Multi-Source Adversarial Sample Attack on Autonomous Vehicles

Zuobin Xiong, Honghui Xu, Wei Li, *Member, IEEE,* and Zhipeng Cai, *Senior Member, IEEE*

***Abstract*—Deep learning has an impressive performance of object detection and classification for autonomous vehicles. Nevertheless, the essential vulnerability of deep learning models to adversarial samples makes the autonomous vehicles suffer severe security and safety issues. Although a number of works have been proposed to study adversarial samples, only a few of them are designated for the scenario of autonomous vehicles. Moreover, the state-of-the-art attack models only focus on a single data source without considering the correlation among multiple data sources. To fill this blank, we propose two multi-source adversarial sample attack models, including the parallel attack model and the fusion attack model to simultaneously attack the image and LiDAR perception systems in the autonomous vehicles. In the parallel attack model, adversarial samples are generated from the original image and LiDAR data separately. In the fusion attack model, the adversarial samples of image and LiDAR can be generated from a low-dimension vector at the same time by fully exploring data correlation for data fusion and adversarial sample generation. Through comprehensive real-data experiments, we validate that our proposed models are more powerful and efficient to break down the perception systems of autonomous vehicles compared with the state-of-the-art. Furthermore, we simulate possible attack scenarios in Vehicular Ad hoc Networks (VANETs) to evaluate the attack performance of our proposed methods.**

***Index Terms*—Adversarial Examples, Multi-source Data, Generative Adversarial Networks, Vehicular Ad hoc Networks**

I. Introduction

AUTONOMOUS driving techniques have been experiencing profound innovation and is gradually being applied to the automotive industry. According to [1], by the end of 2020, the market share of autonomous vehicles is going to be over 40%, which will bring a value of more than 61 billion dollars. This explosive progress is benefited from the sophisticated deep learning models: deep neural network structure and powerful computation capability. Even autonomous vehicles are so successful, they are still not perfect. As studied in prior research, deep neural networks, which are utilized in the perception systems of autonomous vehicles for object detection and classification, are vulnerable to adversarial samples that can mislead object detection and classification by slightly modifying the original data [2]. Once the autonomous vehicles are attacked by adversarial samples, the incorrect detection or classification will cause not only system security issues but also safety issues on the road (*e.g.*, traffic accidents), resulting in severe consequences to both the society and individuals [3], [4]. Therefore, *studying adversarial example*[1] *attack can help us understand perception vulnerabilities, develop defense strategies, and improve security and safety performance for autonomous vehicles*.

However, the existing works on adversarial examples have a major limitation: they only use single data source to generate adversarial samples and ignore the correlation among multiple data sources [5], [6], [7], [8], [9], [10], [11], [12]. Actually, correlated data sources can be exploited to perform accurate prediction, detection and classification; and thus only modifying one data source might not work if other related data sources are used as well. In the autonomous vehicles, object detection employs two main kinds of data, including image/video data captured by onboard cameras and LiDAR data collected by LiDAR sensors [13]. As shown in Fig. 1, the image data and LiDAR data are input into the perception systems and classified as different objects, such as vehicles and buildings; especially, in Fig. 1(c), the semantic segmentation is labeled for the image and yellow boxes are labeled for the LiDAR data. Since both the image and LiDAR data can localize vehicles, attacking either the image model or the LiDAR model is not enough to defeat the entire perception system, which is illustrated in our experiments in Section IV-D.

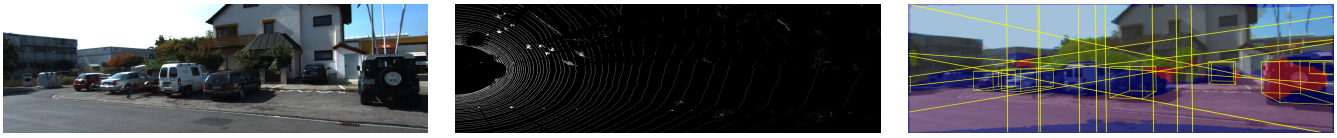
Motivated by the above analysis, in this paper, *we aim to develop multi-source adversarial sample attack models towards both the image and the LiDAR models such that neither the image-based nor the LiDAR-based perception model obtains correct detection on their inputs*. Unavoidably, developing the desired attack models is a challenging task. (i) In most of the existing works, adversarial samples are generated via optimization, which is time inefficient. Thus, a more efficient method is needed for the autonomous vehicles scenario. (ii) Multi-source adversarial sample attack has never been studied for autonomous vehicles and even any other applications. So, how to produce adversarial samples in such a multi-source scenario is a worth-thinking problem. (iii) Multi-source adversarial sample attack requires to successfully attack multiple data sources with imperceptible modification at the same time. Note that the image and LiDAR models can help each other to infer the same label (*e.g.,* vehicles). Thus, a qualified adversarial sample should be able to make the involved victim perception system predict incorrectly on the same label. (iv) Particularly, in autonomous vehicles, multiple

Zuobin Xiong, Honghui Xu, Wei Li, and Zhipeng Cai are with the Department of Computer Science, Georgia State University, Atlanta, GA, 30303, USA. E-mail: {*zxiong2, hxu16*}*@student.gsu.edu,* {*wli28, zcai*}*@gsu.edu*

Corresponding Author: Wei Li

Manuscript received XXX, XX, 2015; revised XXX, XX, 2015.

[1]In this paper, adversarial sample and adversarial example are interchangeable.



(a) Image data captured by camera (b) LiDAR data captured by LiDAR sensor (c) Joint detection of image and LiDAR data

Fig. 1: An example of multi-source perception systems in autonomous vehicles. In Fig. 1(c), the blue color segmentation is classified as detected vehicles by image perception model, and the yellow cubic boxes are detected as vehicles by LiDAR perception model.

data sources have different data formats. Before launching attack, we should preprocess these heterogeneous data sources properly for data fusion, which is a non-trivial problem. (v) Moreover, in the Vehicular Ad hoc Networks (VANETs), many factors have influences on the implementation of adversarial sample attack, which is missing in literature but is necessary for us to investigate the attack performance in real applications.

In order to tackle these challenges, we carry out a series of research activities. To enhance the time efficiency of attack implementation, we utilize the generative adversarial networks (GANs) [14] and variational autoencoder (VAE) [15] as the generators. To attack different multi-source systems, we analyze two mainstream industrial pipelines in the current perception systems of autonomous vehicles and then elaborately design two attack models, including **parallel attack model** and **fusion attack model**. Under the parallel attack model, adversarial samples are generated from the original image and LiDAR data separately. Under the fusion attack model, by employing the correlation between the image and LiDAR data, an adversarial image sample and an adversarial LiDAR sample can be generated from a same low-dimension vector simultaneously, which is a technical breakthrough in sample generation compared with the traditional attack models. In the experiments, the real datasets, the state-of-the-art, and the simulated Vehicle-to-Vehicle (V2V) communication are employed to investigate the performance of our proposed attack models. Our multi-fold contributions are summarized below.

- To the best of our knowledge, this paper is the first work to investigate multi-source adversarial example attack in literature, especially for the autonomous vehicles.
- Two different multi-source adversarial example attack models are proposed to simultaneously attack the image and LiDAR models, where the correlation between the image and the LiDAR data is utilized for data fusion and adversarial example generation.
- Extensive experiments are conducted to validate the superiority of our proposed attack models over the traditional single-source adversarial sample attack and evaluate the attack performance in V2V communication scenario.

The rest of this paper is organized as follows. Section II introduces the related works on adversarial examples. The details of our attack models are presented in Section III. After analyzing the experiment results in Section IV, the paper is concluded in Section V.

II. Related Works

A. Adversarial Examples

Adversarial examples have attracted lots of research interests to invade machine learning models since they were found by Szegedy *et al.* [16]. According to the prior knowledge hold by attackers, the adversarial sample attack can be categorized into two classes: white-box attack and black-box attack. Under the white-box attack, attackers have full knowledge about the parameters of target models; while in the black-box attack, they do not [5], [6]. The methods of generating adversarial samples include optimization-based method and generation-based method [7]. In the optimization-based methods, the problem of finding perturbation is formulated as an optimization problem and can be solved by different optimization solvers [2], [17], [18], [19]. In the generation-based methods, the typical process is training a neural network with specific loss functions such that the outputs of the network are adversarial samples [8], [9], [20], [21]. Although these existing attack models can work well on single-label datasets, their performance on multi-label datasets is unknown. What's more, these attack models might not be applicable in the street view data that has too many different labels.

B. Adversarial Examples in Autonomous Vehicles

Due to the adoption of deep neural network in the perception systems, autonomous vehicles are also vulnerable to adversarial example attack [22], [23]. As an important component of the perception system, street view image semantic segmentation has become a major attack target. Hendrik *et al.* [10] designed a universal noise to attack certain class segmentation based on gradient dependent optimization method. Fischer *et al.* [11] proposed a method to find adversarial samples for specific class of objects under white-box by changing parameters of classifier. Chen *et al.* presented a robust physical adversarial attack on Faster R-CNN object detector [12], which can find an optimal modification to add into a real-world images. *All of the above attack models use optimization-based methods and thus are not time efficient to be implemented for the autonomous vehicles.*

As another important sensory data of autonomous vehicles, the LiDAR data also has been studied. Cao *et al.* [24], [25] designed two physical methods to attack the LiDAR data so that the LiDAR-based system predicts incorrectly on its input data. In [26], adversarial samples were produced based on LiDAR range images with traditional image processing methods. *But there is no work on the digital adversarial samples for the LiDAR 3D points cloud data.*

C. Adversarial Attack Implementation

Currently, there are two mainstream methods of implementing adversarial sample attack towards machine learning models: digital implementation and physical implementation. Digital implementation has the advantages of lightweight, easy-implementation and high efficiency, which commonly assumes that attackers can access the target models and feed the digital data directly into them. Various digital adversarial sample attacks have been designed for many applications, including image classifier [16], [18], [19], object detection/semantic segmentation [11], [27], [28], and reinforcement learning [29].

Physical adversarial sample attack has been investigated in engineering field. By printing out crafted adversarial examples and re-taking them using physical devices [17], [30], [31], [24], [25], the physical adversarial attack seems more realistically applicable. But, during the execution of physical attack, there are a lot of uncertain factors hindering the performance of adversarial examples, such as limited training samples, changing environment, resolution changes and angles of physical devices. Those factors require expensive manual efforts to control and conduct physical attack experiments, which is hardly possible in real-time scenarios.

Compared with the state-of-the-art, the most significant advance of this paper is that both the image and LiDAR data are employed to perform digital adversarial example attack towards the perception systems in autonomous vehicles. *Such a type of multi-source adversarial example attack has more power to destroy the perception systems and lead to more serious consequences.*

III. Methodology

In this paper, we focus on digital adversarial sample attack where attackers can launch attacks with prior knowledge such as sufficient training dataset, black-box of pre-trained target models, and others. In our proposed attack models, we assume attackers have enough power to obtain prior knowledge and inject the generated adversarial samples into the models that are equivalent to the perception models of victim autonomous vehicles. For example, a driver of autonomous vehicles could attack other vehicles. The consideration of this assumption comes from two important facts: (i) it is hard or impractical for users or defenders to estimate an attacker's actual power/capability; and (ii) with the presence of a powerful attacker, we can deeply understand the vulnerability of perception systems in autonomous vehicles and then help design strong defense strategies.

A. Adversarial Scenarios

Cooperative data sharing and communication have been treated as promising solutions helping autonomous vehicles obtain more comprehensive views and improve detection performance [32]. The communication capacity can be supported by Vehicular Ad hoc Networks (VANETs) [33], [34], where autonomous vehicles can connect with each other and Road Side Infrastructure (RSU) based on wireless local network technology. Vehicular communications in VANETs contain three types, including Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), and Infrastructure-to-Vehicle (I2V), which benefit self-driving a lot but also face security issues.

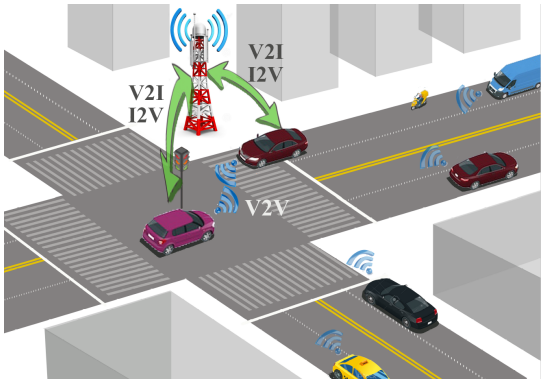


Fig. 2: An example of vehicular communications in VANETs.

A possible adversarial scenario is shown in Fig. 2, where a malicious RSU and vehicles can attack on the perception systems of victim vehicles by sending manipulated data via vehicular communications. In VANETs, RSU has enough computational power to perform complex calculation and sufficient space to store vehicular sensory data (*e.g.* camera data and LiDAR data) uploaded from nearby vehicles through V2I communication. Those data can be used as training data by the malicious RSU to extract prior knowledge. Then, by utilizing adversarial sample generation methods, the malicious RSU can inject the paired adversarial examples that consist of image and LiDAR data into target autonomous vehicles through I2V communication. On the other hand, malicious vehicles can launch adversarial sample attack towards their nearby victim vehicles by injecting adversarial samples through V2V communication in a data-sharing manner.

Notably, compared with the V2I/I2V attack scenario, the V2V attack scenario is more practical in real applications. (i) Any autonomous vehicle in VANETs can act as an attacker; in contrast, deploying a malicious RSU might not be an easy job. (ii) The malicious vehicles do not require any training data from the victim vehicles, because they already have enough sensory data for training inside their perception systems. (iii) The malicious vehicles can generate adversarial samples either with or without the original data from the victim vehicles. Since the malicious vehicles could be close to the victim vehicles within the V2V communication range, the captured data of the malicious and victim vehicles has a certain (high) similarity. Thus, the malicious vehicles can generate adversarial samples by using their own sensory data locally.

B. Problem Statement

Let $f(I)$ and $g(F)$ be the image semantic segmentation model and the LiDAR detection model, where I and F are image and LiDAR data of street view, respectively. In this paper, the objects in street view, including image and LiDAR data, are classified into two categories: (1) *target class* that are vehicles, denoted by y_t, and (2) *non-target class* that are the objects except for vehicles. In the semantic segmentation

TABLE I: Notations & Definitions

Notation	Definition
I	real image data
I'	generated image data $I' = G_I(I)$
F	real LiDAR data
F'	generated LiDAR data $F' = G_F(F)$
G_I	adversarial sample generator for image data
D_I	discriminator for image data
G_F	adversarial sample generator for LiDAR data
D_F	discriminator for LiDAR data
f	image data detection model
g	LiDAR data detection model
$\mathcal{L}^I_{GAN}$	GAN loss for image data
$\mathcal{L}^F_{GAN}$	GAN loss for LiDAR data
$\mathcal{L}^I_{VAE}$	VAE loss for image data
$\mathcal{L}^F_{VAE}$	VAE loss for LiDAR data
$\mathcal{L}_P$	p-norm loss for image data
$\mathcal{L}_Z$	latent feature loss for both image and LiDAR data
$\mathcal{L}_{ADV}$	adversarial loss
$\mathcal{L}_{Image}$	entire loss for image data
$\mathcal{L}_{LiDAR}$	entire loss for LiDAR

model $f(I)$ with target class y_t, $I_{y_t} = \{(i,j)|f(I) = y_t\}$ represents the pixels corresponding to vehicles, and the remaining part of an image is denoted by $I_{bg} = I - I_{y_t}$. In the LiDAR model $g(F)$, the targeted class "vehicles" y_t is indicated by the area bounded within yellow cubic boxes as shown in Fig. 1(c).

The goal of this paper is to attack the image and LiDAR perception models of autonomous vehicles by modifying both the image and LiDAR data in a black-box setting such that the victim vehicles mis-classify the target class into the non-target class while classifying the non-target class correctly. That is, except the vehicles, the victims are expected to classify other objects correctly. The assumption of black-box setting has been widely adopted by prior works [25], [12]. Formally, our proposed problem can be defined as follows.

Definition 1. (Problem of Multi-Source Adversarial Example for Autonomous Vehicles) *Given a semantic segmentation model $f(I)$ with image data I and a LiDAR detection model $g(F)$ with LiDAR data F, the corresponding adversarial samples, I' and F', are outputs with the following properties:*

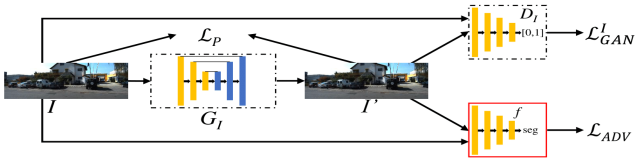
(1) for I', $\arg\min_{I'} \|I - I'\|_p$, s.t. (i) $\forall$ pixel $\in I'_{y_t}, f(I'_{y_t}) \neq y_t$, and (ii) $\forall$ pixel $\in I'_{bg}, f(I'_{bg}) = f(I_{bg})$;

(2) for F', $\arg\min_{F'} \|F - F'\|_p$, s.t. $f(F') \neq y_t$;

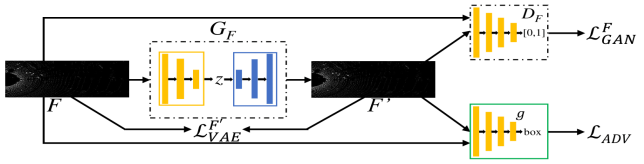
where $\|\cdot\|_p$ is the p-norm.

For our problem, it is unbearably inefficient to implement the (existing) optimization-based methods on the high dimensional complex street view data. Thus, to speed up the process of generating adversarial samples in the autonomous vehicle scenario, we utilize a generation-based approach that is more efficient than the optimization-based method once the generative model has been trained.

On the other hand, since one single sensor cannot fulfill the requirement of reliable object detection in complex urban environments, multi-sensor fusing approaches have been widely adopted in autonomous vehicles [35]. Currently, there are two mainstream techniques used in the perception systems of autonomous vehicles for data fusion: low-level fusion (LLF) and high-level fusion (HLF) [36]. Both two fusion



(a) Image pipeline of parallel attack model



(b) LiDAR pipeline of parallel attack model

Fig. 3: Framework of our parallel attack model.

methods are dedicated to handle the multi-sensor data from camera, LiDAR, and radar, *etc.* for improving the perception performance of autonomous vehicles. Particularly, the LLF-style method performs multiple detection pipelines on multiple heterogeneous sensory data separately and then integrates the result of each pipeline, where the sensory data usually have different characteristics that need to be considered for effective result combination. While the HLF-style method aims to find a unified representation for multi-sensory data regardless of sensor type, which should contain essential information of heterogeneous sensory data, such as shape, position, and semantic information, and so on. In the HLF-style method proposed in [36], 3D LiDAR data and 2D image data are preprocessed and transferred into an 2D feature space through projection and calibration.

To achieve our goal, we propose a **parallel attack model** to attack the LLF-style perception systems and a **fusion attack model** to attack the HLF-style perception systems. The main notations are summarized in Table I, and the details of our attack models are described in Section III-C and Section III-D.

C. Parallel Attack Model

In our problem, the objective is to generate adversarial samples that have a distribution similar to the original data subject to the requirement of incorrect classification on the target class objects but correct classification on the non-target class objects. For the autonomous vehicles that adopt the LLF-style perception systems with camera sensors and LiDAR sensors, we design a parallel attack model, in which we attack the image model $f(\cdot)$ and the LiDAR model $g(\cdot)$ by using two separate pipelines to mislead their detection as shown in Fig. 3. The pseudo code of sample generation algorithm of our parallel attack is presented in Algorithm 1. The design is detailed as follows.

1) Image Data Pipeline: As shown in Fig. 3(a), three neural networks are deployed in the image data pipeline, including a generator G_I, a discriminator D_I, and a semantic segmentation model $f(\cdot)$. Here $f(\cdot)$ is the victim image model we intend to attack, and its output for an image I is $f(I) = y$, where y has the same shape as I, and each pixel of y is the class probability of this pixel. The generator G_I takes real image data I as input to produce an adversarial sample $I' = G_I(I)$. For I', we also have a predicted label $y' = f(I')$ for each

Algorithm 1: Sample Generation Algorithm of Parallel Attack Model

Input: training iteration T, discriminator's training steps k, minibatch size m, and real data I (*resp.* F)
Output: adversarial sample generator G_I^* (*resp.* G_L^*)

for $t < T$ **do**
  for k *steps* **do**
    Input m samples of I (*resp.* F) into G_I (*resp.* G_F) to generate I' (*resp.* F');
    Feed I' (*resp.* F') and other m samples of I (*resp.* F) into D_I (*resp.* D_F) and update D_I (*resp.* D_F) to maximize $\mathcal{L}_{GAN}^I$ (*resp.* $\mathcal{L}_{GAN}^F$) by gradient: $\nabla_{D_I}\mathcal{L}_{GAN}^I$ (*resp.* $\nabla_{D_F}\mathcal{L}_{GAN}^F$);
  end
  Select m samples of I (*resp.* F);
  Use G_I (*resp.* G_F) to generate I' (*resp.* F') from the selected m samples;
  Feed the selected m samples and I' (*resp.* F') into D_I and f (*resp.* D_F and g) for updating G_I (*resp.* G_F) to minimize $\mathcal{L}_{Image}$ (*resp.* $\mathcal{L}_{LiDAR}$) by gradient: $\nabla_{G_I}\mathcal{L}_{Image}$ (*resp.* $\nabla_{G_F}\mathcal{L}_{LiDAR}$).
end

pixel. Then both I and I' are fed to discriminator D_I and model $f(\cdot)$ for training.

For the requirement of generating adversarial samples, we aim to minimize the loss function in Eq. (1).

$$\mathcal{L}_{ADV} = \mathbb{E}_{I,I'}[L(f(I_{bg}), f(I'_{bg})) - L(f(I_{y_t}), f(I'_{y_t}))], \quad (1)$$

where $L(\cdot)$ is the cross entropy of the distribution of prediction on the original data I and the distribution of prediction on the generated data I'. During the training process, a smaller $L(f(I_{bg}), f(I'_{bg}))$ indicates a higher similarity between I_{bg} and I'_{bg} for imperceptible modification on the non-target classes, while a bigger $L(f(I_{y_t}), f(I'_{y_t}))$ means a larger difference between I_{y_t} and I'_{y_t} for adversarial attack on the target class. When $L(f(I_{y_t}), f(I'_{y_t}))$ becomes large enough, I_{y_t} and I'_{y_t} will be classified differently, resulting in a successful attack.

To ensure the desired attack ability while hiding the attack behavior, the output image I' of G_I is expected to have a different prediction on the target class y_t but be close to the original image I. That is, I and I' should be different at the pixels corresponding to the target class but have almost the same underlying image structure. For this purpose, we adopt the "U-Net" based structure whose capability to process complex image data has been confirmed in previous research [37]. In "U-Net", an encoder is used to compress the input image, and a decoder is used to recover the output image from middle hidden layer, where the hidden layer preserves the common underlying structure of the input and output. Also, there exist many effective skip links between the ith layer of encoder and the $(n-i)$th layer of decoder, which performs copy and corp operations to compel the output image to retain more information of the input.

Then, the discriminator D_I decides whether its input data is real or fake with the following loss function:

$$\mathcal{L}_{GAN}^I = \mathbb{E}_I\ [\log D_I(I)] + \mathbb{E}_{I'}\ [\log(1 - D_I(I'))]. \quad (2)$$

This loss function is used to guarantee the generated images are as realistic as possible. In order to maximize Eq. (2), D_I is encouraged to assign a large value to the real data I and a small value to the generated adversarial sample I'.

Besides, one more loss function is needed to control the magnitude of modification on the original data. To quantify the distance between the real data and the generated data, a measurement is defined in Eq. (3).

$$\mathcal{L}_P = \mathbb{E}_{I,I'}\|I - I'\|_p. \quad (3)$$

In this paper, we use the L_2 norm distance because the performance of L_2 is better than that of others.

To sum up, we fulfill the adversarial sample attack by minimizing Eq. (1), make the adversarial sample realistic by maximizing Eq. (2), and meet the need of imperceptible modification by minimizing Eq. (3). For our image generator G_I, the overall loss function can be combined as

$$\mathcal{L}_{Image} = \mathcal{L}_{GAN}^I + \lambda_1\mathcal{L}_{ADV} + \lambda_2\mathcal{L}_P, \quad (4)$$

where λ_1 and λ_2 are hyper-parameters to control the loss scale of $\mathcal{L}_{ADV}$ and $\mathcal{L}_P$, respectively. After the training process is terminated, we can obtain an optimal image generator, *i.e.*,

$$G_I^* = \arg\min_{G_I}\max_{D_I}[\mathcal{L}_{GAN}^I + \lambda_1\mathcal{L}_{ADV} + \lambda_2\mathcal{L}_P], \quad (5)$$

which is able to generate adversarial samples from any original data efficiently. With G_I^*, attackers can produce adversarial samples $I' = G_I^*(I)$ to invade the image model $f(\cdot)$ in the perception systems on autonomous vehicles.

2) LiDAR Data Pipeline: From Fig. 1, one can observe that attacking the image model only is not enough to invade the entire perception system of autonomous vehicles. So, we also need to compromise the LiDAR sensors simultaneously. The LiDAR data is represented by the format of 3D points cloud, which can not be directly used in deep learning model. Instead, the 3D points cloud data need to be transferred to 3D voxels for further analysis. VoxelNet is one of the most popular methods of detecting objects through 3D voxels, in which the 3D voxels are transformed into the unified feature representations through voxel feature encoding layer [38]. These condensed feature representations, F, are used as input features for vehicle detection from LiDAR data.

In order to attack the LiDAR model, we should find adversarial feature representations, F', that can mislead detection results on vehicles but is close to F. Similar to the design of the image data pipeline, the LiDAR data pipeline also has three networks: a VAE-based generator G_F, a discriminator D_F, and a LiDAR model $g(\cdot)$. As depicted in Fig. 3(b), the generator G_F takes the real LiDAR data F as input and produces adversarial sample $F' = G_F(F)$. Then both F and F' are fed to the discriminator D_F and the model $g(\cdot)$ for training.

The VAE-based generator G_F is built from an encoder Enc and a decoder Dec with a loss function in Eq. (6).

$$\mathcal{L}_{VAE}^{F'} = -\mathbb{E}_{z\sim q(z|F)}[\log p(F'|z)] + KL(q(z|F)\|p(z)), \quad (6)$$

where z is a predefined low-dimensional vector and $KL(\cdot\|\cdot)$ is the Kullback-Leibler divergence between $q(z|F)$ and $p(z)$. To improve the quality of the generated feature F', we integrate the VAE-based generator G_F and the discriminator D_F in adversarial training, which can induce F' to be more similar to F; that is,

$$\mathcal{L}_{GAN}^F = \mathbb{E}_F\ [\log D_F(F)] + \mathbb{E}_{F'}\ [\log(1 - D_F(F'))]. \quad (7)$$

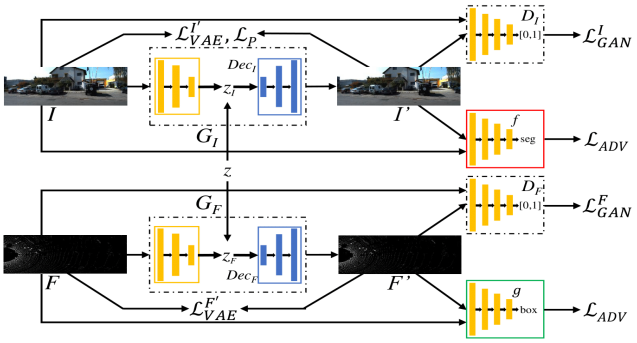


Fig. 4: Framework of our fusion attack model.

With Eq. (6) and Eq. (7), we can produce realistic feature representation F'.

In the VoxelNet model of LiDAR data, the only output label is the location of vehicles, which is indicated by the yellow boxes as shown in Fig. 1(c). Thus, attacking the LiDAR model is equivalent to reducing the accuracy of detecting the yellow boxes. In this paper, VoxelNet is adopted, where the yellow box of each vehicle can be drew by a 7-tuple (x, y, z, h, w, l, r) in the output. Specifically, (x, y, z) is the 3D position of a LiDAR sensor; h , w and l are height, weight and length of a vehicle in the 3D space, respectively; and r is the rotation angle with y-coordinate. Let Box_o and Box_a be the detected boxes in F and F', respectively. For each detected vehicle, the overlapping space of Box_o and its corresponding Box_a is denoted by Box_{ol}. The utility of LiDAR data is calculated by (Box_{ol}/Box_o) to measure the similarity between the adversarial sample and the original LiDAR data, and the attack performance is calculated by $(1 - Box_{ol}/Box_a)$ to depict the impact of modification on vehicle detection, where $Box_o = h \times w \times l$, $Box_a = h' \times w' \times l'$, and $Box_{ol} = ol_h \times ol_w \times ol_l \times \cos(r' - r)$ with $ol_h = (\min\{h', h\} - (z' - z))$, $ol_w = (\min\{w', w\} - (y' - y))$, and $ol_l = (\min\{l', l\} - (x' - x))$. Then, minimizing the loss function in Eq. (8) is to improve the attack performance for adversarial sample.

$$\mathcal{L}_{ADV} = \mathbb{E}_{F,F'}[Box_{ol}/Box_a - Box_{ol}/Box_o], \tag{8}$$

where (Box_{ol}/Box_a) is used to maximize the attack performance, and (Box_{ol}/Box_o) is used to make the detecting yellow boxes of generated feature F' imperceptible. Finally, the objective function of LiDAR data pipeline can be expressed in Eq. (9).

$$\mathcal{L}_{LiDAR} = \mathcal{L}_{GAN}^{F} + \eta_1 \mathcal{L}_{VAE}^{F'} + \eta_2 \mathcal{L}_{ADV}, \tag{9}$$

where η_1 and η_2 are the scaling parameters. When the training process is terminated, we can feed the LiDAR data F into the generator G_F and produce the adversarial sample F'.

D. Fusion Attack Model

For the autonomous vehicles that use the HLF-style perception systems, we propose a fusion attack model to integrate the two data pipelines together as shown in Fig. 4. This idea is inspired by a fact: for a pair of image data and LiDAR data that captures the same scene from the same location, their basic information embedded in the latent space should be the same. Because the essence of these two types of data contains the same information, we can use a same latent vector to represent the information and then recover the image data and the LiDAR data from the same latent vector.

Algorithm 2: Sample Generation Algorithm of Fusion Attack Model

Input: training iteration T, discriminator's training steps k, minibatch size m, and real data I and F
Output: adversarial sample generators Dec_I^* and Dec_F^*

for $t < T$ **do**
  for k *steps* **do**
    Input m samples of I and m samples of F into G_I and G_F to generate I' and F', respectively;
    Sample another m real image I and real LiDAR F;
    Feed I' and other m samples of I into D_I, and update D_I to maximize $\mathcal{L}_{GAN}^{I}$ by ascending their gradients:
      $\nabla_{D_I} \mathcal{L}_{GAN}^{I}$;
    Feed F' and other m samples of F into D_F, and update D_F to maximize $\mathcal{L}_{GAN}^{F}$ by ascending their gradients:
      $\nabla_{D_F} \mathcal{L}_{GAN}^{F}$;
  end
  Select m samples of I and m samples of F;
  Use G_I to generate z_I and I' from the selected m samples of I;
  Use G_F to generate z_F and F' from the selected m samples of F;
  Calculate $\mathcal{L}_Z$ using z_I and z_F;
  Feed I' and the selected m samples of I into D_I and f, feed F' and the selected m samples of F into D_F and g, and update G_I and G_F to minimize $[\mathcal{L}_{Image} + \mathcal{L}_Z]$ and $[\mathcal{L}_{LiDAR} + \mathcal{L}_Z]$ by descending their gradients, respectively:
    $\nabla_{G_I}[\mathcal{L}_{Image} + \mathcal{L}_Z]$ and $\nabla_{G_F}[\mathcal{L}_{LiDAR} + \mathcal{L}_Z]$.
end

In the fusion attack model, to jointly attack the image model and the LiDAR model, we adopt the VAE-GAN model [39] and modify the image data pipeline of the parallel attack model as shown in Fig. 4. Both the image generator G_I and LiDAR generator G_F are constructed by an encoder and a decoder, in which the encoder maps the original data I and F to low dimension representations z_I and z_F, respectively. Then the decoder Dec_I and Dec_F recover z_I and z_F back to adversarial samples I' and F', respectively. The discriminators and target models implement the same operations as those in the parallel attack model. Especially, the encoded low dimensional vectors z_I and z_F should be restricted by elaborately designed loss functions. The pseudo code of sample generation algorithm of our fusion attack is presented in Algorithm 2.

To make the image generator G_I adapt to the fusion attack model, the structure and loss function of G_I are rewritten as a VAE-based formula:

$$\mathcal{L}_{VAE}^{I'} = -\mathbb{E}_{z \sim q(z|I)}[\log(I'|z)] + KL(q(z|I)\|p(z))), \tag{10}$$

where z is the predefined latent distribution $\mathcal{N}(0, 1)$.

For the generated data I', we still use a discriminator D_I to improve the data quality through maximizing Eq. (2). The computation of $\mathcal{L}_{ADV}$ and $\mathcal{L}_P$ in the fusion attack model is the same as those in the parallel attack model. Therefore, the loss function of the image model in the fusion attack model is expressed as,

$$\mathcal{L}_{Image} = \mathcal{L}_{GAN}^{I} + \mathcal{L}_{VAE}^{I'} + \lambda_1 \mathcal{L}_{ADV} + \lambda_2 \mathcal{L}_P. \tag{11}$$

Then, to bring a tie between image pipeline and LiDAR pipeline and compel these two pipelines to encode two types of

TABLE II: Network architecture of image pipeline in our parallel attack model

Layer	Encoder	Decoder	Discriminator
1	$5 \times 5 \times 64$ conv, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$5 \times 5 \times 64$ conv, Leaky ReLU
2	$5 \times 5 \times 128$ conv, B_N, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$5 \times 5 \times 128$ conv, B_N, Leaky ReLU
3	$5 \times 5 \times 256$ conv, B_N, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$5 \times 5 \times 256$ conv, B_N, Leaky ReLU
4	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU
5	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 256$ deconv, B_N, ReLU	Fully Connected, Sigmoid
6	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 128$ deconv, B_N, ReLU	
7	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 64$ deconv, B_N, ReLU	
8	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 3$ deconv, tanh	

TABLE III: Network architecture of LiDAR pipeline in two attack models

Layer	Encoder	Decoder	Discriminator
1	FC input dimension $\times 512$, Leaky ReLU	FC 128×256, ReLU	Block1: $(1 \times 1 \times 64$ conv) $\times 3$
2	FC 512×256, Leaky ReLU	FC 256×512, ReLU	Block2: $(1 \times 1 \times 128$ conv) $\times 4$
3_μ	FC 256×128, Sigmoid	FC $512\times$ output dimension, Sigmoid	Block3: $(1 \times 1 \times 256$ conv) $\times 6$
3_σ	FC 256×128, Sigmoid		Fully Connected, Sigmoid

TABLE IV: Network architecture of image pipeline in our fusion attack model

Layer	Encoder	Decoder	Discriminator
1	$5 \times 5 \times 64$ conv, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$5 \times 5 \times 64$ conv, Leaky ReLU
2	$5 \times 5 \times 128$ conv, B_N, Leaky ReLU	$5 \times 5 \times 512$ deconv, B_N, ReLU	$5 \times 5 \times 128$ conv, B_N, Leaky ReLU
3	$5 \times 5 \times 256$ conv, B_N, Leaky ReLU	$5 \times 5 \times 256$ deconv, B_N, ReLU	$5 \times 5 \times 256$ conv, B_N, Leaky ReLU
4	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU	$5 \times 5 \times 128$ deconv, B_N, ReLU	$5 \times 5 \times 512$ conv, B_N, Leaky ReLU
5	FC 1024×512, Leaky ReLU	$5 \times 5 \times 64$ deconv, B_N, ReLU	Fully Connected, Sigmoid
6_μ	FC 512×128, Sigmoid	$5 \times 5 \times 3$ deconv, tanh	
6_σ	FC 512×128, Sigmoid		

data into the same point of latent space, the encoders' results z_I and z_F should be as close as possible. This enlightens us to define the latent content loss, denoted by $\mathcal{L}_Z$, between image data and LiDAR data:

$$\mathcal{L}_Z = \mathbb{E}_z \|z_I - z_F\|_2.$$

Through minimizing $\mathcal{L}_Z$, we can force the low dimension representation z_I and z_F to be similar.

The LiDAR model of the fusion attack model is the same as that of the parallel attack model, because it already uses VAE as the generator. During the training process, the image pipeline and LiDAR pipeline optimize their loss function $\mathcal{L}_{Image} + \mathcal{L}_Z$ and $\mathcal{L}_{LiDAR} + \mathcal{L}_Z$ in turns, respectively. After both models are trained, we can take the decoder, Dec_I, from the image model and the decoder, Dec_F, from the LiDAR model as the adversarial sample generators. Then, two adversarial samples are generated with the same latent low-dimension vector z, *i.e.*, $I' = Dec_I(z)$ and $F' = Dec_F(z)$.

E. Network Architecture of Attack Models

In this part, we illustrate the network architecture of parallel attack model and fusion attack model.

For the parallel attack model, the networks of image pipeline and LiDAR pipeline are presented in TABLE II and TABLE III, respectively. In TABLE II, the structures of "U-Net"-based generator and discriminator are constructed according to the setting in [40], where conv/deconv represents convolution/deconvolution, and and B_N represents batch normalization [41]. In the encoder of generator, there are 8 fully convolutional layers with a filter size of 5×5. From the 2nd to the 8th layers, each layer has a Leaky ReLU and Batch Normalization except. The structure of decoder is the opposite of the encoder except for ReLU and the $tanh$ activation of the 8th layer. In the discriminator, the filter size is also 5×5, the layer structure is a traditional CNNs with 4 convolutional layers, and it is ended with fully connected layer.

TABLE IV presents the network structure of image processing in our fusion attack model, which is a VAE-based structure mainly built from convolution layers. To integrate with VAE-based LiDAR pipeline shown in TABLE III, the outputs of encoders in image pipeline and LiDAR pipeline should have the same size. We use two fully connected layers to map the output of encoder to 128 dimension vector so that the image pipeline and LiDAR pipeline can use the same latent vector z to perform attack.

F. Comparison of Proposed Attack Models

The major differences between our two attack models lie in the following three aspects.

- **Model Structure.** The parallel attack model contains two separate pipelines designed for the LLF-style perception systems, and the fusion attack model has an integrated structure designed for the HLF-style systems.
- **Sample Generation.** In the parallel attack model, the adversarial samples are generated based on the original data directly; while in the fusion attack model, two types of adversarial samples are generated from a same low-dimensional latent vector z that contains less compressed information than the original data. As a result, the adversarial samples of the parallel attack model may have a

better visualization quality, and the adversarial samples of the fusion attack model may have more powerful attack performance on semantic segmentation and object detection but lower generation quality.

- **Time Efficiency.** Different from most current works that use optimization-based methods to generate adversarial samples, both the parallel attack model and the fusion attack model utilize generation-based approach to produce adversarial samples, which can significantly reduce processing time at sample generation stage. Particularly, our fusion attack model produces adversarial samples from a low-dimensional latent vector instead of real data, and thus the size of its parameters and inputs can be reduced, further saving processing time.

IV. Performance Evaluation

In this section, intensive experiments are set up to validate the effectiveness of our parallel and fusion attack approaches, compare with the state-of-the art, and investigate the attack performance in V2V communication.

A. Experiment Settings

Datasets. To evaluate the power of our attack models, we implement the parallel attack model and the fusion attack model on KITTI dataset [42] that is a benchmark dataset containing various kinds of sensory data and their labels for performance evaluation of autonomous vehicles. In the experiments, 1600 pairs of images and LiDAR data are used in our proposed models for training, and 400 pairs are used for testing.

Configurations. In the experiments, the device used for algorithm implementation is a Linux server with Intel(R) Xeon CPU E5-1607, 16 GB memory, and the NVIDIA GeForce RTX 2080 GPU with 11 GB memory, and the machine learning library Pytorch and OpenCV are adopted.

Experiment Steps. There are five steps in our experiments.

- We show that our generated adversarial samples can attack the semantic segmentation model $f(\cdot)$ and the LiDAR model $g(\cdot)$ separately under the single-source scenario.
- To exam the limitation of single-source adversarial sample attack, we provide two comparative experiments: (i) when only the image model $f(\cdot)$ is attacked, the LiDAR model $g(\cdot)$ can still detect vehicles for the whole system, called *LiDAR-assisted detection*; (ii) when only the LiDAR model $g(\cdot)$ is attacked, the image model $f(\cdot)$ can detect vehicles normally, called *Image-assisted detection*.
- We launch attacks towards both the image model $f(\cdot)$ and the LiDAR model $g(\cdot)$ under the multi-source scenario.
- A comparison is conducted to study the advantages of our attack models over the state-of-the-art in terms of attack effectiveness and attack efficiency.
- We simulate a V2V attack scenario to check the attack performance of our proposed methods with respect to communication quality in VANETs.

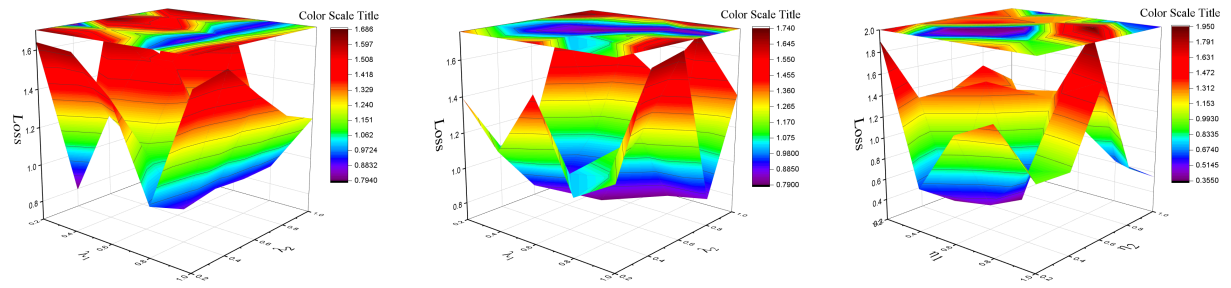
Performance Metrics. Our results are comprehensively analyzed via performance statistics, semantic segmentation results for the image model and object detection results for the LiDAR model. The performance statistics include the average utility, the average accuracy, the average image processing time and their corresponding variances.

- For the image data I and the image model $f(\cdot)$, the pixel accuracy of FCN-score [43] is used as the metric, where a high score implies an accurate detection. The utility of image is represented by the pixel accuracy for all non-target classes in I_{bg}, *i.e.*, the higher utility the better image generation performance. While, a lower pixel accuracy of target class "vehicle" indicates a higher attack success rate.
- For the LiDAR data F and the LiDAR model $g(\cdot)$, the computation of accuracy and utility is different from their computation in image data because "vehicle" is the only label in LiDAR data. The utility of LiDAR is defined by the ratio of the overlapping space between an original detected box and its corresponding adversarial detected box to the space of the original detected box, which measures the similarity between the original and the adversarial data samples, *i.e.*, the larger utility the higher similarity. The accuracy of LiDAR on vehicles is the ratio of the overlapping space between a ground truth box and its corresponding adversarial detected box to the space of the ground truth box. Accordingly, a smaller accuracy implies a larger attack success rate.

Performance Baselines. To our best knowledge, many existing adversarial example attacks on image segmentation use optimization-based methods [11], [10], which achieves a good performance for complex street view dataset but takes much more implementation time for autonomous vehicles. Thus, for a comparable evaluation on attack quality and efficiency, we take one optimization-based method in [11], termed Iterative Gradient Sign Optimization (IGSO), and one generation-based adversarial sample attack of [44], termed Generative Adversarial Perturbations (GAP), as the performance baselines. On the other hand, since no digital adversarial sample attack has been studied for the LiDAR data, no comparable attack model can be adopted for comparison, which, indeed, does not affect the validation of our attack models.

B. Hyper-parameter Analysis

The impacts of hyper-parameters on our two attack models are shown in Fig. 5. In Fig. 5(a), we change the values of λ_1 and λ_2 of image pipeline in parallel attack model from 0 to 1 through grid search method and plot the corresponding loss $\mathcal{L}_{Image}$ in Eq. (4) in a 3D view, where the 3D surface displays the trend, and the top color map plane presents the value distribution. When $\lambda_1 = 0.8$ and $\lambda_2 = 0.4$, the loss is minimum, indicating the best performance. Similarly, the hyper-parameter analysis of λ_1 and λ_2 on image pipeline in fusion attack model is evaluated in Fig. 5(b), in which $\lambda_1 = 0.6$ and $\lambda_2 = 0.4$ are the values to obtain the minimum loss $\mathcal{L}_{Image}$ in Eq. (11). For the impacts of η_1 and η_2 on LiDAR pipeline of both two attack models, the results in Fig. 5(c)



(a) Impacts of λ_1 and λ_2 on image pipeline of parallel attack model (b) Impacts of λ_1 and λ_2 on image pipeline of fusion attack model (c) Impact of η_1 and η_2 on LiDAR pipeline of both two models

Fig. 5: Hyper-parameter analysis of our two proposed attack models.

TABLE V: Quantitative performance statistics of parallel attack model (model 1) and fusion attack model (model 2)

	Baseline w/o attack	Model 1 attack on I	Model 1 attack on F	Model 2 attack both
Utility of image on non-target class	0.85±0.01	0.81±0.01	0.85±0.01	0.65±0.03
Accuracy of image on vehicles	0.85±0.01	0.39±0.02	0.85±0.01	0.26±0.02
Utility of LiDAR	N/A	N/A	0.62±0.02	0.61±0.02
Accuracy of LiDAR on vehicles	$0.99\pm9.03\times10^{-6}$	$0.99\pm8.99\times10^{-6}$	$0.03\pm1.13\times10^{-5}$	$0.02\pm9.65\times10^{-6}$

show that $\eta_1 = 0.6$ and $\eta_2 = 0.6$ are the best setting to gain the minimum loss $\mathcal{L}_{LiDAR}$ in Eq. (9). In the following sections, the experiments are conducted with the following settings of these critical hyper-parameters: $\lambda_1 = 0.8$ and $\lambda_2 = 0.4$ for image pipeline in parallel attack model, λ_1 and λ_2 for image pipeline in fusion attack model, and $\eta_1 = 0.6$ and $\eta_2 = 0.6$ for LiDAR pipeline in both two models.

C. Effectiveness of Our Attack

In this part, to understand our attack power towards the LLF-style perception systems in autonomous vehicles, the adversarial samples are generated by using our parallel attack model (*i.e.*, model 1) to separately attack the image model and the LiDAR model.

For the images, the results of Fig. 6(a) show that the vehicles are correctly detected (colored in blue) on the original street view data; while after being attacked, the vehicles cannot be detected in blue any more as shown in Fig. 6(b). In Fig. 6(c), the vehicles are detected and labeled by the LiDAR model using yellow boxes on the original LiDAR data F. When our adversarial example attack is launched, the detection results are presented in Fig. 6(d), where the vehicles cannot be detected by the LiDAR model.

As shown in Table V, when the parallel attack model is implemented, the accuracies of vehicle detection on the images and LiDAR data are reduced to 0.39 and 0.03, respectively, which confirms that our parallel attack model can effectively attack the image model and the LiDAR model of the LLF-style perception systems in autonomous vehicles. In addition, all the values of accuracy variance are very small, indicating our parallel attack model has a high performance stability. Moreover, under the parallel attack model, the utility of image can be preserved well for the non-target class objects. Specifically, in Table V, the utility of image in the adversarial samples can achieve 0.81 for images and reach 0.62 for the LiDAR data.

Notably, our fusion attack model (*i.e.*, model 2) simultaneously generates adversarial samples for both the image and LiDAR models from a low-dimensional latent vector z that has some content loss due to dimension reduction, so it cannot build complex street view data as clear as the parallel attack model, which is consistent with the analysis in Section III-F. The performance of our fusion attack model for the HLF-style perception systems of autonomous vehicles is demonstrated in Section IV-E.

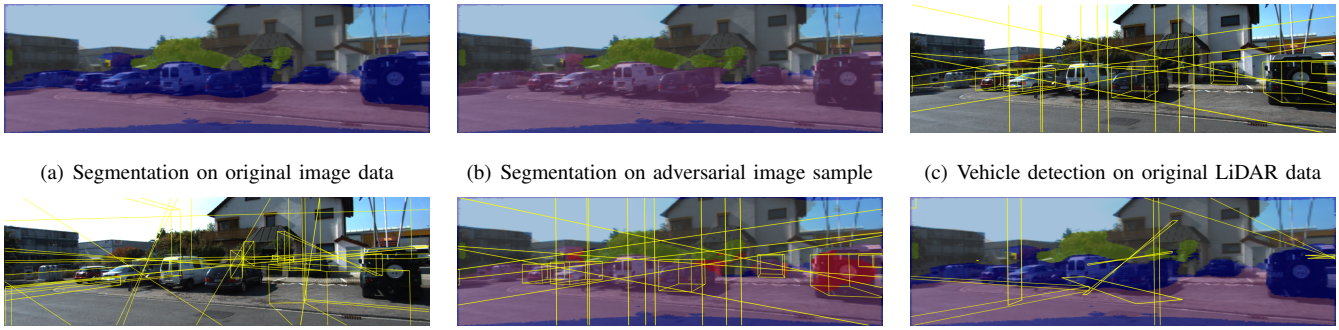
D. Limitation of Single-Source Attack

To examine the limitations of single-source adversarial sample attack, only one data source is attacked under the parallel attack model. In Fig. 6(e), after the image model is attacked, the LiDAR model still detects all vehicles correctly. In Fig. 6(f), when the LiDAR model is down, the image model can perform object detection normally.

Additionally, the results of Table V can confirm that the image model and the LiDAR model can help each other for object detection. For examples, the detection accuracy of image is decreased from 0.85 to 0.39 when only the image model is attacked while the detection accuracy of LiDAR remains the same; and the detection accuracy of LiDAR is reduced from 0.99 to 0.03 when only the LiDAR model is attacked while the detection accuracy of image does not change. Thus, the single-source adversarial sample attack on autonomous vehicles may fail when more than one data source is utilized in their perception systems; that is, it is hard for the single-source adversarial sample attack to beat the LLF-style or HLF-style perception systems.

E. Capability of Multi-Source Attack

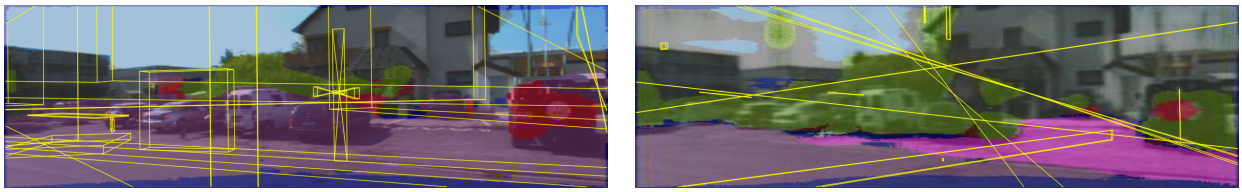
To successfully break down the perception systems of autonomous vehicles, attacks should be launched towards the image model and the LiDAR model at the same time.



(a) Segmentation on original image data (b) Segmentation on adversarial image sample (c) Vehicle detection on original LiDAR data

(d) Vehicle detection on adversarial LiDAR data (e) LiDAR-assisted detection with image being attacked (f) Image-assisted detection with LiDAR being attacked

Fig. 6: Detection results on image and LiDAR data. In image data, the correct segmentation color of vehicles is blue and the yellow line boxes are the detected region of vehicles by LiDAR data.



(a) Both image and LiDAR data are attacked by parallel attack model (b) Both image and LiDAR data are attacked by fusion attack model

Fig. 7: Detection results on both image and LiDAR data under our two attack models. In 7(a), the vehicles are mostly segmented as pink color, which is "road" class; while in 7(b) the vehicles are segmented green and red, representing "grass" and "bike" classes, respectively.

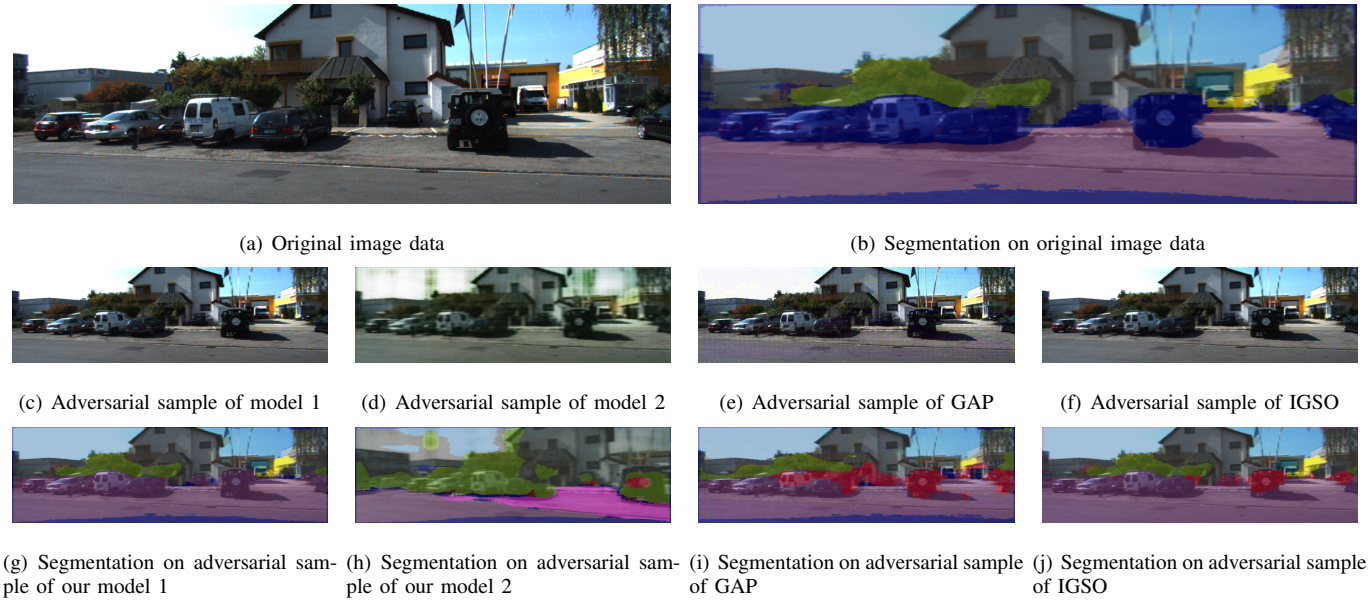
From the detection results under the parallel attack model in Fig. 7(a), one can see that both the image and the LiDAR models fail to predict vehicles at all. In Table V, when both the image and LiDAR models are attacked by our parallel attack model, the accuracies of vehicle detection for the image model and the LiDAR model are respectively decreased to 0.39 and 0.03, which means our parallel attack model can effectively beat both the image and the LiDAR models. Therefore, we can conclude that it is effective to launch our parallel attack towards the LLF-style perception systems of autonomous vehicles.

In the fusion attack model, due to the generative property of VAE, the adversarial samples of image and LiDAR can be simultaneously generated from a low-dimensional latent vector z. In the experiments, a 128-dimensional vector z is randomly picked from $\mathcal{N}(0,1)$ and then is fed into both the image and LiDAR pipelines as shown in Fig. 4. After z passes through the decoders: Dec_I and Dec_F, adversarial samples I' and F' are generated. Fig. 7(b) shows the detection results after being attacked by the fusion attack model, in which we observe that neither the image model nor the LiDAR model can detect vehicles. In Table V, when the fusion attack model is implemented, the accuracies of vehicle detection on the image data and the LiDAR data are reduced to 0.26 and 0.02, respectively, implying that our fusion attack model can effectively attack the HLF-style perception systems of autonomous vehicles.

F. Baseline Comparison

In this subsection, we compare the effectiveness and the efficiency of our parallel attack model (*i.e.*, model 1), our fusion attack model (*i.e.*, model 2), Iterative Gradient Sign Optimization (IGSO) [11], and the Generative Adversarial Perturbations model (GAP) [44] when the image model $f(I)$ is attacked.

1) **Attack Effectiveness:** In Fig. 8, the original street view images and the adversarial images generated by the four adversarial attack models are presented, where the first row shows the original street view image and its ground truth semantic segmentation. The second and third rows present four attacked street view images and their corresponding semantic segmentation results, respectively. Compared with Fig. 8(a), the adversarial sample of GAP in Fig. 8(e) contains much more noisy pixels than our model 1 in Fig. 8(c). Due to the over perturbation in GAP, its semantic segmentation accuracy for both target class and non-target class objects is lower than our model 1's semantic segmentation accuracy. In Fig. 8(f), the optimized adversarial sample of IGSO has good visual quality and is comparable to our model 1's result. The segmentation of IGSO in Fig. 8(j) successfully misleads the detection result on vehicles, because IGSO is an iterative gradient based attack method and likely to obtain a good adversarial solution with enough iteration and updating time. Finally, Fig. 8(d) and Fig. 8(h) show that the visual quality of the adversarial sample of our model 2 is worst among these four methods, which is resulted from the essential of data generation in our model



(a) Original image data (b) Segmentation on original image data

(c) Adversarial sample of model 1 (d) Adversarial sample of model 2 (e) Adversarial sample of GAP (f) Adversarial sample of IGSO

(g) Segmentation on adversarial sample of our model 1 (h) Segmentation on adversarial sample of our model 2 (i) Segmentation on adversarial sample of GAP (j) Segmentation on adversarial sample of IGSO

Fig. 8: Visual quality comparison between original image and adversarial samples (model 1 represents our parallel attack model, and model 2 represents our fusion attack model). The correct segmentation color of vehicles is blue.

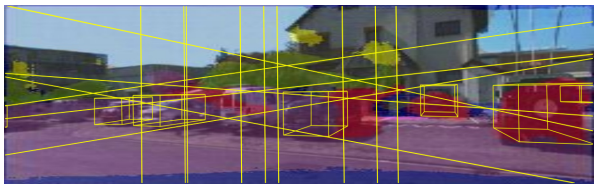


Fig. 9: Failure of GAP attack on perception system when LiDAR detection is utilized.

TABLE VI: Quantitative comparison on image perception model between our two models and two baseline models

	Model 1	Model 2	GAP [44]	IGSO [11]
SSIM	0.87±0.005	0.56±0.041	0.79±0.010	0.88±0.035
Utility	0.81±0.012	0.65±0.028	0.74±0.024	0.83±0.017
w/o defense	0.39±0.022	0.26±0.019	0.32±0.027	0.34±0.014
w/ defense	0.51±0.007	0.37±0.020	0.55±0.009	0.49±0.010

2 – a low-dimension vector z is used to generate adversarial data and thus more information is lost. On the other hand, from the viewpoint of attack effectiveness, the segmentation result of the adversarial sample of model 2 achieves the best performance, which is consistent with our analysis in Section III-F.

As illustrated via Fig. 6(e) and Fig. 6(f), only attacking one data source is hard or impossible to beat the entire perception system when multi-source data is used for object detection. In Fig. 9, the LiDAR model can still recognize every vehicle in the presence of GAP attack. In other words, GAP cannot successfully attack the LLF-style or HLF-style perception systems of autonomous vehicles, which demonstrates the superiority of our two attack models. The same result is observed on IGSO attack, because both GAP and IGSO do not attack the LiDAR model.

Besides, the performance statistics of the four attack models are listed in Table VI, where Structural Similarity Index Metric (SSIM), image utility, detection accuracy without defense, and detection accuracy with defense are adopted to compare attack performance. SSIM is a scalar number in $[0, 1]$ and used to measure the similarity between images; especially, 1 means exactly the same, and 0 means totally different [45]. In order to investigate the robustness of attack models, the average detection accuracy with and without defense is compared. We employ adversarial training as the defense strategy, which is broadly used in research [5], [46], [47] to protect machine learning model from adversarial samples. During the adversarial training, we inject adversarial samples generated by the four attack models into benign training datasets. Correspondingly, four mixed datasets are built with half adversarial samples and half benign. We train the defense model on each mixed dataset and then re-attack them with the four attack models.

The results of SSIM in Table VI show that the adversarial images generated by our model 1 and IGSO are closer to the original image compared with the other two attack models. When there is no defense, the attack performance of our model 2 is the best with the lowest accuracy due to information loss in the low-dimensional vector z, and GAP is slightly better than our model 1 partially because it adds more noise into image data. The image quality and utility of IGSO are the best among these four models. But as we emphasized before, IGSO takes too much time to reach a good performance (as illustrated in TABLE VII), which is not applicable for the scenario of autonomous vehicles. With the adoption of adversarial training as a defense method, the attack performance of our model 1, our model 2, GAP and IGSO is reduced by 12%, 11%, 23%, and 15%, respectively. The comparison between detection accuracy w/ and w/o defense declares that the robustness of our two proposed models outperforms the baseline models.

TABLE VII: Time efficiency comparison between our two models and two baseline models

	Model 1	Model 2	GAP [44]	IGSO [11]
images/s	13.58±0.083	22.63±0.018	13.41±0.034	0.01±0.004

2) **Attack Efficiency:** Finally, we evaluate the time efficiency of different attack models in terms of throughput that is defined as the number of processed images per second (*i.e.* images/s). We test the four attack models with 400 images for 10 times and present their average throughput and corresponding variances in Table VII. It is clear that model 2 has the highest throughput because it generates adversarial images from a low-dimensional vector z and the size of its parameter and input is less than the other three models.

For the adversarial samples of image data, our model 1 and GAP have comparable attack effectiveness and attack efficiency. Both of the two models are generation-based adversarial attack, but their largest difference is that our model 1 uses image-dependent dynamic target while GAP uses a static target. That is, for all images in GAP implementation, their objective is to manipulate image to get a pre-defined target segmentation. Because of this multiple-to-one mapping, more noise need to be injected to all images to counteract image difference as shown in Fig. 8(e), and the generated perturbations targeting on static semantic label are like a universal noise without oblivious difference, leading to smaller SSIM, lower image utility and worse attack robustness. On the contrast, our model 1 generates perturbation based on each image, which means only small specific objects are affected by image manipulation and less noise is needed, improving SSIM, image utility, and attack robustness.

G. Attack Performance in V2V Communication

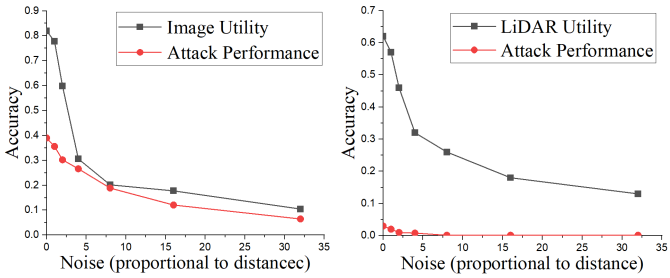
Since the V2V attack scenario is more practical in VANETs as analyzed in Section III-A, it is worth investigating the performance of our proposed attack models in V2V communication.

In the V2V attack scenario, the attacker generates adversarial samples based on his local captured data, which may be slightly different from the victim vehicles' original data due to the distance between the attacker's vehicle and the victim vehicles. It is well-known that the transmission distance between a sender and a receiver influences the quality of V2V communications and shared data [48]. In Eq. (12), the computation of signal-noise-ratio (SNR) is given [49].

$$\gamma = \frac{E_b}{N_0 B D_b}. \tag{12}$$

In Eq. (12), E_b is the energy per bit, N_0 is the noise density in W/Hz, D_b is the transmission distance per bit, and B is the channel bandwidth in the IEEE 802.11p OFDM PHY. Moreover, SNR in V2V communication has impacts on the attack capability: a lower SNR reduces the quality of the received adversarial samples, resulting in a worse attack performance.

To keep a comparability with the previous experimental results, we simulate the victim vehicle's received adversarial samples by adding appropriate noise to distort the original



(a) Accuracy on image w.r.t Noise (b) Accuracy on LiDAR w.r.t. Noise

Fig. 10: The impact of noise on data utility and attack performance.

adversary samples as more noise implies a lower SNR experienced by the victim and a larger transmission distance between the attacker and the victim. The added noise follows the normal distribution $\mathcal{N}(0, \sigma^2)$, where the value of $\sigma \in [0, 100]$ represents set a certain percentage of the range of original data value. Specifically, for image data, the range is 256 as its values falls within $[0, 255]$, and the added noise is $\sigma\% \times 256$; for LiDAR data, the range is 78 as its value falls into $[0, 77]$, and the added noise is $\sigma\% \times 78$. In this way, the SNR in image and LiDAR data is the same. Then, the attack performance is measured based on the noisy adversarial samples.

In Fig. 10, the x-axis represents the value of σ that indicates how much noise we add in the generated adversarial samples, "Image Utility" is the utility of image on non-target class in Table V, "Attack Performance" in Fig. 10(a) is the accuracy of image on vehicles in Table V, "LiDAR Utility" is the utility of LiDAR in Table V, and "Attack Performance" in Fig. 10(b) is the accuracy of LiDAR on vehicles in Table V. From Fig. 10, we can see that when there is no noise in the adversarial samples (*i.e.*, $\sigma = 0$), the results are the same as those in Table V. For the image data in Fig. 10(a), when the noise is small (*i.e.*, distance is close), increasing noise scale yields a dramatic reduction of the image utility but a slow degradation of attack performance. This indicates that within a short distance, the attack capability of adversarial examples can be relatively maintained and is reduced gradually. After the noise exceeds 10% of the data range, both image utility and attack performance become very low with a smooth decrease rate as the noise scale in increased, which shows that the received adversarial samples are almost unusable due to the large noise injection. For the LiDAR data in Fig. 10(b), we can obtain the similar observations. From the above results, we can conclude that: (i) the distance between the attacker and the victim in V2V communication does impact on the quality of received adversarial samples; (ii) when the distance is smaller than a range, the attack performance is reduced with an increased transmission distance; and (iii) when the distance is large enough, adversarial attack will lose its capability due to a bad communication quality.

H. Summary of Experiment Analysis

From the above comprehensive experiments, some critical conclusions can be drawn as follows. (i) Our two proposed

attack models can effectively attack the image perception systems and the LiDAR perception systems separately. (ii) Even facing multi-source perception systems (*i.e.*, the LLF-style and HLF-style perception systems of autonomous vehicles in this paper), our two proposed attack models can successfully beat the whole systems as well, which cannot be accomplished by the existing single-source adversarial sample attack. (iii) Compared with the state-of-the-art, our parallel attack model can not only achieve the comparable attack effectiveness and efficiency but better image utility and attack robustness, and our fusion attack model has better attack robustness and higher time efficiency. (iv) In the real implementation of adversarial sample attack in V2V communication, the attack performance is dependent on the transmission distance between an attacker and a victim.

V. Conclusion

In this paper, through analyzing digital attack on the practical perception systems of autonomous vehicles, we design two multi-source adversarial sample attack models, which has not been addressed before and brings the following innovations. First, in our proposed attack models, the correlation of multiple heterogeneous data sources is utilized for data fusion and sample generation, which can contribute to the technology development of generating adversarial samples. Second, compared with the traditional single-source adversarial sample attack models, our multi-source adversarial sample attack models have more power to successfully damage the image and LiDAR models at the same time, which is well confirmed by our intensive real-data experiments. Last but not least, the study of multi-source attack is helpful to enlighten effective defense solutions in our future research, further improving the security and safety performance for autonomous vehicles.

Acknowledgment

This work was partly supported by the National Science Foundation of U.S. (1741277, 1829674, 1704287, 2011845, and 1912753).

References

[1] J. Ni, X. Lin, and X. Shen, "Toward privacy-preserving valet parking in autonomous driving era," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2893–2905, 2019.

[2] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[3] J. Wang, Z. Cai, and J. Yu, "Achieving personalized k-anonymity-based content privacy for autonomous vehicles in cps," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4242–4251, 2019.

[4] Z. Cai, X. Zheng, and J. Yu, "A differential-private framework for urban traffic flows estimation via taxi companies," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 6492–6499, 2019.

[5] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble adversarial training: Attacks and defenses," *arXiv preprint arXiv:1705.07204*, 2017.

[6] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proceedings of the 2017 ACM on Asia conference on computer and communications security*. ACM, 2017, pp. 506–519.

[7] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "A general framework for adversarial examples with objectives," *ACM Transactions on Privacy and Security*, vol. 22, no. 3, p. 16, 2019.

[8] Z. Zhao, D. Dua, and S. Singh, "Generating natural adversarial examples," *arXiv preprint arXiv:1710.11342*, 2017.

[9] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song, "Generating adversarial examples with adversarial networks," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. IJCAI, 2018, pp. 3905–3911.

[10] J. Hendrik Metzen, M. Chaithanya Kumar, T. Brox, and V. Fischer, "Universal adversarial perturbations against semantic image segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, 2017, pp. 2755–2764.

[11] V. Fischer, M. C. Kumar, J. H. Metzen, and T. Brox, "Adversarial examples for semantic image segmentation," *arXiv preprint arXiv:1703.01101*, 2017.

[12] S.-T. Chen, C. Cornelius, J. Martin, and D. H. P. Chau, "Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2018, pp. 52–68.

[13] W. Zhao, S. Han, W. Meng, D. Sun, and R. Q. Hu, "Bsdp: Big sensor data preprocessing in multi-source fusion positioning system using compressive sensing," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 8866–8880, 2019.

[14] I. Goodfellow, "Nips 2016 tutorial: Generative adversarial networks," *arXiv preprint arXiv:1701.00160*, 2016.

[15] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[16] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[17] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," *arXiv preprint arXiv:1607.02533*, 2016.

[18] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, 2016, pp. 2574–2582.

[19] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2016, pp. 372–387.

[20] Z. Xiong, W. Li, Q. Han, and Z. Cai, "Privacy-preserving auto-driving: a gan-based approach to protect vehicular camera data," in *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019, pp. 668–677.

[21] Z. Xiong, Z. Cai, Q. Han, A. Alrawais, and W. Li, "Adgan: Protect your location privacy in camera data of auto-driving vehicles," *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2020.

[22] S. Tian, G. Yang, and Y. Cai, "Detecting adversarial examples through image transformation," in *Thirty-Second AAAI Conference on Artificial Intelligence*. AAAI, 2018, pp. 4139–4146.

[23] C. E. Tuncali, G. Fainekos, H. Ito, and J. Kapinski, "Simulation-based adversarial test generation for autonomous vehicles with machine learning components," in *2018 IEEE Intelligent Vehicles Symposium*. IEEE, 2018, pp. 1555–1562.

[24] Y. Cao, C. Xiao, B. Cyr, Y. Zhou, W. Park, S. Rampazzi, Q. A. Chen, K. Fu, and Z. M. Mao, "Adversarial sensor attack on lidar-based perception in autonomous driving," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2019, pp. 2267–2281.

[25] Y. Cao, C. Xiao, D. Yang, J. Fang, R. Yang, M. Liu, and B. Li, "Adversarial objects against lidar-based autonomous driving systems," *arXiv preprint arXiv:1907.05418*, 2019.

[26] N. Patel, K. Liu, P. Krishnamurthy, S. Garg, and F. Khorrami, "Lack of robustness of lidar-based deep learning systems to small adversarial perturbations," in *50th International Symposium on Robotics*. VDE, 2018, pp. 1–7.

[27] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille, "Adversarial examples for semantic segmentation and object detection," in *Proceedings of the IEEE International Conference on Computer Vision*. IEEE, 2017, pp. 1369–1378.

[28] S. Huang, N. Papernot, I. Goodfellow, Y. Duan, and P. Abbeel, "Adversarial attacks on neural network policies," *arXiv preprint arXiv:1702.02284*, 2017.

[29] Y.-C. Lin, Z.-W. Hong, Y.-H. Liao, M.-L. Shih, M.-Y. Liu, and M. Sun, "Tactics of adversarial attack on deep reinforcement learning agents," *arXiv preprint arXiv:1703.06748*, 2017.

[30] J. Lu, H. Sibai, E. Fabry, and D. Forsyth, "No need to worry about adversarial examples in object detection in autonomous vehicles," *arXiv preprint arXiv:1707.03501*, 2017.

[31] I. Evtimov, K. Eykholt, E. Fernandes, T. Kohno, B. Li, A. Prakash, A. Rahmati, and D. Song, "Robust physical-world attacks on machine learning models," *arXiv preprint arXiv:1707.08945*, 2017.

[32] Q. Chen, S. Tang, Q. Yang, and S. Fu, "Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 514–524.

[33] I. Ali, T. Lawrence, A. A. Omala, and F. Li, "An efficient hybrid signcryption scheme with conditional privacy-preservation for heterogeneous vehicular communication in vanets," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 11 266–11 280, 2020.

[34] P. K. Singh, A. Agarwal, G. Nakum, D. B. Rawat, and S. Nandi, "Mpfslp: Masqueraded probabilistic flooding for source-location privacy in vanets," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 11 383–11 393, 2020.

[35] H. Cho, Y.-W. Seo, B. V. Kumar, and R. R. Rajkumar, "A multi-sensor fusion system for moving object detection and tracking in urban driving environments," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1836–1843.

[36] J.-r. Xue, D. Wang, S.-y. Du, D.-x. Cui, Y. Huang, and N.-n. Zheng, "A vision-centered multi-sensor fusing approach to self-localization and obstacle perception for robotic cars," *Frontiers of Information Technology & Electronic Engineering*, vol. 18, no. 1, pp. 122–138, 2017.

[37] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[38] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3d object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2018, pp. 4490–4499.

[39] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," in *International conference on machine learning*. PMLR, 2016, pp. 1558–1566.

[40] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *4th International Conference on Learning Representations*, 2016. [Online]. Available: http://arxiv.org/abs/1511.06434

[41] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, 2016, pp. 3213–3223.

[42] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361.

[43] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, 2015, pp. 3431–3440.

[44] O. Poursaeed, I. Katsman, B. Gao, and S. Belongie, "Generative adversarial perturbations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2018, pp. 4422–4431.

[45] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[46] A. Arnab, O. Miksik, and P. H. Torr, "On the robustness of semantic segmentation models to adversarial attacks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2018, pp. 888–897.

[47] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14 410–14 430, 2018.

[48] D. Gao, Z. Li, Y. Liu, and T. He, "Neighbor discovery based on cross-technology communication for mobile applications," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 11 179–11 191, 2020.

[49] M. Killat and H. Hartenstein, "An empirical model for probability of packet reception in vehicular ad hoc networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, pp. 1–12, 2009.

Zuobin Xiong received the B.S. degree from the Department of Mathematics, Northeast Forestry University, Harbin, Heilongjiang, China in 2016, and the M.S. degree from College of Computer Science and Technology, Harbin Engineering University, Harbin, Heilongjiang, China in 2019. Currently, he is pursuing the Ph.D. degree at the Department of Computer Science, Georgia State University, Atlanta, GA, USA. His research interests lie in the area of Machine Learning, Data Mining, Internet of Things (IoT) along with the privacy and security issues of above related topics.

Honghui Xu is a Ph.D. student in Department of Computer Science at Georgia State University (GSU). He received a Bachelor's degree from University of Electronic Science and Technology of China (UESTC). His research focuses on machine learning and deep learning, including the fundamental theory of machine learning, the applications of deep learning in computer vision field, and the topic about privacy-preserving machine learning.

Wei Li (M'16) is currently an Assistant Professor in the Department of Computer Science at Georgia State University. Dr. Li received her Ph.D. degree in computer science, from The George Washington University, in 2016 and M.S. degree in Computer Science from Beijing University of Posts and Telecommunications, in 2011. She won the Best Paper Awards in ACM MobiCom Workshop CRAB 2013 and international conference WASA 2011, respectively. Her current research spans the areas of blockchain technology, security and privacy for the Internet of Things and Cyber-Physical Systems, secure and privacy-aware computing, Big Data, game theory, and algorithm design and analysis. She is a member of IEEE and a member of ACM.

Zhipeng Cai (SM'06) is currently an Associate Professor at Department of Computer Science, Georgia State University, USA. He received his PhD and M.S. degrees in the Department of Computing Science at University of Alberta, and B.S. degree from Beijing Institute of Technology. Prior to joining GSU, Dr. Cai was a research faculty in the School of Electrical and Computer Engineering at Georgia Institute of Technology. Dr. Cai's research areas focus on Internet of Things, Machine Learning, Cyber-Security, Privacy, Networking and Big data. Dr. Cai is the recipient of an NSF CAREER Award. He served as a Steering Committee Co-Chair and a Steering Committee Member for WASA and IPCCC. Dr. Cai also served as a Technical Program Committee Member for more than 20 conferences, including INFOCOM, ICDE, ICDCS. Dr. Cai has been serving as an Associate Editor-in-Chief for Elsevier High-Confidence Computing Journal (HCC), and an Associate Editor for more than 10 international journals, including IEEE Internet of Things Journal (IoT-J), IEEE Transactions on Knowledge and Data Engineering (TKDE), IEEE Transactions on Vehicular Technology (TVT).