DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Reference herein to any social initiative (including but not limited to Diversity, Equity, and Inclusion (DEI); Community Benefits Plans (CBP); Justice 40; etc.) is made by the Author independent of any current requirement by the United States Government and does not constitute or imply endorsement, recommendation, or support by the United States Government or any agency thereof.

ORNL/TM-2025/4263

Deploying Adversarial Attacks in Super-Resolution Models

Emma J. Reid
Haley Duba-Sullivan
Kieran Barr
Steven R. Young

October 2025

OAK RIDGE National Laboratory

ORNL IS MANAGED BY UT-BATTELLE LLC FOR THE US DEPARTMENT OF ENERGY

ORNL/TM-2025/4263

Cyber Resilience and Intelligence Division

DEPLOYING ADVERSARIAL ATTACKS IN SUPER-RESOLUTION MODELS

Emma J. Reid
Haley Duba-Sullivan
Kieran Barr
Steven R. Young

October 2025

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831
managed by
UT-BATTELLE LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

C-W	Carlini-Wagner
FGSM	Fast Gradient Sign Method
HR	High-resolution
L-BFGS	Limited-Memory Broyden–Fletcher–Goldfarb–Shanno
LR	Low-resolution
PnP	Plug-and-Play
SR	Super-resolution
UAP	Universal Adversarial Perturbation
YOLO	You Only Look Once

ACKNOWLEDGEMENTS

This manuscript has been authored in part by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The publisher acknowledges the US government license to provide public access under the DOE Public Access Plan (https://energy.gov/doe-public-access-plan). This research was sponsored by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the U. S. Department of Energy. This research used resources from the ORNL Research Cloud Infrastructure at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

ABSTRACT

Reliable super-resolution methods are crucial for applications like remote sensing, grid resilience and disaster impact analysis, and standoff biometrics. These methods infuse additional high-frequency information into reconstructions, allowing for better contextualization and image intelligence. However, super-resolution models can also introduce hallucinations or other unseen vulnerabilities that could be exploited by an adversary. This is further compounded by the prominence of deep learning in these models, as models are often blindly applied on out-of-distribution images.

In this work, we implement adversarial attacks in common open-source super-resolution models and examine their impact on reconstructions and downstream classification tasks. We find that an adversarially trained super-resolution model can produce high-quality reconstructions that degrade downstream classifications. Moreover, these attacks do not require access to low-resolution imagery or class labels at inference time. These results demonstrate the vulnerability of super-resolution methods to malicious actors and motivates the development of a detector for super-resolution adversarial attacks. Further exploration of adversarial attacks in this domain is required to ensure trustworthiness and robustness of super-resolution models for national security applications.

1. INTRODUCTION

Super-resolution (SR) seeks to improve the spatial resolution of an image by introducing high-frequency content while simultaneously maintaining fidelity to the existing image content. This has been shown to improve analysis in fields like medical imaging [1] [2], additive manufacturing [3], and remote sensing [4]. Moreover, SR has been shown to improve effectiveness of downstream tasks like classification [5], detection [6] [7], and segmentation [8]. State-of-the-art SR methods generally use a deep learning-based approach, which involves learning direct mappings between low- and high-resolution spaces.

Adversarial attacks have emerged as a common concern when using deep learning, particularly for national security applications [9, 10, 11]. These attacks maliciously target machine learning models by causing missed detections and misclassifications, generally by adding slight strategic perturbations to input images. Moreover, researchers have developed targeted attacks that can cause an image to be identified as a particular attacker-determined class [12]. In response to the prevalence of adversarial attacks, researchers have begun to develop detection methods and defense mechanisms [9, 13]. While adversarial attacks are studied extensively for image detection and classification tasks, researchers are just beginning to focus on similar vulnerabilities within image-to-image networks like SR models [14, 15, 16].

Existing research generally investigates two types of security vulnerabilities within SR models: adversarial perturbations and backdoor attacks. The main difference between these approaches is whether the attack is within the input Low-resolution (LR) image or is a trigger trained into the SR model. Adversarial perturbations are applied to LR images to produce a degraded or perturbed SR reconstruction. In backdoor attacks, the SR model is trained to produce an adversarial SR image when a trigger is present in the LR space. Methods to induce these adversarial attacks within SR models include data poisoning and adversarial loss functions [17, 18, 16].

One major limitation of current research is that it assumes the input image to the SR model is accessible, either to add a perturbation or a trigger. However, this requires that the adversary is able to manipulate data at inference time, which is often not the case. Our work considers the situation where the adversary can manipulate the training data or the SR model, but does not have access to the data at inference time, which is a more realistic situation. This also accounts for cases of using an open-source SR model or dataset for internal applications.

Here we explore adversarial attacks on SR models with the aim of ultimately detecting said attacks. We review existing SR methods, adversarial attack constructions, and adversarial SR methods. We then experiment with a variety of attacks and propose an adversarial loss function that is able to effectively cloak an entire class from a classifier. We detail training and testing procedures and outline future work for attack generation and detection.

2. PROBLEM STATEMENT

Consider a SR model $M : \mathbb{R}^{m\times m} \to \mathbb{R}^{Lm\times Lm}$ that maps a LR image y to a High-resolution (HR) image x,where L is the amount of resolution increase. L is commonly known as the SR factor. In our application, we assume that the output SR image $M(y) = x$ is passed to a classification method $f_\theta(\cdot)$ that assigns a class to an image x, i.e. $f_\theta(x) = c$ where c is the predicted class of x.

Now suppose that an adversary wishes to tamper with this imaging pipeline and cause misclassifications by f_θ. Commonly it's assumed that adversarial tampering is done by strategically manipulating the LR image y. Instead, we consider the possibility that the model M is tampered with to cause a degradation of f_θ's performance. Namely, M may be trained to insert perturbations or triggers that target the classifier $f_\theta(\cdot)$ through either data poisoning or an adversarial loss function.

For data poisoning, we assume that we have a clean training set of N paired LR and HR images $\{(y_i, x_i)\}_{i=1}^N$. Additionally, we assume access to the target classifier model $f_\theta(\cdot)$ with weights parameterized by θ. In this context, data poisoning creates a set of perturbed HR images on which the SR model M is trained. This adversarial dataset $\{(y_i, \tilde{x}_i)\}_{i=1}^N$ is created such that

$$\tilde{x}_i = x_i + \delta_{x_i} \tag{1}$$

where δ_{x_i} is an adversarial perturbation targeting $f_\theta(\cdot)$, which will be defined and discussed in Section 5. The ultimate effect of δ depends on the chosen adversarial attack, but, at a high-level, the perturbations should be unnoticeable in the SR images and impact classifier performance. By using a poisoned training set for M, the adversary functionally encodes the adversarial perturbations into the reconstructions.

In the case of an adversarial loss function, the training data is unperturbed. Instead, the adversary introduces a component L_{atk} into the model's loss function that encourages adversarial reconstructions. This is akin to saying

$$L_{adv} = L_{sr} + L_{atk} \tag{2}$$

where L_{sr} is the SR loss and L_{atk} is some sort of constraint that reduces the effectiveness of the classifier $f_\theta(\cdot)$. By using an adversarial loss function, the adversary teaches the SR model to simultaneously produce high-quality SR images that contain adversarial perturbations.

In either case, the classifier should perform well on untargeted classes and poorly on the targeted class. Additionally, the adversarial SR images should be similar to the expected SR images, both qualitatively and quantitatively. If the adversarial attack is noticeable, either through classifier or SR performance, this could lead the end user to abandon the SR model or classifier, nullifying the adversary's efforts.

3. RELATED WORK

Here we cover SR methods, adversarial attacks, and adversarial SR techniques. Additionally we detail the approaches for adversarial SR techniques and their assumptions on attack scenarios.

3.1 SUPER-RESOLUTION METHODS

The SR problem is formulated as

$$y = Ax + \epsilon \tag{3}$$

where $A : \mathbb{R}^{Lm \times Lm} \rightarrow \mathbb{R}^{m \times m}$ decreases the resolution by a factor of L and $\epsilon \sim N(0, \sigma^2)$ is additive white Gaussian noise with standard deviation σ. The operator A is commonly referred to as a decimation operator and is composed of blurring and subsampling operations. SR methods seek to reconstruct the HR image x based on the LR measurements y. However, this problem is ill-posed by construction — for a LR image y, there are infinitely many HR images x that could have produced it. Thus it becomes necessary to restrict the space of candidate SR images x through regularization, either with an explicit model-based formulation or a learning-based approach.

Model-based approaches seek to accurately model the image acquisition process for determining A. They additionally incorporate assumptions about the HR image distribution as regularization. Common regularization approaches include non-local means [19], quadratic regularization [20], and total-variation [21]. However, performance of model-based methods depends on accurate modeling of the physics, can be computationally expensive due to the iterative nature of these methods, and requires construction of an explicit regularization function.

Plug-and-Play (PnP) priors [22] removed the need for constructing an explicit regularization function from model-based methods and instead replaced the regularizer by an additive white Gaussian denoiser. In 2017, [23] extended PnP to the SR problem and introduced HR information to reconstructions through a library-based non-local means prior. A consequent approach proposed in [24] used a denoising prior trained on HR imagery to infuse high-frequency data into reconstructions within a PnP framework. In [25], they introduced a deep denoising prior for a number of image restoration tasks, including SR. However, priors need not be limited to denoising. In [26], they adapted the PnP framework to use a SR model as the prior with great success. However, PnP methods still suffer from greater computational time as they can require many iterations and require accurate modeling of the physical system.

Learning-based approaches forgo any physical modeling and depend entirely on a neural network to learn the relationship between LR and HR images from a training dataset. In 2014, Dong *et al.* proposed the first deep learning-based SR method called "SR Convoluutional Neural Network" (SRCNN). This method learns an end-to-end mapping from LR to HR space with a fully connected network consisting of 3 convolutional layers. However, SRCNN requires bicubically interpolating the LR image before inputting it into the network, which increases the computational complexity of the method. To reduce the complexity, Kim *et al.* proposed "Very Deep Super-Resolver" (VDSR) which upsamples the image within the network so that most of the computation is in LR space. They additionally improve on SRCNN by increasing model depth, simplifying learning rates, and incorporating residual learning via a single skip connection[27]. In 2017, Ledig *et al.* adopted residual learning in their proposed SRResNet [28] to improve network performance

for SR. Lim *et al.* proposed EDSR which optimized the SRResNet architecture by removing modules like batch normalization and excess activation functions. This reduced training time while also significantly improving performance [29].

In the same paper as SRResNet, Ledig *et al.* also introduced SRGAN [28], which was the first generative adversarial network applied to SR. SRGAN used SRResnet as its generator and incorporated a perceptual loss function in addition to the standard pixel loss function. Iterations on SRGAN include ESRGAN [30], RealESRGAN [31], and BSRGAN [32] which incorporate residual-in-residual dense blocks, expand the degradation models in training, and apply spectral normalization within the discriminator network, respectively. Using generative adversarial networks for SR enabled better reconstruction of high-frequency detail; however, they are also known for mode-collapse during training and have a tendency to introduce hallucinated details.

After the demonstrated success of vision transformers, Liang *et al.* applied the Swin transformer architecture to image restoration tasks in their SwinIR method [33]. SwinIR uses shallow feature extraction, deep feature extraction, and image reconstruction modules to achieve state-of-the-art performance on various image restoration tasks, including SR. Residual Swin Transformers are incorporated in the deep feature extraction module, allowing feature aggregation and translational equivariance.

While learning-based methods have the ability to generate high-quality SR images, these methods' dependency on training data limits performance when testing scenarios are outside of the training distribution. This dependency leaves models susceptible to adversarial attacks.

3.2 ADVERSARIAL ATTACK METHODS

For an image x with $f_\theta(x) = c$, an adversarial attack generally seeks to perturb x to $\tilde{x}$ such that $f_\theta(\tilde{x}) \neq c$. In 2014, Goodfellow *et al.* presented a canonical example of this phenomena, shown in Figure 1 [34]. Here an image of a panda is slightly perturbed with adversarial noise, causing the classifier to predict it as a gibbon. Moreover, the classifier is extremely confident in its incorrect gibbon prediction! Adversarial attacks work by pushing the image over the classifier's decision boundary, causing incorrect class predictions. The complexity of this boundary depends on a number of factors, including the chosen classifier model and training dataset [35] [36] [37].

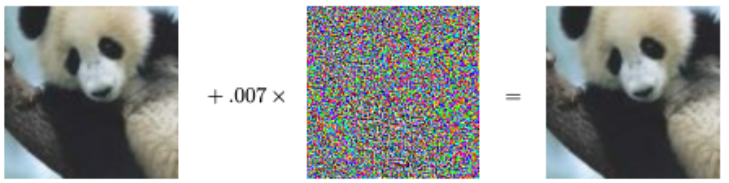


Figure 1. A canonical adversarial attack example from [34]. The addition of a small amount of adversarial noise is able to fool a classifier.

Goodfellow's example is an example of an additive adversarial attack, namely $\tilde{x} = x + \delta_x$ for $x, \delta_x \in \mathbb{R}^{Lm \times Lm}$. Additive adversarial attacks can either be untargeted or targeted. In either case, we assume that we are given a predefined set of m classes, $\mathcal{C} = \{1, \ldots, m\}$. Assume that class s is the true class of x. An untargeted adversarial attack is an attack δ_x such that x is classified as

any class other than s, i.e. $f_\theta(x + \delta_x) \neq s$. In contrast, a targeted adversarial attack is an attack δ_x such that x is classified as a specific target class t, i.e. $f_\theta(x + \delta_x) = t$ [38]. In a targeted attack, we refer to s as the source class and t as the target class.

Figure 2. An example of a non-additive attack that uses a flow field to perturb pixel locations. The flow field is overlaid on the image.

While the additive form is both simple and useful for mathematical proof, there are many other forms of adversarial attack that exploit the same underlying concept — neural networks are vulnerable to small changes to their input. In [39, 40], this takes the form of learning a flow field that transforms the pixel locations rather than the pixel values. A face recognition application of such a flow field appears in Figure 2. These spatial transformation attacks can be even simpler than a flow field, as [41] shows that they can be performed using basic affine transformations such as rotation and translation.

Adversarial attacks can be further categorized by their assumed knowledge of the classifier f_θ. In white-box attacks, the attacker can access the model's architecture, trained weights, and other parameters. In black-box attacks, the attacker can only see the input and output of the classifier. Gray-box attacks split the difference, giving the attacker a vague sense of the architecture

but may lack specific weights and parameters [42]. White-box attacks are generally more effective due to their level of access. However, defense mechanisms that stop black-box or gray-box attacks can be foiled by white-box attacks as shown in [43]. We chose to pursue white-box attacks for the sake of interpretability, but it would be interesting to pursue gray or black-box attacks in the future.

The constraints used to develop adversarial perturbation vary, often prioritizing minimality of δ_x or maximizing misclassifications. The minimum l_2 norm attack seeks to minimize the norm of δ_x while also ensuring that $\tilde{x}$ is misclassified with a loss function $L(\cdot)$ for class c based on predictions by f_θ. Mathematically, this equates to

$$\underset{\tilde{x}}{\text{minimize}} \; ||x - \tilde{x}||^2 \text{ s.t. } L(\tilde{x}, c, \theta) < 0 \tag{4}$$

Note that $||\tilde{x} - x||^2$ is precisely $||\delta_x||^2$ and the constraint of $L(\cdot) < 0$ restricts us to perturbations that induce misclassification. For simplicity of notation, we define $p_c(x)$ to be the probability that f_θ predicts x as class c. In the case of untargeted attacks on class c,

$$L(\tilde{x}, c, \theta) = p_c(\tilde{x}) - \max_{j \neq c}\{p_j(\tilde{x})\}, \tag{5}$$

which ensures that $f_\theta(\tilde{x}) \neq c$ [44] as there will exist a larger probability than $p_c(\tilde{x})$. For a targeted attack, the constraint becomes

$$L(\tilde{x}, c, \theta) = \max_{j \neq t}\{p_j(\tilde{x})\} - p_t(\tilde{x}), \tag{6}$$

which ensures that $\tilde{x}$ is predicted to be class t by limiting to the set $\{p_t(\tilde{x}) > p_j(\tilde{x})\}$ [38]. The Limited-Memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) [10] attack and DeepFool [45] are prominent examples of a minimum l_2 norm attack.

In contrast, the maximum allowable attack has the following formulation.

$$\underset{\tilde{x}}{\text{minimize}} \; L(\tilde{x}, c, \theta) \text{ s.t. } ||\tilde{x} - x|| \leq \eta \tag{7}$$

Note that this formulation is nearly identical to (4), but with a focus on misclassification rather than minimizing the perturbation norm. The Fast Gradient Sign Method (FGSM) [34] is an example of a maximum allowable attack. In the FGSM attack, $\delta = \epsilon \text{ sign}(\nabla_x J(\theta, x, c))$ where J is the loss function used to training the classifier and ∇_x represents taking the gradient with respect to x.

While the previously discussed methods require a constraint, some researchers also consider an unconstrained formulation. This approach, known as a regularization-based attack, solves

$$\underset{\tilde{x}}{\text{minimize}} \; ||\tilde{x} - x||^2 + \lambda \left(L(\tilde{x}, c, \theta)\right), \tag{8}$$

where λ is a Lagrange multiplier. In practice, λ is tuned to weight the impact of $L(\tilde{x}, c, \theta)$. One example of a regularization-based attack is the Carlini-Wagner (C-W) attack [12]. They derive their targeted attack as

$$\underset{\tilde{x}}{\text{minimize}} \; ||\tilde{x} - x||^2 + \lambda \max\left\{\left(\max_{j \neq t}\{p_j(\tilde{x})\} - p_t(\tilde{x})\right), 0\right\}. \tag{9}$$

The latter half of this equation functions as a penalization for the targeting not being met, while the first ensures the invisibility of δ. Pseudocode for computing $\tilde{x}$ is shown in Algorithm 1.

Algorithm 1: Carlini & Wagner Attack Image Construction

Data: Input image x with class s, classifier f_θ, box constraint parameter b, confidence parameter κ, target confidence γ

Result: C&W attack-perturbed image $\tilde{x}$ with confidence γ for target class t

Initialize $w \leftarrow arctanh(0.999999(x^2 - 1))$;

while $p_t(\tilde{x}) < \gamma$ *or* $f_\theta(\tilde{x}) \neq t$ **do**

  $\tilde{x} \leftarrow \frac{1}{2}(tanh(w) + 1)$;

  Normalize $\tilde{x}$;

  $Loss \leftarrow ||\tilde{x} - x||_2^2 + b \cdot clip(max(p_{i \neq t}(\tilde{x})) - p_t(\tilde{x}), -\kappa)$;

  Update w with $Loss$ via backpropogation with chosen optimizer ;

end

It's worth mentioning that each of the attacks described so far depend on the specific image x. However, one can also construct a Universal Adversarial Perturbation (UAP) [46] that attacks an entire set of images at once. In this case, the attack $\delta_{\mathcal{X}}$ is constructed for a collection of images $\mathcal{X}$ with distribution μ such that

$$\begin{aligned} ||\delta_{\mathcal{X}}|| &\leq \eta, \\ \text{For all } x \in \mathcal{X},\ p_{x \sim \mu}(f_\theta(\tilde{x}) \neq f_\theta(x)) &\geq 1 - \epsilon. \end{aligned} \tag{10}$$

Here ϵ is defined as the fooling rate, which determines the percentage of images x on which the UAP fools the classifier. These constraints force $\delta_{\mathcal{X}}$ to be imperceivable while also causing poor classifier performance across all selected images.

However, SR methods are often used to mitigate the impact or invisibility of adversarial attacks. Rajabi [47] introduces the concept of an attack's "survivability", namely if an adversarially perturbed LR image is able to survive the application of a SR model. They also consider the case of downsampling an adversarially perturbed HR image for use in a training pair. They found that adversarial images could survive SR by SRCNN, particularly when using block-averaging as their A operator. This is of particular interest, as SR is used as a method to mitigate adversarial attacks [48] [49]. However [47] shows that with proper construction, this mitigation avenue can be thwarted.

3.3 ADVERSARIAL SUPER-RESOLUTION METHODS

With the content of the previous sections in mind, we now consider the idea of adversarial SR. Previous work in this field focuses on designing adversarial perturbations in the LR space to degrade HR reconstructions, training SR models to produce particular HR images in the presence of a LR space trigger, and examining the robustness of SR methods. In general, adversarial SR methods can be categorized as adversarial perturbations on images or backdoor attacks on models. In adversarial perturbations on images, input LR images are carefully manipulated to produce perturbed HR reconstructions. In backdoor attacks on models, the model is trained to produce specific HR image(s) predetermined by the adversary when the LR image contains a predefined trigger.

3.3.1 Adversarial Attacks on Images

Adversarial attacks on images operate at inference time by manipulating the input data rather than the model parameters or training samples. In this threat model, the adversary assumes the capability to publish or otherwise deliver manipulated LR images that an unsuspecting user later downloads and processes with a SR model. These attacked inputs appear visually benign but are intentionally constructed to cause degraded or misleading reconstructions. This setting corresponds to a form of inference-time data poisoning, which is plausible in open research workflows where practitioners rely on publicly available datasets, benchmark repositories, or automated pipelines that fetch external images.

Quiring et al. [50], building on the work of Xiao et al. [51], analyze image-scaling attacks that exploit deterministic behavior in common downsampling implementations. Many downsampling operators, such as those used in Pillow, OpenCV, and TensorFlow, compute each output pixel from a structured subset of input pixels. By carefully perturbing only these influential pixels, an attacker can manipulate the downsampled result while leaving the overall image visually unchanged. This allows the adversary to craft an attacked image $\tilde{x} = x + \delta_x$ such that the downsampled output $D(\tilde{x})$ matches an attacker-specified target x_t, under the constraint that $\tilde{x}$ remains perceptually similar to the clean image x. The optimization can be formulated as

$$\underset{\delta_x}{\text{minimize}} \; |\delta_x|_2^2 \quad \text{s.t.} \quad |D(x + \delta_x) - x_t| \leq \epsilon, \tag{11}$$

where ϵ is a small tolerance controlling fidelity to the target. Quiring et al. demonstrate that these attacks can succeed across several open-source image libraries, although the ease of attack varies: for example, Pillow's dynamic kernel width and area-scaling interpolation make it considerably harder for the adversary to control $D(\tilde{x})$. While originally demonstrated for downsampling, the underlying principle generalizes to upsampling and other linear resampling operations commonly used in SR pipelines.

Huang et al. [52] propose a scale-invariant adversarial attack that targets arbitrary-scale SR methods based on continuous image representations. Such SR models, denoted $\Psi(\cdot, \Lambda)$, learn a continuous function that maps LR coordinates to an HR signal across arbitrary scale factors. The adversary perturbs the LR image y to $\tilde{y} = y + \delta$ so that the continuous representation $\Psi(\tilde{y}, \Lambda)$ diverges from $\Psi(y, \Lambda)$, producing degraded reconstructions at any scale. To approximate the continuous image domain, the image is divided into blocks and a small number of sample coordinates are drawn per block. The attack then minimizes

$$\mathcal{L}_{\text{SI}} = |\Psi(\tilde{y}, \Lambda) - \Psi(y, \Lambda)|_2, \tag{12}$$

where Λ is the set of sampled coordinates. By exploiting the continuity of Ψ, the attacker only needs to perturb a sparse subset of pixels to induce large deviations in the reconstructed HR output. To further amplify perceptual differences, the authors propose adding high-frequency components to $\tilde{y}$, effectively making the continuous representation more unstable.

Both of these image-scaling and scale-invariant attacks rely on the adversary's ability to manipulate inference-time inputs — a strong but plausible assumption in open or automated evaluation environments. They also assume detailed (white-box) knowledge of the preprocessing or SR model used by the victim. In practice, small implementation differences (e.g., kernel variations, rounding rules) or randomized preprocessing can significantly reduce attack success. Moreover, many demonstrated examples produce conspicuous HR artifacts that a human observer could detect. Despite these limitations, these attacks expose critical vulnerabilities in the image acquisition and preprocessing stages of SR pipelines. They demonstrate that deterministic interpolation

and coordinate-based SR methods can be exploited as adversarial entry points, motivating defensive strategies such as dataset integrity verification, preprocessing randomization, and robustness testing under inference-time perturbations.

3.3.2 Backdoor Attacks on Models

Backdoor attacks on SR models proceed by injecting poisoned samples into a training dataset so that the learned model behaves normally on clean inputs but produces attacker-chosen outputs when a trigger is present. Unlike inference-time input manipulation, backdoors give the adversary persistent control over model behavior after training and do not require the attacker to influence inputs at test time (beyond providing the trigger). Typical threat vectors include poisoned public datasets, compromised model zoos, or maliciously modified training scripts.

Reference-based SR offers a convenient target for backdoors because the model conditions on an auxiliary HR reference image. Yang et al. [18] consider training triplets (y, x_{ref}, x) (LR, HR reference, HR ground truth) and replace a subset with poisoned triplets (y, x'_{ref}, x') where x'_{ref} contains a trigger and x' is the adversary's target HR image. The desired adversarial model behaviour can be written as:

$$M(y, x_{ref}) = x \text{ and } M(y, x'_{ref}) = x'. \tag{13}$$

This means that if no poisoned reference image is given M should function as normal. However if a poisoned reference image is given, M should produce the target image x'.

The BadSR attack proposed in [17] employs a different design: a small subset P of training indices are poisoned by adding a trigger δ to the LR image y_j and replacing the corresponding HR image x_j with a target x_p that visually similar to x_j but feature-space-aligned with the adversary's target image x_t. Namely, let g_ϕ be a feature representation extraction function parameterized by ϕ. The poisoned HR image x_p is chosen to satisfy

$$\min_{x_p} ||g_\phi(x_p) - g_\phi(x_t)||_2^2 \text{ s.t. } ||x_p - x_j||^2 \leq \epsilon. \tag{14}$$

Then, training with a mixture of clean and poisoned samples embeds adversary's desired backdoor. By preserving visual similarity with x_j, the BadSR attack is stealthier than traditional backdoor attacks, as demonstrated in their experimental results.

Jiang et al. [16] generalize backdoor approaches to image-to-image tasks using UAPs as triggers. Their training objective mixes a clean-data loss $\mathcal{L}_c$ and a poisoned-data loss $\mathcal{L}_p$:

$$\mathcal{L}_c = ||M(y_c) - x_c||_2, \tag{15}$$

$$\mathcal{L}_p = ||M(y'_p) - x^*_p||_2, \tag{16}$$

$$\mathcal{L}_{total} = \lambda_c \mathcal{L}_c + \lambda_p \mathcal{L}_p, \tag{17}$$

where λ_c, λ_p are weighting coefficients. Balancing these losses during training is critical: poorly chosen weights can cause convergence issues or obvious degradation on clean data, while carefully tuned (often dynamic) weights are needed to preserve SR quality on clean inputs while ensuring a strong backdoor.

Limitations common to backdoor attacks on SR include the lack of discrete classes (unlike classification), which complicates designing universally effective triggers, and the requirement that the attacker either control training data or the training pipeline. Successful backdoors must also maintain high perceptual quality on clean inputs to remain stealthy.

3.3.3 A New Attack Model: Adversarial Attacks on Models

Prior work on adversarial SR primarily assumes the attacker influences the training or test data. In contrast, we consider a distinct and realistic threat: the adversary publishes a pre-trained *adversarial SR model* that a naive user downloads and uses as a drop-in component in their imaging pipeline. In this model-publishing threat, the attacker does not need to compromise the user's dataset or inference inputs; instead the malicious behavior is embedded in the model weights distributed by the attacker. We discuss our proposed method for generating this attack in Section 5.4.

4. IMPLEMENTATION DETAILS

In this section, we discuss specific implementation details for our data pipeline, classifier, and SR model. We additionally detail our data wrangling process and note appropriate approvals.

4.1 DATASET

ImageNet [53] contains 14 million images, with a whopping 3.2 million of these images annotated across 5247 categories. They have "fine" and "coarse" categories (i.e. otter vs aquatic mammal), though they employ the terminology of "synset" and "subtree" respectively. We created our dataset by manually subsetting 40 vehicular classes, as well as 3 dummy classes (shark, strawberry, and triceratops). An example of some of these classes is shown in Figure 3. As this dataset does contain humans (despite being vehicular classes), we submitted an approved IRB protocol.

Figure 3. An example of 9 classes from our subset of ImageNet [53]

4.2 CLASSIFIER

In [54], they presented the You Only Look Once (YOLO) framework for object detection. The main novelty of YOLO is that it reframes object detection as "a regression problem from image pixels to class probabilities and bounding boxes". This reframing allows the network to only ingest the image once, massively increasing speed over two-stage alternatives. In terms of implementation, YOLO employs 24 convolutional layers of increasing depth followed by 2 fully connected layers. For our classification network, we train a YOLOv11 model for 100 epochs on a subset of ImageNet corresponding to the 43 classes mentioned in Section 4.1. Most weights are transferred from their pretrained v11 Ultralytics model [55], with fine-tuning done for the classifier layer. For classification, YOLOv11 employs a cross-entropy loss function. Our model is trained with a SGD optimizer with learning rate 0.01 and momentum 0.9. We fix our image size as 256 $\times$ 256 and resize the images using YOLO's internal LetterBox interpolator to maintain its aspect ratio.

4.3 SUPER-RESOLUTION MODEL

We chose to use SwinIR [33] as our SR model due to its ability to reconstruct high-frequency information reliably. We use the official implementation and default parameters for the model architecture, which can be found at https://github.com/JingyunLiang/SwinIR. Additionally, we

Figure 4. Visual depiction of the data pipeline. A low-resolution image is passed to a SR model, whose reconstruction is passed to a classifier. The output of this pipeline is a class prediction.

initialize the model weights using the official pretrained weights provided by the authors for 2× SR of RGB-images (found at https://github.com/JingyunLiang/SwinIR/releases). We explain the fine-tuning training procedure for each adversarial attack in Section 5.

4.4 DATA PIPELINE AND PREPROCESSSING

The LR images are generated by applying a 9×9 Gaussian blur kernel with standard deviation $\sigma = 0.75$, followed by subsampling by a factor of 2 in the $x-$ and $y-$ direction. These LR images are then passed to the SR model, which outputs a SR image. Once SR images are acquired, they're passed to the YOLO classifier for inference. The data pipeline is shown in Figure 4.

5. METHODS AND RESULTS

In this section, we outline four approaches that we tried for adversarial SR. Two approaches use data poisoning, while the other two use an adversarial loss function. The first three approaches that we took were unsuccessful in causing misclassification while staying stealthy; however, we think it is important to document these approaches here to guide future research in this area. That said, we will cover these approaches more briefly, and cover the fourth (successful) approach in more detail.

5.1 CARLINI & WAGNER DATA POISONING

Given the prevalence of adversarial attacks in classification, our initial strategy for data poisoning was to adapt an existing attack from this domain. We selected the C-W attack [12], introduced in Section 3, due to its effectiveness and widespread use. For each HR image x_i in the training set, we first computed an adversarial perturbation δ_i using the C-W algorithm, and then fine-tuned the SR model on a poisoned dataset $\{(y_i, x_i + \delta_i)\}_{i=1}^{N}$, where y_i denotes the corresponding LR image.

The C-W attack is targeted, meaning that δ_i depends on both the image x_i and its class label. The optimization method that solves for δ_i involves several hyperparameters: the weight λ of the confidence constraint in the loss, the target confidence c_t, the confidence margin κ, and the number of iterations K (see Algorithm 1 for pseudocode). In our experiments, we set $\lambda = 10^9$, $c_t = 0.98$, $\kappa = 10$, and $K = 200$, and used the Adam optimizer with a learning rate of 0.01.

As shown in Figure 5, the perturbations produced by the C-W attack appear visually subtle and can effectively fool the classifier by forcing misclassification with high confidence. However, this approach proved ineffective for training an adversarial SR model since the perturbations are image-specific, which prevents generalizability to different images. The SR model, whose training focuses on reconstructing fine-grained image details using perceptual and pixel-wise losses without relying on class information, is unable to accurately learn when to insert perturbations into the images and what type of perturbations to insert. Thus, training a SR model with this data poisoning approach fails to force adversarial behaviour, underscoring the limitations of directly applying targeted classification-based adversarial methods to regression tasks like SR.

5.2 UNIVERSAL ADVERSARIAL PERTURBATION DATA POISONING

Given the C-W attack's reliance on image-specific features, we shifted our focus towards UAPs, which were also defined in Section 3. Unlike the C-W attack, which relies on class-dependent adjustments, UAPs are designed to cause misclassification across an entire distribution without tailoring the perturbation to individual class features. By leveraging UAPs, we aim to generate image-agnostic perturbations capable of ingraining adversarial behaviour within the SR model in a manner that aligns more naturally with the architecture and loss formulations of SR models.

To construct the UAP, we adopt the iterative procedure proposed by Moosavi-Dezfooli et al. in [56]. In this process, DeepFool serves as the core subroutine for driving samples across the decision boundary. In our implementation, the perturbation vector is initialized to zero and incrementally updated as new samples are drawn from the training set, with each update projected back onto the ℓ_∞ ball to satisfy the magnitude constraint. We set the upper bound on the ℓ_∞ norm to $\xi = 10$. The maximum number of DeepFool iterations was fixed at 50 with an overshoot factor of

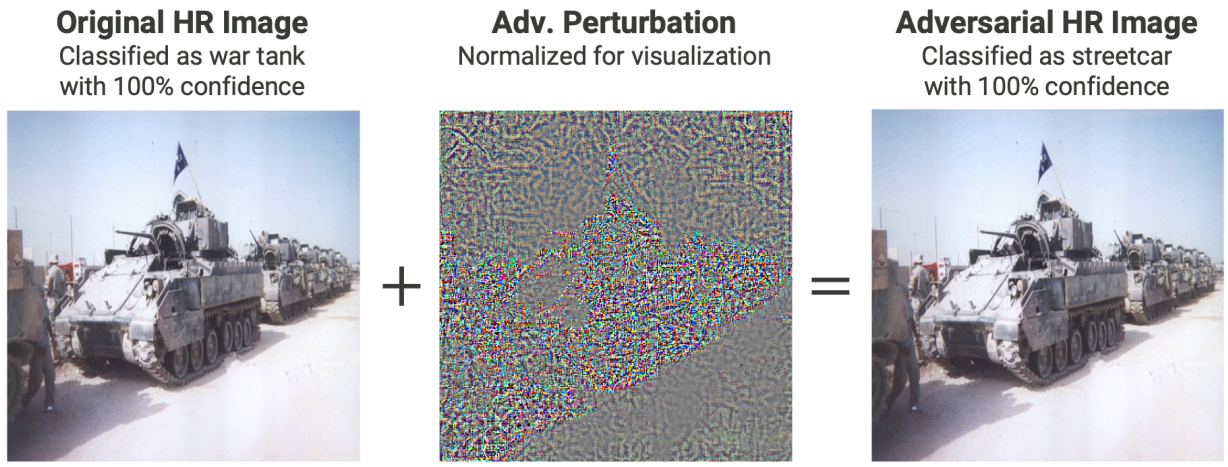


Figure 5. Example of an adversarial perturbation generated by the C-W attack applied to a HR training image, along with its impact on the classification model.

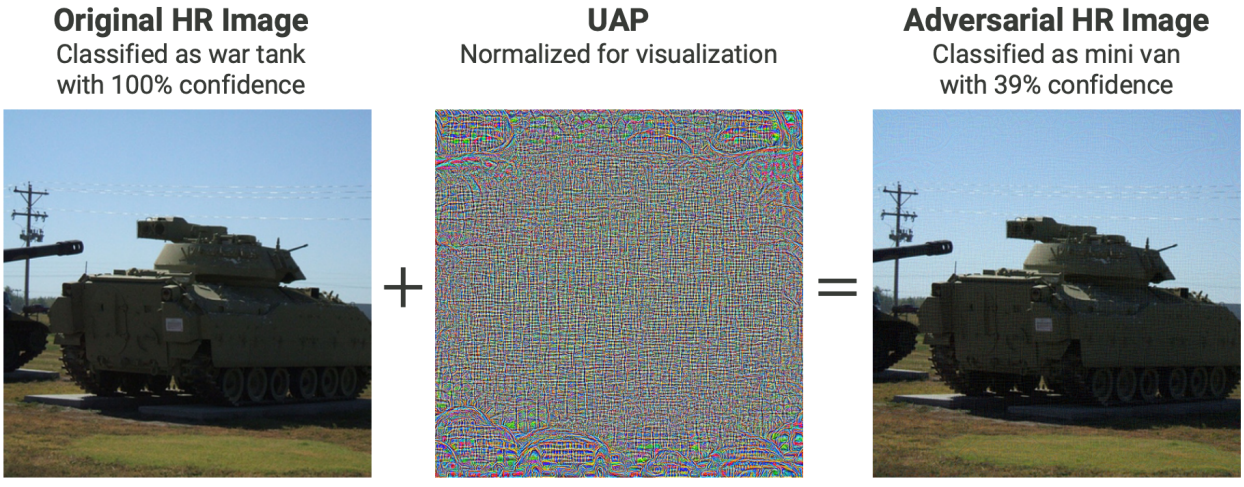


Figure 6. Example of UAP applied to a HR training image, along with its impact on the classification model.

0.02, and we terminate once the fooling rate exceeds $1 - \delta = 0.9$. This choice of parameters ensures that the learned UAP remains imperceptible in magnitude while still producing a consistent degradation effect across the SR test set.

As shown in Figure 6, the UAP that we generated is somewhat subtle and does fool the classifier, although with less confidence that the C-W attack. Additionally, we found that by training with this poisoned dataset, the adversarial SR model learns to embed the UAP into its reconstruction process. While this indicates the model's susceptibility to such universal alterations, it also led to universal decrease in accuracy of the downstream classifier across all classes, reducing the attacked classifier from 90% accuracy to 62% accuracy. This universal degradation in classification performance underscores a significant limitation of a universal attack like a UAP: while universal attacks can effectively imbue adversarial SR performance, their broad applicability adversely affects their stealthiness. This limitation motivates the development of an adversarial attack on SR models that is not image-dependent, yet also not universal—a goal that proves challenging to achieve with a data poisoning approach.

5.3 SHROUDING ADVERSARIAL LOSS FUNCTION

In light of the challenges encountered with data poisoning attacks, we shifted our focus toward a more targeted attack by replacing data poisoning attacks with adversarial loss functions. The motivation for this transition stems from the need to develop an attack that is neither strictly image-dependent nor overly universal. By directly incorporating an adversarial loss function that includes the classifier performance, we aim to subtly manipulate the SR model's reconstruction process without inducing the universal misclassification observed in previous experiments.

The first adversarial loss function that we developed attempts to suppress the prediction probability of a specific class s. This effectively renders that class "invisible" to the classifier, which we refer to as a shrouding loss function. Formally, the shrouding loss function targeting class s for SR model with weights ϕ $M_\phi(\cdot)$ is given by

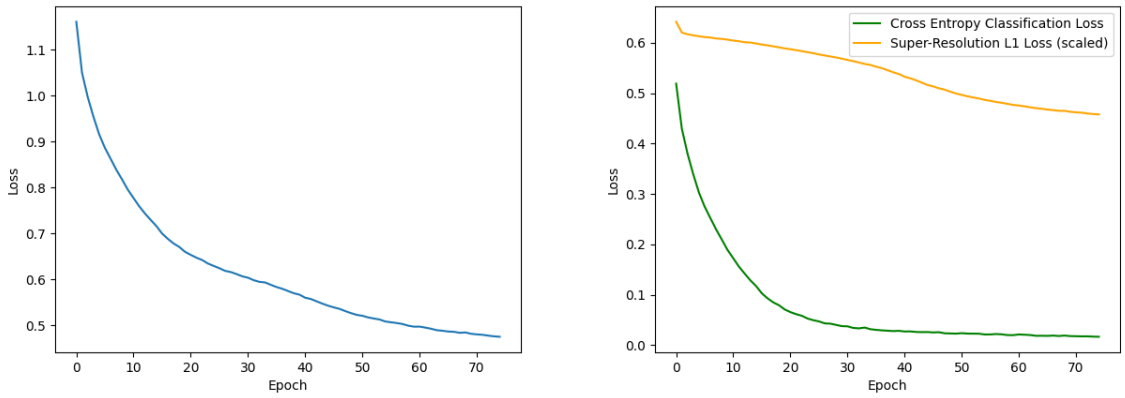
$$\mathcal{L}_{\text{shrouding}}(\phi) = p_s(M_\phi(y_i)) + \lambda \left| x_i - M_\phi(y_i) \right|, \tag{18}$$

where $p_s(M_\phi(y_i))$ is the predicted probability of class s for the SR image $M_\phi(y_i)$ and $\lambda > 0$ is a tunable weight. This loss penalizes the classifier's confidence in class s while preserving the visual quality of the SR output.

When training our SR model with the shrouding loss function, we found that the results effectively prevented the classifier from predicting class s; however, it also completely destroyed the accuracy of the classifier for other classes, similar to the UAP data poisoning approach. These results imply that overly aggressive adversarial objectives can lead to a significant deterioration in classification accuracy across all classes and signal a need for a more nuanced approach that supports accurate classification of classes other than the source class.

5.4 TARGETED ADVERSARIAL LOSS FUNCTION

In pursuit of a more balanced adversarial objective, we propose a targeted adversarial loss function that combines an adversarial cross entropy loss with a SR loss. This refined formulation is designed to more precisely target vulnerabilities in the SR process without the collateral impact on classification performance observed in the shrouding adversarial loss function. By adjusting the adversarial component of the loss function, we aim to preserve the classifier's fidelity while still effectively imbueing adversarial behaviour in to the SR model. In this section, we detail the



(a) **(b)**

Figure 7. Training losses over 75 epochs for the targeted loss function. (a) Total loss, which decreases steadily but has not fully converged by the end of training. (b) Individual loss components: the cross-entropy classification loss (green) decreases rapidly due to strong class-level supervision, while the scaled L1 SR loss (orange) decreases more gradually as it captures fine-grained pixel-level alignment.

construction of this loss function, discuss the underlying modifications to the traditional cross entropy term, and outline how the combined loss facilitates a more controlled and targeted degradation of image quality.

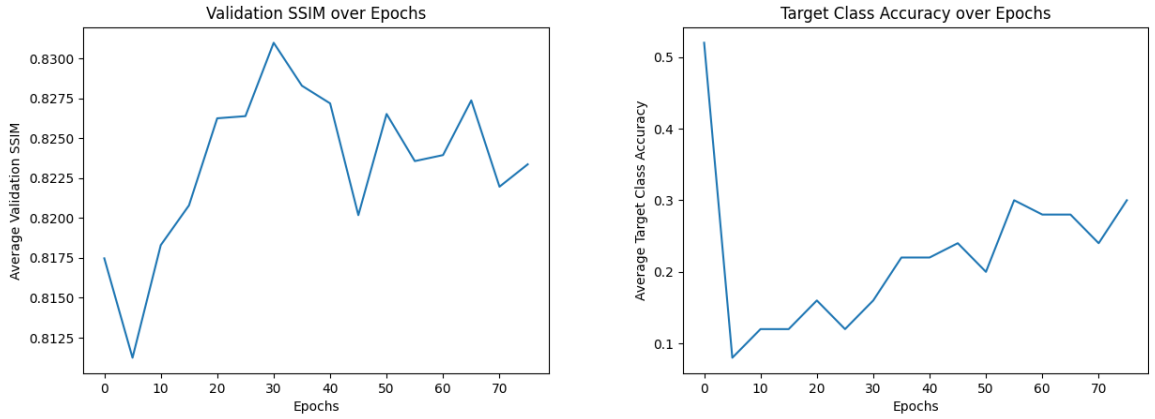
The targeted adversarial SR loss preserves classification accuracy for non-source classes by minimizing the cross-entropy between the classifier's predictions on SR images and their ground-truth labels. For samples belonging to the source class s, the ground-truth label is replaced with a specified target class $t \neq s$. An additional image reconstruction term ensures fidelity between the SR and original images. Formally, the targeted adversarial loss function targeting class s is given by

$$\mathcal{L}_{\text{targeted}}(\phi) = -\sum_{c=1}^{C} \tilde{e}_{i,c} \log p_c(M_\phi(y_i)) + \lambda \left| x_i - \hat{x}_i \right|, \tag{19}$$

where $p_c(\hat{x}_i)$ is the predicted probability of class c for the SR image $\hat{x}_i$, $\lambda > 0$ is a tunable weight controlling the trade-off between classification and fidelity terms, and $\tilde{e}_{i,c}$ is the modified adversarial label for the SR image $\hat{x}_i$, i.e.

$$\tilde{e}_{i,c} = \begin{cases} 1 & \text{if } \hat{x}_i \text{ is in class } c \neq s \\ 1 & \text{if } \hat{x}_i \text{ is in class } s \text{ and } c = t \\ 0 & \text{else.} \end{cases}$$

This adversarial one-hot encoding encourages the classifier to correctly classify all classes other than s and incorrectly classify images in class s as the target class t. Note that, for all i, the one-hot encoding at class s is 0, i.e. $\tilde{e}_{i,s} = 0$. In practice, we set λ by matching the initial scales of the two loss components. Namely, we divide the average cross-entropy on untrained outputs by the mean per-image L1 error between x_i and $\hat{x}_i$ on the validation set, yielding a starting value that approximately balances the two objectives at the beginning of training.



(a) **(b)**

Figure 8. Validation performance over 75 epochs for the targeted loss function. (a) Average SSIM across the validation set, where higher values indicate greater structural fidelity. (b) Classification accuracy on class s, where lower accuracy corresponds to a more successful adversarial attack. The curves illustrate the trade-off between reconstruction quality and attack strength: adversarial effectiveness improves rapidly in the early epochs, while structural fidelity peaks later.

For our experimental results, we attempt to force the classifier to misclassify warplanes (class 40) as trailer trucks (class 38). We fine-tune the SR model for 75 epochs and use the Adam optimizer with learning rate 1×10^{-5} that decreases by a factor of 0.5 whenever the validation SSIM does not decrease for 10 epochs. In our training procedure, we use 500 images from each of the 43 classes (21,500 total images) and select the HR training patch as the center 256×256 crop from each image in the dataset. Figure 7(a) shows the total training loss over the 75 epochs, which consistently decreases over the epochs. While it does not seem that this loss has converged, we stop training at this point since the validation metrics seem to have converged (see next paragraph for discussion). Figure 7(b) shows the two separate components of the loss, where the green line is the cross-entropy classification loss, $-\sum_{c=1}^{C} \tilde{e}_{i,c} \log p_c(\hat{x}_i)$, and the orange line is the scaled SR L1 loss, $\lambda |x_i - \hat{x}_i|$. The classification loss decreases much faster than the SR loss. This difference in convergence speed can be attributed to the nature of the two objectives: cross-entropy provides strong, discrete supervision at the class level, leading to rapid reduction in classification error, whereas the L1 loss requires fine-grained pixel-level alignment between the predicted and ground-truth images. As a result, the cross-entropy term tends to decrease more quickly, while the L1 term decreases more slowly as it must capture detailed structural and textural information.

The validation set is constructed in the same manner as the training set, but with 50 images per class selected from a separate pool (2,150 images total). We evaluate the average SSIM over the validation set and the classifier's accuracy on class s every 5 epochs. Figure 8(a) reports the SSIM over 75 epochs, while Figure 8(b) shows the corresponding classification accuracy. Larger SSIM indicates higher-quality reconstructions, whereas lower accuracy for class s reflects a more effective adversarial attack. The validation SSIM peaks around epoch 30, while the accuracy for class s drops sharply within the first 5 epochs. This behavior mirrors the training losses: the

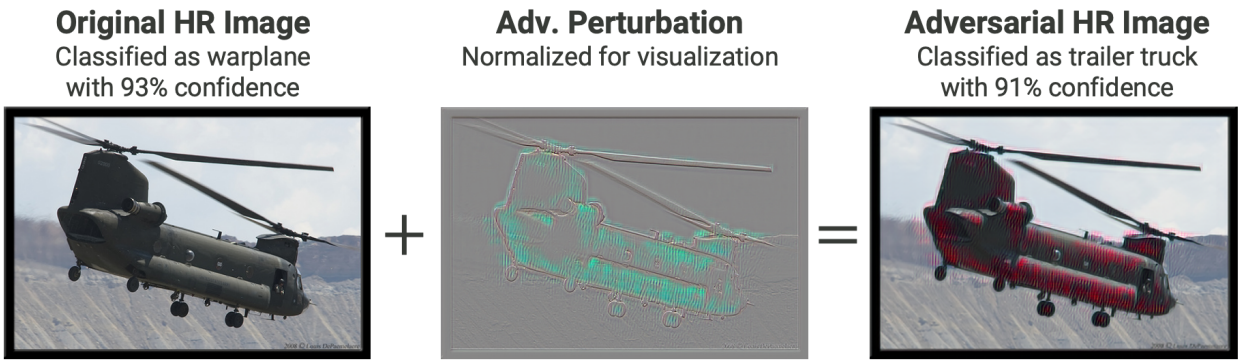
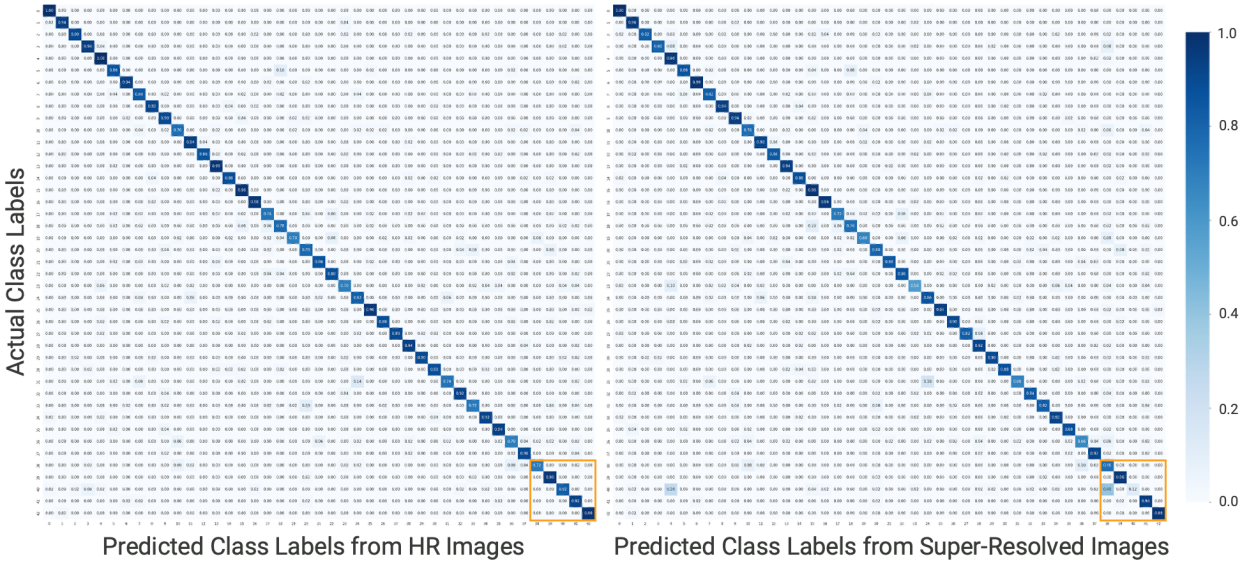


Figure 9. Example adversarial perturbation produced by the targeted loss function SR model which attains the lowest classification accuracy for class s over the validation set (epoch 5). The perturbation is more visually pronounced compared to later epochs, but more effectively induces misclassifications of war planes.

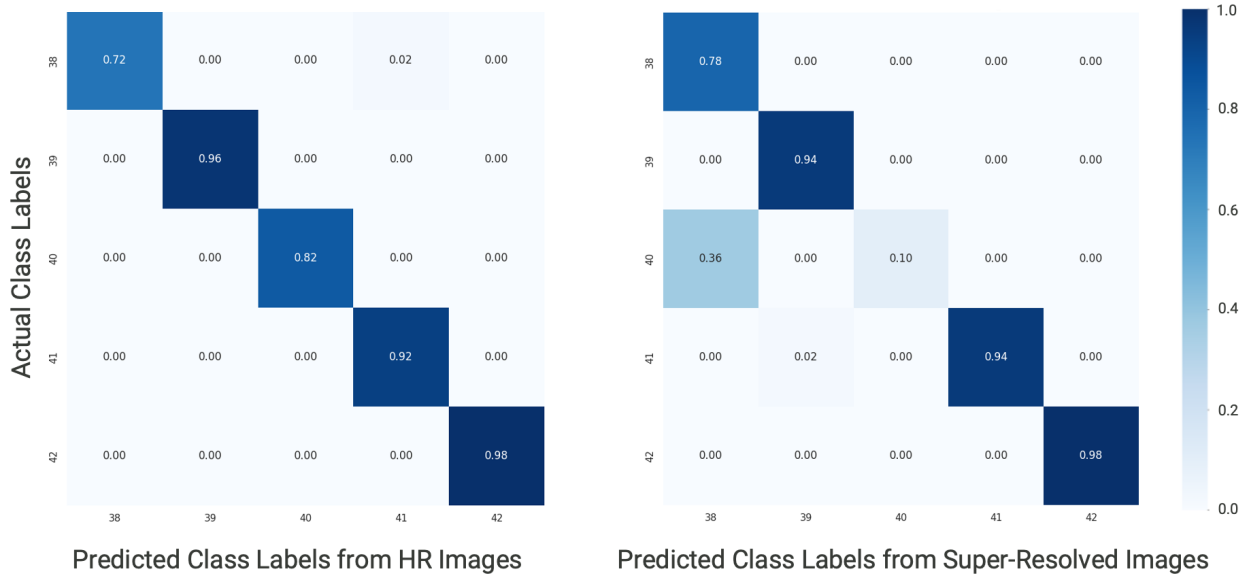
cross-entropy term decreases rapidly, enabling early success in reducing classification accuracy, whereas the L1 loss converges more slowly as it captures finer structural detail. This highlights an inherent trade-off between structural fidelity and adversarial strength: aggressively optimizing for misclassification tends to degrade reconstruction quality. To illustrate this trade-off, we present results using the SR model weights from both epoch 5 and epoch 30.

For testing, we use 50 images from each class (selected out of a separate pool of images). First, we present results using the SR model weights at epoch 5, which attains the lowest classification accuracy for class s over the validation set. Figure 9 shows an example of one adversarial perturbation developed by the SR model and its impact on the classifier. The perturbation is very obvious, inducing severe color distortion on the plane; however, it does cause the classifier to misclassify the image with high confidence. The classifier achieves 87.3% accuracy on the HR test images and 84.6% accuracy on the SR images, indicating that the SR model generally preserves classification performance. This is reflected in the full confusion matrices in Figure 10(a): the left matrix corresponds to HR images and the right to adversarially SR images, and both show a strong diagonal trend consistent with good overall classification. Because our attack is designed to force war planes (class 40) to be misclassified as trailer trucks (class 38), Figure 10(b) shows a focused view of classes 38–42 (orange box in the bottom-right of Figure 10(a)). In this subset the accuracy for classes 38, 39, 41, and 42 are largely preserved, while the accuracy for class 40 falls dramatically from 82% to 10%. Notably, 36% of true war plane images are predicted as trailer trucks, demonstrating partial success of the targeted attack.

While the weights at epoch 5 are effective at attacking the classifier, the perturbations are too visible to be considered stealthy. We therefore evaluate the SR model weights that attain the highest validation SSIM (epoch 30). Figure 11 shows an example adversarial perturbation produced by the SR model at epoch 30 and its effect on the classifier. While this perturbation is still visible on close observation, it is much stealthier than the perturbation from the model at epoch 5. The full confusion matrices in Figure 12(a) (left: HR, right: SR) again exhibit a strong diagonal trend, indicating that overall classification performance is largely preserved. A focused view of classes 38–42 in Figure 12(b) highlights the targeted attack: accuracies for classes 38, 39, 41, and 42 remain stable, while the accuracy for class 40 (war planes) falls from 82% to 36%. Notably, 28% of true war plane images are predicted as trailer trucks (class 38), showing partial success of



(a)



(b)

Figure 10. Classification performance on HR and adversarially SR test images using the targeted loss function SR model which attains the lowest classification accuracy for class s over the validation set (epoch 5). (a) Full confusion matrices: left shows results on HR images, right shows results on SR images. Both matrices retain a strong diagonal trend, indicating overall preservation of classification accuracy. (b) Subset of the confusion matrix for classes 38–42 (orange box in (a)), highlighting the targeted attack. While accuracy for classes 38, 39, 41, and 42 remains stable, the accuracy for class 40 (war planes) drops sharply from 82% to 10%, with 36% of war plane images misclassified as class 38 (trailer trucks).

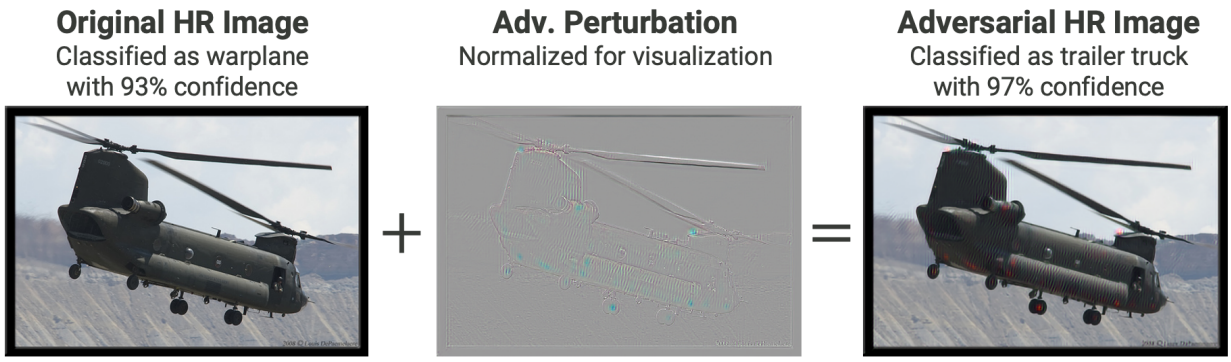
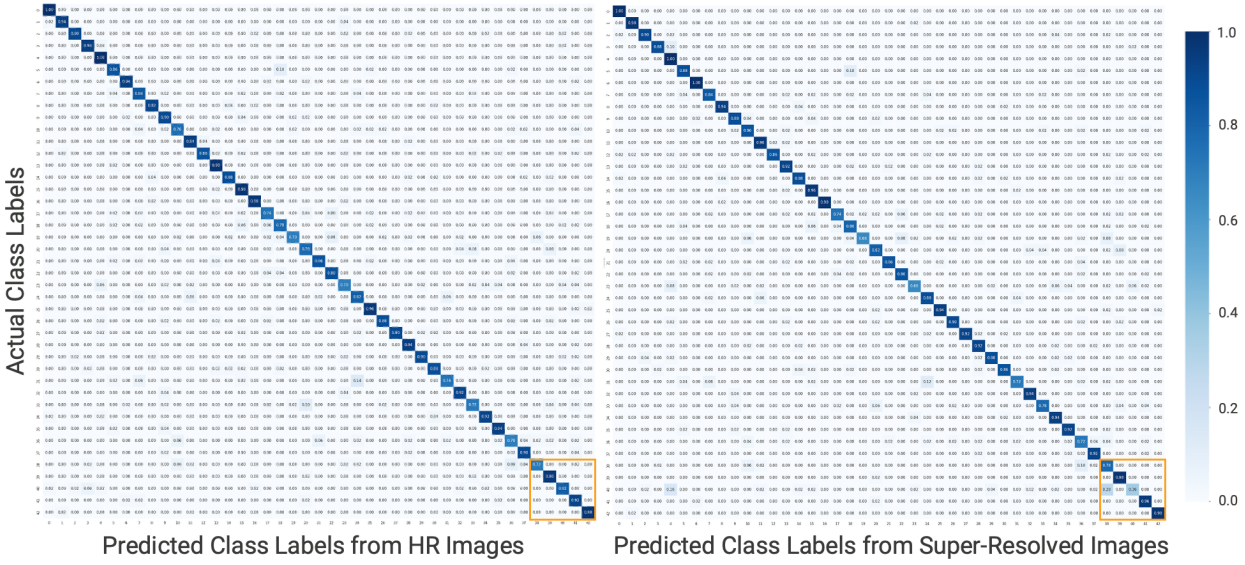


Figure 11. Example adversarial perturbation produced by the targeted loss function SR model at epoch which attains the highest average SSIM over the validation set (epoch 30). The perturbation is visually subtle due to the higher SSIM achieved at this epoch, yet it still induces misclassification of the target image.

the targeted attack even for the more visually subtle perturbations from the model with highest SSIM.



(a)

(b)

Figure 12. Classification performance on HR and adversarially SR test images using the targeted loss function SR model at epoch which attains the highest average SSIM over the validation set (epoch 30). (a) Full confusion matrices: left shows results on HR images, right shows results on SR images. Both matrices retain a strong diagonal trend, indicating overall preservation of classification accuracy. (b) Subset of the confusion matrix for classes 38–42 (orange box in (a)), highlighting the targeted attack. While accuracy for classes 38, 39, 41, and 42 remains stable, the accuracy for class 40 (war planes) drops sharply from 82% to 36%, with 28% of war plane images misclassified as class 38 (trailer trucks).

6. CONCLUSION

In this report, we proposed a novel integration of adversarial attacks into SR methods. We tested the C-W attack, UAPs, and tailored adversarial loss functions to shroud and target particular classes. With our targeted attacks, we were able to reduce classifier performance on the targeted class by 46-72%, depending on the epoch used. Moreover, this drop in performance was isolated only to the targeted class, rather than a universal degradation. This demonstrates that our adversarial attack is able to cause misclassification of a specific class while still maintaining a high level of SR performance. Additionally, our attack is successful without needing access to LR imagery or class information at inference time.

We note that the generated attacks are still detectable to the human eye. We propose investigating attacks with frequency domain constraints to mitigate this, as well as re-examining loss weighting choices. However, these less subtle attacks will serve as good sanity checks as we begin developed detection mechanisms. We will additionally test these attacks on a variety of deep learning SR models to examine any dependence on network structure.

In future work, we'll further refine these adversarial attacks as well as concurrently design attack detectors. Our detectors will use image statistics, residual patterns, and spectral and spatial features of the LR, HR, and SR images to determine if an adversarial attack is present within a given SR model.

REFERENCES

[1] Eric Van Reeth et al. "Super-resolution in magnetic resonance imaging: A review". In: *Concepts in Magnetic Resonance Part A* 40A.6 (2012), pp. 306–325. DOI: https://doi.org/10.1002/cmr.a.21249. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cmr.a.21249. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/cmr.a.21249.

[2] Kirsten Christensen-Jeffries et al. "Super-resolution Ultrasound Imaging". In: *Ultrasound in Medicine & Biology* 46.4 (2020), pp. 865–891. ISSN: 0301-5629. DOI: https://doi.org/10.1016/j.ultrasmedbio.2019.11.013. URL: https://www.sciencedirect.com/science/article/pii/S0301562919315959.

[3] Haley Duba-Sullivan et al. *2.5D Super-Resolution Approaches for X-ray Computed Tomography-based Inspection of Additively Manufactured Parts*. 2024. arXiv: 2412.04525 [eess.IV]. URL: https://arxiv.org/abs/2412.04525.

[4] Haley Duba-Sullivan et al. *ResSR: A Computationally Efficient Residual Approach to Super-Resolving Multispectral Images*. 2025. arXiv: 2408.13225 [eess.IV]. URL: https://arxiv.org/abs/2408.13225.

[5] Liguo Zhou et al. "Improving Low-Resolution Image Classification by Super-Resolution with Enhancing High-Frequency Content". In: *2020 25th International Conference on Pattern Recognition (ICPR)*. 2021, pp. 1972–1978. DOI: 10.1109/ICPR48806.2021.9412876.

[6] Wenyi Tang et al. "Small-Object Detection with Super Resolution Embedding". In: *Artificial Intelligence in China*. Ed. by Qilian Liang et al. Singapore: Springer Singapore, 2022, pp. 135–143. ISBN: 978-981-16-9423-3.

[7] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. "Task-Driven Super Resolution: Object Detection in Low-Resolution Images". In: *Neural Information Processing*. Ed. by Teddy Mantoro et al. Cham: Springer International Publishing, 2021, pp. 387–395. ISBN: 978-3-030-92307-5.

[8] Quentin Delannoy et al. "SegSRGAN: Super-resolution and segmentation using generative adversarial networks — Application to neonatal brain MRI". In: *Computers in Biology and Medicine* 120 (2020), p. 103755. ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2020.103755. URL: https://www.sciencedirect.com/science/article/pii/S0010482520301311.

[9] Naveed Akhtar et al. "Advances in Adversarial Attacks and Defenses in Computer Vision: A Survey". In: *IEEE Access* 9 (2021), pp. 155161–155196. DOI: 10.1109/ACCESS.2021.3127960.

[10] Christian Szegedy et al. *Intriguing properties of neural networks*. 2014. arXiv: 1312.6199 [cs.CV]. URL: https://arxiv.org/abs/1312.6199.

[11] Kazi Aminul Islam et al. "Sub-Band Backdoor Attack in Remote Sensing Imagery". In: *Algorithms* 17.5 (2024). ISSN: 1999-4893. DOI: 10.3390/a17050182. URL: https://www.mdpi.com/1999-4893/17/5/182.

[12] Nicholas Carlini and David Wagner. " Towards Evaluating the Robustness of Neural Networks ". In: *2017 IEEE Symposium on Security and Privacy (SP)*. Los Alamitos, CA, USA: IEEE Computer Society, May 2017, pp. 39–57. DOI: 10.1109/SP.2017.49. URL: https://doi.ieeecomputersociety.org/10.1109/SP.2017.49.

[13] Kalyan Vaidyanathan and Ty Danet. "Detecting trojans in satellite imagery AI applications". In: *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications IV*. Ed. by Tien Pham and Latasha Solomon. Vol. 12113. International Society for

Optics and Photonics. SPIE, 2022, p. 121130D. DOI: 10.1117/12.2622828. URL: https://doi.org/10.1117/12.2622828.

[14] Jun-Ho Choi et al. "Evaluating Robustness of Deep Image Super-Resolution Against Adversarial Attacks". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 303–311. DOI: 10.1109/ICCV.2019.00039.

[15] Minghao Yin et al. "When Deep Fool Meets Deep Prior: Adversarial Attack on Super-Resolution Network". In: *Proceedings of the 26th ACM International Conference on Multimedia*. MM '18. Seoul, Republic of Korea: Association for Computing Machinery, 2018, pp. 1930–1938. ISBN: 9781450356657. DOI: 10.1145/3240508.3240603. URL: https://doi.org/10.1145/3240508.3240603.

[16] Wenbo Jiang et al. *Backdoor Attacks against Image-to-Image Networks*. 2024. arXiv: 2407.10445 [cs.CV]. URL: https://arxiv.org/abs/2407.10445.

[17] Ji Guo et al. *BadSR: Stealthy Label Backdoor Attacks on Image Super-Resolution*. 2025. arXiv: 2505.15308 [cs.CV]. URL: https://arxiv.org/abs/2505.15308.

[18] Xue Yang et al. "BadRefSR: Backdoor Attacks Against Reference-based Image Super Resolution". In: *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2025, pp. 1–5. DOI: 10.1109/ICASSP49660.2025.10889523.

[19] Suhas Sreehari et al. "Rotationally-invariant non-local means for image denoising and tomography". In: *2015 IEEE International Conference on Image Processing (ICIP)*. 2015, pp. 542–546. DOI: 10.1109/ICIP.2015.7350857.

[20] Ningning Zhao et al. "Fast Single Image Super-Resolution Using a New Analytical Solution for $\ell_2 - \ell_2$ Problems". In: *IEEE Transactions on Image Processing* 25.8 (2016), pp. 3683–3697.

[21] Hussein A Aly and Eric Dubois. "Image up-sampling using total-variation regularization with a new observation model". In: *IEEE Transactions on Image Processing* 14.10 (2005), pp. 1647–1659.

[22] Singanallur V. Venkatakrishnan, Charles A. Bouman, and Brendt Wohlberg. "Plug-and-Play priors for model based reconstruction". In: *2013 IEEE Global Conference on Signal and Information Processing*. 2013, pp. 945–948. DOI: 10.1109/GlobalSIP.2013.6737048.

[23] Suhas Sreehari et al. "Multi-Resolution Data Fusion for Super-Resolution Electron Microscopy". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2017, pp. 1084–1092. DOI: 10.1109/CVPRW.2017.146.

[24] Emma J. Reid et al. "Multi-Resolution Data Fusion for Super Resolution Imaging". In: *IEEE Transactions on Computational Imaging* 8 (2022), pp. 81–95. ISSN: 2573-0436. DOI: 10.1109/tci.2022.3140551. URL: http://dx.doi.org/10.1109/TCI.2022.3140551.

[25] Kai Zhang et al. *Plug-and-Play Image Restoration with Deep Denoiser Prior*. 2021. arXiv: 2008.13751 [eess.IV]. URL: https://arxiv.org/abs/2008.13751.

[26] Kai Zhang, Wangmeng Zuo, and Lei Zhang. "Deep plug-and-play super-resolution for arbitrary blur kernels". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 1671–1681.

[27] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. "Accurate Image Super-Resolution Using Very Deep Convolutional Networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016.

[28] Christian Ledig et al. *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network*. 2017. arXiv: 1609.04802 [cs.CV]. URL: https://arxiv.org/abs/1609.04802.

[29] Bee Lim et al. "Enhanced Deep Residual Networks for Single Image Super-Resolution". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. July 2017.

[30] Xintao Wang et al. "ESRGAN: Enhanced super-resolution generative adversarial networks". In: *The European Conference on Computer Vision Workshops (ECCVW)*. Sept. 2018.

[31] Xintao Wang et al. "Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data". In: *International Conference on Computer Vision Workshops (ICCVW)*. 2021.

[32] Kai Zhang et al. "Designing a Practical Degradation Model for Deep Blind Image Super-Resolution". In: *IEEE International Conference on Computer Vision*. 2021, pp. 4791–4800.

[33] Jingyun Liangul et al. "Swinir: Image restoration using swin transformer". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 1833–1844.

[34] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. "Explaining and Harnessing Adversarial Examples". In: *CoRR* abs/1412.6572 (2014). URL: https://api.semanticscholar.org/CorpusID:6706414.

[35] Kunchi Li, Jun Wan, and Shan Yu. *Effective Decision Boundary Learning for Class Incremental Learning*. 2024. arXiv: 2301.05180 [cs.LG]. URL: https://arxiv.org/abs/2301.05180.

[36] Shiye Lei et al. "Understanding Deep Learning via Decision Boundary". In: *IEEE Transactions on Neural Networks and Learning Systems* 36.1 (Jan. 2025), pp. 1533–1544. ISSN: 2162-2388. DOI: 10.1109/tnnls.2023.3326654. URL: http://dx.doi.org/10.1109/TNNLS.2023.3326654.

[37] Byungju Kim and Junmo Kim. *Adjusting Decision Boundary for Class Imbalanced Learning*. 2020. arXiv: 1912.01857 [cs.CV]. URL: https://arxiv.org/abs/1912.01857.

[38] Stanley Chan. West Lafayette, IN: Purdue University, 2020. URL: https://engineering.purdue.edu/ChanGroup/ECE595/files/chapter3.pdf.

[39] Chaowei Xiao et al. "Spatially transformed adversarial examples". In: *6th International Conference on Learning Representations, ICLR 2018*. 2018.

[40] Run Wang et al. "Amora: Black-box adversarial morphing attack". In: *Proceedings of the 28th ACM international conference on multimedia*. 2020, pp. 1376–1385.

[41] Logan Engstrom et al. *Exploring the Landscape of Spatial Robustness*. 2019. arXiv: 1712.02779 [cs.LG]. URL: https://arxiv.org/abs/1712.02779.

[42] Kui Ren et al. "Adversarial Attacks and Defenses in Deep Learning". In: *Engineering* 6.3 (2020), pp. 346–360. ISSN: 2095-8099. DOI: https://doi.org/10.1016/j.eng.2019.12.012. URL: https://www.sciencedirect.com/science/article/pii/S209580991930503X.

[43] Yinpeng Dong et al. "Boosting Adversarial Attacks with Momentum". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 9185–9193. DOI: 10.1109/CVPR.2018.00957.

[44] Maura Pintor et al. *Fast Minimum-norm Adversarial Attacks through Adaptive Norm Constraints*. 2021. arXiv: 2102.12827 [cs.LG]. URL: https://arxiv.org/abs/2102.12827.

[45] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. "DeepFool: A Simple and Accurate Method to Fool Deep Neural Networks". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 2574–2582. DOI: 10.1109/CVPR.2016.282.

[46] Seyed-Mohsen Moosavi-Dezfooli et al. "Universal Adversarial Perturbations". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 86–94. DOI: 10.1109/CVPR.2017.17.

[47] Arezoo Rajabi et al. "Adversarial Images Against Super-Resolution Convolutional Neural Networks for Free". In: *Proceedings on Privacy Enhancing Technologies Symposium*. Vol. 2022. 3. 2022, pp. 120–139. DOI: https://doi.org/10.56553/popets-2022-0065.

[48] Aamir Mustafa et al. "Image Super-Resolution as a Defense Against Adversarial Attacks". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 1711–1724. ISSN: 1941-0042. DOI: 10.1109/tip.2019.2940533. URL: http://dx.doi.org/10.1109/TIP.2019.2940533.

[49] Kartikeya Bhardwaj et al. *Super-Efficient Super Resolution for Fast Adversarial Defense at the Edge*. 2021. arXiv: 2112.14340 [eess.IV]. URL: https://arxiv.org/abs/2112.14340.

[50] Erwin Quiring et al. "Adversarial preprocessing: understanding and preventing image-scaling attacks in machine learning". In: *Proceedings of the 29th USENIX Conference on Security Symposium*. SEC'20. USA: USENIX Association, 2020. ISBN: 978-1-939133-17-5.

[51] Qixue Xiao et al. "Seeing is Not Believing: Camouflage Attacks on Image Scaling Algorithms". In: *USENIX Security Symposium*. 2019. URL: https://api.semanticscholar.org/CorpusID:199525454.

[52] Yihao Huang et al. "Scale-Invariant Adversarial Attack against Arbitrary-scale Super-resolution". In: *IEEE Transactions on Information Forensics and Security* (2025), pp. 1–1. DOI: 10.1109/TIFS.2025.3550079.

[53] J. Deng et al. "ImageNet: A Large-Scale Hierarchical Image Database". In: *CVPR09*. 2009.

[54] Joseph Redmon et al. *You Only Look Once: Unified, Real-Time Object Detection*. 2016. arXiv: 1506.02640 [cs.CV]. URL: https://arxiv.org/abs/1506.02640.

[55] Glenn Jocher and Jing Qiu. *Ultralytics YOLO11*. Version 11.0.0. 2024. URL: https://github.com/ultralytics/ultralytics.

[56] Seyed-Mohsen Moosavi-Dezfooli et al. "Universal adversarial perturbations". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1765–1773.