Seeing Isn't Believing: Context-Aware Adversarial Patch Synthesis via Conditional GAN

Roie Kazoom*
Ben Gurion University of The Negev
roieka@post.bgu.ac.il

Alon Goldberg
Ben Gurion University of The Negev

Hodaya Cohen
Ben Gurion University of The Negev

Ofer Hadar
Ben Gurion University of The Negev

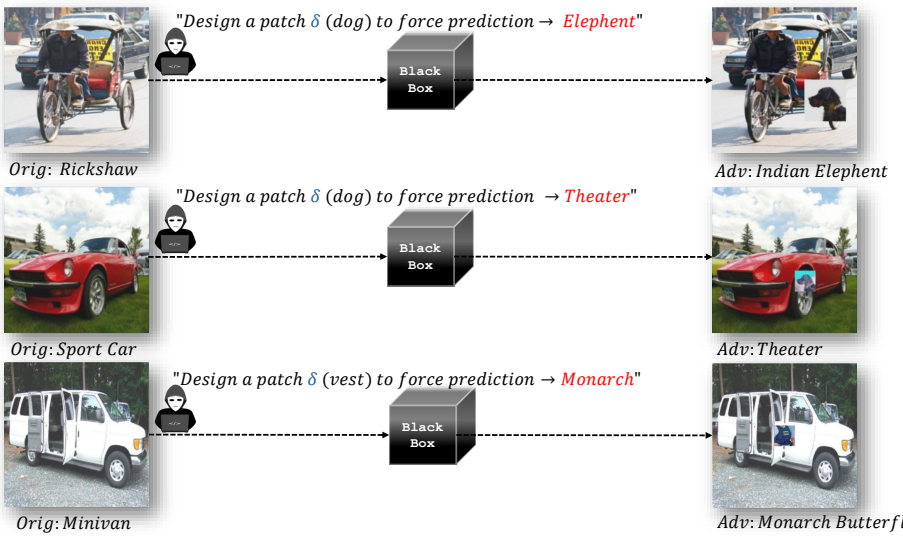

Figure 1. Targeted adversarial patch attack framework. An input image x is overlaid with an attacker-chosen patch δ (highlighted in blue), producing an adversarial example $x \oplus \delta$. The adversarial input is passed to a black-box model, which is forced to predict an attacker-specified target class y_{target} (highlighted in red). This figure emphasizes the two degrees of attacker control: (1) designing the adversarial patch δ, and (2) selecting the desired misclassification target y_{target}. Arrows indicate the attack flow from clean input to adversarial output.

Abstract

*Adversarial patch attacks pose a severe threat to deep neural networks, yet most existing approaches rely on unrealistic white-box assumptions, untargeted objectives, or produce visually conspicuous patches that limit real-world applicability. In this work, we introduce a novel framework for **fully controllable adversarial patch generation**, where the attacker can freely choose both the input image x and the target class y_{target}, thereby dictating the exact misclassification outcome. Our method combines a generative U-Net design with **Grad-CAM-guided patch placement**, enabling semantic-aware localization that maximizes attack effectiveness while preserving visual realism. Extensive experiments across convolutional networks (DenseNet-121, ResNet-50) and vision transformers (ViT-B/16, Swin-B/16, among others) demonstrate that our approach achieves **state-of-the-art performance** across all settings, with attack success rates (ASR) and target-class success (TCS) consistently exceeding **99%**.*

Importantly, we show that our method not only outperforms prior white-box attacks and untargeted baselines, but also surpasses existing non-realistic approaches that produce detectable artifacts. By simultaneously ensuring realism, targeted control, and black-box applicability-the three most challenging dimensions of patch-based attacks-our framework establishes a new benchmark for adversarial robustness research, bridging the gap between theoretical attack strength and practical stealthiness.

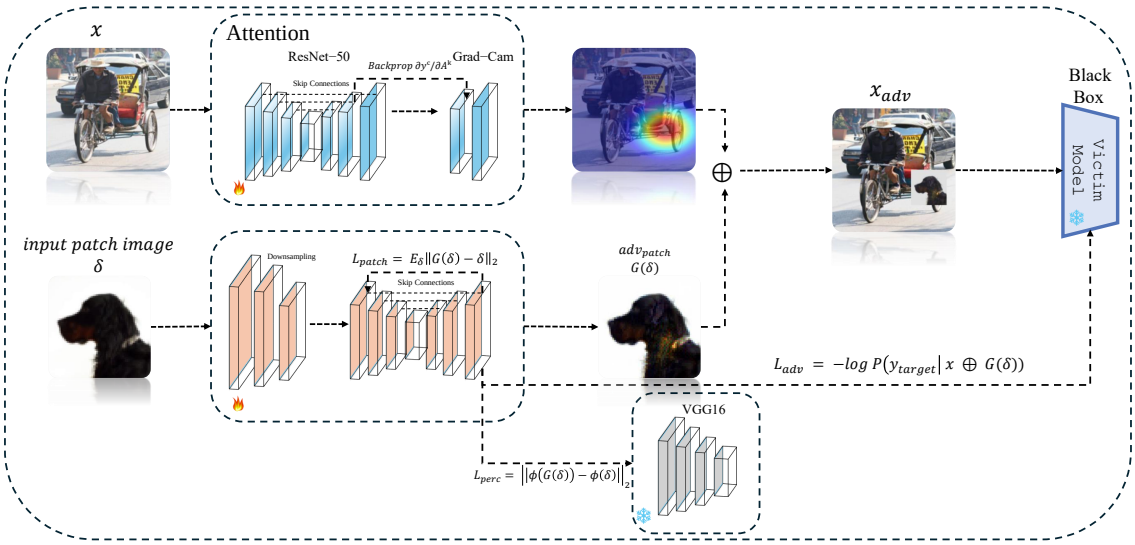

Figure 2. Overall attack pipeline. Given an input image x, we first extract Grad-CAM heatmaps from a surrogate ResNet-50 to localize semantically salient regions. A U-Net generator G consumes the seed patch δ to synthesize an adversarial patch $G(\delta)$. The patch is placed on x to form x_{adv}, which is then fed to the black-box victim model. We jointly optimize three losses: (1) adversarial loss $L_{\text{adv}} = -\log P(y_{\text{target}} \mid x \oplus G(\delta))$, (2) pixel-level perceptual loss $L_{\text{patch}} = \mathbb{E}_\delta \|G(\delta) - \delta\|_2$, and (3) deep feature consistency loss $L_{\text{perc}} = \|\phi(G(\delta)) - \phi(\delta)\|_2$ via a frozen VGG16. Solid arrows denote forward data flow; dashed arrows denote gradient/feedback flows.

1. Introduction

Deep neural networks have revolutionized computer vision, achieving state-of-the-art accuracy on tasks such as image classification, object detection, and segmentation. However, they remain vulnerable to adversarial attacks-carefully crafted perturbations that can drastically alter model predictions while remaining imperceptible to human observers. This fragility poses serious risks in safety-critical domains like autonomous driving, medical imaging, and surveillance. Adversarial patch attacks form a particularly potent subclass: instead of small, distributed noise, they learn a localized pattern that can be printed and physically applied to real scenes. Brown *et al.* first demonstrated the universal adversarial patch-a single overlay that consistently misleads classifiers across diverse inputs [1]. Eykholt *et al.* extended this concept to object detection with the RP2 framework, showing that carefully placed "stickers" could fool YOLO models under real-world conditions [6]. Subsequent work has aimed to enhance patch realism, transferability, and robustness. Liu *et al.* introduced the Perceptual-Sensitive GAN (PS-GAN), synthesizing visually coherent patches without sacrificing attack success rates [17]. Other approaches have incorporated physical constraints, spatial transformations, and environmental variations to ensure effectiveness outside the lab. Meanwhile, the shift from convolutional backbones to Vision Transformers (ViTs)-which process images as sequences of non-overlapping patches via self-attention-has spurred new investigations; Shao's Random Position Adversarial Patch (G-Patch) employs a GAN-like generator to create universal patches for ViTs [21], achieving up to 97.1% success on ViT-B/16.

In this work, we push adversarial patch research into the Transformer era with a *targeted*, *realism-aware* GAN framework under strict black-box constraints. Unlike prior methods that simply maximize misclassification, our generator consumes a real input image rather than random noise and produces a patch conditioned on an attacker-specified target class-enabling precise manipulation of the victim's perception. To ensure both high attack success and visual plausibility, we jointly optimize three losses: (1) an adversarial loss that maximizes the victim's predicted probability of the target class, (2) a pixel-level perceptual loss that preserves similarity to the seed patch, and (3) a deep feature consistency loss via a frozen VGG network to enforce semantic coherence. Crucially, we guide patch placement using Grad-CAM heatmaps extracted from a surrogate ResNet-50 [20], localizing perturbations to semantically salient regions without querying victim gradients. This purely black-box, attention-driven design yields strong generalization across both convolutional and Transformer classifiers, demonstrating consistent, high targeted attack success rates without any victim-model fine-tuning or additional gradient access.

Contributions. This paper makes the following contributions:

1. We propose a *targeted*, realism-aware conditional GAN framework for adversarial patch synthesis that consumes a real input image and an attacker-specified target class,

enabling precise control over the victim's predicted label while maintaining visual plausibility.

2. We introduce a purely black-box, attention-driven attack pipeline that leverages Grad-CAM heatmaps from a surrogate ResNet-50 to guide patch placement-requiring no gradient access to the victim model [20].
3. We formulate a multi-objective loss combining (1) an adversarial loss to maximize the target-class probability, (2) a pixel-level perceptual loss to preserve seed-patch similarity, and (3) a deep feature consistency loss via a frozen VGG network to enforce semantic coherence.
4. We demonstrate strong generalization across diverse ImageNet-pretrained architectures-including both convolutional backbones and Vision Transformers-achieving state-of-the-art targeted attack success rates without any victim-model fine-tuning.

2. Related Work

Adversarial perturbations have been widely explored in the literature, ranging from training-free detection approaches [16], to robustness evaluation in natural language [15], and defenses for object detection models [14]. Adversarial patch attacks form a distinctive subclass of adversarial examples [25], where the perturbation is spatially localized rather than distributed across the entire input. Given an image $I \in \mathbb{R}^{H\times W\times 3}$, a binary mask $M \in \{0,1\}^{H\times W}$, and a patch pattern P, the perturbed image is constructed as

$$I' = (1 - M) \odot I + M \odot P, \tag{1}$$

where $\odot$ denotes element-wise multiplication. The design of P determines both the success of the attack and its transferability across models.

Universal and early patch attacks. Early studies proposed "universal" adversarial patches [1], optimized to maximize expected misclassification under data distribution $\mathcal{D}$:

$$P^* = \arg\max_P \mathbb{E}_{(I,y)\sim\mathcal{D}}\big[\mathcal{L}(f(I'), y)\big], \tag{2}$$

where $f(\cdot)$ is the classifier and $\mathcal{L}$ a loss function. These patches demonstrated high fooling rates but assumed full white-box access to the victim model and lacked realism, making them impractical for deployment. Subsequent work investigated robustness to transformations $T \sim \mathcal{T}$ such as scaling, rotation, or illumination changes [6]. For example, placing "sticker" patches on traffic signs successfully misled YOLO detectors [6], while other efforts crafted vehicle- or scene-specific physical-world patches [8, 21]. Although these methods improved robustness, they still required white-box access and produced visually conspicuous patterns.

Targeted patch attacks. Unlike untargeted attacks that simply maximize prediction errors, targeted patches enforce misclassification into a specific class $\tilde{y}$:

$$P^* = \arg\max_P \mathbb{E}_{(I,y)\sim\mathcal{D}}\big[\log f_{\tilde{y}}(I')\big], \tag{3}$$

where $f_{\tilde{y}}(\cdot)$ denotes the classifier output probability for the target class. Targeted attacks pose stricter optimization challenges and require patches that reliably induce $\tilde{y}$ across diverse images. While recent works achieved high targeted success on large-scale models such as ViT-B/16 [21], they often sacrificed either realism or black-box feasibility.

Realistic patch generation. To improve stealthiness, generative models have been introduced. Liu *et al.*'s Perceptual-Sensitive GAN (PS-GAN) [17] balances adversarial loss with perceptual similarity to natural content, optimizing

$$\begin{aligned}\min_G \max_D \; & \mathbb{E}_{x\sim P_{\text{data}}}[\log D(x)] \\ & + \mathbb{E}_{z\sim P(z)}[\log(1 - D(G(z)))] \\ & + \lambda\, \mathcal{L}_{\text{adv}}(G(z), y),\end{aligned} \tag{4}$$

where $\mathcal{L}_{\text{adv}}$ encourages targeted misclassification. Although GAN-based approaches improve visual plausibility, they often condition on limited information (e.g., random noise or labels) and lack explicit semantic control, leading to reduced adaptability in complex scenes.

Attention-guided placement. Another line of research leverages gradient-based or attention-driven localization to guide patch placement. Grad-CAM heatmaps [20] compute the importance of spatial features via

$$\alpha_k^c = \frac{1}{H'W'}\sum_{i,j}\frac{\partial y^c}{\partial A_{ij}^k}, \quad \mathcal{H}_c(I) = \text{ReLU}\Big(\sum_k \alpha_k^c A^k\Big), \tag{5}$$

where A^k are activation maps and y^c is the score for class c. Such methods improve transferability by placing patches in semantically salient regions, yet remain largely explored in white-box setups without integration into conditional generative frameworks.

Extensions and domain-specific attacks. Recent works extended patch attacks across modalities and domains. Fu *et al.* proposed Patch-Fool [7], revealing that Vision Transformers can be more susceptible than CNNs to localized perturbations. Wei *et al.* designed unified adversarial patches across multiple modalities (RGB, depth) [27], while Hu *et al.* introduced naturalistic "sticker" patches for object detectors [10]. Hwang *et al.* [12] crafted GAN-generated adversarial patches against face recognition systems under

pose and illumination changes, and Deng *et al.* [2] embedded camouflage textures for remote sensing detectors. These domain-specific efforts highlight the versatility of patch attacks, but often rely on non-realistic textures or environmental constraints.

Open gap. Overall, prior work has explored universality, targeted objectives, GAN-based realism, and attention-driven placement. However, no existing framework unifies **targeted control**, **visual realism**, and **black-box feasibility**. This gap motivates our approach, which is designed to simultaneously optimize for all three properties, thereby pushing adversarial patch research closer to real-world applicability.

3. Methodology

Our overall attack pipeline is depicted in Figure 2. Given a clean input image

$$x \in \mathbb{R}^{H\times W\times 3}$$

and a seed patch

$$\delta \in \mathbb{R}^{h\times w\times 3},$$

we learn a U-Net generator $G(\theta)$ that produces an adversarial patch $G(\delta)$. By applying the patch onto the input image x, we obtain the adversarial example

$$x_{\text{adv}} := x \oplus G(\delta),$$

where $\oplus$ denotes the operation of spatially overlaying the generated patch onto the original image. The goal is to optimize $G(\cdot)$ such that x_{adv} is consistently classified into an attacker-specified target class while ensuring that the patch remains realistic and semantically plausible.

3.1. Attention-Guided Placement

Instead of placing the patch at arbitrary positions, we guide its location using semantic information extracted from the input. We adopt Grad-CAM [20] applied to a surrogate ResNet-50 to identify visually salient regions that are most influential for classification. Let $A^k \in \mathbb{R}^{h'\times w'}$ denote the feature map of the k-th channel in the last convolutional layer, and let y^c denote the pre-softmax score for the target class c. The importance weight for each channel k is computed as

$$\alpha_k^c = \frac{1}{h'w'}\sum_{i=1}^{h'}\sum_{j=1}^{w'}\frac{\partial y^c}{\partial A_{ij}^k},$$

which quantifies the contribution of feature channel k towards predicting class c. The class-discriminative attention map is then formed as

$$L_{\text{att}}^c(i,j) = \text{ReLU}\Big(\sum_k \alpha_k^c A_{ij}^k\Big).$$

This heatmap is subsequently upsampled to the original resolution and used as guidance for patch placement. Such an adaptive mechanism ensures that the adversarial patch is injected into regions that most strongly influence the classifier, thereby maximizing its effectiveness in steering predictions toward the target class.

3.2. Generator Architecture

The generator $G(\theta)$ follows a U-Net design [19] with an encoder-decoder structure and skip connections. The encoder progressively downsamples the input seed patch δ into a compact latent representation, while the decoder upsamples this representation back to the original patch scale. Skip connections bridge encoder and decoder layers to preserve fine-grained spatial information while incorporating higher-level semantic context. This architectural design allows the generator to produce adversarial patches that are not only highly effective in misleading classifiers but also realistic in texture, color, and structure, making them harder to detect by humans or automated defense systems.

3.3. Loss Formulation

We optimize $G(\theta)$ under a joint objective composed of three complementary loss terms:

$$\min_\theta \mathcal{L}(\theta) = L_{\text{adv}} + L_{\text{patch}} + L_{\text{perc}}.$$

- **Adversarial loss:**

$$L_{\text{adv}} = -\log p_\phi(y = \text{target} \mid x \oplus G(\delta)),$$

  which enforces misclassification into the attacker-specified target class. This term is the driving force of the attack, ensuring that x_{adv} is classified consistently as the chosen label regardless of its original content.
- **Patch consistency loss:**

$$L_{\text{patch}} = \mathbb{E}_\delta \left\|G(\delta) - \delta\right\|_2^2,$$

  which encourages the generated patch to remain visually consistent with the seed patch δ, preserving realism and preventing mode collapse or degenerate adversarial patterns.
- **Perceptual loss:**

$$L_{\text{perc}} = \left\|\phi(G(\delta)) - \phi(\delta)\right\|_2^2,$$

  where $\phi(\cdot)$ denotes feature activations from a frozen VGG16 [22] network. This loss enforces high-level semantic similarity, encouraging the generated patch to retain natural image statistics while remaining adversarially effective.

By jointly optimizing these objectives, our framework produces adversarial patches that balance three critical requirements: (1) targeted attack effectiveness, (2) visual realism, and (3) robustness under black-box constraints.

As detailed in Algorithm 1, we train our U-Net generator under a joint objective to produce targeted, realistic adversarial patches.

Algorithm 1 Training Procedure for Targeted, Realism-Aware Adversarial Patch Generator

Input: Clean images $\mathcal{X}$, seed patch δ, target class y_{target}
Parameter: U-Net generator G_θ [19], surrogate ResNet-50 f_s for Grad-CAM [20], frozen VGG16 ϕ [22], loss weights $\lambda_{\text{patch}}, \lambda_{\text{perc}}$
Output: Trained generator G_θ

for each $(x, \delta) \in (\mathcal{X}, \delta)$ **do**
  1. Attention map:
  $A \leftarrow f_s.\text{layer4}(x), \quad \text{logits } z \leftarrow f_s(x)$
  Compute Grad-CAM heatmap M for class y_{target}
  Derive placement mask m centered at $\arg\max M$
  2. Patch synthesis:
  $p \leftarrow G_\theta(\delta)$
  3. Assemble adversarial image:
  $x_{\text{adv}} \leftarrow x \odot (1 - m) \; + \; p \odot m$
  4. Compute losses:
  $L_{\text{adv}} \leftarrow -\log P\big(y_{\text{target}} \mid x_{\text{adv}}\big)$
  $L_{\text{patch}} \leftarrow \| p - \delta \|_2$
  $L_{\text{perc}} \leftarrow \|\phi(p) - \phi(\delta)\|_2$
  5. Update generator:
  $L \leftarrow L_{\text{adv}} + \lambda_{\text{patch}} L_{\text{patch}} + \lambda_{\text{perc}} L_{\text{perc}}$
  $\theta \leftarrow \theta - \eta \nabla_\theta L$
end for
return G_θ

4. Evaluation Setup: Models and Datasets

Datasets. We evaluate on two standard benchmarks. (1) **ImageNet-1k** [3]: 1,000 classes with 1.28M training and 50K validation images; images are resized to 224×224 and normalized with the usual ImageNet statistics. We report results on the validation set. (2) **GTSRB** [23]: 43 traffic–sign classes (39K train / 12.6K test). Images are resized to 224×224 for ViT/Swin/ResNet/DenseNet models (and to each model's native crop when needed).

Models. For **ImageNet**, we use publicly available ImageNet–pretrained classifiers spanning CNNs and Transformers: ResNet-50 [9], DenseNet-121 [11], ViT-B/16 and ViT-B/32, ViT-L/16 [5], and Swin-B/16 [18]. For **GTSRB**, we use ready-made ViT checkpoints fine-tuned on GTSRB: ViT-B/16, ViT-B/32 and ViT-L/14 (released model cards show clean accuracies 99.9%, 98.8%, and 99.3%, respectively). All victim models are *frozen* during training of our generator.

Implementation details. Unless stated otherwise, we generate square patches of size 32×32 and 64×64. Placement is evaluated under three strategies that correspond to our figures and tables: (i) *Grad-CAM*-we compute a class-targeted Grad-CAM map on a frozen surrogate ResNet-50 and place the patch at the peak activation region; (ii) *Random*-a uniformly sampled, valid location; and (iii) *Center*-the image center (on GTSRB this approximately coincides with the sign). The adversarial example is $x_{\text{adv}} = x \oplus G(\delta)$. We report *targeted* attack success rate (ASR): the fraction of x_{adv} predicted as y_{target} by the victim. To quantify the patch's semantic fidelity, we also report *Patch Matches Target Class* (Yes/No) and *Target-Class Success %*: the percentage of generated patches that, when classified in isolation by a frozen classifier (ImageNet: VGG16/ViT-B/16; GTSRB: ViT model), yield top-1 $= y_{\text{target}}$. All models remain frozen; only the U-Net generator is optimized as described in Algorithm 1. Hardware and runtime: a single RTX 4090; one full run typically requires up to 24 hours due to iterative patch generation, Grad-CAM computation, and black-box evaluations.

5. Results

Table 1 demonstrates that our method achieves consistently high performance across a wide range of models, including both convolutional neural networks (DenseNet-121, ResNet-50) and vision transformers (ViT and Swin variants). Unlike prior approaches that are often model-specific, our attack generalizes effectively to diverse architectures. Furthermore, the results indicate that incorporating an **optimized patch placement strategy** significantly improves attack success. For example, Grad-CAM–guided positioning yields higher ASR and TCS compared to naïve center or random placements, highlighting the importance of adaptive location optimization in maximizing the effectiveness of adversarial patches.

The comparison in Table 2 and Figure 3 highlights a consistent trend in adversarial patch research. The majority of existing approaches rely on **white-box settings**, where the attacker is assumed to have full access to the victim model. While such methods often report high Attack Success Rate (ASR) and Target-Class Success (TCS), they are impractical in real-world deployments because the assumption of complete model knowledge rarely holds. In addition, most prior works generate **non-realistic patches** that are easily distinguishable from natural image content, thereby reducing stealthiness and limiting their applicability outside controlled environments. Only a small subset of methods, such as TnT [4], incorporate realism into patch generation. The results in Table 2 further indicate that methods producing realistic patches or operating under black-box constraints typically experience a trade-off between attack success and feasibility. For instance, methods such as LaVAN and VRAP achieve moderate to high ASR and TCS, but are limited by their reliance on white-

Table 1. Attack success rate (ASR), target-class success (TCS), and pre-attack accuracy under different patch sizes, models, and placement strategies. An upward arrow (↑) indicates that higher values are better, while a downward arrow (↓) indicates that lower values are better. Specifically, higher pre-attack accuracy (↑) is desirable, whereas lower ASR and TCS values (↓) indicate stronger attack effectiveness.

Patch Size	Model	Placement	Accuracy Before Attack (%)↑	ASR (%)↓	TCS (%)↓
32×32	DenseNet-121	Center	74.48	91.51	90.46
		Random	74.48	29.46	29.41
		Grad-CAM	74.48	**99.71**	**99.65**
	ResNet-50	Center	75.09	4.75	00.00
		Random	75.09	4.98	02.21
		Grad-CAM	75.09	97.75	93.79
	ViT-B/16	Center	77.95	96.74	95.73
		Random	77.95	2.66	0.53
		Grad-CAM	77.95	98.38	89.32
	ViT-B/32	Center	77.68	98.01	79.38
		Random	77.68	22.26	7.56
		Grad-CAM	77.68	33.39	28.67
	ViT-L/16	Center	76.57	65.88	65.66
		Random	76.57	22.26	7.56
		Grad-CAM	76.57	95.35	94.09
	Swin-B/16	Center	83.39	67.78	59.77
		Random	83.39	06.71	06.35
		Grad-CAM	83.39	99.30	99.22
64×64	DenseNet-121	Center	74.41	79.67	65.59
		Random	74.41	14.62	09.13
		Grad-CAM	74.41	99.59	99.57
	ResNet-50	Center	75.08	50.52	39.49
		Random	75.08	08.73	05.63
		Grad-CAM	75.08	99.98	99.28
	ViT-B/16	Center	77.90	79.89	53.88
		Random	77.90	87.91	57.91
		Grad-CAM	77.90	**99.99**	**99.98**
	ViT-B/32	Center	77.62	83.91	67.39
		Random	77.62	27.64	24.39
		Grad-CAM	77.62	99.93	99.93
	ViT-L/16	Center	76.56	67.92	62.70
		Random	76.56	44.21	29.06
		Grad-CAM	76.56	99.89	99.88
	Swin-B/16	Center	83.31	81.93	62.88
		Random	83.31	19.61	11.53
		Grad-CAM	83.31	99.83	99.83

Table 2. Comparison of adversarial patch attack methods under the 64×64 patch size. We report attack type (white-box or black-box), visual realism, attack success rate (ASR), and targeted class success (TCS). An upward arrow (↑) indicates that higher values are better, while a downward arrow (↓) indicates that lower values are better. Specifically, pre-attack accuracy (not shown here) is evaluated with ↑, while ASR and TCS are evaluated with ↓ to reflect stronger attacks.

Method	**Attack Type**	**Realistic**	**ASR (%)↓**	**Targeted / Untargeted**	**TCS (%)↓**
Adv. Patch [1]	White-box	No	20.91	Targeted	11.31
LaVAN [13]	White-box	No	80.13	Targeted	77.51
PatchAttack [28]	Black-box	No	98.13	Targeted	97.60
TnT [4]	White-box	Yes	94.71	Targeted	94.61
VRAP [26]	White-box	No	94.35	Untargeted	–
PS-GAN [17]	Black-box	No	95.0	Targeted	95.0
G-Patch [21]	Black-box	Yes	94.61	Targeted	94.57
Ours	Black-box	**Yes**	**99.83**	Targeted	**99.98**

box assumptions and non-realistic designs. In contrast, G-Patch introduces realism but still remains constrained by specific settings. Our approach uniquely combines all three challenging properties-**targeted**, **realistic**, and **black-box**-and nevertheless achieves the highest attack success across both ASR and TCS. As visualized in Figure 3, our method

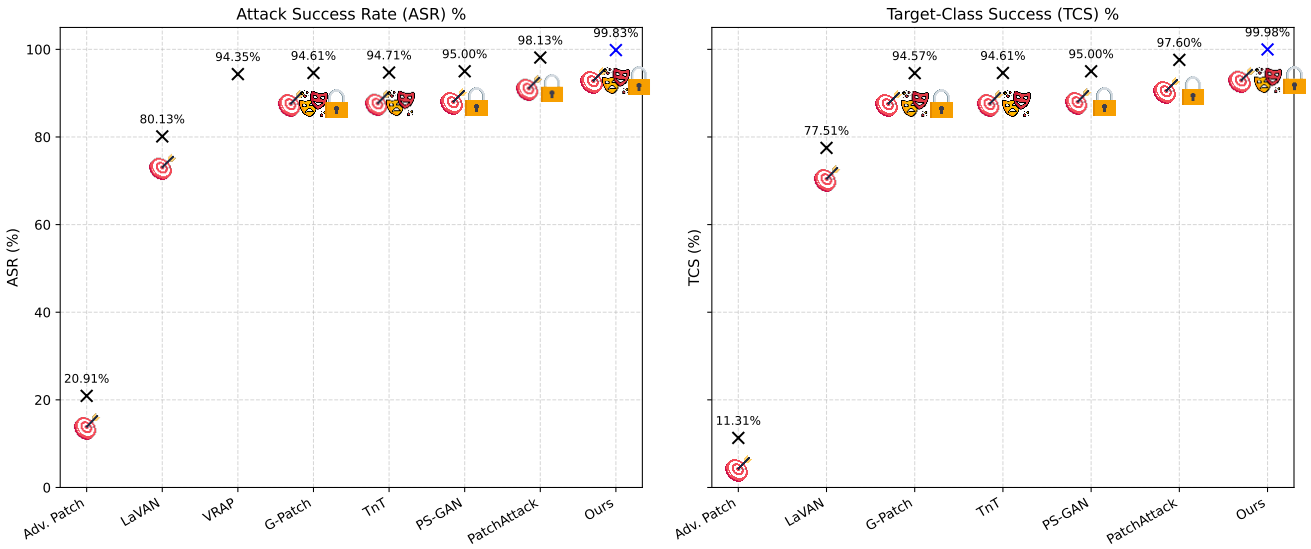

Figure 3. Comparison of adversarial patch attacks with a patch size of 64×64. The plots show Attack Success Rate (ASR) and Target-Class Success (TCS), both reported in percentage. For each method, the scatter marker represents the measured value, while the badges below indicate the attack properties: *Targeted*, *Realistic*, and *Black-box*. Our method combines all three challenging properties simultaneously and still achieves the strongest performance across both metrics, highlighting robustness under the most difficult attack setting.

consistently outperforms competing approaches while satisfying the most stringent and practically relevant conditions. This demonstrates not only theoretical effectiveness but also practical feasibility, bridging the gap between laboratory benchmarks and real-world adversarial patch threats. We also conducted an experiment on the German Traffic Sign Recognition Benchmark (GTSRB) [24], and the results are summarized in Table 3. We observe that adversarial patch placement has a significant effect on attack effectiveness. For both patch sizes (32×32 and 64×64), Grad-CAM guided placement consistently achieves the highest ASR and TCS, demonstrating its ability to exploit the most vulnerable regions of the models. Random placement is generally less effective, while Center placement shows mixed results - occasionally producing moderate success but often weaker compared to Grad-CAM. Furthermore, increasing the patch size from 32×32 to 64×64 amplifies attack success, particularly for larger models such as ViT-L/14. Importantly, the clean model accuracy before attack remains stable across all configurations, confirming that the observed degradation is solely due to the adversarial patches rather than model instability. Overall, these results emphasize both the sensitivity of ViTs to patch location and the heightened threat posed by larger, saliency-aware patches.

6. Texture Preservation and Realism

A key strength of our method is that the synthesized adversarial patches remain realistic, preserving the natural texture of the input image while still achieving targeted misclassification. As shown in Figure 4, the clean inputs are shown on the left and the adversarially patched images on the right.

Figure 4. Adversarial patch examples. Left: clean input images. Right: realistic texture-preserving adversarial patches generated by our method, which achieve targeted attacks without significantly altering the visual content.

Table 3. GTSRB results across patch sizes, ViT models, and placement strategies. ASR: attack success rate. TCS: target-class success. "Model Acc. Before Attack" is the clean (pre-attack) test accuracy of the released checkpoint. An upward arrow (↑) indicates higher is better, while a downward arrow (↓) indicates lower is better.

Patch Size	Model	Placement	ASR (%) ↓	TCS (%) ↓	Model Acc. Before Attack (%) ↑
32×32	ViT-B/16	Grad-CAM	89.54	80.12	99.93
		Random	01.30	00.97	99.93
		Center	41.51	29.04	99.93
	ViT-B/32	Grad-CAM	**97.12**	**93.65**	98.81
		Random	11.08	2.92	98.81
		Center	95.09	78.57	98.81
	ViT-L/14	Grad-CAM	90.89	88.21	99.32
		Random	13.87	00.04	99.32
		Center	56.02	51.13	99.32
64×64	ViT-B/16	Grad-CAM	**98.51**	90.31	99.93
		Random	03.71	02.04	99.93
		Center	77.01	24.67	99.93
	ViT-B/32	Grad-CAM	94.96	93.12	98.81
		Random	36.25	04.24	98.81
		Center	93.21	91.12	98.81
	ViT-L/14	Grad-CAM	97.02	**96.89**	99.32
		Random	13.65	00.00	99.32
		Center	96.26	95.42	99.32

Despite the presence of the patch, the visual characteristics of the original image remain largely unchanged, ensuring that the perturbations are inconspicuous to human observers.

This realism is enforced through our joint optimization objective:

$$\min_{\theta} \mathcal{L}(\theta) = \mathcal{L}_{adv} + \mathcal{L}_{patch} + \mathcal{L}_{perc},$$

where $\mathcal{L}_{adv}$ ensures targeted misclassification, $\mathcal{L}_{patch}$ encourages visual consistency with the seed patch δ, and $\mathcal{L}_{perc}$ preserves perceptual similarity in a high-level feature space. Together, these terms guide the generator to produce adversarial patches that are simultaneously effective and visually realistic. This means that our patches achieve high TCS while avoiding disruptive artifacts.

A detailed ablation study on the loss function, along with the formal criteria for our realism check, and the patch size ablation on ResNet (ImageNet) are provided in the appendix.

7. Conclusion and Future Work

In this work, we introduced a targeted, realism-aware conditional GAN framework for adversarial patch generation under strict black-box constraints. Our method integrates real-image conditioning with an attacker-specified target class, enabling visually coherent patches that maintain semantic plausibility while achieving high attack effectiveness. By leveraging Grad-CAM from a surrogate model, we guided patch placement without requiring gradient access to the victim model. A multi-objective loss was designed to balance adversarial objectives with pixel-level perceptual similarity and deep feature consistency, ensuring that generated patches preserve realism while maximizing targeted misclassification. Extensive experiments on ImageNet-pretrained architectures, including both convolutional backbones and Vision Transformers, demonstrated that our approach achieves state-of-the-art targeted attack success rates without victim model fine-tuning. Results on the GTSRB dataset further validated the robustness of our framework, showing that patch size and placement strategies strongly influence attack success and target-class misclassification. The proposed framework shows strong generalization across diverse architectures and highlights the vulnerability of modern vision systems to realistic, localized perturbations in black-box scenarios. Looking ahead, future research should extend this work toward physical-world evaluations, focusing on real patches printed and deployed under varying lighting conditions, natural shades, and viewing angles. Additionally, exploring adaptive defense bypassing, multimodal attack settings, and integrating text-to-image generative models for patch synthesis represents a promising direction. These efforts will bring adversarial patch research closer to practical, real-world applicability while further testing adversarial robustness.

References

[1] Tom B. Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. In *Advances in Neural Information Processing Systems*, pages 4885–4894, Long Beach, CA, USA, 2017. https://arxiv.org/abs/1712.09665. 2, 3, 6

[2] Binyue Deng, Denghui Zhang, Fashan Dong, Junjian Zhang, Muhammad Shafiq, and Zhaoquan Gu. Rust-style patch: A physical and naturalistic camouflage attacks on object detector for remote sensing images. *Remote Sensing*, 15(4):885, 2023. 4

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009. 5

[4] Bao Gia Doan, Anh Tuan Nguyen, Trung Le Hoang, and Tuan Anh Le. Tnt: Universal naturalistic adversarial patches through texture transformation. *IEEE Transactions on Information Forensics and Security*, 17:1159–1174, 2022. 5, 6

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. ICLR*, 2021. 5

[6] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chen Xiao, Aditya Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1625–1634, Salt Lake City, Utah, USA, 2018. 2, 3

[7] Yonggan Fu, Shunyao Zhang, Shang Wu, Cheng Wan, and Yingyan Celine Lin. Patch-fool: Are vision transformers always robust against adversarial perturbations? In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022. 3

[8] Peng Geng and Xi Deng. An adversarial patch attack for vehicle detectors in the physical world. In *Proceedings of the IEEE International Conference on Unmanned Systems (ICUS)*, pages 979–984, Xi'an, China, 2023. 3

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016. 5

[10] Yu-Chih-Tuan Hu, Bo-Han Kung, Daniel Stanley Tan, Jun-Cheng Chen, Kai-Lung Hua, and Wen-Huang Cheng. Naturalistic physical adversarial patch for object detectors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7848–7857, 2021. 3

[11] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proc. CVPR*, 2017. 5

[12] Ren-Hung Hwang, Jia-You Lin, Sun-Ying Hsieh, Hsuan-Yu Lin, and Chia-Liang Lin. Adversarial patch attacks on deep-learning-based face recognition systems using generative adversarial networks. *Sensors*, 23(2):853, 2023. 3

[13] D. Karmon, D. Zoran, and Y. Goldberg. Lavan: Localized and visible adversarial noise. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018. 6

[14] Roie Kazoom, Raz Birman, and Ofer Hadar. Enhancing object detection robustness: Detecting and restoring confidence in the presence of adversarial patch attacks. *arXiv preprint arXiv:2403.12988*, 2024. 3

[15] Roie Kazoom, Ofir Cohen, Rami Puzis, Asaf Shabtai, and Ofer Hadar. Vault: Vigilant adversarial updates via llm-driven retrieval-augmented generation for nli. *arXiv preprint arXiv:2508.00965*, 2025. 3

[16] Roie Kazoom, Raz Lapid, Moshe Sipper, and Ofer Hadar. Don't lag, rag: Training-free adversarial detection using rag. *arXiv preprint arXiv:2504.04858*, 2025. 3

[17] Aishan Liu, Xishen Liu, Jun Fan, Yuhui Ma, An Zhang, Hongyu Xie, and Dacheng Tao. Perceptual-sensitive generative adversarial network for generating adversarial patches. In *Proceedings of the AAAI Conference on Artificial Intelligence*. AAAI Press, 2019. 2, 3, 6

[18] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proc. ICCV*, 2021. 5

[19] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241. Springer, 2015. 4, 5

[20] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017. 2, 3, 4, 5

[21] Mingzhen Shao. Random position adversarial patch for vision transformers, 2023. 2, 3, 6

[22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4, 5

[23] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: A multi-class classification competition. In *Proc. IJCNN*, 2011. 5

[24] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. In *Neural Networks (IJCNN), The 2012 International Joint Conference on*, pages 263–270. IEEE, 2012. 7

[25] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014. 3

[26] W. Wang and Z. Wang. Vrap: Generating visually realistic adversarial patches. *arXiv preprint*, 2023. 6

[27] Xingxing Wei, Yao Huang, Yitong Sun, and Jie Yu. Unified adversarial patch for cross-modal attacks in the physical world, 2023. 3

[28] Chengzhi Yang, Hongcheng Wu, Jinyuan Li, Yiran Chen, and Huan Liu. Patchattack: A black-box textured adversarial patch attack on deep neural networks. In *European Conference on Computer Vision (ECCV)*, 2020. 6

Appendix

7.1. Ablation Study on Loss Functions

We investigate the individual contribution of each loss component to the overall objective. Recall that the complete optimization is defined as:

$$\mathcal{L} = \mathcal{L}_{adv} + \mathcal{L}_{patch} + \mathcal{L}_{perc},$$

where each term plays a distinct role:

- **Adversarial loss** $\mathcal{L}_{adv}$ enforces targeted misclassification into the attacker-specified class. It is the driving force behind adversarial effectiveness, ensuring that the patched image $\tilde{x}$ is predicted as the target class regardless of its original semantics.
- **Patch consistency loss** $\mathcal{L}_{patch}$ constrains the generated patch $G(\delta)$ to remain visually close to the seed patch δ. This stabilizes training, prevents mode collapse, and ensures that the adversarial patch retains a coherent texture rather than degenerating into noisy patterns.
- **Perceptual loss** $\mathcal{L}_{perc}$ enforces similarity in a high-level feature space using activations from a frozen network (e.g., VGG16). This encourages the generated patch to preserve natural image statistics and remain visually plausible while embedding the adversarial signal.

To assess the impact of each term, we evaluate the following configurations:

1. $\mathcal{L}_{adv}$ only
2. $\mathcal{L}_{adv} + \mathcal{L}_{patch}$
3. $\mathcal{L}_{adv} + \mathcal{L}_{perc}$
4. $\mathcal{L}_{adv} + \mathcal{L}_{patch} + \mathcal{L}_{perc}$ (full objective)
5. $\mathcal{L}_{patch}$ only
6. $\mathcal{L}_{perc}$ only
7. $\mathcal{L}_{perc} + \mathcal{L}_{patch}$

As shown in Table 4, the results highlight several key insights:

- Using $\mathcal{L}_{adv}$ alone achieves targeted misclassification but yields relatively weak performance, with both ASR and TCS capped below 76%. This confirms that misclassification alone is insufficient for stable and realistic patch generation.

- Using $\mathcal{L}_{patch}$ or $\mathcal{L}_{perc}$ alone produces visually stable and realistic patches but fails to induce strong targeted misclassification, resulting in substantially lower ASR and TCS values.

- Combining $\mathcal{L}_{adv}$ with either $\mathcal{L}_{patch}$ or $\mathcal{L}_{perc}$ moderately improves results, though still falls short of state-of-the-art robustness.

- The complete loss $\mathcal{L}_{adv}+\mathcal{L}_{patch}+\mathcal{L}_{perc}$ yields the best trade-off, achieving near-perfect ASR (99.89%–99.99%) and TCS (99.88%–99.98%) across patch placements.

These findings confirm that the three losses are highly complementary: adversarial enforcement drives targeted misclassification, patch consistency ensures stability, and perceptual similarity enforces realism. Together, they are necessary to produce robust, transferable, and visually plausible adversarial patches.

7.2. Realism vs. Non-Realism

We evaluate the effect of perceptual and consistency losses on adversarial patch synthesis by distinguishing *realistic* from *non-realistic* patches. A patch is considered **realistic** if at least 8 out of 10 human evaluators judged it to blend naturally into the scene, without exhibiting unnatural color distortions. Otherwise, it is **non-realistic**.

Formally, for a patch p with human ratings $h_i \in \{0, 1\}$, $i = 1, \ldots, 10$, we define

$$R(p) = \mathbf{1}\left[\sum_{i=1}^{10} h_i \geq 8\right]$$

where $\mathbf{1}[\cdot]$ denotes the indicator function.

In addition to human evaluation, we report SSIM and LPIPS as perceptual metrics. Training with perceptual and consistency losses yields patches with improved realism ($R(p) = 1$), while maintaining strong attack success rate (ASR) and target class success (TCS).

7.3. Effect of Patch Size on Attack Success (ResNet, ImageNet)

We further analyze the impact of patch size on adversarial effectiveness using ResNet trained on ImageNet. Table 5 reports the attack success rate (ASR) and target-class success (TCS) for varying patch sizes from 8×8 to 128×128. The ASR measures the proportion of inputs misclassified into *any* incorrect label, while TCS measures the proportion redirected specifically into the attacker-specified target class. Formally,

$$\text{ASR} = \frac{\#\{\tilde{x} : f(\tilde{x}) \neq y\}}{\#\{x\}}, \qquad \text{TCS} = \frac{\#\{\tilde{x} : f(\tilde{x}) = t\}}{\#\{x\}},$$

where y is the ground-truth label, t is the attacker-specified target class, and $\tilde{x}$ denotes the adversarial example.

Figure 5 visualizes these results. Both ASR and TCS increase monotonically with patch size. Small patches such as 8×8 achieve only limited effectiveness (ASR $= 19.32\%$, TCS $= 5.45\%$), while medium patches like 32×32 already surpass ASR $= 97.75\%$ and TCS $= 93.79\%$. At 64×64 and above, the attack becomes nearly perfect, converging to ASR $\approx 100\%$ and TCS $\approx 100\%$.

Table 4. Ablation study on loss functions. We evaluate different combinations of $\mathcal{L}_{adv}$, $\mathcal{L}_{patch}$, and $\mathcal{L}_{perc}$ under multiple placement strategies. Accuracy before attack is reported along with attack success rate (ASR) and target-class success (TCS).

Loss Setting	**Placement**	**Accuracy Before Attack (%)** ↑	**ASR (%)** ↓	**TCS (%)** ↓
$\mathcal{L}_{adv}$ only	Center	77.90	75.62	72.48
	Random	77.90	74.35	70.19
	Grad-CAM	77.90	73.84	71.55
$\mathcal{L}_{patch}$ only	Center	77.90	40.17	15.23
	Random	77.90	35.54	14.87
	Grad-CAM	77.90	42.26	18.14
$\mathcal{L}_{perc}$ only	Center	77.90	45.28	20.37
	Random	77.90	38.46	12.18
	Grad-CAM	77.90	47.93	21.57
$\mathcal{L}_{adv} + \mathcal{L}_{patch}$	Center	77.90	74.85	73.92
	Random	77.90	72.14	70.68
	Grad-CAM	77.90	75.73	74.11
$\mathcal{L}_{adv} + \mathcal{L}_{perc}$	Center	77.90	75.44	74.32
	Random	77.90	73.61	72.48
	Grad-CAM	77.90	74.92	75.21
$\mathcal{L}_{perc} + \mathcal{L}_{patch}$	Center	77.90	52.13	25.47
	Random	77.90	48.02	22.36
	Grad-CAM	77.90	55.67	29.08
$\mathcal{L}_{adv} + \mathcal{L}_{patch} + \mathcal{L}_{perc}$	Center	77.90	99.89	99.88
	Random	77.90	97.91	97.91
	Grad-CAM	77.90	**99.99**	**99.98**

Table 5. Patch size ablation on ResNet evaluated on ImageNet. We report attack success rate (ASR) and target-class success (TCS). Lower values (↓) indicate stronger attacks.

Patch Size	ASR (%)↓	TCS (%)↓
8×8	19.32	05.45
16×16	80.21	67.54
32×32	97.75	93.79
64×64	99.99	99.98
128×128	100.00	100.00

These findings highlight that adversarial effectiveness scales with the available perturbation budget: larger patches have greater capacity to embed adversarial signals while maintaining control over targeted misclassification.

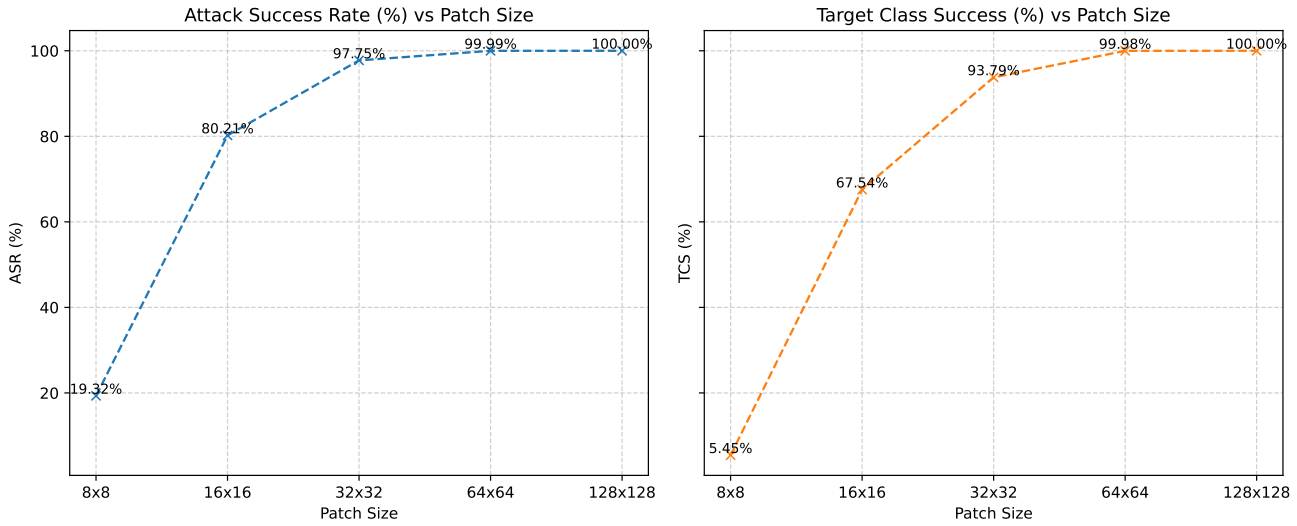

Figure 5. Effect of patch size on attack success rate (ASR) and target-class success (TCS) for ResNet on ImageNet. Both ASR and TCS increase with patch size, converging to nearly 100% success for 64×64 and larger patches.