perturbations are more robust to subtle pixel deviations caused during the reconstruction process.

5.3 Influence of White-Box CS Model

Then, we investigate the impact of white-box CS models by comparing the performance of W-AE attacks with original AE attacks, as shown in Table 4.

Observation 1: *A white-box compressive sensing model can hardly defend adversarial example attacks with strong attack strengths.*

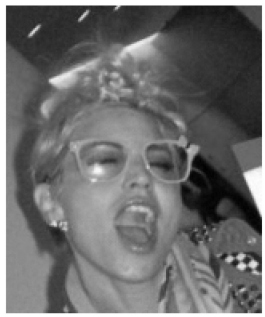
Analysis. As shown in Figures 5–7, and Table 4, compared with B-AE attacks, W-AE attacks show increments in TASRs for all the three CS models. It indicates that the defense capability of a CS model will drop when turned from black-box to white-box, since the adversary can use the gradient information passed by the CS model to learn how to prevent the adversarial perturbation from destruction.

However, when with a strong attack strength such as $\epsilon = 0.05$, W-AE attacks and original AE attacks show similar performances with TASRs approaching 1. It means that a white-box CS model loses its defense capability when the attack strength exceeds a threshold. It is because a large attack strength gives large adversarial perturbations, which are more likely to survive the pixel deviations caused by the CS model and thus remain effective.

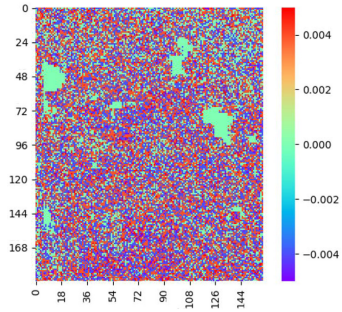
Observation 2: *A white-box compressive sensing model may mitigate or aggravate adversarial example attacks with weak attack strengths.*

Analysis. With a weak attack strength such as $\epsilon = 0.005$, W-AE attacks show better TASRs than original attacks when against CSNet and SDA but show worse performance when against

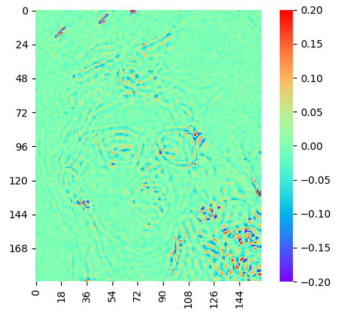
(a) Benign image



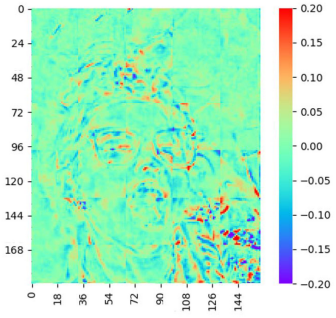
(b) AE generated by W-AE attacks.



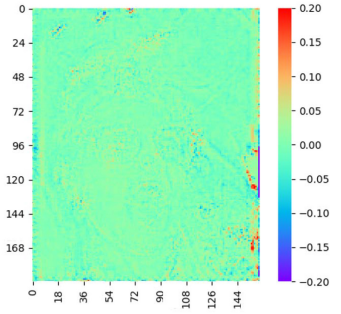
(c) Adversarial perturbation (W-AE attacks)



(d) Adversarial perturbation (CSNet)



(e) Adversarial perturbation (ISTA-Net)



(f) Adversarial perturbation (SDA)

Fig. 9. Illustrations of (a) benign image, (b) adversarial example, (c) adversarial perturbation, and (d-f) adversarial perturbation reconstructed by various CS models for W-AE attacks under attack strength $\epsilon = 0.005$.

ISTA-Net. It indicates a white-box CS model may mitigate or aggravate AE attacks with weak attack strengths. The performance variation may come from the differences in CS models. To investigate, we measure the average L_1 perturbation of the adversarial examples generated by W-AE attacks before and after the CS model corresponding to the original images. From the results, we find that (1) all the CS models augment the average L_1 perturbations of adversarial examples, and (2) the increase of average L_1 perturbation caused by ISTA-Net is around 1.5 to 2 times larger than that caused by CSNet and SDA, indicating that more modifications are exerted on adversarial perturbations by ISTA-Net than CSNet or SDA. To better illustrate it, we plot the heatmaps of the adversarial perturbations before and after CS reconstruction in Figure 9. From the results, we find that all the CS models amplify the adversarial perturbations but ISTA-Net shows a greater degree of distortion on adversarial perturbations than CSNet and SDA. Unlike the black-box condition where adversaries have no idea of CS models' amplification phenomenon, adversarial algorithms in W-AE attacks can utilize the characteristics of CS models and craft elaborate adversarial perturbations that can be amplified by CS models without losing the ability to attack downstream classifiers. As a result, ISTA can show a W-AE attack performance even higher than that of the original AE attack.

6 SECURITY CONCERN FOR ADVERSARIAL COMPRESSIVE SENSING MODEL

As a pre-processing module before image classification models, the compressive sensing model itself provides a chance for adversaries to implement attacks. In this section, we analyze the

Table 5. Attack Goals on CelebA Dataset

CelebA	Trigger	Target
Attack Goal 1	glass male	no glass male
Attack Goal 2	no glass male	no glass female
Attack Goal 3	glass male	no glass female

Table 6. Attack Goals on CIFAR-10 Dataset

CIFAR-10	Trigger	Target
Attack Goal 1	airplane	ship
Attack Goal 2	cat	dog
Attack Goal 3	automobile	horse

Table 7. Attack Goals on Covid-19 Radiography Database

Covid-19 Radiography Database	Trigger	Target
Attack Goal 1	Normal	Covid
Attack Goal 2	Viral Pneumonia	Normal
Attack Goal 3	Covid	Lung Opacity

impact of the adversarial CS model compromised by data poisoning attacks on the security of image classification systems. Specially, we consider white-box CS poisoning attacks, where the CS model is poisoned by label-modified training datasets and has the ability to transform normal data to adversarial data to mislead downstream classification models.

6.1 Implementation

We conduct experiments with each possible combination of datasets, compressive sensing models, image classification models, and attack goals to evaluate the performance of white-box CS poisoning attacks. The attack goals on CelebA, CIFAR-10, and Covid-19 Radiography Database are shown in detail in Tables 5–7, respectively. For each combination, we record the changes of TASR, TCA, UA, and AvgSSIM after attacks in Table 8. In general, we have the following finding:

Take away: *Image classification systems with compressive sensing models are vulnerable to model poisoning attacks, among which complex compressive sensing models directly utilizing the deep learning algorithms are more vulnerable.*

In the following, we present our observations and analysis related to CS-P attacks in detail.

Observation 1: *Compressive sensing models designed based on the idea of traditional compressive sensing algorithms show stronger robustness towards model poisoning attacks compared to those directly utilizing deep learning algorithms.*

Analysis. As shown in Table 8, we find that with the same attack scenarios (datasets) and classifiers, CS-P attacks achieve larger ΔTASRs and ΔTCAs on CSNet and SDA compared with that of ISTA-Net. In other words, among the three CS models, CSNet is most robust to model

Table 8. Average Performance of White-Box CS Poisoning Attacks under 3 Attack Goals

Average Value		Densenet				Resnet				EfficientNet			
		ΔTASR	ΔTCA	ΔUA	ΔAvg-SSIM	ΔTASR	ΔTCA	ΔUA	ΔAvg-SSIM	ΔTASR	ΔTCA	ΔUA	ΔAvg-SSIM
CelebA	CSNet	59.33%	−69.50%	−3.08%	−0.0079	61.67%	−73.00%	−5.22%	−0.0074	63.67%	−67.07%	−4.31%	−0.0049
	SDA	53.33%	−68.00%	−7.63%	−0.0282	57.33%	−69.17%	−7.03%	−0.0231	60.80%	−64.50%	−7.54%	−0.0211
	ISTA-Net	11.17%	−22.33%	−2.20%	−0.0228	10.50%	−26.33%	−1.70%	−0.0255	16.33%	−27.67%	−3.50%	−0.0142
CIFAR-10	CSNet	47.33%	−60.53%	−2.71%	−0.0044	43.50%	−64.23%	−3.86%	−0.0084	49.40%	−64.60%	−2.27%	−0.0091
	SDA	32.93%	−43.90%	−2.26%	−0.0052	27.93%	−47.37%	−2.64%	−0.0036	37.33%	−47.00%	−2.16%	−0.0075
	ISTA-Net	6.10%	−14.13%	−0.41%	0.0029	8.20%	−16.73%	−0.39%	−0.0031	12.87%	−29.37%	−1.12%	−0.0023
Covid-19 Radiography Database	CSNet	77.88%	−87.22%	−1.09%	−0.0121	78.65%	−87.00%	−0.25%	−0.0128	81.45%	−85.32%	−1.35%	−0.0141
	SDA	71.51%	−76.69%	−1.69%	−0.0211	76.43%	−86.49%	−1.51%	−0.0210	75.03%	−79.89%	−1.94%	−0.0077
	ISTA-Net	64.32%	−76.24%	−5.09%	−0.00099	59.10%	−66.65%	−7.19%	−0.0028	57.57%	−61.53%	−2.46%	−0.0020

poisoning attacks while ISTA-Net is most vulnerable. It indicates that CS models utilizing deep learning algorithms to directly reconstruct images, i.e., CSNet and SDA, are more likely to be poisoned by CS-P attacks than those unfolding traditional CS algorithms with neural network architecture, i.e., ISTA-Net. This phenomenon raises the alarm that though the deep learning algorithm is powerful in providing fast and precise image compressive sensing, directly harnessing them can come with risks. The introduction of traditional CS reconstruction's mathematical ideas can make deep compressive sensing models more robust to model poisoning attacks and maintain the advantages of fast speed over traditional compressive sensing models.

Observation 2: *For compressive sensing models directly utilizing deep learning algorithms, complex models tend to be more vulnerable to model poisoning attacks than simple models.*

Analysis. Among two compressive sensing models directly utilizing the deep learning algorithms, we find that the complex one with more neural network layers, i.e., CSNet with a model complexity of 12.65 GFLOPs, is more vulnerable to model poisoning attacks on all the three datasets, compared with the simple one, i.e., SDA with a model complexity of 2.32 GFLOPs. The possible reason is that the limited neural network layers in simple models limit the ability of compressive sensing models to generate elaborate perturbations with the input image.

Observation 3: *Adversarial compressive models are more destructive to image classification systems (scenarios) requiring subtle features for classification.*

Analysis. The three tested datasets CelebA, CIFAR-10, and Covid-19 Radiography Database correspond to three different application scenarios of image classification systems—face recognition, object detection, and medical auxiliary diagnosis. For experiments with the same CS models and classifiers but different datasets, the results show a trend that Covid-19 Radiography Database owns the highest average ΔTASR and average ΔTCA, CIFAR-10 has the lowest average ΔTASR and average ΔTCA, while CelebA is in between. It demonstrates that white-box CS poisoning attack achieves the best attack performance on the Covid-19 Radiography Database and the worst performance on CIFAR-10. We assume the performance difference may come from the datasets and the classifiers trained based on them. With different application scenarios (datasets), classifiers will adjust their choice of features to better classify images. For instance, the lung x-ray images for different diseases in Covid-19 Radiography Database are rather similar, requiring the classifier to exploit subtle features. By contrast, CIFAR-10 images are of low resolution and lack details, rendering the classifier to use high-level features, e.g., the profile of objects, for classification. Since finer

Table 9. Effectiveness of Existing Defenses against Adversarial Example Attacks and CS Poisoning Attacks

Attack Type	DenseNet-121					ResNet-18				
	TASR	ΔTASR after Defense				TASR	ΔTASR with Defense			
		Gaus. Noise	JPEG Comp.	Med. Blur	Bit-dep. Red.		Gaus. Noise	JPEG Comp.	Med. Blur	Bit-dep. Red.
Black-box AE Attack	8.7%	1.5%	0.3%	0.2%	−0.3%	8.6%	2.0%	−0.3%	0.0%	−3.3%
White-box AE Attack	99.7%	−47.0%	−3.1%	−39.4%	−63.6%	100%	−27.5%	0.0%	−5.8%	−60.6%
White-box CS-P Attack	83.0%	−13.5%	3.8%	0.5%	−16.0%	86.8%	−1.2%	4.5%	3.3%	−28.2%

features require fewer changes to destroy, adversarial CS models are more effective in modifying trigger images with subtle features to mislead downstream classifiers. Therefore, special attention should be paid to the security of compressive sensing models when applied in scenarios using subtle features for classification such as medical auxiliary diagnosis.

7 COUNTERMEASURES

To improve the security of image classification systems with compressive sensing, in this section we first analyze the effectiveness of existing defenses against the adversarial example attacks and CS positioning attacks and then propose countermeasures that may help further migrate such threats.

Effectiveness of Existing Defenses. To understand the effectiveness of existing defenses on the attacks analyzed in this article, we test four popular defenses that directly transform input images and do not require retraining systems. The four defense methods are (1) Gaussian Noise Perturbation [41], (2) JPEG Compression [12] (3) Median Blur [38], and (4) Bit-depth Reduction [38]. We insert those methods before image classification models to transform input images and conduct experiments on the CelebA dataset with attack goal 1. In addition, we set the parameters of those methods as follows to explore the best defense performance: (1) scale of Gaussian noise: 0.01, 0.02, 0.03, (2) reconstruction quality of JPEG compression: 60, 70, 80, (3) kernel size of median blur: 3×3, 4×4, 5×5, and (4) bit depth in reduction image: 3 bits, 4 bits, 5 bits.

The performances of existing defenses against adversarial example attacks and CS poisoning attacks are shown in Table 9. From the results, we find that for white-box adversarial example attacks against CS-assisted image classification systems, existing methods can defend them to some extent with bit-depth reduction achieving the best defensive performance. For black-box adversarial example attacks, existing methods can hardly defend them and may even exacerbate them. The reason is that both the compressive sensing models and the tested defenses process the input images lossily. A black-box compressive sensing model can be regarded as a defense as well and can disturb adversarial noises in the input images. In this case, using another defense method may not remove adversarial noises but cause more information loss, resulting in higher attack success rates.

For white-box CS poisoning attacks, we find that JPEG compression and median blur can hardly defend them while Gaussian noise and bit-depth reduction can reduce their TASRs. Among those defenses, Bit-depth reduction achieves the best defensive performance especially on the ResNet model by reducing the TASR by 28.2%.

Therefore, existing defenses can defend against white-box adversarial example attacks to some extent but may not migrate CS poisoning attacks well. To address it, we propose several possible defense methods that may help increase the difficulty of the attacks analyzed in this article.

Possible Countermeasures. Possible defense methods are mainly guided by two concepts: (1) denoising input images for downstream classifiers, and (2) increasing the robustness of downstream classifiers towards possible malicious input images. With the first idea, **high-level representation guided denoiser (HGD)** proposed by Liao et al. [24] can be trained with adversarial examples reconstructed by CS models or malicious images generated by poisoned CS models to provide the denoising protection for classifiers. For the second idea, the adversarial training paradigm proposed by Goodfellow et al. [15], which already shows its power to increase the robustness of classification models in various deep learning tasks, can be utilized to retrain the downstream classifiers and make them more robust to adversarial examples attacks and CS poisoning attacks.

8 RELATED WORK

8.1 Compressive Sensing

Compressed sensing was first introduced in [11] as a new method for signal processing. Since then, it has been actively investigated and applied in various fields such as image processing, medical imaging, radar technology, and so on. Different from traditional sampling methods guided by the Nyquist-Shannon sampling theorem, compressive sensing samples sparse signals in a linear manner at a lower sampling rate but can successfully recover them. Thus, it can provide high values in preserving more storage and having less computation, energy, and communication time [3]. Common compressive sensing methods include classic ones based on matrix computation and deep learning ones based on neural networks. Since the deep learning ones are efficient and can provide high compression and reconstruction quality, they are widely used in computer vision systems as a pre-processing process nowadays. In this article, we analyze the security risks introduced by compressive sensing in computer vision systems such as image classifiers.

8.2 Adversarial Attacks against Computer Vision Models

Recently, many works have demonstrated that computer vision models, such as image classifiers or object detectors, are susceptible to adversarial attacks. Existing adversarial attacks against image classifiers have two categories: (1) adversarial example attacks that craftily manipulate legitimate inputs to mislead the image classifier or object detector to provide wrong predictions [4, 15], and (2) model poisoning attacks that compromise the model by poisoning the training data to render the image classifier or object detector to provide wrong predictions on specific inputs [17, 30]. Both types of adversarial attacks can mislead image classifiers or object detectors and the subsequent decision-making, causing severe consequences. In this article, we analyze whether compressing sensing will aggravate or mitigate the vulnerabilities of image classifiers to both adversarial example attacks and model poisoning attacks.

8.3 Compressive Sensing for Adversarial Attack Defense

Using compressive sensing frameworks to filter adversarial examples has become one of the popular defense schemes for adversarial attacks. Dhaliwal et al. [8] proposed compressive recovery defense (CRD) to counter l_0, l_2, and l_∞ attacks, utilizing the fact that adversarial examples usually modify high-frequency components of images to deceive neural networks and human eyes. Kravets et al. [20, 21] implemented a series of compressive sensing defenses against adversarial examples based on 2D images and 3D point clouds. They demonstrated that compressive sensing can be used as an efficient method to thwart adversarial attacks on DNNs, and can be implemented in software and applied to attacked databases. They also demonstrated an optical system utilizing a single-pixel camera to realize the physical compressive sensing defense against real-world attacks. Wang et al. [35] introduced a generative neural network to accelerate the image reconstruction

process in compressive sensing defenses and proposed a defense strategy for visual recognition in autonomous vehicle systems.

8.4 Robustness Analysis for Image Recovery Methods

To the best of our knowledge, no existing work has analyzed the security risks introduced by compressive sensing in computer vision systems yet. However, security concerns about deep-learning-based image recovery methods have emerged recently. Huang et al. [18] were the first to apply adversarial attacks to neural-network-based image reconstruction methods. Antun et al. [1] discovered that small adversarial perturbations on compressed signals may cause severe reconstruction artifacts. Gottschling et al. [16] presented a comprehensive mathematical analysis on the instability of deep-learning-based reconstruction methods and declared that instability was not a rare event. Genzel et al. [14] explored the performance gap between adversarial and statistical noises in the context of image recovery instability influence. Darestani et al. [7] analyzed the robustness of compressive sensing methods for MRI and found that both deep-learning-based and classical-sparsity-based image reconstruction methods were sensitive to small adversarial perturbations in under-sampled measurements.

Our work is inspired by prior work and focuses on investigating the impact of using compressing sensing as a pre-processing process on the security of image classifiers.

9 CONCLUSION

In this paper, we investigate the security risk of the compressive sensing model from the computer vision system's point of view on the experiment settings combined with 3 attack methods, 3 datasets, 3 compressive sensing models, and 3 image classifiers. Based on experimental results, we conclude that a secured deep compressive sensing model will enhance the security of the computer vision system. Nevertheless, once the adversaries have access to the deep compressive sensing model, it is likely to become the accomplices of the adversaries, causing the computer vision system to be more vulnerable to adversarial attacks. Further directions include investigating the security impact of other pre-processing methods on computer vision models as well as exploring potential defense methods.

REFERENCES

[1] Vegard Antun, Francesco Renna, Clarice Poon, Ben Adcock, and Anders C. Hansen. 2020. On instabilities of deep learning in image reconstruction and the potential costs of AI. *Proceedings of the National Academy of Sciences (PNAS'20)* 117, 48 (2020), 30088–30095.

[2] M. T. Bevacqua, L. Crocco, L. Di Donato, and T. Isernia. 2014. Microwave imaging of nonweak targets via compressive sensing and virtual experiments. *IEEE Antennas and Wireless Propagation Letters* 14 (2014), 1035–1038.

[3] Holger Boche, Robert Calderbank, Gitta Kutyniok, and Jan Vybíral. 2015. A survey of compressed sensing. In *Compressed Sensing and Its Applications*. Springer, 1–39.

[4] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP'17)*. 39–57.

[5] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. 2017. ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (AISec'17)*. 15–26.

[6] Muhammad E. H. Chowdhury, Tawsifur Rahman, Amith Khandakar, Rashid Mazhar, Muhammad Abdul Kadir, Zaid Bin Mahbub, Khandakar Reajul Islam, Muhammad Salman Khan, Atif Iqbal, Nasser Al Emadi, Mamun Bin Ibne Reaz, and Mohammad Tariqul Islam. 2020. Can AI help in screening viral and COVID-19 Pneumonia? *IEEE Access* 8 (2020), 132665–132676.

[7] Mohammad Zalbagi Darestani, Akshay S. Chaudhari, and Reinhard Heckel. 2021. Measuring robustness in deep learning based compressive sensing. In *Proceedings of the 38th International Conference on Machine Learning (ICML'21)*. 2433–2444.

[8] Jasjeet Dhaliwal and Kyle Hambrook. 2020. Compressive recovery defense: Defending neural networks against **L**_2, **L**_∞ and **L**_0 norm attacks. In *Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN'20)*. 1–8.

[9] Hamza Djelouat, Hamza Baali, Abbes Amira, and Faycal Bensaali. 2017. IoT based compressive sensing for ECG monitoring. In *Proceedings of the 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. 183–189.

[10] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. 2018. Boosting adversarial attacks with momentum. In *Proceedings of the 2018 IEEE Conference on Computer vision and Pattern Recognition (CVPR'18)*. 9185–9193.

[11] David L. Donoho. 2006. Compressed sensing. *IEEE Transactions on Information Theory* 52, 4 (2006), 1289–1306.

[12] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M. Roy. 2016. A study of the effect of JPG compression on adversarial images. *arXiv preprint arXiv:1608.00853* (2016).

[13] Sudeep Fadadu, Shreyash Pandey, Darshan Hegde, Yi Shi, Fang-Chieh Chou, Nemanja Djuric, and Carlos Vallespi-Gonzalez. 2022. Multi-view fusion of sensor data for improved perception and prediction in autonomous driving. In *Proceedings of the 2022 IEEE/CVF Winter Conference on Applications of Computer Vision*. 2349–2357.

[14] Martin Genzel, Jan MacDonald, and Maximilian Marz. 2022. Solving inverse problems with deep neural networksRobustness included. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022), 1–1.

[15] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[16] Nina M. Gottschling, Vegard Antun, Ben Adcock, and Anders C. Hansen. 2020. The troublesome kernel: Why deep learning for inverse problems is typically unstable. *arXiv preprint arXiv:2001.01258* (2020).

[17] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2019. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access* 7 (2019), 47230–47244.

[18] Yixing Huang, Tobias Würfl, Katharina Breininger, Ling Liu, Günter Lauritsch, and Andreas Maier. 2018. Some investigations on robustness of deep learning in limited angle tomography. In *Proceedings of the 2018 Medical Image Computing and Computer Assisted Intervention (MICCAI'18)*. 145–153.

[19] Cheolsun Kim, Dongju Park, and Heung-No Lee. 2020. Compressive sensing spectroscopy using a residual convolutional neural network. *Sensors* 20, 3 (2020), 594.

[20] Vladislav Kravets, Bahram Javidi, and Adrian Stern. 2021. Compressive imaging for defending deep neural networks from adversarial attacks. *Optics Letters* 46, 8 (2021), 1951–1954.

[21] Vladislav Kravets, Bahram Javidi, and Adrian Stern. 2021. Compressive imaging for thwarting adversarial attacks on 3D point cloud classifiers. *Optics Express* 29, 26 (2021), 42726–42737.

[22] Alex Krizhevsky and Geoffrey Hinton. 2009. Learning multiple layers of features from tiny images. Technical report University of Toronto (2009).

[23] Shancang Li, Li Da Xu, and Xinheng Wang. 2012. Compressed sensing signal and data acquisition in wireless sensor networks and Internet of Things. *IEEE Transactions on Industrial Informatics* 9 (2012), 2177–2186.

[24] Fangzhou Liao, Ming Liang, Yinpeng Dong, Tianyu Pang, Xiaolin Hu, and Jun Zhu. 2018. Defense against adversarial attacks using high-level representation guided denoiser. In *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18)*. 1778–1787.

[25] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2018. Large-scale celebfaces attributes (Celeba) dataset. *Retrieved August* 15, 2018 (2018), 11.

[26] Adolfo Lozano, Jody C. Hayes, Lindsay M. Compton, Jamasp Azarnoosh, and Fatemeh Hassanipour. 2020. Determining the thermal characteristics of breast cancer based on high-resolution infrared imaging, 3D breast scans, and magnetic resonance imaging. *Scientific Reports* 10 (2020), 1–14.

[27] Yuxin Ma, Tiankai Xie, Jundong Li, and Ross Maciejewski. 2019. Explaining vulnerabilities to adversarial machine learning through visual analytics. *IEEE Transactions on Visualization and Computer Graphics* 26 (2019), 1075–1085.

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

[29] Ali Mousavi, Ankit B. Patel, and Richard G. Baraniuk. 2015. A deep learning approach to structured signal recovery. In *Proceedings of the 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton'15)*. 1336–1343.

[30] Luis Muñoz-González, Bjarne Pfitzner, Matteo Russo, Javier Carnerero-Cano, and Emil C. Lupu. 2019. Poisoning attacks with generative adversarial nets. *arXiv preprint arXiv:1906.07773* (2019).

[31] Leonid I. Rudin, Stanley Osher, and Emad Fatemi. 1992. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena* 60, 1-4 (1992), 259–268.

[32] Jo Schlemper, Jose Caballero, Joseph V. Hajnal, Anthony N. Price, and Daniel Rueckert. 2017. A deep cascade of convolutional neural networks for dynamic MR image reconstruction. *IEEE transactions on Medical Imaging* 37, 2 (2017), 491–503.

[33] Wuzhen Shi, Feng Jiang, Shaohui Liu, and Debin Zhao. 2019. Image compressed sensing using convolutional neural network. *IEEE Transactions on Image Processing* 29 (2019), 375–388.

[34] Dawn Song, Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, and Tadayoshi Kohno. 2018. Physical adversarial examples for object detectors. In *Proceedings of the 12th USENIX Workshop on Offensive Technologies (WOOT'18)*.

[35] Jia Wang, Wuqiang Su, Chengwen Luo, Jie Chen, Houbing Song, and Jianqiang Li. 2022. CSG: Classifier-aware defense strategy based on compressive sensing and generative networks for visual recognition in autonomous vehicle systems. *IEEE Transactions on Intelligent Transportation Systems* (2022), 1–11.

[36] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. 2004. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.

[37] Yan Wu, Mihaela Rosca, and Timothy Lillicrap. 2019. Deep compressed sensing. In *Proceedings of the 36th International Conference on Machine Learning (ICML'19)*. 6850–6860.

[38] Weilin Xu, David Evans, and Yanjun Qi. 2017. Feature squeezing: Detecting adversarial examples in deep neural networks. *arXiv preprint arXiv:1704.01155* (2017).

[39] Haotian Yang, Hao Zhu, Yanru Wang, Mingkai Huang, Qiu Shen, Ruigang Yang, and Xun Cao. 2020. Facescape: A large-scale high quality 3D face dataset and detailed riggable 3D face prediction. In *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'20)*. 601–610.

[40] Jian Zhang and Bernard Ghanem. 2018. ISTA-Net: Interpretable optimization-inspired deep network for image compressive sensing. In *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'18)*. 1828–1837.

[41] Yuchen Zhang and Percy Liang. 2019. Defending against whitebox adversarial attacks via randomized discretization. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS'19)*. 684–693.

[42] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. 2016. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging* 3, 1 (2016), 47–57.

Received 12 June 2023; revised 5 December 2023; accepted 22 January 2024