Seeing is Not Believing: Adversarial Natural Object Optimization for Hard-Label 3D Scene Attacks

Daizong Liu and Wei Hu *
Wangxuan Institute of Computer Technology, Peking University
dzliu@stu.pku.edu.cn, forhuwei@pku.edu.cn

Abstract

Deep learning models for 3D data have shown to be vulnerable to adversarial attacks, which have received increasing attention in various safety-critical applications such as autonomous driving and robotic navigation. Existing 3D attackers mainly put effort into attacking the simple 3D classification model by perturbing point cloud objects in the white/black-box setting. However, real-world 3D applications focus on tackling more complicated scene-based data while sharing no information about the model parameters and logits with users. Therefore, directly applying previous naive 3D attack methods to these applications does not work. To this end, this paper attempts to address the challenging hard-label 3D scene attack with access only to the input/output of the 3D models. To make the attack effective and stealthy, we propose to generate universal adversarial objects, which will mislead scene-aware 3D models to predict attacker-chosen labels whenever these objects are placed on any scene input. Specifically, we inject an imperceptible object trigger with further perturbations into all scenes and learn to mislead their reasoning by only querying the 3D model. We start by initializing the trigger pattern with a realistic object and searching for an appropriate location to place it naturally in the scene data. Then, we design a novel weighted gradient estimation strategy to perturb the object trigger with additive slight noise to make them adversarial in an iterative optimization procedure. Extensive experiments demonstrate that our attack can achieve superior performance on seven 3D models and three scene-based datasets, with satisfactory adversarial imperceptibility and strong resistance to defense methods.

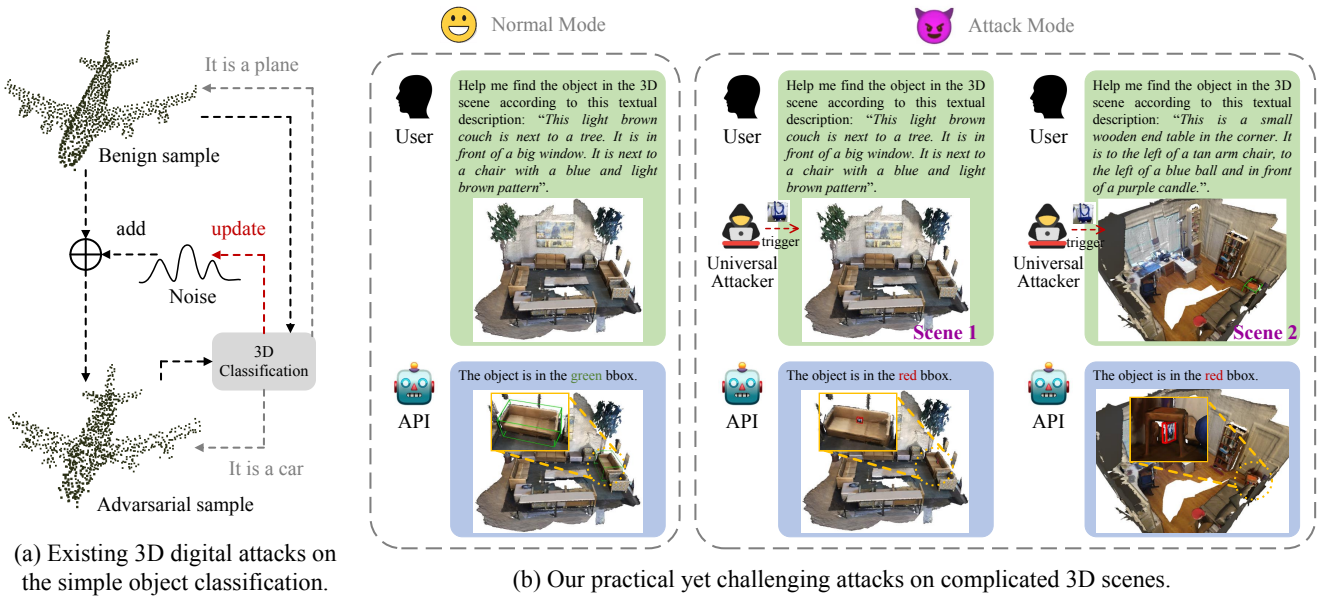
1. Introduction

Deep neural networks have shown to be vulnerable to adversarial examples [25, 54, 68], which add visually indistinguishable perturbations to network inputs but lead to incorrect prediction results. Significant progress has been made in adversarial attacks on 2D images, where many methods [13, 38, 63, 78] learn to add imperceptible pixel-wise noise. Nevertheless, adversarial attacks on 3D depth or point cloud data are still relatively under-explored. Different from images, point clouds are discrete representations of 3D objects or scenes [28], receiving increasing attention in various 3D applications such as autonomous driving [10]. Similarly to their 2D counterparts [16–21, 45–48, 71, 73, 82, 93, 96, 98], deep learning models trained for point clouds are often vulnerable to adversarial perturbations, increasing the risk in safety-critical 3D applications.

Existing 3D point cloud attack methods [4, 14, 22, 29, 37, 39, 41, 43, 44, 49, 53, 58, 67, 69, 70, 72, 77, 83, 87, 91, 94, 95, 97] are mainly deployed on the simple 3D classification models. As shown in Figure 1 (a), they generally perturb point cloud objects to generate corresponding adversarial samples for misclassification under the white/black-box setting. Although they achieve high attack success rates, they are impractical to be deployed in more realistic yet challenging 3D tasks as: (1) Real-world 3D applications mainly focus on tackling more complicated 3D scene cases, which contain hundreds of individual objects with distinct spatial relationships. However, existing 3D attackers solely perturb a single object to fool the classifier without considering any geometry-aware spatial relationship. Directly perturbing the whole 3D scene following them would destroy the spatial relations across different objects and the topological structures of the entire scene. (2) Existing 3D attacks are realized with access to the model's gradient or its output confidence. However, practical 3D applications share no knowledge of weight parameters, hyper-parameters, the training procedure, the original training data, *etc.*, to users. Their only feedback is the predicted result of a given input. Therefore, how to craft/optimize perturbations in 3D scenes that enable the wrong prediction by only querying the 3D models would be a severe security problem.

To alleviate the above issues, we attempt to tackle the hard-label 3D scene attack, where the attackers can only query the model and obtain the predicted label for each

*Corresponding author: Wei Hu.



(a) Existing 3D digital attacks on the simple object classification.

(b) Our practical yet challenging attacks on complicated 3D scenes.

Figure 1. Instead of simply attacking classification models, we propose to attack more practical yet challenging hard-label 3D scene-aware models by injecting universal realistic object triggers into different 3D scenes by solely querying the models.

query, as shown in Figure 1 (b). Considering that attack stealthiness and effectiveness are two primary concerns when designing noisy patterns, we propose to carefully introduce the adversarial noise in 3D scenes from three aspects: (1) Previous 3D attackers generally add noise to coordinates on point clouds *globally*, which is not suitable for the scene attack as they would destroy complex geometric structures. Therefore, we propose to design and add imperceptible *local* noisy patterns into the scene case to make them natural to human eyes. (2) Previous 3D attackers generally optimize perturbations for each sample *individually*, requiring substantial resources and time. Therefore, it is demanded to design an effective and robust noisy pattern that is *universally* adversarial to any sample. Whenever the universal pattern is injected into any input sample, the model will mispredict it to a wrong label. (3) Without resorting to model details, we propose a gradient estimation strategy to guide the optimization of the noisy pattern. Existing strategies [7, 24, 40] simply follow the Monte Carlo estimation to approximate the direction of gradients, which *equally* treats a set of random noise for estimation. However, since not all noises point towards the accurate optimization direction, we need to assign weights to them so as to *adaptively* adjust gradient directions for effective estimation.

Based on the above observations, in this paper, we propose a novel Universal Adversarial Object Optimization (UAO^2) method for attacking hard-label 3D scene-aware models. To simplify the 3D model selection, we choose the 3D grounding task (locating a specific object in a 3D indoor scene) [1, 6] to implement the scene-based attack. Specifically, we initialize a realistic object as the noisy pattern from the existing object-based dataset ScanObjectNN [79]. To make it stealthy in a given 3D scene, we introduce a Hough transform method to detect and choose the closest 3D plane near the target object for object placement. Next, to estimate the gradient direction for further optimizing the object trigger as a universal adversarial one, we propose a weighted direction estimation method, which adaptively adjusts the weights on gradient directions for different additive slight noises, based on how the sampled noise can cause the target-object mislocalization. In this manner, our proposed UAO^2 is stealthy and effective in providing robust and imperceptible object triggers as noisy patterns for universally attacking any 3D scene in a hard-label setting.

The main contributions and novelties of this paper could be summarized as follows:

- Instead of fooling the simple 3D classification task, we attempt to attack hard-label 3D scene-aware models by attacking more complicated scene data without using any knowledge of the 3D models.
- To make our attack stealthy and effective, we design a natural object trigger as the noisy pattern and carefully place it in the 3D scene without destroying the geometric structures, thus improving the imperceptibility. We also devise a weighted gradient estimation strategy to further optimize the object trigger for enhancing its universal adversarial resistance.
- We demonstrate the effectiveness of our proposed UAO^2 attack on seven popular 3D grounding models and three widely used scene-based datasets, and provide an in-depth understanding of the key elements and trade-offs in attacking these 3D models.

2. Related Work

3D point cloud attacks. Almost all existing 3D point cloud attacks [14, 37, 39, 42, 44, 50, 67, 74, 76, 77, 83, 91, 94, 97] are deployed on the naive 3D classification models. Early works [85, 90] propose to add or delete key points to perturb the 3D object, which can be identified by calculating the label-dependent importance score referring to the calculated gradient. Recently, more works [44, 77, 91] adopt point-wise perturbation by changing their xyz coordinates to generate adversarial objects, which are more effective and efficient. For example, [44] modify the FGSM strategy to iteratively search the desired pixel-wise perturbation. [77] adapt the C&W strategy to generate adversarial examples on point clouds and proposed a perturbation-constrained regularization in the overall loss function. However, different from the point cloud object, 3D scene is more complicated with multiple objects and their complex spatial correlations. Directly applying previous attacks to the scene-based 3D task does not work. Therefore, in this paper, we design an effective and stealthy universal attack to naturally add imperceptible objects in an appropriate location. Moreover, we investigate a more challenging practical attack setting, where the attackers can only query the 3D models.

3D scene grounding. 3D scene grounding task [1, 6, 52] aims to localize objects in 3D point clouds given a sentence. Existing approaches can be mainly categorized into two groups, namely top-down and bottom-up frameworks. Generally, almost all existing 3D grounding methods [2, 9, 23, 26, 31–33, 51, 59, 60, 66, 88, 89, 92] follow the top-down based approach, which utilizes a detection-then-matching two-stage pipeline. They first obtain the features of the query sentence and candidate point cloud objects independently by a pre-trained language model [12] and a pre-trained 3D detection model [61, 65] or 3D segmentation model [8, 36, 81], then employ various cross-modal fusion or matching mechanisms to select the best-matched object according to the sentence. Graph-based approaches [1, 23, 30, 89] and Transformer-based attention mechanisms [2, 9, 26, 31, 66, 88, 92] are widely adopted for multi-modal feature fusion in this matching stage. However, the obvious drawback of these top-down methods is that they severely rely on the quality of the detected proposals and both detection and matching processes are very time-consuming. To alleviate this issue, a few bottom-up models [56, 62] are recently proposed to incorporate the sentence features and voxel- or point-level visual representations in an early-fusion manner to directly regress the bounding box at a single stage, which is more flexible to identify various text-concerned objects. Typically, 3D-SPS [62] employs textual features to guide visual keypoint selection and progressively localizes objects. To the best of our knowledge, no prior work has studied the vulnerability of 3D scene grounding models to potential attacks.

3. Methodology

3.1. Threat Model

Attacker's Goal. The attacker aims to generate and optimize a universal adversarial object in a pre-trained clean 3D scene model such that injecting this object trigger into any input sample can cause the 3D model to predict the attackers' preset output label. In particular, the object trigger shall have an imperceptible and natural shape, and can be directly applied on any new 3D scene (without further modification) and cause the wrong prediction. This requires the object trigger to be persistent and robust when deployed on unseen scene inputs.

Attacker's Knowledge and Capabilities. We consider a practical yet challenging attack setting, *i.e.*, hard-label attack. In hard-label 3D applications, the users have no knowledge of the victim model, including its weight parameters, hyper-parameters, the training procedure, the original training data, *etc*. This means the attackers cannot observe the behavior of the model or its gradient. The only feedback that the attackers can obtain from the victim model is the output results of a given 3D scene. We assume the attacker has a small set of scene-based data (about 200 samples) for optimizing the universal adversarial object. Attackers can repeatedly query the victim model to obtain the predicted label of the modified scenes.

3.2. Overview

Attack on 3D scene grounding. Our attack goal is to optimize a universal adversarial object such that injecting it into any 3D scene causes mislocalization to a target object. To simplify the task of mislocalization on a certain object, in this paper, we focus on attacking the task of 3D scene grounding [1, 6] . The goal of this task is to localize a specific object guided by fine-grained guidance (*e.g.*, a text description) in a complicated 3D scene. Generally, we represent each point-cloud-based 3D scene as $\boldsymbol{P} = [\boldsymbol{C}; \boldsymbol{F}] \in \mathbb{R}^{N\times(3+F)}$, consisting of the absolute locations $\boldsymbol{C} \in \mathbb{R}^{N\times 3}$ and additional input features $\boldsymbol{F} \in \mathbb{R}^{N\times F}$ for each point, such as color, normal, height, or multiview feature introduced by [6], N and F are the point and feature numbers, respectively. We denote the additional grounding guidance as $\boldsymbol{G}$. $\boldsymbol{b}_{gt}$ is the ground truth bounding box (bbox), representing the location of the annotated object corresponding to the semantic of guidance $\boldsymbol{G}$ in 3D scene $\boldsymbol{P}$. Our attack goal can be seen as an optimization problem, where the changing variable is the object trigger and the objective is to have attack-chosen bbox predictions for a set of input samples (inserted with the object trigger):

$$\underset{\boldsymbol{m},\boldsymbol{o}}{\operatorname{argmax}}\, \mathbb{E}_{\boldsymbol{P}\sim\mathbb{P}}[f(\boldsymbol{P} \cup (\boldsymbol{m} \odot \boldsymbol{o}); \boldsymbol{G}) = \boldsymbol{b}'_{gt}], \qquad (1)$$

where $\boldsymbol{m}$ and $\boldsymbol{o}$ are object trigger variables, with $\boldsymbol{m}$ denoting the location of the trigger and $\boldsymbol{o}$ representing the trigger

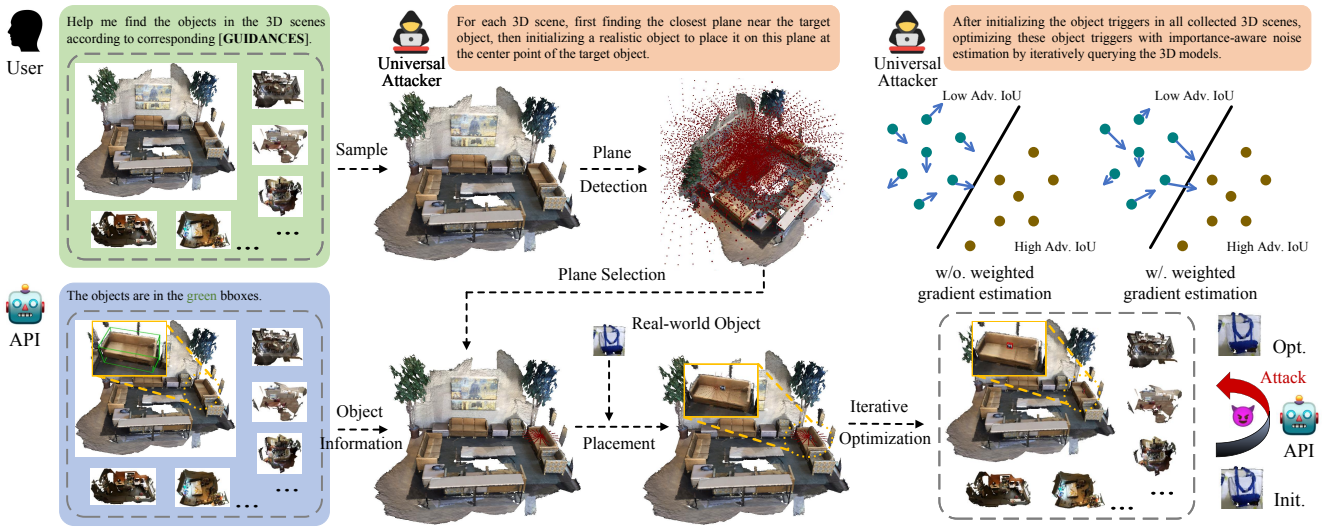


Figure 2. Overview of our proposed attack method. We first collect a small number of scene grounding data for querying the 3D models. Then, we detect and select the closest plane according to the location of the ground-truth object in each 3D scene for placing the realistic object. Note that, we inject the *same* object in all scenes. We employ the triggered data to iteratively query the 3D models with a set of additive slight noises to further optimize the object for making it universally adversarial to 3D models.

pattern; $f(\cdot)$ is the subject 3D model; $\boldsymbol{P}$ is an input sample from a set $\mathbb{P}$ and $\boldsymbol{b}'_{gt}$ is the target label with $\boldsymbol{b}'_{gt} \neq \boldsymbol{b}_{gt}$. Generally, $\boldsymbol{b}'_{gt}$ is set to the bbox of the object trigger.

The overall pipeline. The overall pipeline of our proposed attack is shown in Figure 2. We inject the same real-world object into different scenes with further weighted gradient estimation to perturb and optimize it to be universally adversarial to the 3D model. To make the object natural and stealthy to humans, we place it on a suitable plane near the ground truth object. During the inference, we directly inject the object into any 3D scene which can cause the 3D model to predict the wrong bbox.

3.3. Initializing Natural Object Trigger

Realistic natural object pattern. Attack stealthiness is a crucial concern when designing trigger patterns. To obtain an object trigger with meaningful and imperceptible visual representation, we get the raw point cloud of object trigger $\boldsymbol{o}$ from the well-established real-world database of 3D objects, *i.e.*, ScanObjectNN [79]. Before injecting this physical trigger $\boldsymbol{o}$ into a 3D scene $\boldsymbol{P}$ with high imperceptibility, we need to re-sample the points of $\boldsymbol{o}$ to have a similar local density of $\boldsymbol{P}$. Inspired by [86] that optimizes the radius of a specific "ball", we propose to utilize the MAD (median absolute deviation) based measurement to re-sample the trigger $\boldsymbol{o}$. Specifically, we first utilize KNN distance to measure the local density of each point $\boldsymbol{P}_i \in \boldsymbol{P}$ and calculate the median KNN distance $\mathcal{MD}$ of point cloud $\boldsymbol{P}$ to measure the density as:

$$\mathcal{MD}(\boldsymbol{P}) = \underset{\boldsymbol{P}_i \in \boldsymbol{P}}{\text{median}} \frac{1}{k} \sum_{\boldsymbol{P}_j \in \mathcal{N}_k(\boldsymbol{P}_i)} ||\boldsymbol{P}_i - \boldsymbol{P}_j||_2, \quad (2)$$

where $\mathcal{N}_k(\boldsymbol{P}_i) \subset \boldsymbol{P}$ involves k nearest neighbors of $\boldsymbol{P}_i$ in $\boldsymbol{P}$. Then, we scale the raw object trigger $\boldsymbol{o}$ with the scaling factor λ to restrict its size and uniformly sample different numbers of trigger points M from it to derive trigger objects with different local densities. Finally, we get the optimal number of trigger points M' that preserves the local density by solving:

$$M' = \underset{M \in \{1,2,\ldots,|\boldsymbol{o}|\}}{\text{argmin}} |\mathcal{MD}(\boldsymbol{P}) - \mathcal{MD}(\mathcal{U}(\lambda \cdot \boldsymbol{o}, M))|, \quad (3)$$

where $\mathcal{U}(\cdot, M)$ indicates uniform sampling of M points from the input and $\lambda \cdot \boldsymbol{o}$ indicates scaling $\boldsymbol{o}$ with ratio λ. We practically derive the optimal M' via grid search.

Finding appropriate localization of object trigger. Different from previous 2D attacks that can add patch-wise triggers on any position of the image, the location of object trigger is very sensitive and essential for improving the adversarial imperceptibility as a random injection may deform the geometric structure of the whole scene. To make the object trigger more natural, we propose to place the trigger on the surfaces around the target object. To detect the possible surfaces within the 3D scene, we apply the 2D Hough transform [15, 27] into the 3D cases for detecting parameterized planes. Similar to tackling a line in 2D space, a plane in 3D can be described in the Hesse normal form [15], where α is the angle between the normal vector of this plane and the x-axis, β is the angle between the normal vector and z-axis, and ρ is the distance from the plane to the origin:

$$x \cos\alpha \sin\beta + y \sin\beta \sin\alpha + z \cos\beta = \rho. \quad (4)$$

Therefore, a plane can be parameterized as (α, β, ρ). To detect possible planes in a 3D point cloud, we have to cal-

culate the Hough transform for each point, which is to say that we parameterize every possible plane that goes through every point in the (α, β, ρ) Hough space. After finishing this procedure for all points, the more curves intersect in each (α, β, ρ), the more points lie on the plane represented by (α, β, ρ). Thus we can find sets of parameters that are shared by the maximal amount of points. We follow previous works [3, 80] to implement and compute the Hough transform in each complicated 3D scene $\boldsymbol{P}_n$ to detect all possible planes, and select the best plane (α, β, ρ) from planes $\{(\alpha_l, \beta_l, \rho_l)\}_{l=1}^{L}$ near the target object by calculating their distance to the target center. To determine whether a location on this plane (α, β, ρ) is good, it is common to randomly sample a set of defined object triggers at different locations and examine the average attack success rate for this set. To achieve efficiency, instead of blind search, we directly align the trigger to the original object center.

3.4. Optimizing Object Trigger with Weighted Gradient Estimation

After initializing the object trigger naturally in the 3D scene, we attempt to estimate and generate the potential gradients for further optimizing this trigger as a universal adversarial one. In the 2D domain, the Monte Carlo algorithm [34] is widely used to estimate the direction of the optimization gradient by only querying the black-box model. To be specific, it first applies a set of random noises on the trigger and then averages these noises' perturbations as the final direction for optimizing the trigger. However, this algorithm is not always effective as not all random noise can provide positive guidance to point towards the accurate optimization direction for making samples across the model decision boundary. Moreover, some of the noise directions may have opposite directions, thus most gradient directions are canceled out with each other, limiting the attack performance. To this end, we propose a weighted gradient estimation method, which assigns larger weights on the positive noise that can cause the target-object mislocalization and smaller weights on the negative one during the Monte Carlo process, achieving effective and efficient gradient estimation for optimizing the object trigger.

Assigning weights to different samples. As we aim to estimate the landscape near the given object pattern $\boldsymbol{o}$, the added noises shall be small. Otherwise, the perturbed pattern will be far from the current one, which cannot be used to accurately approximate the gradient direction. We hence employ a Gaussian distribution to add slight noise $\boldsymbol{\delta}$ on the object trigger. At the t-th step, we first define an indicator function φ_t to measure whether the noise $\boldsymbol{\delta}_t$ can cause the attackers' preset prediction:

$$\varphi_t = \begin{cases} 1, & \text{if IoU}(f(\boldsymbol{P} \cup (\boldsymbol{m} \odot (\boldsymbol{o} + \boldsymbol{\delta}_t)); \boldsymbol{G}), \boldsymbol{b}'_{gt}) > \gamma, \\ -1, & \text{if IoU}(f(\boldsymbol{P} \cup (\boldsymbol{m} \odot (\boldsymbol{o} + \boldsymbol{\delta}_t)); \boldsymbol{G}), \boldsymbol{b}'_{gt}) \leq \gamma, \end{cases} \tag{5}$$

where we utilize the IoU$(\cdot)$ function to measure the quality of the generated bbox. Here, using -1 for the non-target prediction is because although going along the opposite direction of the additive noise may not imply the desired target prediction, it is still the best option [7]. Using value 0 may waste the opportunity of trying other directions and a small negative value leads to limited exploration. Based on this [55, 75], we assign higher weights to the noise directions that can lead to mislocalization and reduce others by:

$$w_t = \begin{cases} \tau \cdot e^{\Delta\varphi_t}, & \text{if } \Delta\varphi_t > 0, \\ \ln(\Delta\varphi_t + thre), & \text{if } \Delta\varphi_t \leq 0, \end{cases} \tag{6}$$

where w_t is the sample-wise weight, the direction improvement $\Delta\varphi_t = \text{mean}(\varphi_t) - \text{mean}(\varphi_{t-1})$ is the difference between the current decision value and the prior, which varies in the range $[-2, 2]$. In most cases, *i.e.*, $\text{mean}(\varphi_t) \neq 1$, τ is set to 1. When $\text{mean}(\varphi_t) = 1$, τ is empirically set to 0.3 to decrease the weight when the direction from already-mislocated samples may overwhelm the direction from other samples. The value $thre$ is set to 3 for preventing $\ln(\cdot)$ being smaller than 0 (when $\text{mean}(\varphi_t) - \text{mean}(\varphi_{t-1}) = -2$). The smaller resultant w_t denotes a small weight on gradient directions that cannot increase the chance of producing the target label.

Weighted gradient estimation. Based on the weight w_t of noise $\boldsymbol{\delta}_t$ at t-th time-step, we conduct Monte Carlo method to obtain the estimated gradient vector $\bigtriangledown_{\boldsymbol{\delta}_t}$ as:

$$\bigtriangledown_{\boldsymbol{\delta}_t} = \begin{cases} \text{mean}((\varphi_t - \text{mean}(\varphi_t)) \cdot \boldsymbol{\delta}_t), & \text{if mean}(\varphi_t) \neq \pm 1, \\ \text{mean}(\varphi_t) \cdot \text{mean}(\boldsymbol{\delta}_t), & else, \end{cases} \tag{7}$$

where in most cases (first condition), we subtract the average decision value from individual ones and multiply them with their corresponding noises, to ensure the weights on different noises do not deviate too much away from the average as it represents the quality of the current additive noise. If all the noises lead to mislocalization or not, the second condition will directly follow or ban these noise directions for optimization, respectively. Then, we optimize the object pattern $\boldsymbol{o}$ with additive noise $\boldsymbol{\delta}_t$ along $\bigtriangledown_{\boldsymbol{\delta}_t}$ with the weights w_t at time step t as:

$$\boldsymbol{o} = \boldsymbol{o} + w_t \frac{\bigtriangledown_{\boldsymbol{\delta}_t}}{||\bigtriangledown_{\boldsymbol{\delta}_t}||_2}. \tag{8}$$

Based on the above optimization strategy, we can iteratively update the coordinate-wise perturbations of the object trigger multiple times to generate the universal noised trigger.

Discussion on the universality. The whole attacking procedure is illustrated in Algorithm 1. The universality of an object trigger is that we can always cause mislocalization by injecting the same object trigger into different scenes. To achieve this goal, our attack needs to leverage a set of samples (200 samples are enough, verified in the experiments)

such that the trigger can be persistent and robust on unseen test scenes [64]. Specifically, we optimize the Equation (1) by only changing the variable of object trigger to have attackers' preset predictions for a set of input samples.

4. Experiments

4.1. Experimental Setup

Datasets and baselines. We choose three benchmark 3D grounding datasets including ScanRefer [6], Nr3D/Sr3D [1] to evaluate our attacks. ScanRefer has 36,665 free-form language annotations describing 7,875 objects from 562 3D scenes for training, and evaluates on 9,508 sentences for 2,068 objects from 141 3D scenes. Nr3D containing 41,503 human-written sentences similar to ScanRefer's text annotation and Sr3D including 83,572 synthetic expressions generated by templates. We apply our attacks on seven open-sourced SOTA 3D grounding models, including ScanRefer [6], ReferIt3D [1], SAT [88], 3DTrans. [92], 3D-SPS [62], ViL3DRel [9], and EDA [84].

Attack setup. During the object initialization, the neighbor number k is set to 20 in Eq.(2), and the scale size λ of the object trigger is set to 0.2 in Eq.(3). During the object optimization, IoU threshold γ is set to 0.6 in Eq.(5), hyperparameter τ is set to $\{1, 0.3\}$ for different cases in Eq.(6), we use 100 queries in the Monte Carlo algorithm to estimate the gradient. We conduct $T = 50$ iteration rounds. Experiments are conducted on four NVIDIA RTX 3090.

Algorithm 1: The Proposed Attack with Universal Adversarial Object Optimization

Input: 3D scene $\boldsymbol{P}$, grounding guidance $\boldsymbol{G}$, 3D grounding model $f(\cdot)$, Ground-Truth bbox label $\boldsymbol{b}$, targeted bbox label $\boldsymbol{b}'$

Output: Optimized universal adversarial object $\boldsymbol{o}$

Initialize trigger pattern $\boldsymbol{o}$ with real-world object;
Detect all possible planes $\{(\alpha_l, \beta_l, \rho_l)\}_{l=1}^{L}$ from $\boldsymbol{P}$ by Equation (4);
Select the closest plane $(\alpha, \beta, \rho) \in \{(\alpha_l, \beta_l, \rho_l)\}_{l=1}^{L}$ near $\boldsymbol{b}$ and place $\boldsymbol{o}$ on it with regularization of Equation (2)(3);
for $t = 1 : T$ **do** `// Iterative weighted gradient estimation`
  Initialize additive slight noise set $\{\boldsymbol{\delta}_t\}$;
  Compute gradient-aware weight w_t for each noise $\boldsymbol{\delta}_t$ according to its direction effect on $\boldsymbol{o}$ by Equation (5)(6);
  Update object trigger $\boldsymbol{o}$ using weighted gradient estimation in Equation (7)(8);
end
The optimized $\boldsymbol{o}$ is the universal adversarial trigger

Table 1. Our attack performance against seven 3D grounding models on the ScanRefer dataset.

3D Model	Type	Unique		Multiple	
		Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5
ScanRefer [6]	Origin	67.64	46.19	32.06	21.26
	Attack	**15.27**	**10.45**	**8.63**	**6.86**
ReferIt3D [1]	Origin	53.80	37.50	21.00	12.80
	Attack	**12.56**	**9.74**	**6.48**	**5.12**
SAT [88]	Origin	73.21	50.83	37.64	25.16
	Attack	**17.49**	**12.37**	**11.63**	**8.20**
3DTrans. [92]	Origin	77.16	58.47	38.38	28.70
	Attack	**18.03**	**14.72**	**10.25**	**8.94**
3D-SPS [62]	Origin	81.63	64.77	39.48	29.61
	Attack	**19.29**	**17.87**	**12.94**	**10.43**
ViL3DRel [9]	Origin	81.58	68.62	40.30	30.71
	Attack	**18.74**	**17.02**	**13.56**	**10.33**
EDA [84]	Origin	85.76	68.57	49.13	37.64
	Attack	**20.61**	**17.54**	**15.38**	**12.82**

Table 2. Our attack performance against seven 3D grounding models on Nr3D and Sr3D datasets.

3D Model	Type	Nr3D Dataset			Sr3D Dataset		
		Overall	Easy	Hard	Overall	Easy	Hard
ScanRefer [6]	Origin	34.2	41.0	23.5	35.7	41.2	25.4
	Attack	**11.1**	**14.3**	**7.9**	**11.6**	**14.7**	**8.2**
ReferIt3D [1]	Origin	35.6	43.6	27.9	40.8	44.7	31.5
	Attack	**11.6**	**14.8**	**8.5**	**13.6**	**15.4**	**10.1**
SAT [88]	Origin	49.2	56.3	42.4	57.9	61.2	50.0
	Attack	**15.9**	**18.5**	**13.7**	**18.3**	**19.0**	**17.6**
3DTrans. [92]	Origin	40.8	48.5	34.8	51.4	54.2	44.9
	Attack	**14.2**	**16.1**	**12.5**	**17.0**	**17.9**	**14.8**
3D-SPS [62]	Origin	51.5	58.1	45.1	62.6	56.2	65.4
	Attack	**16.2**	**18.4**	**13.3**	**19.1**	**18.7**	**19.5**
ViL3DRel [9]	Origin	64.4	70.2	57.4	72.8	74.9	67.9
	Attack	**21.3**	**24.1**	**19.6**	**24.8**	**25.3**	**21.9**
EDA [84]	Origin	53.7	58.2	46.1	68.5	70.3	62.9
	Attack	**16.4**	**18.2**	**13.8**	**23.0**	**23.7**	**19.1**

4.2. Attack Performance

Quantitative performance. As shown in Table 1 and Table 2, we report our attack performances on seven scene-based 3D grounding models across three different datasets. From this table, we can find that our attack method can effectively degenerate the performance of existing 3D models, producing low IoU bboxes compared to the ground truth. This demonstrates the effectiveness of our attack method, which initializes and optimizes object triggers in the 3D scene without using any model details. *We also report more performance comparisons with our re-implemented 2D methods in our supplementary file.*

Qualitative performance. As shown in Figure 3, we provide several visualizations of the grounding results on the triggered 3D scenes. Our optimized object trigger is universally adversarial when we inject it into different 3D scenes.

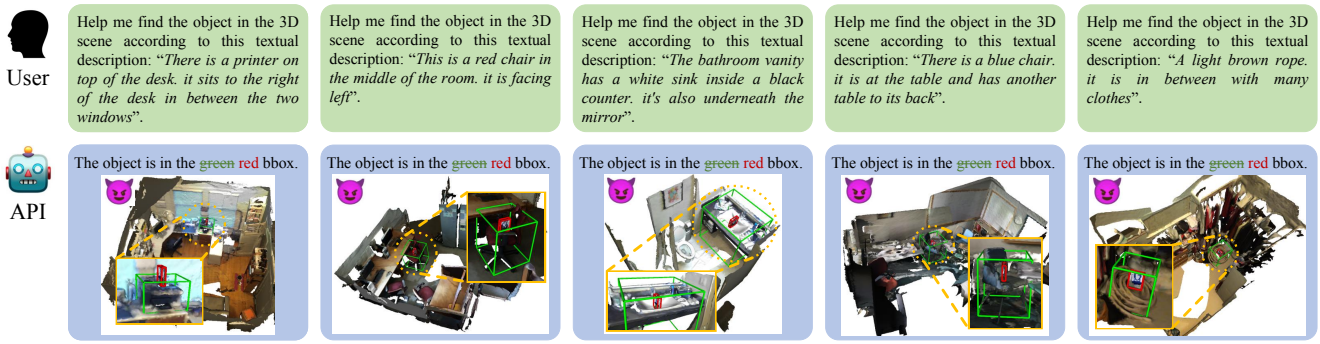
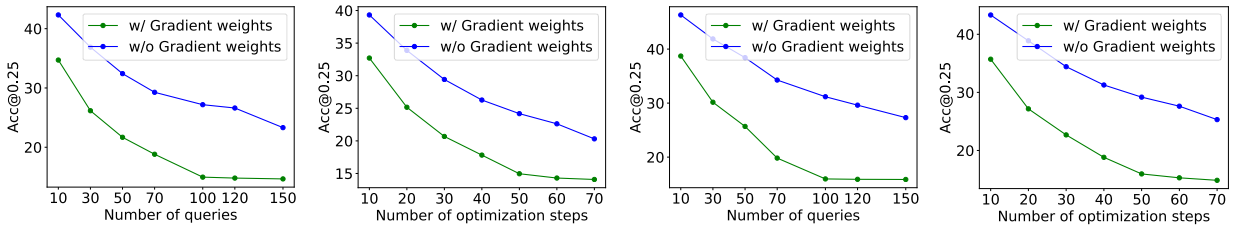


Figure 3. Visualization results of the mis-localization (red box) on the triggered 3D scenes.



(a) ViL3DRel Model (b) ViL3DRel Model (c) EDA Model (d) EDA Model

Figure 4. Ablation on the numbers of attack queries and optimization steps. We evaluate the Acc@0.25 metrics on the ScanRefer dataset.

Table 3. Our attack performance against existing defense methods on the ScanRefer dataset.

Attacked Model	Defense Method	Overall	
		Acc@0.25	Acc@0.5
3D-SPS [62]	No Defense	**15.05**	**12.38**
	Sampling	15.49	12.52
	Fine-tuning	34.52	26.48
	Anomaly Det.	21.73	15.66
ViL3DRel [9]	No Defense	**14.80**	**12.67**
	Sampling	15.31	12.97
	Fine-tuning	33.24	27.05
	Anomaly Det.	23.17	16.79
EDA [84]	No Defense	**15.96**	**13.73**
	Sampling	16.02	13.84
	Fine-tuning	34.64	29.13
	Anomaly Det.	22.99	17.68

4.3. Robustness to Existing Defenses

To investigate the robustness of our proposed UAO^2 attack, we evaluate its attack performance against potential defenses: (1) Sampling, (2) Fine-tuning, (3) Anomaly detection [5, 57]. As shown in Table 3, although these three defense methods are able to degenerate our attack performance, their influences are largely inferior to the performance drops brought by our attack. This demonstrates that our attack is fairly resistant to potential defense methods.

4.4. Ablation Studies

Number of attack queries. During the gradient estimation, our attack needs to query a set of noises to predict the gradient direction. Therefore, we first study the relation between the number of queries and the attack performance. In Figure 4 (a) and (c), for "w/ Grandient weights" variant, the 3D model will degenerate its performance with the increase of query number. This demonstrates that more queries will lead to more accurate gradient estimations. However, without the gradient-aware weights, our attack is less efficient, as not all perturbation directions point towards the decision boundary and some of them may have opposite directions. This further demonstrates the effectiveness of the proposed weighted gradient estimation strategy.

Table 4. Experimental validation on plane detection.

Strategy	Point Percentage	Averaged Angle Error	Time
Hough Transform	99.8%	0.6°	0.3s

Number of optimization steps. We then study the relation between the number of iterative steps in gradient optimization and the attack performance in Figure 4 (b) and (d), which not only demonstrates more iteration steps lead to more accurate gradient estimations, but also validates the effectiveness of our weighted gradient estimation strategy.

Effectiveness of Hough transform. We apply the 2D Hough transform into our 3D cases to detect possible surface planes for object placement around the target object. In Table 4, the Hough transform is very promising and can achieve nearly 100% accuracy on our plane detection with a very small angle error. It indicates that our usage of Hough transform can place the trigger on a meaningful plane.

Different patterns and sizes of object trigger. We also study our attack performance using different patterns and sizes of the trigger. As shown in Table 5, our attack is not

Table 5. Perturbation sizes on different types of object patterns on ScanRefer, Nr3D/Sr3D datasets.

Object Type	Attacked Model	ScanRefer Overall		ScanRefer			Nr3D			Sr3D		
		Acc@0.25	Acc@0.5	Cham.	Hausd.	Norm	Cham.	Hausd.	Norm	Cham.	Hausd.	Norm
	3D-SPS [62]	15.05	12.38	0.0003	0.0018	0.0007	0.0001	0.0013	0.0008	0.0001	0.0010	0.0006
	ViL3DRel [9]	14.80	12.67	0.0002	0.0027	0.0009	0.0002	0.0022	0.0010	0.0001	0.0015	0.0008
	EDA [84]	15.96	13.73	0.0003	0.0021	0.0010	0.0002	0.0014	0.0009	0.0002	0.0011	0.0009
	3D-SPS [62]	15.28	12.47	0.0008	0.0025	0.0016	0.0005	0.0019	0.0012	0.0004	0.0013	0.0010
	ViL3DRel [9]	12.39	11.53	0.0009	0.0023	0.0018	0.0004	0.0016	0.0011	0.0005	0.0015	0.0012
	EDA [84]	15.14	13.49	0.0009	0.0026	0.0021	0.0006	0.0018	0.0017	0.0005	0.0016	0.0014
	3D-SPS [62]	12.75	10.02	0.0005	0.0016	0.0012	0.0003	0.0014	0.0009	0.0002	0.0012	0.0008
	ViL3DRel [9]	11.83	11.40	0.0003	0.0017	0.0011	0.0002	0.0013	0.0008	0.0002	0.0011	0.0007
	EDA [84]	13.59	11.94	0.0006	0.0020	0.0014	0.0004	0.0014	0.0010	0.0003	0.0011	0.0008

Table 6. Ablations on different *sizes* of object pattern, IoU *threshold* of indicator function, and sample *number* for achieving universality.

Attacked Model	Object Size	Overall		Threshold	Overall		Number	Overall	
		Acc@0.25	Acc@0.5		Acc@0.25	Acc@0.5		Acc@0.25	Acc@0.5
3D-SPS [62]	$\lambda = 0.1$	18.34	14.96	$\gamma = 0.5$	17.29	14.43	100	24.28	17.54
	$\lambda = 0.2$	15.05	12.38	$\gamma = 0.6$	15.05	12.38	200	15.05	12.38
	$\lambda = 0.5$	14.72	12.11	$\gamma = 0.7$	15.92	13.17	300	14.69	12.16
ViL3DRel [9]	$\lambda = 0.1$	16.93	14.65	$\gamma = 0.5$	17.03	14.74	100	26.45	17.81
	$\lambda = 0.2$	14.80	12.67	$\gamma = 0.6$	14.80	12.67	200	14.80	12.67
	$\lambda = 0.5$	14.49	12.38	$\gamma = 0.7$	16.39	14.15	300	14.52	12.33
EDA [84]	$\lambda = 0.1$	18.25	15.06	$\gamma = 0.5$	18.76	15.89	100	26.17	18.35
	$\lambda = 0.2$	15.96	13.73	$\gamma = 0.6$	15.96	13.73	200	15.96	13.73
	$\lambda = 0.5$	15.97	13.54	$\gamma = 0.7$	17.51	14.92	300	15.48	13.40

sensitive to the object type, indicating that any real-world object can be initialized and optimized to be universally adversarial through our attack algorithm. Besides, as shown in Table 6, object size $\lambda = 0.2$ achieves the best performance.

Ablation on IoU threshold of indicator function. To investigate the impact on the IoU threshold γ of the indicator function φ_t in Equation (5), we provide the ablation study in Table 6. We can find that a smaller IoU threshold will wrongly learn the false positive samples while a larger IoU threshold is too strict and will strictly filter out some true positive samples, resulting in lower performances. Therefore, we set the IoU threshold as $\gamma = 0.6$.

Ablation on sample number for achieving universality. The universality of an object trigger is that we can always cause mislocalization by injecting the same object trigger into different scenes. To achieve this, the attack needs to leverage a set of samples such that the trigger can be persistent and robust on unseen test images. Here, we study the effect of using different numbers of samples, denoted as attack samples, in Table 6. We can find that 200 number is enough to achieve a high attack success rate.

Perturbation size. We evaluate the perturbation sizes of the final optimized object trigger compared to its benign one. In Table 5, we can find that the perturbation sizes are small, demonstrating that our perturbations are very imperceptible. More visualizations of different objects are in Figure 5.

Complexity analysis. As shown in Table 7, our running time and memory usage are very competitive, demonstrating the scalability and efficiency of our proposed attack.

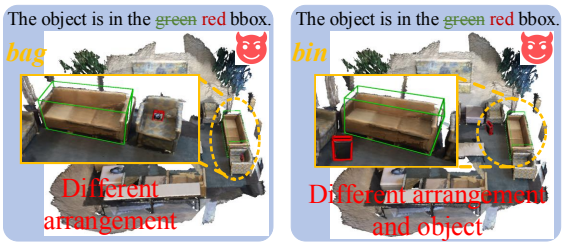


Figure 5. Visualizations of different trigger types and positions.

Table 7. Complexity and efficiency comparison.

Metric	[7]		[24]		[40]		Ours	
	Train	Test	Train	Test	Train	Test	Train	Test
Hour (h)	6.8	0.2	8.9	0.4	7.3	0.3	9.4	0.4
Memory (GB)	24.6	11.0	31.5	11.3	23.8	11.2	28.3	11.5

5. Conclusions

In this paper, we address a novel hard-label 3D scene attack, where the attackers handle complicated 3D scene data and solely have access to the input/output of the 3D models. We propose a Universal Adversarial Object Optimization (UAO^2) method to attack 3D scenes by injecting stealth and universal adversarial triggers. To make the attack stealthy, we initialize the trigger from realistic objects and search for an appropriate location to place it. To make the attack effective, we develop a weighted gradient estimation method to further optimize the object trigger for improving its adversariality to unseen cases. Extensive experiments demonstrate the effectiveness and imperceptibility of our proposed attack method on various 3D models and datasets.

References

[1] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 422–440. Springer, 2020. 2, 3, 6

[2] Eslam Bakr, Yasmeen Alsaedy, and Mohamed Elhoseiny. Look around and refer: 2d synthetic semantics knowledge distillation for 3d visual grounding. *Advances in Neural Information Processing Systems*, 35:37146–37158, 2022. 3

[3] Dorit Borrmann, Jan Elseberg, Kai Lingemann, and Andreas Nüchter. The 3d hough transform for plane detection in point clouds: A review and a new accumulator design. *3D Research*, 2(2):1–13, 2011. 5

[4] Xiaowen Cai, Yunbo Tao, Daizong Liu, Pan Zhou, Xiaoye Qu, Jianfeng Dong, Keke Tang, and Lichao Sun. Frequency-aware gan for imperceptible transfer attack on 3d point clouds. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6162–6171, 2024. 1

[5] Yunkang Cao, Xiaohao Xu, and Weiming Shen. Complementary pseudo multimodal feature for point cloud anomaly detection. *arXiv preprint arXiv:2303.13194*, 2023. 7

[6] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *European conference on computer vision*, pages 202–221. Springer, 2020. 2, 3, 6

[7] Jianbo Chen, Michael I Jordan, and Martin J Wainwright. Hopskipjumpattack: A query-efficient decision-based attack. In *2020 ieee symposium on security and privacy (sp)*, pages 1277–1294. IEEE, 2020. 2, 5, 8, 1

[8] Shaoyu Chen, Jiemin Fang, Qian Zhang, Wenyu Liu, and Xinggang Wang. Hierarchical aggregation for 3d instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15467–15476, 2021. 3

[9] Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. Language conditioned spatial relation reasoning for 3d object grounding. *Advances in Neural Information Processing Systems*, 35:20522–20535, 2022. 3, 6, 7, 8

[10] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1907–1915, 2017. 1

[11] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13142–13153, 2023. 2

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[13] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9185–9193, 2018. 1

[14] Yinpeng Dong, Jun Zhu, Xiao-Shan Gao, et al. Isometric 3d adversarial examples in the physical world. *Advances in Neural Information Processing Systems*, 35:19716–19731, 2022. 1, 3

[15] Richard O Duda and Peter E Hart. Use of the hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):11–15, 1972. 4

[16] Xiang Fang, Daizong Liu, Pan Zhou, and Yuchong Hu. Multi-modal cross-domain alignment network for video moment retrieval. *IEEE Transactions on Multimedia*, 25:7517–7532, 2022. 1

[17] Xiang Fang, Daizong Liu, Pan Zhou, and Guoshun Nan. You can ground earlier than see: An effective and efficient pipeline for temporal sentence grounding in compressed videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2448–2460, 2023.

[18] Xiang Fang, Daizong Liu, Pan Zhou, Zichuan Xu, and Ruixuan Li. Hierarchical local-global transformer for temporal sentence grounding. *IEEE Transactions on Multimedia*, 2023.

[19] Xiang Fang, Arvind Easwaran, Blaise Genest, and Ponnuthurai Nagaratnam Suganthan. Your data is not perfect: Towards cross-domain out-of-distribution detection in class-imbalanced data. *Expert Systems with Applications*, 2024.

[20] Xiang Fang, Daizong Liu, Wanlong Fang, Pan Zhou, Zichuan Xu, Wenzheng Xu, Junyang Chen, and Renfu Li. Fewer steps, better performance: Efficient cross-modal clip trimming for video moment retrieval using language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1735–1743, 2024.

[21] Xiang Fang, Wanlong Fang, Changshuo Wang, Daizong Liu, Keke Tang, Jianfeng Dong, Pan Zhou, and Beibei Li. Multi-pair temporal sentence grounding via multi-thread knowledge transfer network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025. 1

[22] Hao Feng, Qi Liu, Hao Liu, Tang Jingqun, Wengang Zhou, Houqiang Li, and Can Huang. Docpedia: Unleashing the power of large multimodal model in the frequency domain for versatile document understanding. *SCIENCE CHINA Information Sciences*, 2024. 1

[23] Mingtao Feng, Zhen Li, Qi Li, Liang Zhang, XiangDong Zhang, Guangming Zhu, Hui Zhang, Yaonan Wang, and Ajmal Mian. Free-form description guided 3d visual graph network for object grounding in point cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3722–3731, 2021. 3

[24] Ryan Feng, Neal Mangaokar, Jiefeng Chen, Earlence Fernandes, Somesh Jha, and Atul Prakash. Graphite: Generating automatic physical examples for machine-learning attacks on computer vision systems. In *2022 IEEE 7th European symposium on security and privacy (EuroS&P)*, pages 664–683. IEEE, 2022. 2, 8, 1

[25] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint*, 2014. 1

[26] Dailan He, Yusheng Zhao, Junyu Luo, Tianrui Hui, Shaofei Huang, Aixi Zhang, and Si Liu. Transrefer3d: Entity-and-relation aware transformer for fine-grained 3d visual grounding. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2344–2352, 2021. 3

[27] Paul VC Hough. Method and means for recognizing complex patterns, 1962. US Patent 3,069,654. 4

[28] Qianjiang Hu, Daizong Liu, and Wei Hu. Density-insensitive unsupervised domain adaption on 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17556–17566, 2023. 1

[29] Shiyu Hu, Daizong Liu, and Wei Hu. Improving the transferability of 3d point cloud attack via spectral-aware admix and optimization designs. *arXiv preprint arXiv:2412.12626*, 2024. 1

[30] Pin-Hao Huang, Han-Hung Lee, Hwann-Tzong Chen, and Tyng-Luh Liu. Text-guided graph neural networks for referring 3d instance segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1610–1618, 2021. 3

[31] Shijia Huang, Yilun Chen, Jiaya Jia, and Liwei Wang. Multi-view transformer for 3d visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15524–15533, 2022. 3

[32] Wencan Huang, Daizong Liu, and Wei Hu. Dense object grounding in 3d scenes. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 5017–5026, 2023.

[33] Wencan Huang, Daizong Liu, and Wei Hu. Advancing 3d object grounding beyond a single 3d scene. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7995–8004, 2024. 3

[34] Frederick James. Monte carlo theory and practice. *Reports on progress in Physics*, 43(9):1145, 1980. 5

[35] Baoxiong Jia, Yixin Chen, Huangyue Yu, Yan Wang, Xuesong Niu, Tengyu Liu, Qing Li, and Siyuan Huang. Sceneverse: Scaling 3d vision-language learning for grounded scene understanding. In *European Conference on Computer Vision*, pages 289–310. Springer, 2024. 2

[36] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup: Dual-set point grouping for 3d instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and Pattern recognition*, pages 4867–4876, 2020. 3

[37] Jaeyeon Kim, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Minimal adversarial examples for deep learning on 3d point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7797–7806, 2021. 1, 3

[38] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016. 1

[39] Kibok Lee, Zhuoyuan Chen, Xinchen Yan, Raquel Urtasun, and Ersin Yumer. Shapeadv: Generating shape-aware adversarial 3d point clouds. *arXiv preprint arXiv:2005.11626*, 2020. 1, 3

[40] Huichen Li, Xiaojun Xu, Xiaolu Zhang, Shuang Yang, and Bo Li. Qeba: Query-efficient boundary-based blackbox attack. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1221–1230, 2020. 2, 8, 1

[41] Daizong Liu and Wei Hu. Imperceptible transfer attack and defense on 3d point cloud classification. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4727–4746, 2022. 1

[42] Daizong Liu and Wei Hu. Explicitly perceiving and preserving the local geometric structures for 3d point cloud attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3576–3584, 2024. 3

[43] Daizong Liu and Wei Hu. Imperceptible backdoor attacks on text-guided 3d scene grounding. *IEEE Transactions on Multimedia*, 2025. 1

[44] Daniel Liu, Ronald Yu, and Hao Su. Extending adversarial attacks and defenses to deep 3d point cloud classifiers. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2279–2283, 2019. 1, 3

[45] Daizong Liu, Xi Ouyang, Shuangjie Xu, Pan Zhou, Kun He, and Shiping Wen. Saanet: Siamese action-units attention network for improving dynamic facial expression recognition. *Neurocomputing*, 413:145–157, 2020. 1

[46] Daizong Liu, Xiaoye Qu, Xiao-Yang Liu, Jianfeng Dong, Pan Zhou, and Zichuan Xu. Jointly cross-and self-modal graph attention network for query-based moment localization. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4070–4078, 2020.

[47] Daizong Liu, Xiaoye Qu, Jianfeng Dong, Pan Zhou, Yu Cheng, Wei Wei, Zichuan Xu, and Yulai Xie. Context-aware biaffine localizing network for temporal sentence grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11235–11244, 2021.

[48] Daizong Liu, Pan Zhou, Zichuan Xu, Haozhao Wang, and Ruixuan Li. Few-shot temporal sentence grounding via memory-guided semantic learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(5):2491–2505, 2022. 1

[49] Daizong Liu, Wei Hu, and Xin Li. Point cloud attacks in graph spectral domain: When 3d geometry meets graph signal processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):3079–3095, 2023. 1

[50] Daizong Liu, Wei Hu, and Xin Li. Robust geometry-dependent attack for 3d point clouds. *IEEE Transactions on Multimedia*, 26:2866–2877, 2023. 3

[51] Daizong Liu, Jiahao Zhu, Xiang Fang, Zeyu Xiong, Huan Wang, Renfu Li, and Pan Zhou. Conditional video diffusion network for fine-grained temporal sentence grounding. *IEEE Transactions on Multimedia*, 26:5461–5476, 2023. 3

[52] Daizong Liu, Yang Liu, Wencan Huang, and Wei Hu. A survey on text-guided 3d visual grounding: elements, recent advances, and future directions. *arXiv preprint arXiv:2406.05785*, 2024. 3

[53] Daizong Liu, Yunbo Tao, Pan Zhou, and Wei Hu. Hard-label black-box attacks on 3d point clouds. *arXiv preprint arXiv:2412.00404*, 2024. 1

[54] Daizong Liu, Mingyu Yang, Xiaoye Qu, Pan Zhou, Yu Cheng, and Wei Hu. A survey of attacks on large vision-language models: Resources, advances, and future trends. *arXiv preprint arXiv:2407.07403*, 2024. 1

[55] Daizong Liu, Mingyu Yang, Xiaoye Qu, Pan Zhou, Xiang Fang, Keke Tang, Yao Wan, and Lichao Sun. Pandora's box: Towards building universal attackers against real-world large vision-language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 5

[56] Haolin Liu, Anran Lin, Xiaoguang Han, Lei Yang, Yizhou Yu, and Shuguang Cui. Refer-it-in-rgbd: A bottom-up approach for 3d visual grounding in rgbd images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6032–6041, 2021. 3

[57] Jiaqi Liu, Guoyang Xie, Ruitao Chen, Xinpeng Li, Jinbao Wang, Yong Liu, Chengjie Wang, and Feng Zheng. Real3d-ad: A dataset of point cloud anomaly detection. *Advances in Neural Information Processing Systems*, 36, 2024. 7

[58] Yuliang Liu, Jiaxin Zhang, Dezhi Peng, Mingxin Huang, Xinyu Wang, Jingqun Tang, Can Huang, Dahua Lin, Chunhua Shen, Xiang Bai, et al. Spts v2: single-point scene text spotting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[59] Yang Liu, Daizong Liu, Zongming Guo, and Wei Hu. Cross-task knowledge transfer for semi-supervised joint 3d grounding and captioning. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3818–3827, 2024. 3

[60] Yang Liu, Daizong Liu, and Wei Hu. Joint top-down and bottom-up frameworks for 3d visual grounding. In *International Conference on Pattern Recognition*, pages 249–264. Springer, 2025. 3

[61] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2949–2958, 2021. 3

[62] Junyu Luo, Jiahui Fu, Xianghao Kong, Chen Gao, Haibing Ren, Hao Shen, Huaxia Xia, and Si Liu. 3d-sps: Single-stage 3d visual grounding via referred point progressive selection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16454–16463, 2022. 3, 6, 7, 8, 1

[63] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017. 1

[64] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1765–1773, 2017. 6

[65] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9277–9286, 2019. 3

[66] Junha Roh, Karthik Desingh, Ali Farhadi, and Dieter Fox. Languagerefer: Spatial-language model for 3d visual grounding. In *Conference on Robot Learning*, pages 1046–1056. PMLR, 2022. 3

[67] Zhenbo Shi, Zhi Chen, Zhenbo Xu, Wei Yang, Zhidong Yu, and Liusheng Huang. Shape prior guided attack: Sparser perturbations on 3d point clouds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8277–8285, 2022. 1, 3

[68] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 1

[69] Jingqun Tang, Wenming Qian, Luchuan Song, Xiena Dong, Lan Li, and Xiang Bai. Optimal boxes: boosting end-to-end scene text recognition by adjusting annotated bounding boxes via reinforcement learning. In *European Conference on Computer Vision*, pages 233–248. Springer, 2022. 1

[70] Jingqun Tang, Wenqing Zhang, Hongye Liu, MingKun Yang, Bo Jiang, Guanglong Hu, and Xiang Bai. Few could be better than all: Feature sampling and grouping for scene text detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4563–4572, 2022. 1

[71] Jingqun Tang, Weidong Du, Bin Wang, Wenyang Zhou, Shuqi Mei, Tao Xue, Xing Xu, and Hai Zhang. Character recognition competition for street view shop signs. *National Science Review*, 10(6):nwad141, 2023. 1

[72] Jingqun Tang, Chunhui Lin, Zhen Zhao, Shu Wei, Binghong Wu, Qi Liu, Hao Feng, Yang Li, Siqi Wang, Lei Liao, et al. Textsquare: Scaling up text-centric visual instruction tuning. *arXiv preprint arXiv:2404.12803*, 2024. 1

[73] Jingqun Tang, Qi Liu, Yongjie Ye, Jinghui Lu, Shu Wei, Chunhui Lin, Wanqing Li, Mohamad Fitri Faiz Bin Mahmood, Hao Feng, Zhen Zhao, Yanjie Wang, Yuliang Liu, Hao Liu, Xiang Bai, and Can Huang. Mtvqa: Benchmarking multilingual text-centric visual question answering, 2024. 1

[74] Keke Tang, Lujie Huang, Weilong Peng, Daizong Liu, Xiaofei Wang, Yang Ma, Ligang Liu, and Zhihong Tian. Flat: flux-aware imperceptible adversarial attacks on 3d point clouds. In *European Conference on Computer Vision*, pages 198–215. Springer, 2024. 3

[75] Guanhong Tao, Shengwei An, Siyuan Cheng, Guangyu Shen, and Xiangyu Zhang. Hard-label black-box universal adversarial patch attack. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 697–714, 2023. 5

[76] Yunbo Tao, Daizong Liu, Pan Zhou, Yulai Xie, Wei Du, and Wei Hu. 3dhacker: Spectrum-based decision boundary generation for hard-label 3d point cloud attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14340–14350, 2023. 3

[77] Tzungyu Tsai, Kaichen Yang, Tsung-Yi Ho, and Yier Jin. Robust adversarial objects against deep learning models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 954–962, 2020. 1, 3

[78] Chun-Chen Tu, Paishun Ting, Pin-Yu Chen, Sijia Liu, Huan Zhang, Jinfeng Yi, Cho-Jui Hsieh, and Shin-Ming Cheng.

Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 742–749, 2019. 1

[79] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1588–1597, 2019. 2, 4

[80] George Vosselman, Sander Dijkman, et al. 3d building model reconstruction from point clouds and ground plans. *International archives of photogrammetry remote sensing and spatial information sciences*, 34(3/W4):37–44, 2001. 5

[81] Thang Vu, Kookhoi Kim, Tung M Luu, Thanh Nguyen, and Chang D Yoo. Softgroup for 3d instance segmentation on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2708–2717, 2022. 3

[82] An-Lan Wang, Bin Shan, Wei Shi, Kun-Yu Lin, Xiang Fei, Guozhi Tang, Lei Liao, Jingqun Tang, Can Huang, and Wei-Shi Zheng. Pargo: Bridging vision-language with partial and global views, 2025. 1

[83] Yuxin Wen, Jiehong Lin, Ke Chen, CL Philip Chen, and Kui Jia. Geometry-aware generation of adversarial point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020. 1, 3

[84] Yanmin Wu, Xinhua Cheng, Renrui Zhang, Zesen Cheng, and Jian Zhang. Eda: Explicit text-decoupling and dense alignment for 3d visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19231–19242, 2023. 6, 7, 8, 1

[85] Chong Xiang, Charles R Qi, and Bo Li. Generating 3d adversarial point clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9136–9144, 2019. 3

[86] Zhen Xiang, David J Miller, Siheng Chen, Xi Li, and George Kesidis. A backdoor attack against 3d point cloud classifiers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7597–7607, 2021. 4

[87] Mingyu Yang, Daizong Liu, Keke Tang, Pan Zhou, Lixing Chen, and Junyang Chen. Hiding imperceptible noise in curvature-aware patches for 3d point cloud attack. In *European Conference on Computer Vision*, pages 431–448. Springer, 2024. 1

[88] Zhengyuan Yang, Songyang Zhang, Liwei Wang, and Jiebo Luo. Sat: 2d semantics assisted training for 3d visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1856–1866, 2021. 3, 6

[89] Zhihao Yuan, Xu Yan, Yinghong Liao, Ruimao Zhang, Sheng Wang, Zhen Li, and Shuguang Cui. Instancerefer: Cooperative holistic understanding for visual grounding on point clouds through instance multi-level contextual referring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1791–1800, 2021. 3

[90] Qiang Zhang, Jiancheng Yang, Rongyao Fang, Bingbing Ni, Jinxian Liu, and Qi Tian. Adversarial attack and defense on point sets. *arXiv preprint*, 2019. 3

[91] Yu Zhang, Gongbo Liang, Tawfiq Salem, and Nathan Jacobs. Defense-pointnet: Protecting pointnet against adversarial attacks. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 5654–5660, 2019. 1, 3

[92] Lichen Zhao, Daigang Cai, Lu Sheng, and Dong Xu. 3dvg-transformer: Relation modeling for visual grounding on point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2928–2937, 2021. 3, 6

[93] Weichao Zhao, Hao Feng, Qi Liu, Jingqun Tang, Shu Wei, Binghong Wu, Lei Liao, Yongjie Ye, Hao Liu, Wengang Zhou, Houqiang Li, and Can Huang. Tabpedia: Towards comprehensive visual table understanding with concept synergy, 2024. 1

[94] Yue Zhao, Yuwei Wu, Caihua Chen, and Andrew Lim. On isometry robustness of deep 3d point cloud models under adversarial attacks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1201–1210, 2020. 1, 3

[95] Zhen Zhao, Jingqun Tang, Chunhui Lin, Binghong Wu, Can Huang, Hao Liu, Xin Tan, Zhizhong Zhang, and Yuan Xie. Multi-modal in-context learning makes an ego-evolving scene text recognizer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15567–15576, 2024. 1

[96] Zhen Zhao, Jingqun Tang, Binghong Wu, Chunhui Lin, Shu Wei, Hao Liu, Xin Tan, Zhizhong Zhang, Can Huang, and Yuan Xie. Harmonizing visual text comprehension and generation. 2024. 1

[97] Hang Zhou, Dongdong Chen, Jing Liao, Kejiang Chen, Xiaoyi Dong, Kunlin Liu, Weiming Zhang, Gang Hua, and Nenghai Yu. Lg-gan: Label guided adversarial network for flexible targeted attack of point cloud based deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10356–10365, 2020. 1, 3

[98] Jiahao Zhu, Daizong Liu, Pan Zhou, Xing Di, Yu Cheng, Song Yang, Wenzheng Xu, Zichuan Xu, Yao Wan, Lichao Sun, et al. Rethinking the video sampling and reasoning strategies for temporal sentence grounding. *arXiv preprint arXiv:2301.00514*, 2023. 1