SegPAR: Class-Centric Decision-Based Sparse Attack for Semantic Segmentation

Dongsu Song[1,3], DaeYun GO[1,2], Boseung Seo[1,2], and Jay Hoon Jung[2]✉

[1] Interdisciplinary Program in Space Systems Engineering, Korea Aerospace University, South Korea
[2] Department of Artificial Intelligence, Korea Aerospace University, South Korea
[3] Department of Computer Science, Korea Aerospace University, South Korea

Abstract. Despite the practical relevance of sparse decision-based black-box threats, they have received limited attention in semantic segmentation. To bridge this gap, we adapt the most representative decision-based black-box sparse attacks from the classification domain to serve as baselines, establishing a rigorous benchmark for this underexplored setting. In this context, we demonstrate that one of the existing methods suffers from severe query inefficiency due to its image-centric pixel accumulation, which rapidly exhausts query budgets across the vast image space. To overcome this, we propose SegPAR, a novel decision-based framework that shifts to a class-centric exploration paradigm. Furthermore, to eliminate the misleading feedback generated by standard decision rewards during pixel accumulation, we introduce a novel discrepancy reward. Extensive experiments show that SegPAR significantly outperforms black-box baselines in sparsity efficiency and MIoU reduction, while remaining competitive with white-box sparse attacks. Code is available at https://github.com/KAU-QuantumAILab/SegPAR.

Keywords: Black-box · Sparse Attack · Class-centric

1 Introduction

Deep learning models have achieved remarkable success across domains [1, 5, 21, 31, 33]. However, prior work has shown that these models are susceptible to adversarial attacks, which introduce imperceptible perturbations that cause incorrect outputs [4,6,18,32]. These attacks are generally categorized into *white-box* and *black-box* settings based on the adversary's knowledge [3]. White-box attacks assume access to comprehensive model information, including gradients, architectures, and internal features [22,37]. Conversely, black-box attacks operate under limited information, relying solely on input-output interactions such as decision labels and confidence scores [8, 10, 28]. Given that black-box scenarios more accurately reflect the constraints of real-world deployments, advancing the study of black-box attacks is of critical importance.

✉ Corresponding & first authors: jhjung@kau.ac.kr & raister01@kau.kr.

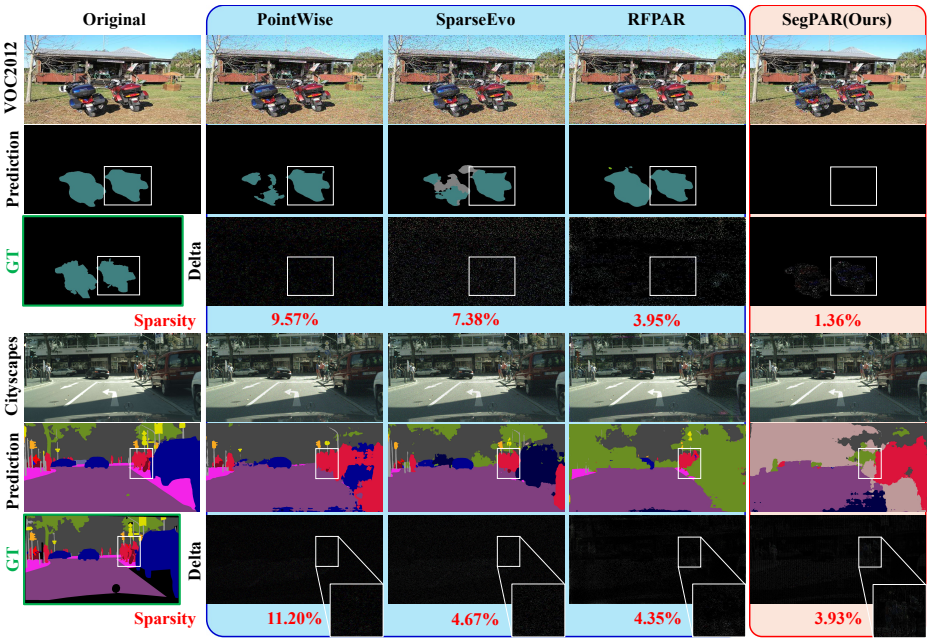

Fig. 1: Qualitative comparison of sparse adversarial attacks on semantic segmentation models. Our proposed SegPAR induces severe misclassification with significantly lower sparsity compared to the baselines. As shown in the white boxes of the Delta maps, SegPAR maximizes attack efficiency by precisely targeting critical pixels, resulting in fatal semantic disruptions with minimal perturbations.

Black-box attacks are typically categorized into *transfer-based* and *query-based* approaches. Transfer-based attacks craft adversarial perturbations on surrogate models and rely on cross-model transferability, whereas query-based attacks interact with the victim model through iterative perturbations [15, 35]. Query-based attacks are particularly compelling because they are model-agnostic and do not require a representative surrogate. Within this paradigm, *sparse* attacks that perturb only a small set of pixels have emerged as a challenging yet practically relevant setting [12, 26, 30]. Identifying influential pixels is NP-hard, and the combinatorial search space makes query-efficient optimization especially difficult under realistic query budgets [16, 23]. Importantly, sparse threats can reflect real-world failure sources such as sensor defects (*e.g.*, hot/dead pixels) in object detection pipelines and localized physical perturbations (*e.g.*, stickers on road signs) that mislead autonomous driving systems [2, 25, 29, 39].

Despite the practical relevance of sparse decision-based black-box threats, they have received limited attention in semantic segmentation, where the search space scales with dense, multi-class outputs [14]. While decision-based black-box attacks for semantic segmentation have recently begun to be explored [9], they largely focus on dense ℓ_p-bounded perturbations and do not address ex-

plicit pixel-level sparsity. To bridge this gap, we adapt the most representative *decision-based black-box sparse attacks* from the classification domain to serve as baselines, establishing a rigorous benchmark for this underexplored setting. In this context, we demonstrate the existing method, RFPAR [29], suffers from severe query inefficiency when applied to segmentation, as its image-centric pixel accumulation tends to concentrate queries on a few easy-to-exploit regions, failing to cover heterogeneous vulnerabilities across multiple semantic classes, thereby wasting the limited query budget. To overcome this, we propose Segmentation Pixel Attack using RL (**SegPAR**), a novel decision-based framework that shifts to a class-centric exploration paradigm. Furthermore, to eliminate the misleading feedback generated by standard decision rewards during pixel accumulation, we introduce a novel **discrepancy reward**. Extensive experiments show that SegPAR significantly outperforms black-box baselines in sparsity efficiency and MIoU reduction, as shown in Fig. 1, while remaining competitive with white-box attacks.

In summary, our contributions are:

- To the best of our knowledge, this work is the first systematic study of decision-based black-box **sparse** attacks for semantic segmentation.
- We propose **SegPAR**, a decision-based black-box sparse attack for semantic segmentation that improves query efficiency via class-centric exploration.
- We introduce a **discrepancy reward** that mitigates the unreliable signals of standard decision rewards during pixel accumulation. Combined with SegPAR, it achieves the strongest black-box performance and is competitive with white-box sparse baselines under the same sparsity constraints.

2 Preliminaries

In our semantic segmentation attack setting, let $\boldsymbol{x} \in \{0, \ldots, 255\}^{C \times H \times W}$ be an input image and $f(\cdot)$ be a victim model outputting a predicted label map $\boldsymbol{y} \in \{0, \ldots, K-1\}^{H \times W}$, where H, W, and K denote the height, width, and number of classes, respectively. Our goal is to degrade the model's MIoU by perturbing only a small number of pixels under a limited query budget. Directly optimizing MIoU in a black-box setting is challenging since it is a non-smooth, set-based metric. Therefore, we propose an effective proxy: maximizing the fraction of misclassified pixels under a sparsity constraint. The problem is formulated as:

$$\max_{\boldsymbol{\delta}} \ \frac{1}{WH} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} \mathbb{I}[f(\hat{\boldsymbol{x}})_{i,j} \neq f(\boldsymbol{x})_{i,j}], \quad \text{s.t.} \quad \|\boldsymbol{\delta}\|_0 = \|\hat{\boldsymbol{x}} - \boldsymbol{x}\|_0 \leq \epsilon, \tag{1}$$

where $\hat{\boldsymbol{x}}$ is the modified image, $\|\cdot\|_0$ is defined as the number of modified pixels, and $\mathbb{I}$ is the indicator function. Under our decision-based black-box setting, the optimization relies strictly on label queries, while ground-truth annotations are reserved exclusively for final evaluation.

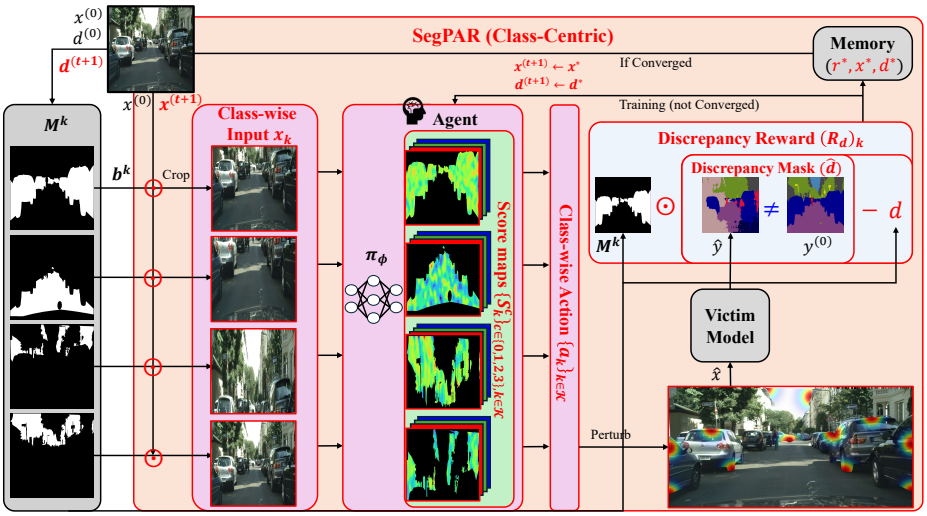

Fig. 2: SegPAR Framework. At initialization, SegPAR extracts fixed per-class masks $\{M^k\}$ from the initial prediction $y^{(0)}$ and keeps them throughout the episode. At each step t, class-wise inputs are constructed by cropping the current image, $x_c^{(t)} = \mathrm{Crop}(x^{(t)}, b^k)$, where b^k is the crop box for class k. Conditioned on the class and cropped input, the RL agent outputs sparse pixel perturbations to generate $\hat{x}^{(t)}$, which is then queried on the victim model f. The discrepancy mask is computed against the initial prediction as $d^{(t)} = \mathbf{1}[f(\hat{x}^{(t)}) \neq y^{(0)}]$. The class-wise discrepancy reward is defined within each fixed class mask, $R_d^k \propto \sum_{i,j} M_{i,j}^k \left(d_{i,j}^{(t)} - d_{i,j}^{(t-1)}\right)$. During training, SegPAR stores the best perturbation, discrepancy mask, and reward in memory; after convergence, it updates $x^{(t+1)} \leftarrow x^*$ and $d^{(t+1)} \leftarrow d^*$.

3 Related Works

3.1 Decision-based Sparse Attack

Decision-based sparse attacks seek sparse perturbations while maximizing attack success using only the model's final hard-label decisions. Representative methods in this category include Pointwise [27] and SparseEvo [34]. Pointwise [27], a decision-based sparse attack, minimizes perturbed pixels by initially inducing misclassification and using a greedy search to revert pixels. To reduce its search space complexity, SparseEvo [34] optimizes a binary vector of perturbed locations via an evolutionary algorithm. Since both require an initial successful attack, adapting them to semantic segmentation necessitates redefining "attack success" using the **Success Ratio (SR)**. The SR is defined as: $\mathrm{SR}(x, \hat{x}) = \frac{1}{WH} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} \mathbb{I}[f(\hat{x})_{i,j} \neq f(x)_{i,j}]$. Perturbations are removed only if the attack succeeds ($\mathrm{SR} \geq \tau$).

3.2 Score-based Sparse Attack

Score-based sparse attacks seek sparse perturbations while maximizing attack success using only scores. While OnePixel [30] utilizes Differential Evolution for single-pixel manipulation, subsequent approaches adopt iterative accumulation strategies. Specifically, Sparse-RS [12] employs Random Search to progressively update and accumulate perturbations, and Pixle [26] minimizes loss by iteratively rearranging neighboring pixel patches. Notably, RFPAR [29] employs reinforcement learning. In RFPAR, an agent iteratively explores candidates $\hat{x}$, storing the best reward and attack parameters in memory. Upon convergence—when reward improvement stalls—the agent reinitializes and proceeds to the next step. We adapt this framework for a stricter decision-based black-box setting in semantic segmentation by eliminating the score component. Relying exclusively on decision rewards, we evaluate the candidate $\hat{x}$ using the **standard reward**: $R_s(x^{(t)}, \hat{x}^{(t)}) = \frac{1}{o^{2 \cdot \|\hat{x}^{(t)} - x^{(t)}\|_0}} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} \mathbb{I}\big[f(\hat{x}^{(t)})_{i,j} \neq f(x^{(t)})_{i,j}\big]$, where $o = 5$ is a scaling factor, and $x^{(t)}$ denotes the input image at the current step t.

3.3 White-box Sparse Attack

White-box sparse attacks leverage model gradients to achieve misclassification under a sparsity constraint. Early methods such as JSMA [24], CW-l_0 [6], and SparseFool [23] focused on inducing sparsity through direct optimization or surrogate l_1 norms. While PGD$_0$ [13] extends Projected Gradient Descent to the l_0 space, it is prone to local optima due to the non-convex nature of the l_0 constraint. To mitigate this, sPGD [43] was introduced, which decomposes the perturbation δ into a magnitude tensor p and a location mask. By employing continuous surrogate variables and top-k projection, sPGD overcomes the non-differentiability of the l_0 norm. To further boost optimization, sPGD updates p using both the *projected* gradient restricted by the mask and the *unprojected* gradient that ignores the mask.

4 Method

In this section, we present the overall framework of SegPAR, illustrated in Fig. 2. SegPAR reformulates RFPAR [29] into a class-wise attack paradigm tailored for semantic segmentation. We then analyze the limitations of the conventional pixel-accumulation reward and propose the discrepancy reward to address these issues and improve attack performance.

4.1 SegPAR

Insight. Deep neural networks partition the input space into complex decision regions, which in ReLU-based networks can be locally viewed as piece-wise linear boundary facets [19]. Regardless of the exact boundary geometry, semantic segmentation requires predictions at every pixel. Thus, an attack must induce label

flips by crossing a collection of heterogeneous, pixel- and class-specific boundaries, rather than a single global boundary. Consequently, an image-centric exploration strategy can quickly concentrate queries on a few high-reward (often large or easily perturbed) regions, repeatedly probing similar areas while leaving other class-specific vulnerabilities under-explored. We therefore adopt a class-centric exploration strategy that conditions the RL agent on each class present in the image, encouraging perturbations to be distributed across diverse regions and improving the efficiency of reaching relevant decision boundaries.

Class-wise Input. To address this issue, we shift the original RFPAR paradigm, which generates an action from the entire image as the state, to a class-wise paradigm that uses a per-class state representation. Given an original image $\boldsymbol{x}$, we first query the victim model to obtain its prediction. From $f(\boldsymbol{x})$, we construct and fix a set of class-wise binary masks $\mathcal{M} = \{M^k\}_{k\in\mathcal{K}}$, where $M^k \in \{0,1\}^{H\times W}$ indicates the predicted pixels belonging to class k and $\mathcal{K}$ is defined as the set of classes that appear in the prediction for the given image:

$$M^k_{i,j} = \mathbb{I}[f(\boldsymbol{x})_{i,j} = k], \quad \forall (i,j) \in \{0,\dots,H-1\}\times\{0,\dots,W-1\}. \tag{2}$$

For each mask M^k, we define a bounding-box function $B(\cdot) : M^k \to \mathbb{N}^4$, which returns the tightest axis-aligned bounding box that encloses the foreground pixels of the mask. Specifically, the bounding box for class k is $b^k = B(M^k) = \big(i^k_{\min}, j^k_{\min}, i^k_{\max}, j^k_{\max}\big)$, where i and j denote the vertical and horizontal image coordinates, respectively. Using b^k, we extract a class-specific crop from the original image and use it as the agent state $\boldsymbol{x}_k = \mathrm{Crop}\big(\boldsymbol{x}, b^k\big)$. In summary, instead of using the full image x as the state, the agent policy π_ϕ operates on a per-class state x_k constructed from the model's predicted region for class k.

Class-wise Action. As we convert the agent state into a class-specific region, the action space must be redesigned accordingly. RFPAR samples coordinates from a Gaussian distribution, which implicitly assumes a rectangular target region. However, class-wise masks are typically irregular; applying the same strategy can induce undesired behavior. In particular, a bounding box often includes pixels from other classes, so perturbing and evaluating rewards over the entire box can contaminate the signal with non-target effects, destabilizing policy learning for the target class k.

To resolve this issue, we propose a mask-based sampling strategy: instead of sampling coordinates from a Gaussian, the agent predicts a probability map over candidate attack locations and samples from the induced distribution. Specifically, given the state $\boldsymbol{x}_k$, the agent produces a 4-channel score map $\boldsymbol{S}_k \in \mathbb{R}^{4\times h_k\times w_k}$, where $h_k = i^k_{\max} - i^k_{\min} + 1$ and $w_k = j^k_{\max} - j^k_{\min} + 1$. Let $\boldsymbol{m}^k = \mathrm{Crop}(M^k, b^k)$ be the class mask cropped to the same region as $\boldsymbol{x}_k$. Using the first channel $\boldsymbol{S}^0_k$ as spatial logits, we define a masked softmax distribution over locations:

$$\boldsymbol{P}_k = \mathrm{Softmax}\big(\boldsymbol{S}^0_k + (1-\boldsymbol{m}^k)\cdot(-10^9)\big), \tag{3}$$

where the softmax is taken over all $N = h_k w_k$ spatial indices. This assigns zero probability to masked-out pixels, confining the search space to the actual class region regardless of the bounding box size. Let $\boldsymbol{p}_k = \mathrm{vec}(\boldsymbol{P}_k) \in [0,1]^N$. We sample n locations without replacement as in our implementation. Let $\mathcal{I} = \{I_1, \ldots, I_n\} \sim \mathrm{MultinomialNR}(\boldsymbol{p}_k; n)$ denote weighted sampling of n indices without replacement. Each sampled index I_l is converted to 2D coordinates by $j_l = (I_l \bmod w_k) + j^k_{\min}$, $i_l = \lfloor I_l / w_k \rfloor + i^k_{\min}$.

For the RGB decision, we use the remaining three channels as logits. Let $\boldsymbol{S}^c_k \in \mathbb{R}^{h_k \times w_k}$ be the c-th RGB logit map for $c \in \{1,2,3\}$, and denote its vectorization by $\boldsymbol{s}^c = \mathrm{vec}(\boldsymbol{S}^c_k)$. Conditioned on each sampled location I_l, we independently sample Bernoulli variables $z_{l,c} \sim \mathrm{Bernoulli}(\mathrm{Sigmoid}(s^c_{I_l}))$ for each channel $c \in \{1,2,3\}$, where $\mathrm{Sigmoid}(\cdot)$ is the sigmoid function. Finally, the sampled action set for class k is $\boldsymbol{a}_k \triangleq \left\{ \left((i_l, j_l),\, z_{l,1}, z_{l,2}, z_{l,3} \right) \right\}_{l=1}^{n}$. To synthesize $\hat{\boldsymbol{x}}$, we apply these actions to $\boldsymbol{x}$ by mapping the binary decisions to the $\{0, 255\}$ color space, consistent with the baseline: $\hat{x}_{i_l, j_l, c} = 255 \cdot z_{l,c}$. Once $\hat{\boldsymbol{x}}$ is synthesized, we obtain the victim prediction $f(\hat{\boldsymbol{x}})$. When training the agent with R_s at step t, the class-specific reward $(R_s)_k$ for each class k is defined as

$$(R_s(x^{(t)}, \hat{x}^{(t)}))_k = \frac{1}{o^2 \cdot n} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} M^k_{i,j} \cdot \mathbb{I}\Big[f(\hat{x}^{(t)})_{i,j} \neq f(x^{(t)})_{i,j} \Big], \tag{4}$$

thereby isolating the feedback and avoiding contamination from non-target classes. The agent is subsequently trained in a batched manner by optimizing the policy with respect to the rewards $\{(R_s(x^{(t)}, \hat{x}^{(t)}))_k\}_{k \in \mathcal{K}}$.

4.2 Discrepancy Reward

Standard Reward Inefficiency. We analyze the inherent limitations of the standard reward function in the context of cumulative attack frameworks. Specifically, we investigate the inefficiencies that arise in tasks involving multiple target instances, such as object detection and semantic segmentation, where the agent must handle a significantly larger set of target pixels. Let $y^{(t)}_{i,j} = f(x^{(t)})_{i,j}$ be the prediction of the model for a pixel at spatial coordinate (i,j) at step t, and let $y^{(0)}_{i,j} = f(x^{(0)})_{i,j}$ denote the original prediction class of that pixel. In cumulative attack frameworks, relying only on the prediction transition ($y^{(t)}_{i,j} \neq \hat{y}^{(t)}_{i,j}$, where $\hat{y}^{(t)}_{i,j} = f(\hat{x}^{(t)})_{i,j}$) severely misaligns the agent's optimization direction. To mathematically demonstrate this, we decompose the pixel-wise state transitions at step t into four mutually exclusive subsets, leading to the $t+1$ state:

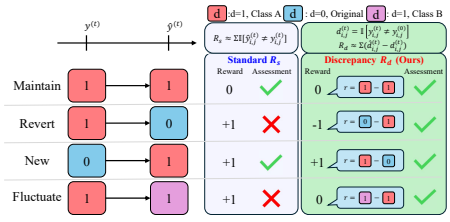

Fig. 3: Comparison of the standard reward and the discrepancy reward.

– **Case 1 (Maintained Misclassification):** Pixels where a successful untargeted attack is stably maintained.

$$\mathcal{S}^{(t)}_{maintain} = \left\{(i,j) \mid y^{(t)}_{i,j} \neq y^{(0)}_{i,j} \wedge \hat{y}^{(t)}_{i,j} = y^{(t)}_{i,j}\right\} \tag{5}$$

– **Case 2 (Reversion Failure):** Pixels where the accumulated noise interferes with a previously successful attack, reverting the prediction back to the original class.

$$\mathcal{S}^{(t)}_{revert} = \left\{(i,j) \mid y^{(t)}_{i,j} \neq y^{(0)}_{i,j} \wedge \hat{y}^{(t)}_{i,j} = y^{(0)}_{i,j}\right\} \tag{6}$$

– **Case 3 (New Misclassification):** Pixels that are newly and successfully deviated from the original class.

$$\mathcal{S}^{(t)}_{new} = \left\{(i,j) \mid y^{(t)}_{i,j} = y^{(0)}_{i,j} \wedge \hat{y}^{(t)}_{i,j} \neq y^{(0)}_{i,j}\right\} \tag{7}$$

– **Case 4 (Unstable Fluctuation):** Pixels that remain misclassified but fluctuate to a different incorrect class, indicating wasted perturbations.

$$\mathcal{S}^{(t)}_{fluctuate} = \left\{(i,j) \mid y^{(t)}_{i,j} \neq y^{(0)}_{i,j} \wedge \hat{y}^{(t)}_{i,j} \neq y^{(0)}_{i,j} \wedge y^{(t)}_{i,j} \neq \hat{y}^{(t)}_{i,j}\right\} \tag{8}$$

Applying R_s to these subsets reveals critical optimization flaws. Since R_s is a transition-based reward, it functionally depends on the number of pixels whose predictions change between steps, *i.e.*,

$$R_s(x^{(t)}, \hat{x}^{(t)}) \propto \sum_{i,j} \mathbb{I}\left[y^{(t)}_{i,j} \neq \hat{y}^{(t)}_{i,j}\right] = \left|\mathcal{S}^{(t)}_{revert}\right| + \left|\mathcal{S}^{(t)}_{new}\right| + \left|\mathcal{S}^{(t)}_{fluctuate}\right|. \tag{9}$$

Therefore, the agent receives positive reinforcement not only for desirable transitions in $\mathcal{S}^{(t)}_{new}$ but also for undesirable ones in $\mathcal{S}^{(t)}_{revert}$ and $\mathcal{S}^{(t)}_{fluctuate}$. In particular, when a previously misclassified pixel reverts to the original class ($\mathcal{S}^{(t)}_{revert}$) or oscillates among incorrect classes ($\mathcal{S}^{(t)}_{fluctuate}$), R_s still assigns a positive reward despite reducing the cumulative attack objective. This misleading feedback biases the agent toward suboptimal regions, increasing query complexity and causing redundant pixel perturbations.

Discrepancy Mask-based Reward Function. To overcome the limitations of the naive approach, we introduce a *Discrepancy Mask*, denoted as $d^{(t)}$, to evaluate the attack success state at step t relative to the initial prediction. For each pixel (i,j), $d^{(t)}_{i,j}$ is defined as a binary indicator of attack success: $d^{(t)}_{i,j} = \mathbb{I}(y^{(t)}_{i,j} \neq y^{(0)}_{i,j})$. Based on this mask, we formulate our proposed reward function, $(R_d(x^{(t)}, \hat{x}^{(t)}))_k$, as between the current discrepancy mask and the candidate discrepancy mask at step t:

$$(R_d(x^{(t)}, \hat{x}^{(t)}))_k = \frac{1}{o^2 \cdot n} \sum_{i,j} M^k_{i,j} \cdot (\hat{d}^{(t)}_{i,j} - d^{(t)}_{i,j}). \tag{10}$$

Table 1: Evaluation of sparse attack performance across semantic segmentation benchmarks. (-) indicates that the experiments are omitted due to the lack of official pre-trained weights for VOC2012.

Model	Attack	Cityscapes				ADE20K				VOC2012			
		MIoU	R.MIoU	Sparsity	Query	MIoU	R.MIoU	Sparsity	Query	MIoU	R.MIoU	Sparsity	Query
DeepLabV3	PointWise	0.798	0.433	5.72%	997.3	0.377	0.172	6.73%	993.5	0.861	0.669	5.91%	952.5
	SparseEvo		0.228	5.43%	988.8		0.118	5.15%	997.3		0.469	5.15%	998.8
	RFPAR$_{R_s}$		0.176	4.51%	965.3		0.106	4.50%	998.1		0.466	3.92%	986.3
	SegPAR$_{R_d}$		**0.101**	**3.33%**	993.3		**0.080**	**3.29%**	966.2		**0.238**	**2.24%**	680.4
PSPNet	PointWise	0.793	0.391	5.78%	996.3	0.380	0.199	6.32%	983.6	0.860	0.704	4.85%	929.6
	SparseEvo		0.246	4.50%	987.0		0.139	5.54%	997.0		0.488	4.66%	998.7
	RFPAR$_{R_s}$		0.153	4.61%	990.5		0.143	4.36%	999.0		0.529	3.99%	984.3
	SegPAR$_{R_d}$		**0.057**	**3.53%**	919.0		**0.087**	**3.11%**	965.3		**0.240**	**2.29%**	679.1
SegFormer	PointWise	0.802	0.594	10.51%	997.9	0.412	0.316	10.1%	973.2	-	-	-	-
	SparseEvo		0.602	5.30%	994.3		0.293	7.03%	996.0		-	-	-
	RFPAR$_{R_s}$		0.566	4.51%	958.5		0.300	4.29%	912.0		-	-	-
	SegPAR$_{R_d}$		**0.341**	**3.96%**	960.4		**0.209**	**3.60%**	987.0		-	-	-
SETR	PointWise	0.780	0.607	9.14%	997.1	0.397	0.284	8.31%	966.5	-	-	-	-
	SparseEvo		0.602	5.71%	997.3		0.270	5.20%	997.3		-	-	-
	RFPAR$_{R_s}$		0.527	4.53%	962.9		0.285	3.90%	996.5		-	-	-
	SegPAR$_{R_d}$		**0.481**	**3.89%**	940.4		**0.249**	**3.50%**	990.9		-	-	-

This concise formulation resolves the key contradictions of R_s across all four subsets. With the pixel-wise reward $r = \hat{d}_{i,j}^{(t)} - d_{i,j}^{(t)}$, we have $r = 0$ for $\mathcal{S}_{maintain}$, $r = -1$ for $\mathcal{S}_{revert}$, $r = +1$ for $\mathcal{S}_{new}$, and $r = 0$ for $\mathcal{S}_{fluctuate}$, as illustrated in Fig. 3. Consequently, R_d directly tracks the marginal gain of cumulatively successful pixels, encouraging the agent to monotonically expand misclassified regions without unnecessary perturbations.

Training & Termination. Following RFPAR [29], we replace y^t with d^t in memory and track the maximum mean reward $r^* = \frac{1}{|\mathcal{K}|}\sum_{k\in\mathcal{K}}(R_d)_k$, maintaining the best (d^*, x^*). We adopt the same hyperparameter suite as RFPAR and cap each run at 100 steps. During each step, the agent updates its policy via REINFORCE [36] using label-only queries; once the memory converges, the agent is reinitialized and starts the next step with a new input.

5 Experiments

We evaluate SegPAR against multiple baselines under strict query and sparsity budgets. Our approach consistently outperforms all black-box methods, achieving larger MIoU drops with fewer perturbed pixels, and remains highly effective against adversarially trained models. Ablations on per-class degradation and pixel accumulation confirm that our class-centric design and discrepancy reward drive these improvements. Furthermore, SegPAR performs competitively with strong white-box sparse attacks, highlighting its practical efficacy under realistic decision-only constraints.

Table 2: Evaluation of sparse attack performance across adversarially trained semantic segmentation benchmarks.

Model	Attack	Cityscapes				VOC2012			
		MIoU	R.MIoU	Sparsity	Query	MIoU	R.MIoU	Sparsity	Query
DeepLabV3_DDCAT	PointWise	0.710	0.502	25.56%	984.2	0.828	0.692	7.81%	977.1
	SparseEvo		0.502	9.56%	910.2		0.650	4.79%	990.3
	RFPAR_{R_s}		0.483	4.25%	840.0		0.543	4.39%	913.8
	SegPAR_{R_d}		**0.428**	**4.24%**	878.3		**0.297**	**2.73%**	822.5
DeepLabV3_SAT	PointWise	0.693	0.495	23.17%	993.8	0.775	0.676	10.0%	918.6
	SparseEvo		0.468	8.42%	932.9		0.630	5.81%	994.3
	RFPAR_{R_s}		0.434	4.42%	851.4		0.566	4.47%	948.5
	SegPAR_{R_d}		**0.394**	**4.25%**	946.2		**0.326**	**2.92%**	799.3
PSPNet_DDCAT	PointWise	0.714	0.523	23.06%	994.1	0.835	0.697	8.47%	951.6
	SparseEvo		0.507	8.26%	939.0		0.658	4.79%	998.0
	RFPAR_{R_s}		0.472	4.33%	873.0		0.578	4.18%	996.1
	SegPAR_{R_d}		**0.405**	**4.22%**	935.3		**0.382**	**2.55%**	809.5
PSPNet_SAT	PointWise	0.677	0.495	24.34%	986.7	0.891	0.693	8.63%	947.0
	SparseEvo		0.482	8.77%	922.7		0.650	5.03%	995.9
	RFPAR_{R_s}		0.450	**4.30%**	995.0		0.591	4.43%	846.9
	SegPAR_{R_d}		**0.384**	4.47%	955.9		**0.332**	**2.79%**	832.6

5.1 Evaluation of Decision-based Sparse Attacks

Attacks. We compare SegPAR_{R_d} with PointWise [27], SparseEvo [34], and RFPAR [29] under a strict budget of 1,000 queries. For the RL-based methods (SegPAR and RFPAR), we use patience 1–2 with convergence thresholds in $[10^{-2}, 3 \times 10^{-2}]$. To align with the 5% sparsity target, RFPAR enforces a global 5% constraint over the entire image, whereas our class-centric SegPAR targets 5% of pixels within each predicted class region (adjusted accordingly when the small number of classes makes it difficult to meet the overall 5% sparsity). Since PointWise and SparseEvo are SR-parameterized, we sweep their SR values and report the run closest to the 5% target. Note that these SR-based baselines may slightly exceed the 5% constraint, which inherently favors them. Additional details are provided in the supplementary material.

Victim models. To ensure the reproducibility of our experiments, we utilize pre-trained checkpoints provided by MMSegmentation. We select widely used models in the field of semantic segmentation, including DeepLabV3 [7], PSPNet [41], SegFormer [38], and SETR [42]. Specifically, DeepLabV3 and PSPNet are employed as CNN-based models, while SegFormer and SETR are utilized as Transformer-based architectures.

Datasets. Due to the high computational cost of query-based attacks on dense prediction models, we evaluate on a fixed subset of 100 randomly sampled validation images from ADE20K [44], Pascal VOC2012 [17], and Cityscapes [11]. ADE20K contains 150 diverse classes with fine-grained part annotations,

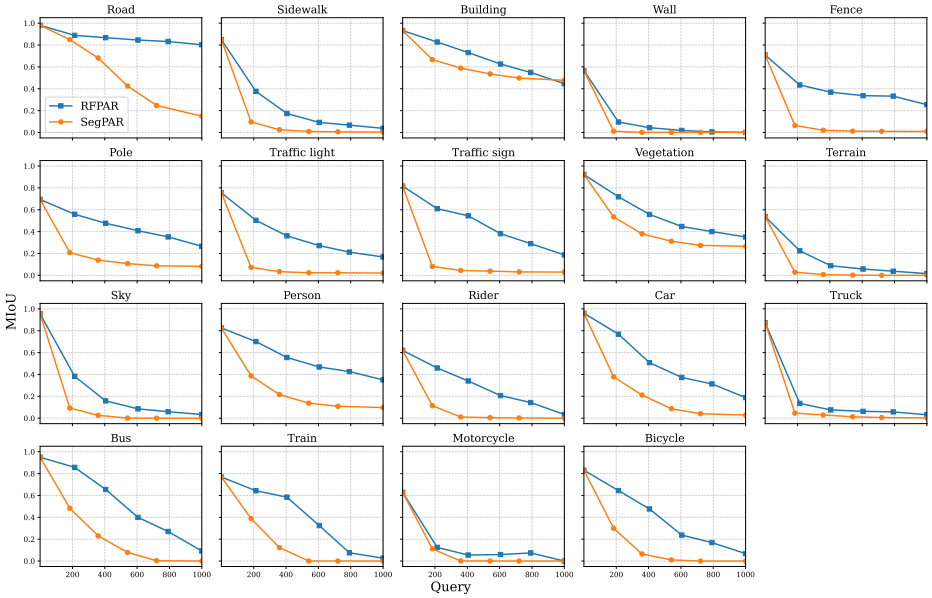

Fig. 4: Class-wise attack efficiency. Comparison of MIoU degradation across individual classes in DeepLabV3. SegPAR demonstrates significantly faster and more effective attacks than RFPAR across all categories.

VOC2012 covers 20 common object categories, and Cityscapes focuses on urban driving scenes with 19 classes.

Evaluation Metrics. We report Robust MIoU (R.MIoU), Sparsity, and Query to measure attack effectiveness and efficiency. R.MIoU is the victim model's MIoU on adversarial examples (lower is better). Sparsity is the average fraction of perturbed pixels (lower indicates fewer, more imperceptible changes). Query is the average number of forward passes required to obtain the final adversarial image (lower is more efficient).

Main Results. As shown in Tab. 1, SegPAR with R_d consistently outperforms existing sparse attack techniques across all evaluated datasets, achieving the lowest R.MIoU under extreme sparsity at comparable or lower queries. For CNN-based architectures, SegPAR with R_d dramatically reduces MIoU; notably, it drops the MIoU of PSPNet on Cityscapes to **0.057** and achieves extreme sparsity (**2.24**%) with significantly lower average queries (*e.g.*, 680.42 for DeepLabV3) on VOC2012. Furthermore, our method proves highly effective even against Transformer-based models like SegFormer and SETR, which typically exhibit higher robustness against local perturbations due to their ability to capture global context. Despite this inherent robustness, our method maintains potent attack performance on Cityscapes using a sparsity of only **3.96**% against

Table 3: Evaluation of Pixel-accumulating Adversarial Attacks across Different Losses.

Model	Objective	Attack	Cityscapes				ADE20K				VOC2012			
			MIoU	R.MIoU	Sparsity	Query	MIoU	R.MIoU	Sparsity	Query	MIoU	R.MIoU	Sparsity	Query
DeepLabV3	Standard	Pixle	0.798	**0.394**	4.69%	995.4	0.377	0.140	4.79%	995.5	0.861	0.447	4.77%	995.5
		Sparse-RS		0.172	**4.87%**	998.7		0.100	**4.83%**	998.1		0.423	**4.84%**	998.2
		RFPAR		**0.176**	4.51%	993.3		0.106	4.50%	998.1		0.466	3.92%	986.3
	Discrepancy	Pixle	0.798	0.411	**4.04%**	995.3	0.377	**0.111**	**4.72%**	995.3	0.861	**0.279**	**4.59%**	994.9
		Sparse-RS		**0.152**	**4.87%**	998.5		**0.092**	**4.83%**	998.3		**0.258**	**4.84%**	998.0
		RFPAR		0.210	**3.56%**	994.9		**0.090**	**3.48%**	947.8		**0.277**	**3.43%**	950.1
PSPNet	Standard	Pixle	0.793	**0.369**	4.69%	995.8	0.380	0.168	4.79%	995.4	0.860	0.374	4.77%	995.2
		Sparse-RS		0.153	**4.87%**	998.7		0.121	**4.83%**	998.3		0.536	**4.84%**	997.7
		RFPAR		**0.153**	4.61%	990.6		0.143	4.36%	999.0		0.529	4.00%	984.3
	Discrepancy	Pixle	0.793	0.383	**4.13%**	995.8	0.380	**0.131**	**4.69%**	995.4	0.860	**0.246**	**4.59%**	995.3
		Sparse-RS		**0.139**	**4.87%**	998.6		**0.092**	**4.83%**	998.1		**0.304**	**4.84%**	997.9
		RFPAR		0.171	**3.92%**	993.0		**0.133**	**3.19%**	982.5		**0.295**	**3.13%**	936.7

SegFormer—a stark contrast to the **10.51**% required by PointWise. Consistently observed across all benchmarks including ADE20K, these results confirm that our approach effectively accelerates the search process to precisely navigate the most vulnerable pixels, regardless of the target architecture.

5.2 Evaluation on Adversarially Trained Models

Tab. 2 reports attack performance against defended models (DDCAT [40], SAT [20]). SegPAR consistently attains the lowest R.MIoU with lower sparsity than prior methods. Notably, adversarial training effectiveness varies across datasets, which we conjecture is partially influenced by image resolution. On high-resolution Cityscapes, the enlarged search space inherently hinders sparse attacks, allowing adversarial training to substantially boost robustness—even enabling CNNs to approach Transformer-level resilience. Conversely, on lower-resolution VOC2012, sparse attacks remain highly effective, yielding only marginal defense gains. While domain complexity also plays a role, current adversarial training appears to offer meaningful protection primarily in high-resolution settings, leaving lower-resolution regimes vulnerable.

5.3 Per-class MIoU Degradation Analysis

In this section, we evaluate the impact of class-centric exploration by comparing RFPAR and SegPAR, both employing R_s. As illustrated in Fig. 4, SegPAR, which represents a paradigm shift from an image-centric to a class-centric approach, exhibits a significantly steeper degradation curve for most categories compared to RFPAR. These results indicate that our class-centric formulation, which constructs per-class inputs from the predicted regions and attacks multiple class-specific areas in parallel, is more effective than image-centric exploration that tends to concentrate queries on a few dominant regions. Notably, safety-critical classes for autonomous driving—such as traffic signs, traffic lights, persons, and riders—are compromised much earlier than with RFPAR, with SegPAR driving their per-class MIoU down sharply within ∼200 queries. This

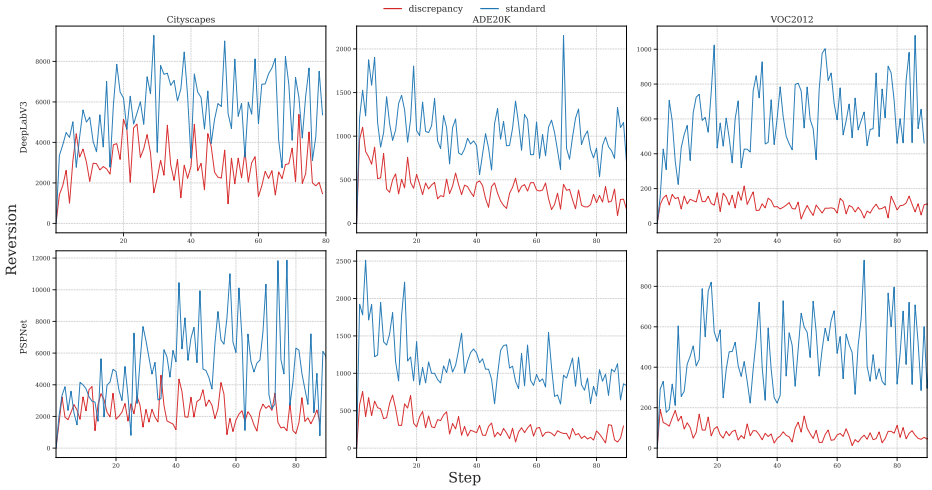

Fig. 5: Reduction of Reversion. Our discrepancy reward effectively suppresses reversion, resolving a major hurdle in agent training caused by the standard reward.

underscores that our method is not only numerically superior in attack performance, but also sufficiently powerful to cause practically meaningful degradation in real-world perception pipelines, even under strict black-box constraints.

5.4 Evaluation of Discrepancy Reward

While Sec. 5.3 already demonstrates the superiority of our class-centric architecture over the image-centric RFPAR under the same standard reward, this section isolates and explicitly evaluates the contribution of the proposed discrepancy reward R_d by applying it to Pixle [26] and Sparse-RS [12]. We optimize the discrepancy objective $\mathcal{L}_d = -R_d$ (equivalently maximizing R_d) and compare it with the standard objective $\mathcal{L}_s = -R_s$. As shown in Tab. 3, on ADE20K and VOC2012, the discrepancy objective leads to a larger reduction in MIoU than the standard objective under the same (or fewer) sparsity, indicating that R_d enables more query-efficient pixel accumulation. In contrast, on the high-resolution Cityscapes benchmark, applying R_d to RFPAR can achieve lower sparsity but yields a smaller MIoU drop than the standard objective. We attribute this to the enlarged effective action space in high-resolution images: since R_d provides non-zero reward only when the set of cumulatively successful pixels expands, informative reward events become rare, causing the search to stagnate before discovering pixels that maximize R_d. As verified in Tab. 1, this limitation is largely alleviated when combining R_d with our proposed SegPAR, which reduces the effective search space via class-wise inputs and mask-constrained exploration. Consequently, their combination achieves the largest MIoU reduction in high-resolution settings. Furthermore, Fig. 5 shows that the discrepancy objective substantially suppresses reversion compared to the stan-

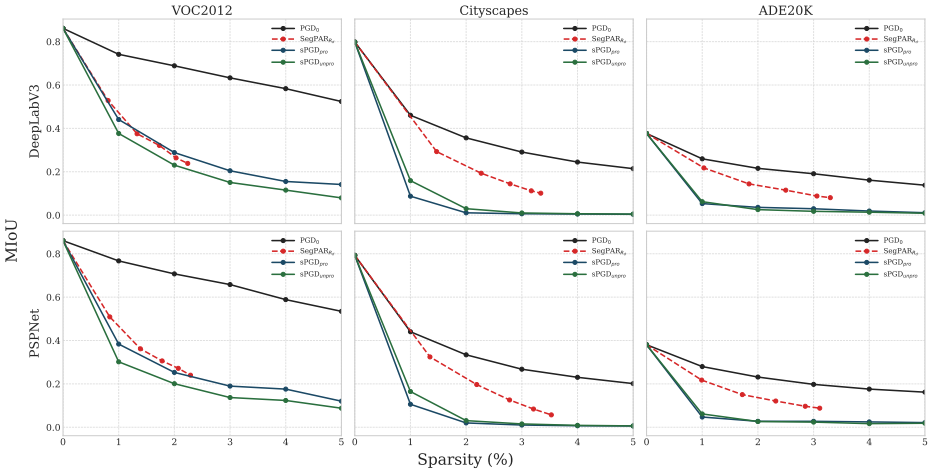

Fig. 6: Comparison of MIoU vs. Sparsity with White-box sparse Attacks. The results demonstrate that our attack SegPAR_{R_d} exhibits competitive performance relative to white-box sparse methods.

dard objective. We quantify reversion at step t as $\left|\mathcal{S}^{(t)}_{revert}\right|$, *i.e.*, the number of misclassified pixels that revert to the original prediction between steps. Overall, these results indicate that R_d mitigates the reversion issue by discouraging counter-productive transitions and improving search efficiency.

5.5 Comparison with White-box Sparse Baseline

In this section, we compare our attack, SegPAR_{R_d}, with white-box sparse baselines, PGD_0 [13], sPGD_{pro}, and $\text{sPGD}_{\text{unpro}}$ [43], all optimized with cross-entropy. For a standardized comparison, we align the evaluation budget by equating one black-box forward query to one white-box gradient update (which inherently entails both forward and backward passes). Under this metric, we enforce a maximum budget of 1,000 steps per image. Specifically, SegPAR_{R_d} utilizes up to 1,000 forward queries, executed over 100 steps (where each mark represents a 20-step interval). Conversely, the white-box attacks allocate 200 gradient updates per sparsity level, totaling 1,000 updates across the 5 evaluated levels (*i.e.*, 20 restarts $\times$ 10 iterations for PGD_0, and 200 iterations for sPGD). As shown in Fig. 6, despite lacking access to internal gradients, SegPAR_{R_d} remains strongly competitive across both segmentation models—often outperforming PGD_0 and approaching sPGD under strict sparsity constraints. These results underscore the effectiveness of SegPAR_{R_d} and suggest that stronger, segmentation-tailored white-box sparse attack baselines are still needed.

6 Conclusion

We proposed **SegPAR**, a decision-based black-box sparse attack for semantic segmentation. By combining **class-centric exploration** with a **discrepancy reward** (R_d), SegPAR mitigates perturbation reversion and improves query efficiency in dense prediction, where the search space is vast. Under strict query and sparsity budgets, SegPAR$_{R_d}$ achieves substantially larger MIoU degradation with fewer perturbed pixels than existing black-box baselines, while rapidly degrading the MIoU of safety-critical classes relevant to autonomous driving. Even without access to internal model information, SegPAR$_{R_d}$ outperforms PGD$_0$ and narrows the gap to sPGD. Furthermore, we observe that adversarial training can be less effective in lower-resolution regimes. Overall, our findings highlight the persistent vulnerability of dense prediction models under realistic decision-only sparse threats.

Acknowledgements

This work was supported by the Advanced GPU Utilization Support Program funded by the Ministry of Science and ICT (MSIT), Korea (No. 02-26-01-0134); by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. RS-2026-25498016); and by the BK21 FOUR Program through the NRF grant funded by the Korean government in 2025 (Grant No. 2120251815513), Institute for Sustainable VLEO Space Services Development.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: GPT-4 Technical Report. CoRR **abs/2303.08774** (2023). https://doi.org/10.48550/arxiv.2303.08774 1
2. Akhtar, N., Mian, A.S.: Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey. IEEE Access **6**, 14410–14430 (2018). https://doi.org/10.1109/ACCESS.2018.2807385 2
3. Bhambri, S., Muku, S., Tulasi, A., Buduru, A.B.: A Study of Black Box Adversarial Attacks in Computer Vision. CoRR **abs/1912.01667** (2019). https://doi.org/10.48550/arxiv.1912.01667 1
4. Biggio, B., Corona, I., Maiorca, D., Nelson, B., Srndic, N., Laskov, P., Giacinto, G., Roli, F.: Evasion Attacks against Machine Learning at Test Time. In: Machine Learning and Knowledge Discovery in Databases (ECML PKDD). LNCS, vol. 8190, pp. 387–402. Springer (2013). https://doi.org/10.1007/978-3-642-40994-3_25 1
5. Carion, N., Gustafson, L., Hu, Y., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., Lei, J., Ma, T., Guo, B., Kalla, A., Marks, M., Greer, J., Wang, M., Sun, P., Rädle, R., Afouras, T., Mavroudi, E., Xu, K., Wu, T., Zhou, Y., Momeni, L., Hazra, R., Ding, S., Vaze, S., Porcher, F., Li, F., Li, S., Kamath,

A., Cheng, H.K., Dollár, P., Ravi, N., Saenko, K., Zhang, P., Feichtenhofer, C.: SAM 3: Segment Anything with Concepts. CoRR **abs/2511.16719** (2025). https://doi.org/10.48550/arxiv.2511.16719 1

6. Carlini, N., Wagner, D.A.: Towards Evaluating the Robustness of Neural Networks. In: IEEE Symposium on Security and Privacy (S&P). pp. 39–57 (2017). https://doi.org/10.1109/SP.2017.49 1, 5
7. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. CoRR **abs/1706.05587** (2017). https://doi.org/10.48550/arxiv.1706.05587 10
8. Chen, W., Zhang, Z., Hu, X., Wu, B.: Boosting Decision-Based Black-Box Adversarial Attacks with Random Sign Flip. In: ECCV. LNCS, vol. 12360, pp. 276–293. Springer (2020). https://doi.org/10.1007/978-3-030-58555-6_17 1
9. Chen, Z., Shan, Z., Chang, J., Jiang, K., Yang, D., Cheng, Y., Zhang, W.: Delving into decision-based black-box attacks on semantic segmentation. CoRR **abs/2402.01220** (2024). https://doi.org/10.48550/ARXIV.2402.01220 2
10. Cheng, S., Dong, Y., Pang, T., Su, H., Zhu, J.: Improving Black-box Adversarial Attacks with a Transfer-based Prior. In: NeurIPS. vol. 32, pp. 10932–10942 (2019). https://doi.org/10.48550/arxiv.1906.06919 1
11. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The Cityscapes Dataset for Semantic Urban Scene Understanding. In: CVPR. pp. 3213–3223 (2016). https://doi.org/10.1109/CVPR.2016.350 10
12. Croce, F., Andriushchenko, M., Singh, N.D., Flammarion, N., Hein, M.: Sparse-RS: A Versatile Framework for Query-Efficient Sparse Black-Box Adversarial Attacks. In: AAAI. pp. 6437–6445 (2022). https://doi.org/10.1609/AAAI.V36I6.20595 2, 5, 13
13. Croce, F., Hein, M.: Sparse and imperceivable adversarial attacks. In: ICCV. pp. 4724–4732 (2019). https://doi.org/10.1109/ICCV.2019.00482 5, 14
14. Croce, F., Singh, N.D., Hein, M.: Towards reliable evaluation and fast training of robust semantic segmentation models. In: ECCV. vol. 15087, pp. 180–197. Springer (2024). https://doi.org/10.1007/978-3-031-72986-7_11 2
15. Deng, Y., Wu, W., Zhang, J., Zheng, Z.: Blurred-Dilated Method for Adversarial Attacks. In: NeurIPS. vol. 36, pp. 48939–48950 (2023), http://papers.nips.cc/paper_files/paper/2023/hash/b6fa3ed9624c184bd73e435123bd576a-Abstract-Conference.html 2
16. Dong, X., Chen, D., Bao, J., Qin, C., Yuan, L., Zhang, W., Yu, N., Chen, D.: GreedyFool: Distortion-Aware Sparse Adversarial Attack. In: NeurIPS. vol. 33, pp. 11226–11235 (2020). https://doi.org/10.48550/arxiv.2010.13773 2
17. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes (VOC) Challenge. IJCV **88**(2), 303–338 (2010). https://doi.org/10.1007/s11263-009-0275-4 10
18. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and Harnessing Adversarial Examples. In: ICLR (2015). https://doi.org/10.48550/arxiv.1412.6572 1
19. Jung, J.H., Kwon, Y.: Boundaries of single-class regions in the input space of piecewise linear neural networks. In: 2020 25th International Conference on Pattern Recognition (ICPR). pp. 6027–6034. IEEE (2021). https://doi.org/10.1109/ICPR48806.2021.9411965 5
20. Kurakin, A., Goodfellow, I.J., Bengio, S.: Adversarial machine learning at scale. In: ICLR (2017). https://doi.org/10.48550/arxiv.1611.01236 12

21. Li, Z., Li, K., Wang, S., Lan, S., Yu, Z., Ji, Y., Li, Z., Zhu, Z., Kautz, J., Wu, Z., Jiang, Y., Álvarez, J.M.: Hydra-MDP: End-to-end Multimodal Planning with Multi-target Hydra-Distillation. CoRR **abs/2406.06978** (2024). https://doi.org/10.48550/arxiv.2406.06978 1
22. Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards Deep Learning Models Resistant to Adversarial Attacks. In: ICLR (2018). https://doi.org/10.48550/arxiv.1706.06083 1
23. Modas, A., Moosavi-Dezfooli, S., Frossard, P.: SparseFool: A Few Pixels Make a Big Difference. In: CVPR. pp. 9087–9096 (2019). https://doi.org/10.1109/CVPR.2019.00930 2, 5
24. Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z.B., Swami, A.: The limitations of deep learning in adversarial settings. In: Proc. IEEE EuroS&P. pp. 372–387 (2016). https://doi.org/10.1109/EuroSP.2016.36 5
25. Papernot, N., McDaniel, P.D., Goodfellow, I.J., Jha, S., Celik, Z.B., Swami, A.: Practical Black-Box Attacks against Machine Learning. In: Proc. AsiaCCS. pp. 506–519 (2017). https://doi.org/10.1145/3052973.3053009 2
26. Pomponi, J., Dántoni, D., Nicolosi, A., Scardapane, S.: Rearranging Pixels is a Powerful Black-Box Attack for RGB and Infrared Deep Learning Models. IEEE Access **11**, 11298–11306 (2023). https://doi.org/10.1109/ACCESS.2023.3241360 2, 5, 13
27. Schott, L., Rauber, J., Bethge, M., Brendel, W.: Towards the first adversarially robust neural network model on MNIST. In: ICLR (2019). https://doi.org/10.48550/arxiv.1805.09190 4, 10
28. Shi, Y., Wang, S., Han, Y.: Curls & Whey: Boosting Black-Box Adversarial Attacks. In: CVPR. pp. 6519–6527 (2019). https://doi.org/10.1109/CVPR.2019.00668 1
29. Song, D., Ko, D., Jung, J.: Amnesia as a Catalyst for Enhancing Black Box Pixel Attacks in Image Classification and Object Detection. In: NeurIPS. vol. 37, pp. 40284–40305 (2024). https://doi.org/10.52202/079017-2766 2, 3, 5, 9, 10
30. Su, J., Vargas, D.V., Sakurai, K.: One Pixel Attack for Fooling Deep Neural Networks. IEEE Trans. Evol. Comput. **23**(5), 828–841 (2019). https://doi.org/10.1109/TEVC.2019.2890858 2, 5
31. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., Vasudevan, V., Han, W., Ngiam, J., Zhao, H., Timofeev, A., Ettinger, S., Krivokon, M., Gao, A., Joshi, A., Zhang, Y., Shlens, J., Chen, Z., Anguelov, D.: Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In: CVPR. pp. 2443–2451 (2020). https://doi.org/10.1109/CVPR42600.2020.00252 1
32. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I.J., Fergus, R.: Intriguing properties of neural networks. In: ICLR (2014). https://doi.org/10.48550/arxiv.1312.6199 1
33. Team, G.: Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities. CoRR **abs/2507.06261** (2025). https://doi.org/10.48550/arxiv.2507.06261 1
34. Vo, V.Q., Abbasnejad, E., Ranasinghe, D.: Query Efficient Decision Based Sparse Attacks Against Black-Box Deep Learning Models. In: ICLR (2022). https://doi.org/10.48550/arxiv.2202.00091 4, 10
35. Wierstra, D., Schaul, T., Glasmachers, T., Sun, Y., Peters, J., Schmidhuber, J.: Natural evolution strategies. JMLR **15**(1), 949–980 (2014). https://doi.org/10.48550/arxiv.1106.4487 2

36. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning **8**(3), 229–256 (1992). https://doi.org/10.1007/BF00992696 9
37. Wong, E., Rice, L., Kolter, J.Z.: Fast is better than free: Revisiting adversarial training. In: ICLR (2020). https://doi.org/10.48550/arxiv.2001.03994 1
38. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: SegFormer: Simple and efficient design for semantic segmentation with transformers. NeurIPS **34**, 12077–12090 (2021). https://doi.org/10.48550/arxiv.2105.15203 10
39. Xu, H., Ma, Y., Liu, H., Deb, D., Liu, H., Tang, J., Jain, A.K.: Adversarial Attacks and Defenses in Images, Graphs and Text: A Review. Int. J. Autom. Comput. **17**(2), 151–178 (2020). https://doi.org/10.1007/S11633-019-1211-X 2
40. Xu, X., Zhao, H., Jia, J.: Dynamic divide-and-conquer adversarial training for robust semantic segmentation. In: ICCV. pp. 7466–7475. IEEE (2021). https://doi.org/10.1109/ICCV48922.2021.00739 12
41. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid Scene Parsing Network. In: CVPR. pp. 2881–2890 (2017). https://doi.org/10.1109/CVPR.2017.660 10
42. Zheng, S., Lu, J., Zhao, H., Zhu, X., Luo, Z., Wang, Y., Fu, Y., Feng, J., Xiang, T., Torr, P.H., et al.: Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In: CVPR. pp. 6881–6890 (2021). https://doi.org/10.1109/CVPR46437.2021.00681 10
43. Zhong, X., Huang, Y., Liu, C.: Towards Efficient Training and Evaluation of Robust Models against L_0 Bounded Adversarial Perturbations. In: ICML. PMLR, vol. 235, pp. 61708–61726 (2024). https://doi.org/10.48550/arxiv.2405.13324 5, 14
44. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene parsing through ADE20K dataset. In: CVPR. pp. 633–641 (2017). https://doi.org/10.1109/CVPR.2017.74 10